Mestrado em Engenharia Eletrotécnica e de Computadores

Automação e Sistemas

CONTROLO DE ACESSOS

SISTEMA DE CREDENCIAÇÃO

Sérgio Fernando Dias Martins

Departamento de Engenharia Electrotécnica

Instituto Superior de Engenharia do Porto

2014

Este relatório apresenta a Tese do Mestrado em Engenharia Electrotécnica e de Computadores

Candidato: Sérgio Fernando Dias Martins, Nº 1120987,
1120987@isep.ipp.pt; sfmartins@ana.pt

Orientação científica e pedagógica: Prof. Cecília Reis, cmr@isep.ipp.pt

Departamento de Engenharia Electrotécnica

Instituto Superior de Engenharia do Porto

Junho 2014

Resumo

Os sistemas de controlo de acessos são ferramentas usadas para garantir níveis de segurança, seletividade e controlabilidade no acesso a áreas ou recursos privados minimizando constrangimentos provocados por inspeções.

O Aeroporto Francisco Sá Carneiro é uma infraestrutura composta por vários edifícios, com áreas de acesso restrito caracterizadas por diversas especificidades. Para fazer a gestão de acessos, alguns edifícios estão equipados com sistemas de controlo automáticos para rastreio de pessoas e viaturas, noutros existe um elemento humano a desempenhar essa tarefa. Operacionalmente, no Aeroporto são usados conjuntos de ferramentas de *software*, desenvolvidas por várias entidades e em períodos de tempo diferentes que sustentam os procedimentos relacionados com o controlo de acessos.

Por questões de suporte das aplicações, questões de inclusão de novas funcionalidades e adaptação a exigências do dia-a-dia, pretende-se substituir as várias ferramentas atuais por uma plataforma única que agregue a satisfação de todas as necessidades decorrentes do processo de controlo de acessos e de credenciação de pessoas e viaturas.

Este documento de dissertação descreve o estudo, o projeto e a implementação de um protótipo funcional da nova plataforma de *software* de controlo de acessos para o Aeroporto Francisco Sá Carneiro.

Palavras-Chave

Controlo de Acessos; Controlo de Viaturas; Credenciação; Portaria Eletrónica; Identificação; Autenticação.

Agradecimentos

Á Professora Cecília Reis, por aceitar o desafio me orientar num tema tão diverso e pelas sugestões importantes para que este trabalho atingisse a qualidade que tem.

Aos colegas da minha equipa de trabalho na ANA, cujas conversas me levam sempre mais longe.

E à Dina. Por tudo. Pelos anos de companheirismo. Pelo apoio irredutível. Pelo carinho nos momentos mais difíceis. E pelo amor de sempre.

Sérgio Martins

"*I believe in the hands that work,*

in the brains that think,

and in the hearts that love"

Richard Branson

Índice

Índice de Figuras

Índice de Tabelas

Acrónimos

2FA	-	*Two Factor Authentication*
ANA	-	ANA – Aeroportos de Portugal S.A.
ASC	-	Aeroporto Francisco Sá Carneiro
CCTV	-	*Closed Circuit Television.*
CDMA	-	*Code Division Multiple Access*
CNPD	-	Comissão Nacional de Protecção de Dados
Dpi	-	*Dots per inch*
EEPROM	-	*Electric Erasable and Programmable Read-Only Memory*
EER	-	*Equal Error Rate*
ETSI	-	*European Telecommunications Standards Institute*
FAR	-	*False Acceptance Rate*
FER	-	*Failure to Enroll Rate*
FRAM	-	*Ferroelectric Random Access Memory*
FRR	-	*False Rejection Rate*
FTC	-	*Failure to Capture Rate*
FTIR	-	*Frustrated total internal reflection*
GSM	-	*Global System for Mobile Communication*

I^2C	-	*Inter-Integrated Circuit, Inter-IC*
IATA	-	*International Air Transport Association*
IP	-	*Internet Protocol*
LINQ	-	*Language-Integrated Query*
LVO	-	*Low Visibility Operation*
MAC	-	*Media Access Control*
MFA	-	*Multifactor Authentication*
NFC	-	*Near Field Communication*
NPU	-	*Numeric Processing Unit* (*cryptoprocessor*)
PACS	-	*Physical Access Control System*
PDF	-	*Portable Document Format*
PIN	-	*Personal Identification Number*
PIV	-	*Personal Identity Verification*
PoE	-	*Power over Ethernet*
Ppi	-	*Pixels per inch*
PSIM	-	*Physical Security Information Management*
PSP	-	Polícia de Segurança Publica
RADAR	-	*Radio Detecting and Ranging*
RAM	-	*Random Access Memory*
RDBMS	-	*Relational Database Management System.*

RFID	-	*Radio-Frequency IDentification.*
SACA	-	Sistema Automático de Controlo de Acessos
SEF	-	Serviço de Estrangeiros e Fronteiras
SFA	-	Single-factor authentication
SGO	-	Sistema de Gestão Operacional
SIM	-	*Subscriber Identity Module*
SMS	-	*Short Message Service*
SO	-	Sistema Operativo
SOC	-	*Storage On Card.*
SOPs	-	*Standard Operating Procedures*
SWP	-	*Single Wire Protocol*
UART	-	*Universal Asynchronous Receiver –Transmitter*
UCA	-	Unidade de Controlo de Acessos
UML	-	*Unified Modeling Language*
UMTS	-	*Universal Mobile Telecommunication System*
USB	-	*Universal Serial Bus*
USIM	-	*Universal Subscriber Identity Module*
VLAN	-	*Virtual Local Area Network*

Glossário

Aerogare	Terminal aeroportuário de passageiros.
Lado ar	"Zona de movimento dos aeroportos e seus terrenos e edifícios adjacentes, ou parte destes cujo acesso é controlado." [1]
Lado Terra	"Zonas dos aeroportos e os terrenos adjacentes ou partes destes, não incluídos no lado ar". [1]
Taxiway	Caminho de circulação de aeronaves entre as pistas de aterragem/descolagem e as plataformas de estacionamento.
SIM	*Subscriber Identity Module* é um cartão usado nos dispositivos de comunicação móvel que permite identificar e autenticar de forma segura o subscritor do serviço de comunicações GSM. Os cartões com as mesmas características mas que acedem a serviços de comunicação UTMS denominam-se USIM – *Universal Subscriber Identity Module.*
GSM	*Global System for Mobile Communication*, sistema de telecomunicações móveis digitais, celulares de segunda geração que opera nas frequências de 900, 1800 e 1900MHz.
UMTS	*Universal Mobile Telecommunication System*, sucessor para a Europa do sistema de comunicações GSM, que opera na frequência de 2000MHz: A maior diferença relativamente ao GSM é o uso da tecnologia CDMA que proporciona maiores taxas de transferência
CDMA	*Code Division Multiple Access*, é um método de acesso concorrente para transmissão de dados de vários transmissores para um recetor usando uma frequência.

1. INTRODUÇÃO

O presente documento é o relatório da Tese do Mestrado em Engenharia Eletrotécnica e de Computadores, ramo Automação e Sistemas apresentado no Departamento de Engenharia Eletrotécnica do Instituto Superior de Engenharia do Porto - ISEP. O trabalho apresenta o desenvolvimento de uma plataforma de credenciação e controlo de acessos para implementação no Aeroporto Francisco Sá Carneiro.

Este capítulo apresenta um enquadramento do âmbito da tese de mestrado, fazendo uma descrição histórica dos últimos anos do controlo de acessos no Aeroporto Francisco Sá Carneiro e definindo os objetivos do trabalho.

O conceito de segurança é um requisito que se pretende em todas as atividades. Seja sob a vertente preventiva ou mais pró-ativa em todos os âmbitos de atuação e do conhecimento a segurança é um valor indiscutível.

Os locais onde as pessoas se movem pelo seu estado de propriedade, por questões fronteiriças, por questões de confidencialidade ou por questões de perigosidade classificam-se em vários níveis de restrição de acessos. Uma das áreas em que se aplica o

termo *segurança* prende-se com o controlo do acesso de pessoas a zonas restritas. Os aeroportos são um caso particular de espaços onde coexistem zonas com diferentes níveis de restrição de acessos nos quais é necessário garantir que apenas as pessoas com permissões acedem às respetivas zonas.

1.1. CREDENCIAÇÃO NO AEROPORTO FRANCISCO SÁ CARNEIRO

O Aeroporto Francisco Sá Carneiro - ASC, por mandato regulamentar da atividade que nele se exerce, está dividido em várias áreas de acesso reservado ou condicionado classificadas em, [1]:

- Áreas públicas: áreas às quais o público tem acesso;
- Áreas reservadas: áreas de acesso condicionado;
- Áreas restritas: áreas de acesso controlado a fim de garantir a segurança da aviação civil sendo necessária a existência de cartões de acesso para pessoas e dísticos para viaturas. Pessoas e veículos são controlados e rastreados a 100%[2]. Fazem parte das áreas restritas as salas de embarque e desembarque, os caminhos de circulação de passageiros em trânsito, as áreas operacionais com pistas, *taxiways*, plataformas de estacionamento de aeronaves, terminais de carga e edifícios de manutenção.

Atualmente no ASC, pode-se obter credenciação com autorização de acesso para pessoas e para viaturas. As credenciais para pessoas podem ser dos seguintes tipos [2]:

- Cartão de acesso único permanente, Figura 1: emitido a Funcionários de Entidades que operam no Aeroporto com vínculo laboral de efetividade, ou quando a empresa está sujeita a um contrato de concessão temporário que garanta o período de vigência do cartão. Válido, no máximo por cinco anos.

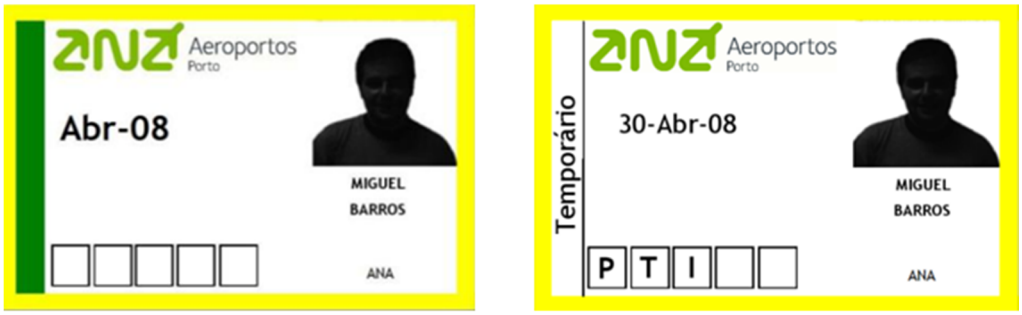


Figura 1 – Cartão de identificação permanente e temporário, adaptado de [1].

- Cartão de acesso único temporário: emitido a Funcionários de Entidades que operam no Aeroporto com contrato de trabalho temporário, ou quando a empresa está sujeita a um contrato de concessão temporário que garanta o período de vigência do cartão. Válido, no máximo por um ano.

- Cartão de acesso pontual: podem ser de dois tipos ou para fazer face a uma necessidade de carater pontual como visitas, auditorias, etc. em que os possuidores deste cartão deverão ser acompanhados por um Funcionário da Entidade requerente, Figura 2. Válido no máximo por cinco dias. Ou se emitem cartões de acesso pontual para permitir que passageiros utilizadores do serviço de chegadas, possam aceder a áreas restritas para reclamação de bagagem perdida, Figura 3.

Figura 2 – Cartões de identificação pontual, adaptado de [1].

Figura 3 – Cartões de identificação pontual para passageiros, adaptado de [1].

As credenciais para viaturas podem ser dos seguintes tipos [2]:

- Identificação lateral fixa: emitida para viaturas afetas permanentemente a atividades que se processam nas áreas restritas, é constituída por dois conjuntos de números e por um dístico. Válida no máximo por um ano, Figura 4 e Figura 5.

Figura 4 – Exemplos de dísticos laterais de identificação de viaturas.

Figura 5 – Dísticos permanente de viaturas, adaptado de [1].

- Dísticos de acesso temporário: emitidos para viaturas com necessidade de circulação temporário nas áreas restritas. Valido no máximo por dois meses, Figura 6.

Figura 6 – Dísticos temporário de viaturas, adaptado de [1].

Os vários cartões de identificação numa das faces apresentam impressas informações relevantes para a credenciação como: data de validade, nome ou matriculas e empresas que representam, uma cor e/ou um conjunto de até cinco letras que representam as áreas a que lhes é concedido acesso conforme descrito no Anexo A .

1.2. EVOLUÇÃO HISTÓRICA DA CREDENCIAÇÃO NO ASC

A atividade do Aeroporto Francisco Sá Carneiro, [5], remonta a 1945 no então chamado Aeroporto de Pedras Rubras. Em 1956 recebeu o seu primeiro voo internacional e ao longo dos anos tem sido alvo de várias intervenções de melhoramentos e ampliações. Atualmente, está equipado com uma pista com 3.480m de comprimento, e por um terminal de passageiros com área bruta 150.000m^2 que no ano de 2013 serviu e 6,3 milhões de passageiros. Desde 2006[1] o Aeroporto Francisco Sá Carneiro já foi classificado sete vezes num dos três primeiros lugares de melhor aeroporto da Europa em qualidade de serviço, pelo *Airport Council International*, [3].

[1] 2006, 2007, 2008, 2009, 2010, 2011 e 2013.

O perímetro do ASC, implementado numa área de 320 hectares, além do terminal de passageiros contém diversos edifícios como torre de controlo, terminal de carga, quartel de bombeiros, edifícios de manutenção, *hangars*, edifícios de apoio ao abastecimento de combustível, edifícios administrativos, edifícios comerciais e diversas áreas de estacionamento, Figura 7, que tem características concretas de restrição de acessos.

1- Terminal de passageiros; 2 – Terminal de Carga ; 3 – Edifícios Administrativos; 4 – Edifícios de manutenção; 5 – Edifícios *Rent-a-car*

Figura 7 – Aeroporto Francisco Sá Carneiro – vista aérea parcial, [4].

Ao longo dos anos a forma como se efetuou a gestão e controlo de acessos mudou e evoluiu para acompanhar as exigências e necessidades de cada época. Antes da implementação do atual terminal de passageiros as pessoas que desenvolviam atividade profissional no ASC eram detentoras de um cartão oficial de identificação com fotografia que permitia o acesso às áreas reservadas.

Durante os anos 2002 a 2005 foi desenvolvida e consecutivamente melhorada, por técnicos operacionais e técnicos informáticos da empresa uma aplicação de *software* denominada "Sistema de Gestão Operacional" – SGO, que entre outras funcionalidades fazia a gestão dos cartões de identificação de pessoas, das licenças de condução de viaturas no lado ar do

aeroporto e dos dísticos de identificação de viaturas, Figura 8. Este foi o primeiro passo para o registo de informação sobre acessos em formato digital.

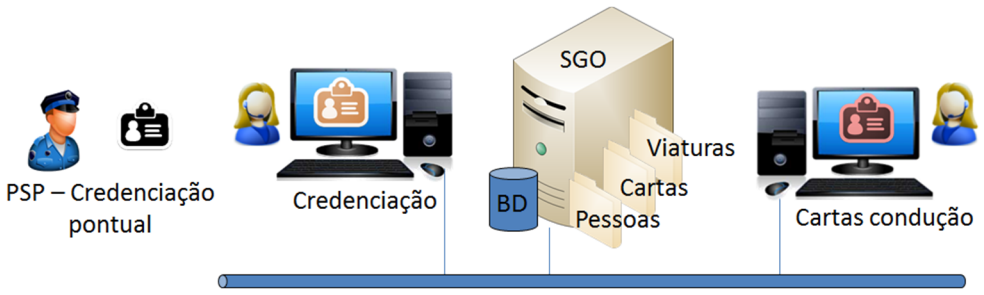


Figura 8 – A credenciação no Sistema de Gestão Operacional – SGO.

Nos anos 2000 a 2005 o Aeroporto Francisco Sá Carneiro foi alvo de um grande projeto de ampliação que introduziu novos edifícios, novos parques de estacionamento de viaturas, novas posições de estacionamento de aeronaves e um novo terminal de passageiros. A nova aerogare foi então equipada com um sistema de abertura e controlo eletrónico automático de portas denominado Sistema Automático de Controlo de Acessos – SACA, Figura 9. A partir de setembro de 2005, o *staff* utilizador da aerogare passou a ser portador de um cartão de funcionamento por proximidade que lhes permitia abrir portas no edifício segundo o seu perfil de acesso.

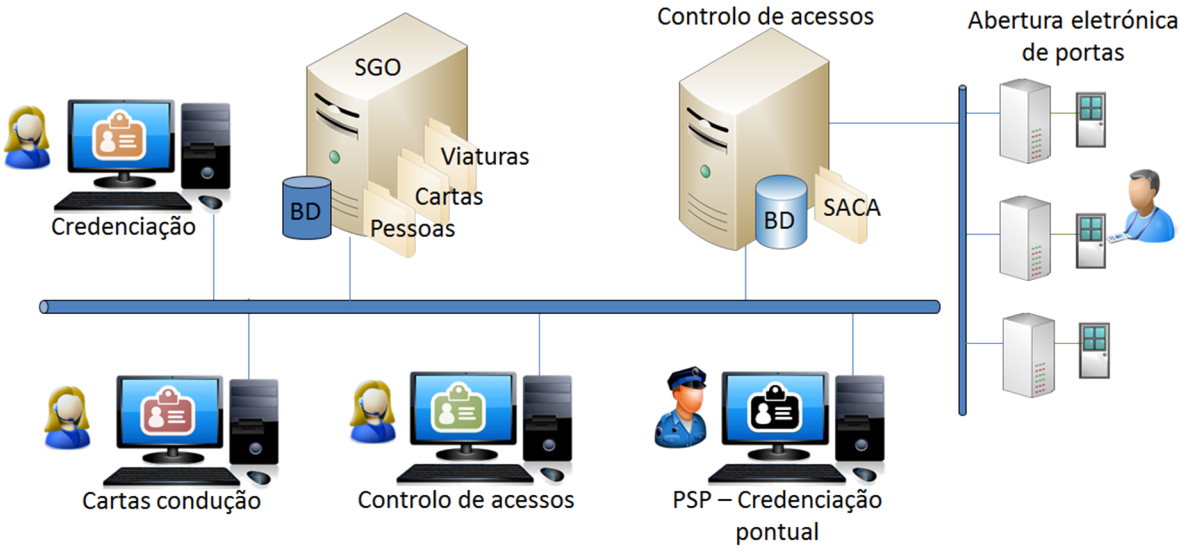


Figura 9 – Introdução do sistema automático de controlo de acessos.

Neste período o *staff* usava um cartão de credenciação, uma licença de condução, um cartão de abertura de portas e os colaboradores da ANA também tinham um cartão de

registo de picagens para controlo de assiduidade. Do ponto de vista de serviços existiam quatro entidades distintas a emitir cartões. Este cenário evoluiu rapidamente para a criação do "Cartão único" que consistiu na agregação das várias funcionalidades, apenas num cartão, Figura 10, e na criação um serviço incumbido da gestão do cartão, Figura 11.

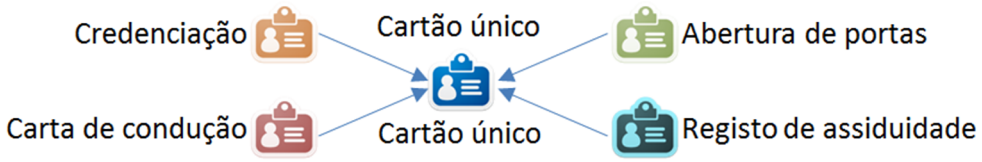


Figura 10 – Cartão único.

Na evolução para o cartão único foi também implementado o sistema de portarias eletrónicas, que fazem a validação do acesso de pessoas através do cartão único. Esta funcionalidade foi instalada em locais onde há necessidade de restringir acessos e também de controlar os haveres que as pessoas transportam. Exemplos destes locais são os pontos de rastreio de *staff* que se deslocam a pé ou em viaturas e as entradas para fornecedores.

Com a solução do cartão unico, as pessoas passaram a ser detentoras de um cartão RFID que as identifica do ponto de vista de acesso, valida a permissão de condução, permite abrir portas de controlo automático e para colaboradores ANA permite registar a assiduidade.

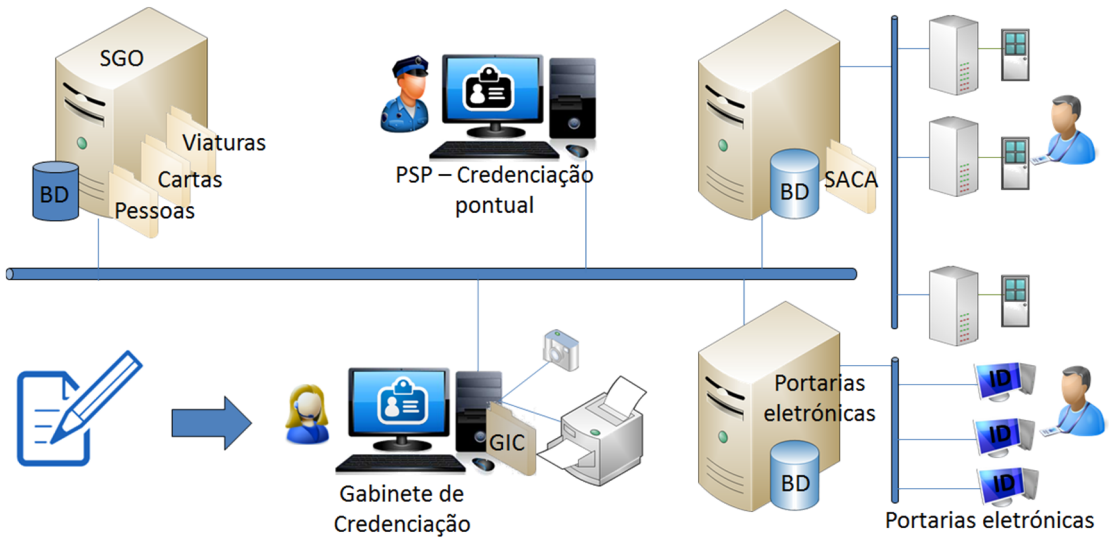


Figura 11 – Gabinete de Credenciação, no ciclo do cartão único.

Posteriormente, por dificuldades operacionais, a licença de condução foi separada do cartão único e passou a ser representada por um cartão de uso exclusivo para esse fim.

Durante o ano de 2011 foi desenvolvido, a nível de toda a empresa ANA, um portal baseado em tecnologia *web* onde os interessados podem solicitar cartões de acesso para qualquer aeroporto, introduzindo a documentação solicitada e posteriomente acompanhamento do processo remotamente, Figura 12. Após a validação processual do pedido, o portal envia, automaticamente, para o respetivo aeroporto, a informação de emissão de cartões, seguindo depois o processo local adaptado a cada realidade.

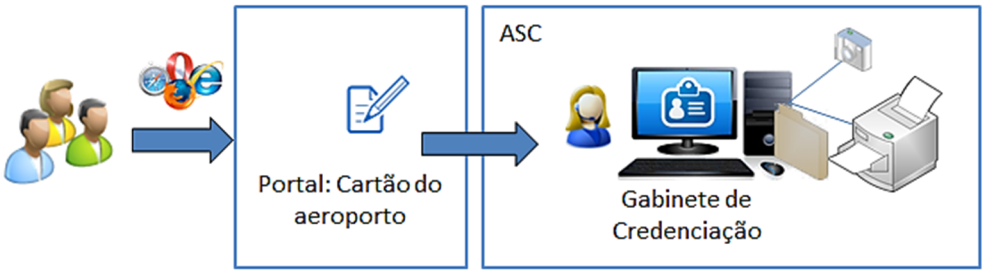


Figura 12 – Portal eletrónico de credenciação.

1.3. ÂMBITO DO TRABALHO

A solução para controlo e gestão de credenciação e acessos atualmente implementada no Aeroporto Francisco Sá Carneiro, envolve várias ferramentas desenvolvidas e instaladas em diversos momentos e de origens várias, estas ferramentas são descritas na Tabela 1.

Tabela 1 – Credenciação/Controlo de acessos: soluções instaladas.

Sistema	Descrição	Empresa de desenvolvimento
SGO – Sistema de Gestão operacional	Sistema composto por: • Base de dados em SQL *Server* que contem a informação de credenciação de pessoas e viaturas. • Aplicação de *software*, parte da aplicação original SGO, que gere a informação da base de dados.	ANA

Sistema	Descrição	Empresa de desenvolvimento
Pro-Watch®	Plataforma de controlo de acessos que controla a abertura de portas através de identificação por cartão RFID.	*Honeywell*
GIC – R	Gestão de Informação em Credenciais: aplicação de *software* usada na criação do cartão RFID, nomeadamente na aquisição de fotografia, impressão de cartão e operações de registo na base de dados do *Pro-Watch* e do SGO.	Logica
GIC – C	Gestão de Informação em Credenciais - Controlo: aplicação de *software* usada na verificação de credenciais nas portarias eletrónicas.	Logica

O passar do tempo conduziu: à desatualização das tecnologias em uso, à falta de suporte de algumas soluções implementadas e à necessidade de inclusão de novas funcionalidades, levantando o cenário de repensar toda a implementação no sentido de se criar uma solução integrada que inclua as valências de existentes, as novas funcionalidades pretendidas e cuja implementação faça a migração tecnológica para ferramentas atuais. Na migração para o novo sistema pretende-se que a ANA seja detentora do conhecimento, capacidade de suporte e evolução da solução. Desta corrente de pensamento surgiu a necessidade que o presente trabalho pretende satisfazer.

1.4. OBJETIVO

Do explanado no subcapítulo 1.3 - Âmbito, o objetivo deste trabalho é o desenvolvimento de uma plataforma de *software* que integre todas as funcionalidades de administração, gestão, configuração e auditoria, inseridas no âmbito de rastreio e controlo de acesso de pessoas e viaturas para o Aeroporto Francisco Sá Carneiro.

Relativamente aos sistemas existentes, a solução pretendida por este projeto deve:

- Usar o portal corporativo para entrada de informação, sistema padrão a toda a empresa.
- Usar a solução *Pro-Watch* para controlo automático de abertura de portas.

- Substituir os módulos: SGO, GIC-R e GIC-C
- Integrar os diversos procedimentos atualmente efetuados em formato de papel ou via *e-mail*.

1.5. ORGANIZAÇÃO DO RELATÓRIO

Este documento é dividido em capítulos que abordam as três grandes vertentes do trabalho: o estado da arte de controlo de acesos, o projeto da solução e o desenvolvimento de protótipo funcional.

O presente capítulo faz o enquadramento de uma necessidade concreta, relativa numa situação real que conduz à definição do objetivo do trabalho.

No capítulo 2 é apresentado o estado da arte do âmbito "controlo de acessos". São descritas as tecnologias usadas: arquiteturas de sistemas, tipos de dispositivos, protocolos de comunicação, etc.

O capítulo 3 é o capítulo de projeto onde são apresentadas as definições, as escolhas efetuadas e os vários modelos para implementação da solução.

No capítulo 4 apresenta a implementação do protótipo do projeto e descrevem-se detalhes específicos de como as soluções foram construídas.

Por fim, no capítulo 5 são apresentadas conclusões sobre o trabalho efetuado e são lançadas pistas para desenvolvimentos futuros.

No final do documento são apresentados diversos anexos com informação complementar relativa aos vários capítulos.

1.6. FASES DO TRABALHO

O desenvolvimento e implementação do projeto contempla várias fases, Figura 13. A primeira é constituída pela compilação das funcionalidades existentes nos diversos sistemas, pela compilação dos processos ligados ao âmbito que decorrem em forma de papel ou *e-mail* e pelo de levantamento da novas funcionalidades pretendidas pelos administradores do sistema atual.

A fase seguinte prende-se com o processo de projeto da solução onde é necessário selecionar as ferramentas de trabalho, construir os modelos de armazenamento de dados e os modelos de implementação das aplicações para execução das funcionalidades, assim como definir as soluções de *hardware* que forem necessárias.

A terceira fase refere-se à implementação do projeto desenvolvido num protótipo funcional.

O trabalho termina com a execução de testes do protótipo e avaliação da solução obtida.

Figura 13 – Fases do projeto.

2. ESTADO DA ARTE

Este capítulo apresenta o estado da arte do tema "Controlo de acesos" iniciando-se com uma descrição dos conceitos gerais e posteriormente desenvolve-se focando os aspetos mais relevantes dos sistemas de controlo de acessos físicos, com destaque para a descrição de arquiteturas de sistemas, tecnologias e metodologia envolvidas e apresentando soluções de mercado.

2.1. CONCEITOS TEÓRICOS SOBRE CONTROLO DE ACESSOS

O instinto de proteção da propriedade é uma característica intrínseca ao ser humano e faz parte das suas preocupações do dia-a-dia. Para satisfazer esta necessidade desde sempre foram desenvolvidas formas para tentar garantir que o acesso, o uso ou o conhecimento de um recurso privado fosse restringido e acessível, apenas a um grupo selecionado de indivíduos. Com o desenvolvimento tecnológico, além do controlo de acesso a recursos físicos como edifícios, equipamentos ou locais, levantou-se também a questão do acesso a

recursos lógicos, normalmente em formato digital, como contas bancárias, correio eletrónico, servidores de informação, etc.

A eficiência dos mecanismos de proteção e segurança assentam em três pilares fundamentais: as pessoas, as tecnologias e os processos. O equilíbrio entre estes três fatores é essencial para garantir os níveis de proteção pretendidos. Depois de implementadas as soluções tecnológicas, a definição de procedimentos, a formação, e a sensibilização, são vitais para que as ações humanas não conduzam a fatores de risco e vulnerabilidades nas soluções como um todo.

A análise das soluções de proteção começa sempre, por um lado, com a consideração dos riscos previstos e por outro, com a consideração dos níveis de segurança pretendidos. A norma europeia EN50131 define quatro graus de risco em que os espaços físicos podem ser classificados:

- Baixo risco: quando se considera pouco provável a tentativa de acessos não autorizados e quando acontecem são frequentemente efetuados pela violação de portas ou janelas.

- Risco baixo a médio: este tipo de risco caracteriza a maioria dos espaços residenciais ou comerciais de baixo valor. Nestes casos as intrusões são normalmente efetuadas por pontos de acesso desprotegidos.

- Risco médio a elevado: nesta categoria estão classificadas a maioria das instalações industriais e comerciais. Nestes casos as intrusões são efetuadas por indivíduos com experiencia e com conhecimentos em sistemas de controlo de acessos e deteção de intrusão.

- Risco elevado: esta categoria caracteriza as instalações com conteúdos de elevado valor. As intrusões espectáveis nestes locais são efetuadas por equipas de indivíduos treinados, com elevados conhecimentos multidisciplinares e fortemente motivados.

Por definição os sistemas de controlo de acessos[2] são ferramentas de implementação eletrónica e/ou informática que usando mecanismos de identificação e autenticação de utilizadores, permitem ou inibem o acesso a recursos de uso restrito.

Num ambiente empresarial os sistemas de controlo de acessos estão envolvidos em todos os níveis de propriedade [6]:

- Instalações - o controlo de acessos controla as entradas e os movimentos nos locais físicos protegendo as pessoas, os equipamentos e a informação contida nos espaços;
- Sistemas de suporte - o acesso as instalações de suporte técnico, como locais de gestão de energia, locais de armazenamento de água, de produção de ar-condicionado, sistemas de combate contra incêndios, etc;
- Sistemas de informação - as várias camadas existentes nos sistemas de informação têm de ser protegidas por questões de confidencialidade, criticidade e operacionalidade dos sistemas;
- Pessoas - a gestão do acesso e movimentação das pessoas dentro da organização devem garantir que apenas as pessoas com permissões acedem aos locais definidos.

As necessidades de controlo de acesso consideradas de forma genérica são inúmeras e dependentes de múltiplos fatores como valor dos recursos a proteger, nível de ameaça, gravidade das perdas causadas por intrusões, etc. Na satisfação das múltiplas necessidades de controlo de acesso podemos categorizar os mecanismos de controlo nos seguintes tipos:

- Diretivos: constituídos por normas e regras que se definem como guia de comportamento esperado. Este mecanismo é implementado normalmente com a instituição de políticas e procedimentos, são exemplo do controlo de acesso

[2] Na literatura inglesa os sistemas de controlo de acesso físicos são frequentemente referidos como PACS – *Physical Access Control System*.

diretivos as instruções que proíbem pessoas do departamento administrativo de acederem às zonas fabris.

- Dissuasores: são mecanismos que pelo simples facto de existirem previnem situações indesejadas, por exemplo o facto de estar uma camara de vigilância num corredor onde não é suposto uma pessoa passar, pode ser motivo suficiente para que essa pessoa não tente passar nesse local.

- Preventivos: estes sistemas garantem o controlo de acessos por imposições que não são facilmente transpostas, por exemplo uma porta fechada ou pontos de controlo e inspeção.

- Detetores: estes sistemas geram notificações de presença quando os sistemas preventivos são violados.

- Corretivos: são sistemas usados quando outros falham, por exemplo um sistema que deteta que uma unidade de controlo de portas entrou em colapso, bloqueia as portas com recursos próprios.

Do ponto de vista de administração, a definição da estratégia de controlo de acessos decorre da análise do problema, abordando as seguintes questões [6], [12].

- Que recursos são controlados – a definição dos recursos a ser controlados é fundamental para a definição da estratégia de controlo de acessos.

- Como é que se rastreia os eventos da solução – qualquer sistema de segurança têm de garantir um elevado grau de auditabilidade sobre os eventos que gere para que o seu funcionamento possa ser analisado e validado. Esta funcionalidade permite atribuir responsabilidades aos utilizadores não só dos espaços físicos mas também da equipa de gestão do próprio sistema.

- Que utilizadores são autorizados – a seleção de utilizadores com permissões de acesso é feita considerando as necessidades que cada utilizador tem para desempenhar as suas atividades e considerando o grau de confiança que se deposita em cada utilizador.

- Especificação de uso – nesta vertente definem-se que utilizadores têm permissões de acesso a que recursos e que operações os operadores podem efetuar sobre esses recursos, por exemplo um utilizador pode ter permissões de uso de um elevador mas apenas com acesso a determinados pisos, ou ter acesso a uma área apenas durante um período de tempo do dia. A especificação de uso deve ser gerida pelos seguintes princípios, [6]:

    - Princípio do menor privilégio, este princípio define o conjunto mínimo de privilégios que um utilizador dever ter para cumprir as suas funções. Apenas o grupo mínimo de privilégios deve ser atribuído ao utilizador e não mais.

    - Princípio da compartimentalização, este conceito define que os recursos ou as zonas de acesso restrito devem ser compartimentadas para que são haja fluxos entre zonas com diferentes características. Por exemplo, numa instalação de processamento de pedras preciosas, as zonas físicas de receção, de manufatura e de armazenamento são perfeitamente compartimentadas e não há passagem de pessoas de um local para outro. No entanto, é necessário analisar os espaços porque pode acontecer a existência de espaços dentro de outros espaços e o facto de se conceder permissões de acesso a uns espaços implicitamente pode-se estar a permitir o acesso a outros.

    - Princípio de herança reversa, este princípio indica que para atribuir o acesso a determinado espaço, é necessário garantir que o utilizador tem acesso a pelo menos um conjunto de espaços que conduzem ao primeiro, esta questão impacta diretamente com o princípio da compartimentação e pode ser um problema na disposição física dos espaços.

    - Princípio do domínio, o domínio é uma noção conceptual que agrega várias compartimentações e onde se desenrola um conjunto muito definido de processos. Por exemplo a oficina de trabalho em pedras preciosas referido anteriormente é um domínio de segurança, completamente independente do domínio onde se desenrolam os processos comerciais.

A implementação de um sistema de controlo de acessos requere um estudo detalhado dos objetivos que se esperam alcançar e de um planeamento cuidadoso para que o sistema seja integrado na vertente operacional da instalação. As soluções a implementar devem ser dimensionadas considerando diversos fatores, [6]:

- O valor dos bens a ser protegidos. Bens de maior valor justificam soluções mais complexas e mais dispendiosas.

- O nível de ameaça. Dependendo de situação há um conjunto de ameaças reais ou espectáveis que justificam o uso de determinada solução.

- O potencial das medidas de segurança. Definido o nível de ameaça, pode-se identificar medidas de segurança que diminuem o risco em determinado sentido, deve-se definir a melhor seleção de combinação de medidas de forma a solução final conduza à maior redução possível de riscos das ameaças.

- O custo das medidas de segurança. Tem de existir um balanceamento entre o valor do bem, os níveis de ameaça, o custo da implementação dos sistemas segurança e o custo da operacionalidade desses sistemas na vertente humana e tecnológica.

- Impacto na atividade corrente. A solução selecionada deve ter o menor impacto possível na operacionalidade dos espaços. A potencial quebra no fluxo provocada pelos sistemas pode ser um impedimento para o uso de uma solução específica.

Em instalações de grandes dimensões, acedidas por um número elevado de pessoas, os sistemas de controlo de acessos desempenham um papel fundamental, por um lado para garantir a segurança dos espaços e por outro para simplificar os procedimentos de gestão de acessos. Os sistemas de controlo de acessos atualmente disponíveis no mercado são ferramentas multifuncionais, com grande capacidade de configuração e adaptação às mais variadas exigências permitindo:

- Facilidade de uso, normalmente o utilizadores apenas tem de ser detentores de mecanismos de identificação para aceder a vários locais, ao contrário por exemplo das chaves mecânicas em que se usa uma para cada porta, permitindo ao utilizadores desempenharem as suas atividades com o mínimo de impacto.

- Aumento do nível de segurança, tornando o acesso independente de erros humanos e impondo uma maior dificuldade de reprodução de identificações ou de usos indevidos.
- Facilidade de gestão de acessos permitindo incluir ou retirar permissões sem impacto de carga de trabalho significativa.
- Capacidade de gestão de perfis de acesso para múltiplos utilizadores.
- Possibilidade de apenas conceder acessos para determinados períodos do dia.
- Rastreabilidade do histórico de passagem pelos locais de controlo e criação de relatórios estatísticos sobre os eventos do sistema.
- Integridade, garantindo que a informação associada ao sistema não possa ser alterada de forma não autorizada.
- Escalabilidade das soluções com reaproveitamento dos equipamentos instalados, mesmo quando se pretende níveis de segurança mais elevados.
- Integração com outros sistemas, como deteção de intrusão e videovigilância, etc.

2.2. IDENTIFICAÇÃO E AUTENTICAÇÃO DE PESSOAS

O sistema de controlo de acesso perante uma pessoa que se identifica para aceder a um recurso, autentica a sua identificação e analisa as permissões de acesso da pessoa ao recurso e no caso de a pessoa ter permissões executa o procedimento de autorização de acesso. Genericamente, os sistemas de controlo de acessos usam quatro mecanismos no seu funcionamento: identificação, autenticação[3], autorização e registo [6], [11].

- Identificação – a identificação [8], é o processo de adquirir a identidade de uma pessoa que se apresenta para aceder a um recurso. O processo de identificação procura a identidade de uma pessoa desconhecida dando resposta à pergunta:

[3] Na literatura inglesa os mecanismos de autenticação são frequentemente referidos como PIV - *Personal Identity Verification*.

"Quem é esta pessoa?". Um exemplo de identificação de uma pessoa é a leitura do seu nome num documento de identificação.

- Autenticação – a autenticação [8], é o mecanismo através do qual a pessoa prova a sua identidade, neste processo a pessoa apresenta-se e o sistema vai verificar a validade da informação respondendo à pergunta: "Esta pessoa é quem afirma ser?" o objetivo deste processo é garantir, sob determinadas condições, a autenticidade da identificação. Por exemplo a foto apresentada num passaporte comparada com a face do portador é um processo que permite autenticar a identificação da pessoa.

- Autorização – A autorização [8], é o procedimento que verifica as permissões de um utilizador e que concede, ou não, o acesso da pessoa identificada e autenticada ao recurso restrito.

- Registo, Rastreabilidade – Os sistemas de controlo de acessos como subgrupo de sistemas de segurança, para finalidades de auditoria, análise ou inspeções, têm implementadas funcionalidades de registo dos acontecimentos que ocorrem no sistema ao longo do tempo. Os registos devem versar sobre todos os eventos relativos às ações dos utilizadores, ações das aplicações, ações dos sistemas e ações das redes de comunicação. Estes registos podem ser filtrados e agregados para rastrear os diversos aspetos de funcionamento do sistema quer do ponto de vista técnico quer do ponto de vista operacional.

Normalmente os processos de identificação e de autenticação, são usados em conjunto. Comparando o processo de identificação como o de autenticação [9], verifica-se que a identificação é um processo mais pesado computacionalmente porque tem de comparar as características da identificação apresentadas com todas a identificações registadas no sistema até encontrar uma semelhança, por exemplo num sistema de abertura de portas por impressão digital, a pessoa identifica-se com o dedo no leitor e o sistema compara a impressão digital obtida com as impressões digitais com permissões de acesso a essa porta, no caso de haver uma coincidência a porta é desbloqueada. No processo de autenticação conhece-se a identificação da pessoa e o sistema apenas tem de consultar o registo dessa pessoa e verificar se a informação de autenticação apresentada é semelhante à informação registada, por exemplo num sistema de autenticação de uma conta de correio eletrónico o

utilizador identifica-se com a indicação de *user* e o sistema no processo de autenticação verifica se a *password* apresentada corresponde à *password* guardada, apenas, no registo desse utilizador.

2.2.1. MECANISMOS DE IDENTIFICAÇÃO

A forma de identificação está intimamente ligada à solução de controlo de acessos implementada, o método mais comum para acessos físicos é o uso de um cartão de identificação, normalmente este cartão possui o nome e a fotografia do portador que são apresentados num suporte e configuração que são reconhecidos oficialmente nos pontos de controlo e que atribui um conjunto de privilégios ao portador. Nas instalações de maiores dimensões pode haver mais de um tipo de cartões, por exemplo, uns que são usados pelos colaboradores e outros de carater mais temporário para serem atribuídos a visitas.

Em soluções com caracter mais tecnológico o cartão de identificação tem embebidos meios eletrónicos de armazenamento de informação e/ou de processamento para conter mecanismos de identificação e validação de autenticação. Na última década, tem-se assistido ao desenvolvimento massivo da tecnologia de identificação/autenticação denominada RFID - *Radio Frequency Identification*, que consiste numa tecnologia que usa campos eletromagnéticos como meio de transmissão de informação.

Outra forma de identificação, esta mais usada para acessos lógicos e acessos a sistemas, é o número de conta ou o PIN - *Personal Identification Number*. Ainda dentro dos acessos lógicos por vezes a identificação é efetuada usando o endereço de correio eletrónico, o endereço MAC - *Media Access Control* ou o endereço IP- *Internet Protocol*.

Em qualquer dos casos os mecanismos de identificação devem possuir as seguintes características [6]:

- Ser único: a identificação dever ser única para destituir inequivocamente o utilizador.

- Ser não descritivo: a identificação deve revelar o menos possível sobre a pessoa, e nunca deve apresentar a função, as tarefas que o utilizador executa e muito menos deve apresentar os privilégios de acesso.

- Ser de emissão segura: o processo de criação da identificação deve garantir os níveis de segurança necessários ao fim a que se destina para minimizar o risco de criação de cópias de identificações.

2.2.2. MECANISMOS DE AUTENTICAÇÃO

Quando às formas de autenticação de utilizadores, os sistemas de controlo de acessos são classificados em três tipos, também denominados fatores de autenticação [6], [10]:

- Fator baseado no conhecimento, que é algo que o utilizador sabe, por exemplo uma palavra-chave ou um código.

- Fator baseado na posse, que é algo que o utilizador tem, por exemplo um cartão de acesso.

- Fator biométrico, que é uma característica intrínseca do utilizador.

A autenticação da identidade usando um fator denomina-se autenticação de fator-único. A autenticação pelo conhecimento baseia-se em algo que o utilizador sabe, enquadram-se nesta categoria por ordem crescente de garantia de nível de segurança:

- Códigos de acesso, tipicamente conjuntos de caracteres numéricos.

- Palavras-chave simples, normalmente combinação de letras.

- Padrões gráficos.

- Palavras-chave complexas, combinação de letras, números e símbolos que podem constituir frases completas para se tornarem mais fáceis de recordar.

O maior problema deste método de autenticação é garantir a confidencialidade do código. Por um lado, tecnicamente os códigos nunca devem ser transmitidos nas redes de comunicação sem serem encriptados, por exemplo o FTP - *File Transmition Protocol* faz a transmissão de dados em texto simples e apesar de ser um protocolo popular na transmissão de dados não deve ser usado quando a informação contem códigos de acesso. Por outro, lado sob a perspetiva humana, os códigos devem ser decorados e se forem escritos devem ser guardados em locais seguros. Ainda sob a perspetiva humana a necessidade de criar códigos não óbvios leva à criação com conjuntos de carateres difíceis de memorizar que levantam, também, problemas na apresentação do código.

A autenticação baseada na posse usa um dispositivo físico externo para garantir que a pessoa é quem diz ser. Exemplos de dispositivos de autenticação:

- Cartão de coordenadas, quando um utilizador se identifica, o sistema solicita-lhe a introdução de alguns dados desse cartão que só o utilizador possui e em cada autenticação pede dados diferentes.

- Dispositivos eletrónicos que aceitam perguntas efetuadas pelo sistema de controlo de acessos e geram respostas que garante a autenticação.

- Dispositivos de memória ou de processamento que contem informação ou algoritmos de autenticação.

Um dos maiores problemas deste tipo de fator de autenticação prende-se com o extravio do dispositivo que garante a própria autenticação e com a possibilidade do dispositivo ser clonado.

A autenticação baseada no ser funciona usando uma ou mais características físicas do portador para o autenticar, como por exemplo:

- Características físicas: Impressão digital, reconhecimento da face, reconhecimento da íris, reconhecimento do padrão de veias, ADN, etc.

- Características fisiológicas: padrão das tensões cardíacas, ritmo de oscilação da pupila, etc.

- Características comportamentais: forma de andar, forma de escrever, reconhecimento da voz, reconhecimento da digitação, etc.

As maiores vantagens deste método são o elevado grau de unicidade das características, a dificuldade de duplicação e a possibilidade de criar uma autenticação em que o nível de sofisticação seja compatível com o nível de segurança pretendido. Os maiores problemas residem na implementação de sistemas de leitura das características que são comparativamente mais complexos quer a nível de aquisição de dados quer a nível de processamento e armazenamento de informação e são consequentemente mais dispendiosos.

Relativamente às soluções biométricas, surgem por vezes problemas na aceitação da tecnologia por parte dos utilizadores devido essencialmente a receios de integridade física e por dúvidas sobre a violação de privacidade, normalmente mais agravadas quando estão envolvidas relações laborais. Sobre estas questões, em Portugal, a Constituição da República, no artigo 35º define os conceitos fundamentais da utilização de dados informáticos. O Decreto de Lei 67/98 aborda a temática do tratamento e circulação de dados. E a Comissão Nacional de Protecção de Dados, entidade que regula os sistemas de aquisição, tratamento e armazenamento de dados pessoais, define um conjunto de regras relacionados com o assunto, nomeadamente:

- "Princípio sobre a utilização de dados biométricos no âmbito do controlo de acessos e de assiduidade".

- Deliberação nº 1638/2013 "Aplicável aos tratamentos de dados pessoais decorrentes do controlo da utilização para fins privados das tecnologias de informação e comunicação no contexto laboral".

Numa tentativa de mitigar as possíveis vulnerabilidades em cada um dos tipos de autenticação, em instalações com níveis se segurança mais elevados usa-se [13] uma conjugação de dois ou mais fatores de autenticação que tem de ser verificados

simultaneamente, tipicamente um cartão de acesso e uma característica biométrica, ou um cartão, um código e uma característica biométrica, Figura 14. Esta metodologia denomina-se autenticação multi-factor[4]. Nestas aplicações há uma maior garantia que quem se identifica é a pessoa a quem foram atribuídas credenciais. Por exemplo no *data center* que a Portugal Telecom instalou na Covilhã o controlo de acessos é feito por reconhecimento do padrão das veias da mão e pela verificação do peso da pessoa [15] .

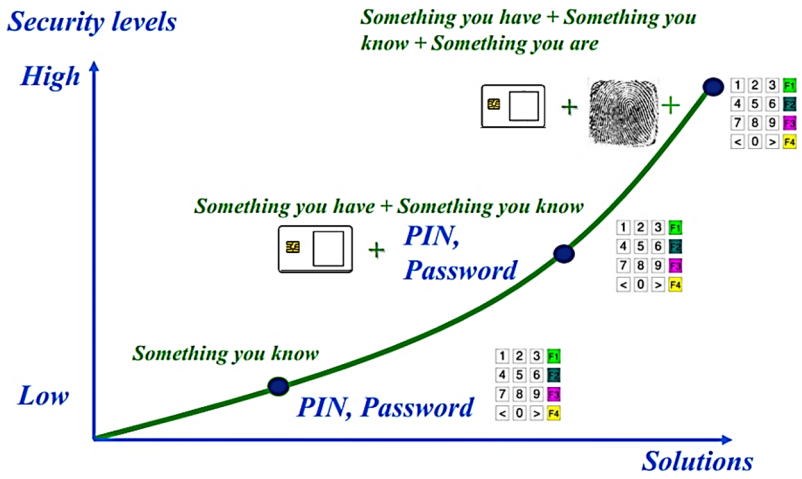


Figura 14 – Comparação dos níveis de segurança na autenticação multi-factor, [13].

Recentemente [6] tem surgido soluções que usam a localização como um quarto fator de autenticação. Estas soluções baseiam-se em referenciação GPS ou localização geográfica da rede de comunicações de origem, para detetar por exemplo, intrusões usando credenciais aparentemente válidas mas despoletadas de locais fora da área geográfica esperada.

Sendo os sistemas de controlo de acessos implementações efetuadas para garantir níveis de segurança, a escolha da solução a implementar para um projeto concreto depende do nível

[4] Na literatura inglesa os mecanismos de autenticação multi-factor são frequentemente referidos como MFA - *Multifactor Authentication*, em constraste com SFA - *Single-factor authentication*, ou 2FA - *Two Factor Authentication.*

de proteção pretendida, mas nas soluções devem ser considerado que todos os elementos que constituem o sistema tem de apresentar o nível mínimo de garantia que suporta os requisitos do sistema como um todo. Assim deve-se considerar:

- Segurança nos cartões como elemento de identificação: existem soluções de impressão do cartão que evitam a contrafação como impressão de hologramas, impressão de imagens com tintas visíveis apenas quando expostas a radiação ultravioleta, impressão de micro-gravações, uso de imagens tridimensionais, etc.

- Segurança nos cartões como elemento de autenticação: usando cartões com armazenamento de dados encriptados ou com capacidade de processamento para fazer a autenticação com mecanismos de emparelhamento de chaves ou cálculos algorítmicos.

- Proteção de dados: deve-se considerar a proteção da informação que circula em todos os troços do sistema sejam entre os cartões e os leitores de cartões, entre os leitores de cartões e as unidades de controlo ou entre a unidades e os servidores. Deve-se ter em conta que cada canal deve apenas transmitir informação codificada e que os meios de transmissão devem ser o mais controlados e de difícil acesso e/ou interceção possível.

2.3. ARQUITETURA DE SISTEMAS DE CONTROLO DE ACESSOS

O trabalho a desenvolver no âmbito desta tese versa sobre sistemas de controlo de acesso físico como subgrupo dos sistemas de controlo de acesso que também incluem os sistemas de controlo de acesso lógicos. A partir deste parágrafo, por simplicidade de terminologia, as referencias a "sistemas de controlo de acesso" referem-se, salvo indicações em contrário, apenas a sistemas de controlo de acesos físicos. Neste subcapítulo apresentam-se os blocos de componentes e dispositivos usados na implementação de sistemas de controlo de acessos físicos e exemplos concretos de sistemas disponíveis no mercado.

A arquitetura dos sistemas de controlo de acessos é composta por três grandes grupos de equipamento: pontos de controlo, sistemas informáticos de serviços e de armazenamento de informação e sistemas de configuração e operação, como mostrado na Figura 15.

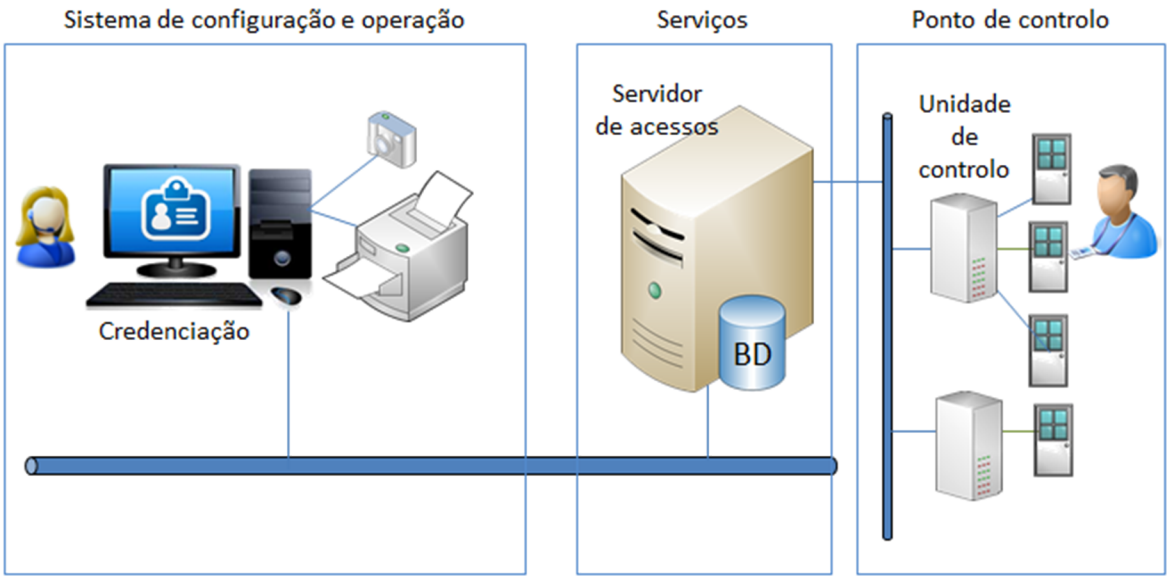


Figura 15 – Arquitetura geral dos sistemas de controlo de acessos

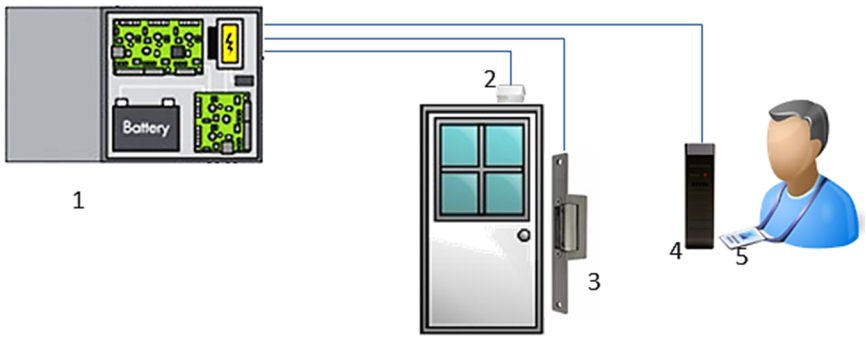
Os sistemas de serviços e de armazenamento de informação, são aplicações informáticas que são executadas num ou mais servidores e que disponibilizam todos os serviços funcionais do sistema de controlo de acessos e fazem a gestão e armazenamento da informação do sistema disponibilizando mecanismos de salvaguarda, consistência e restrição de dados.

Os sistemas de configuração e operação são aplicações informáticas, normalmente aplicações cliente das aplicações que são executadas nos servidores, onde existem vários perfis de utilizador que efetuam as tarefas de configuração e operação do sistema, nomeadamente:

- Ao nível de configuração
    - Configuração de zonas e áreas restritas.
    - Configuração de perfis de abertura de portas.
    - Configuração do equipamento de campo.
    - Configuração dos meios de credenciação.
- A nível de operação
    - Registo de dados de pessoas.
    - Aquisição de fotografia.
    - Aquisição de dados biométricos.
    - Atribuição cartões às pessoas.
    - Atribuição de permissões às pessoas/cartões.
    - Impressão de cartões.
    - Procura e relacionamento de dados de eventos gerados pelo sistema.
    - Consultas e cruzamento de informação estatística.

Os pontos de controlo são os locais fronteira entre áreas de níveis de acesso diferente e que são usados para restringir o acesso às zonas mais restritas. Os pontos de controlo, genéricos em edifícios, Figura 16, são constituídos por vários tipos de equipamento, normalmente por:

- Obstáculos físicos que impedem o acesso às áreas reservadas, por exemplo: portas, torniquetes ou barreiras. Neste documento será usado o termo "porta" para referir este tipo de equipamento.
- Equipamentos de autenticação da identificação dos utilizadores, podendo ser teclados para digitar códigos, leitores de cartões, leitores biométricos ou uma combinação destes.
- Mecanismos de abertura dos obstáculos físicos, como trincos elétricos, retentores eletromagnéticos, motores, etc.
- Equipamentos para verificação do estado de abertura do obstáculo físico como contactos magnéticos, botões de pressão ou detetores de posição.
- Unidades de operação dos equipamentos que constituem o ponto de controlo.



1- Unidade de controlo de *n* portas; 2 – Sistema de verificação da posição da porta;
3- Testa elétrica que permite a abertura da porta por ação da unidade de controlo;
4 – Equipamento de leitura de credenciais; 5 – Credencial

Figura 16 – Ponto de controlo de um sistema de controlo de acessos genérico.

Do ponto de vista de tomada de decisão de autorização de acesso, os sistemas podem ser do tipo centralizado ou distribuído, [14].

Nos sistemas centralizados as unidades de controlo enviam para um servidor a informação das credenciais que lhe foram apresentadas, o servidor compara a informação recebida com a que tem armazenada na sua base de dados, toma uma decisão sobre o acesso e transmite instruções à unidade de controlo para execução de abertura de portas ou para as manter bloqueadas.

Nos sistemas distribuídos o servidor central a cada alteração de permissões de acesso efetuada pelas aplicações de gestão, envia para as unidades de controlo a informação que lhes permite decidir sobre a permissão ou a negação de acessos das credencias e a decisão

é efetuada nas unidades de controlo. Este tipo de solução em instalações de grandes dimensões aumenta o desempenho a nível de tempos de resposta e diminui significativamente o trafego de dados. Nestes sistemas, para efeitos rastreabilidade de eventos, as unidades de controlo transmitem periodicamente para os servidores as informações sobre a sua atividade.

Nas unidades de controlo a funcionar em modo distribuído, ligadas a sistemas centrais, no caso de falhas de comunicação podem funcionar de uma das seguintes formas:

- Continuam a validar as credencias em modo autónomo, podendo neste período de tempo estarem a efetuar decisões baseadas em listas desatualizadas.

- Colocar todas as posições em modo de bloqueio até as comunicações serem restabelecidas. Esta situação tem um maior impacto na operacionalidade dos espaços mas é a metodologia usada em instalações que exigem maiores níveis de segurança

2.3.1. UNIDADES DE CONTROLO DE ACESSO

As unidades de controlo de acesso são equipamentos eletrónicos que gerem todos os dispositivos associados a postos de controlo, normalmente o leitores de cartões, teclados de introdução de código ou leitores biométricos, trincos ou retentores de portas e sensores de deteção do estado de abertura da portas, como mostrado na Figura 16. Tipicamente as unidades de controlo de acesso disponibilizadas no mercado têm capacidade para gestão de duas, quatro ou oito portas.

Genericamente os sistemas de controlo de acesso funcionam da seguinte forma: num posto de controlo, por exemplo numa porta controlada, o utilizador apresenta o seu meio de identificação e autenticação ao dispositivo de leitura. O dispositivo obtém as credenciais do utilizador e remete-as a uma unidade de controlo de acesso. Esta unidade verifica a identificação, autentica-a e consulta a sua lista de credenciais com as permissões de acesso ao local protegido pelo posto de controlo. Se as credenciais apresentadas, tem acesso ao

local, a unidade de controlo instrui o mecanismo de bloqueio da porta para a abrir, caso contrário mantem-na fechada. Normalmente o sistema informa o utilizador com um sinal sonoro e/ou luminoso se a credencial foi aceite ou rejeitada e sem tem ou não permissões de passagem.

As unidades de controlo podem funcionar autonomamente, sendo programadas no local ou podem estar inseridas em sistemas computacionais de controlo, ligadas a servidores a funcionar em modo centralizado ou distribuído.

As operações de programação das unidades de controlo fazem a configuração dos dispositivos que lhe estão ligados fisicamente e nas unidades autónomas introduzem as credenciais de acesso dos utilizadores autenticados para passagem.

As ligações de dados entre as unidades de controlo e os servidores centrais nos sistemas mais antigos são implementadas em topologias estrela ou barramento RS-485 mas tem-se assistido à evolução para comunicações usando redes com protocolo IP. O sistema *SiPass* da *Siemens* [16] é um exemplo de implementação em barramento RS-485 e as primeiras gerações do sistema *NexWatch-Star* é um exemplo de implementação de estrela RS-485, [17]. Neste momento, as duas marcas, também apresentam soluções baseadas em redes IP.

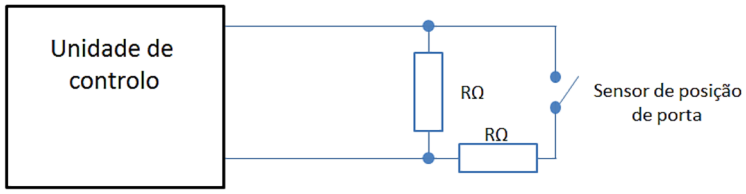


Figura 17 – Sensor de vigia de portas.

Em cada momento a posição da porta, fechada/aberta, é analisada pela unidade de controlo através do respetivo equipamento de monitorização: contacto magnético, botão de posição, etc. Esta monitorização é efetuada normalmente por leitura de resistência elétrica como mostrado na Figura 17, em estado normal - porta fechada, a unidade de controlo lê RΩ, nos terminais do cabo do sensor de posição da porta, quando a porta abre o sensor de posição da porta, fecha o contacto e a unidade de controlo lê $\frac{R}{2}\Omega$, ficando com o conhecimento que a porta está aberta, se a instalação do sensor for alterada como por exemplo o cabo for

cortado ou curto-circuitado, ou o sensor avariar, a unidade de controlo apercebe-se de uma alteração da instalação de vigia da porta e pode gerar alarmes a reportar o facto.

Quando a porta está num estado diferente do esperado por períodos de tempo superiores aos definidos, a unidade de controlo gera um alarme de porta aberta ou entrada forçada, que se pode traduzir apenas num alarme no sistema de monitorização central mas também pode atuar um sinalizador ótico e/ou acústico na porta ou na geração de um alerta remoto para um local onde se encontra uma equipa de vigilância. Por exemplo, a plataforma *Pro-Watch* [18] da marca HONEYWELL é um exemplo de aplicação onde se pode configurar este tipo de funcionamento.

Do ponto de vista de segurança, as unidades de controlo vigiam do estado das portas dos armários onde estão fisícamente instaladas, para que se forem abertas de forma não autorizada entram em modo de segurança imitindo alarmes para os sistemas centrais e deixam de fazer validação de acesso às portas que controlam.

Do ponto de vista de recursos, as unidades de controlo estão equipadas com baterias e os respetivos carregadores que permitem o funcionamento da unidade mesmo com falta de alimentação elétrica, podendo efetuar nestas condições cerca de mil operações de acesso, [17].

As unidades de controlo de acesso mais evoluídas estão equipadas com conjuntos de pontos de entrada e saída elétricos genéricos que podem ser usados para integração do controlo de acessos com outros sistemas, nomeadamente sistemas de deteção de intrusão ou sistemas de vídeo vigilância - CCTV.

Atualmente tem surgido sistemas de controlo de portas em que as unidades de controlo apenas comandam uma porta, estas unidades são instaladas fisicamente o mais próximo possível da porta. Os dispositivos de leitura, comando e controlo são ligados diretamente à unidade de controlo, Figura 18 como nas unidades que controlam várias portas, mas neste caso, como a unidade fica próximo da porta minimizam-se os custos da instalação de "grandes" troços de cabos.

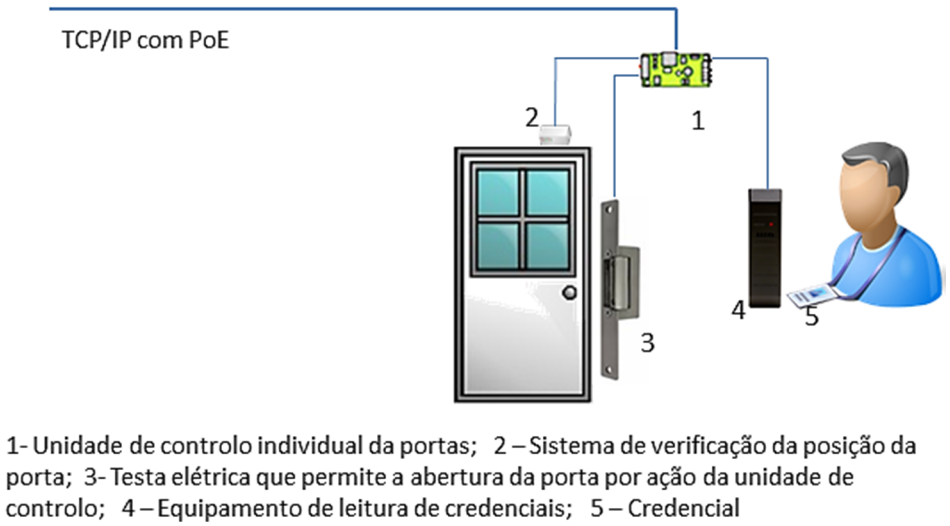


Figura 18 – Unidade de controlo de uma porta individual.

Por sua vez a unidade de controlo liga-se ao sistema central por um pondo de rede TCP/IP com recursos PoE - *Power over Ethernet*[5]. Esta abordagem necessita da instalação do ponto de rede com PoE que à partida é um custo, mas apresenta algumas vantagens: primeiro o só é necessário instalar um cabo, esse cabo permite a comunicação de dados e o transporte da energia, depois a própria unidade de controlo pode ser mais simples, sem circuitos de alimentação nem baterias para o caso de falhas, porque normalmente os sistemas PoE são alimentados por equipamentos suportados em sistemas centrais de energia ininterrupta. Um exemplo deste tipo de soluções é o sistema *FUSION System*, da empresa Borer.

A comunicação entre as unidades de controlo e os leitores de credenciais, no caso de leitores biométricos ou leitores de dispositivos eletrónicos é efetuada usando protocolos

[5] PoE – *Power over Ethernet*- é uma tecnologia definida na norma IEEE802.3af, que permite o transporte de energia elétrica no cabo de dados da rede *Ethernet.* Esta facilidade é usada para alimentação elétrica dos dispositivos apenas com a instalação do cabo de dados

proprietários das respetivas marcas com informação encriptada. No caso da comunicação das unidades de controlo com leitores de cartões normalmente é usado o protocolo *Wiegand*, com dados encriptados ou não, dependendo da aplicação.

2.3.1.1. PROTOCOLO *WIEGAND*

O protocolo *Wiegand* [19] é um protocolo de comunicação serie, assíncrono, que usa três sinais elétricos, um é o sinal de referência e os outros dois são sinais ativos de dados, denominados Wiegand0 e Wiegand1, ou Data0 e Data1.

A transmissão de informação é efetuada em níveis TTL 0-5V e admite distâncias até 150m. Em estado de repouso as duas linhas de dados estão no nível lógico alto, 5V. Quando é transmitido um "0" lógico a linha Data0 é colocada no nível lógico baixo e a linha Data1 mantem-se no nível lógico alto. Quando é transmitido um "1" lógico a linha Data1 é colocada no nível lógico baixo e a linha Data0 mantem-se no nível alto como ilustrado na Figura 19. As duas linhas simultaneamente no nível lógico baixo geram um erro de comunicações que pode ser indicativo por exemplo de cabo cortado.

Temporalmente, os impulsos que representam os caracteres duram 50 microssegundos e o espaçamento entre eles é de 1000 microssegundos [19].

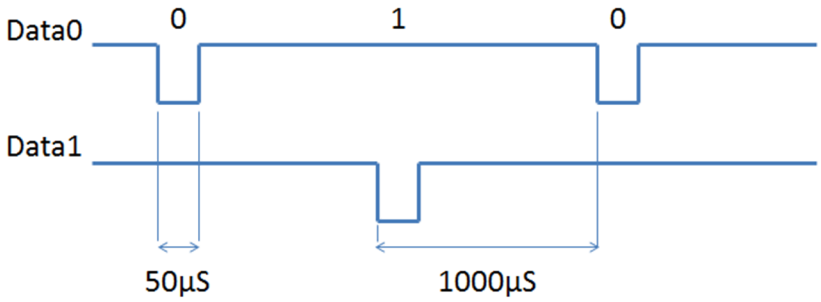


Figura 19 – Diagrama temporal do protocolo *Wiegand*.

Com a denominação "*Wiegand*" existe também uma tecnologia usada em sistemas de controlo de acessos que não é um protocolo de comunicação mas uma norma de domínio publico, de codificação de dados binários. A norma denominada por H10301, também conhecida por "formato *Wiegand* original", [19], que define o formato de codificação de números de série cartões. O formato é constituído por blocos de vinte e seis *bit* com a seguinte descrição e ilustrado na Figura 20:

- A trama é iniciada com um *bit* de paridade[6].
- Os oito *bits* seguintes representam a identificação de local - F.
- Os dezasseis *bits* seguintes representam o código de dispositivo que é o número do cartão - U.
- A trama termina com um stop *bit* de paridade.

O *bit* de início da trama é de paridade par, calculado com base nos primeiros doze *bits* da trama, marcados a azul na Figura 20. O stop *bit* é de paridade ímpar e calculado com base nos últimos doze *bits*, marcados a vermelho na Figura 20.

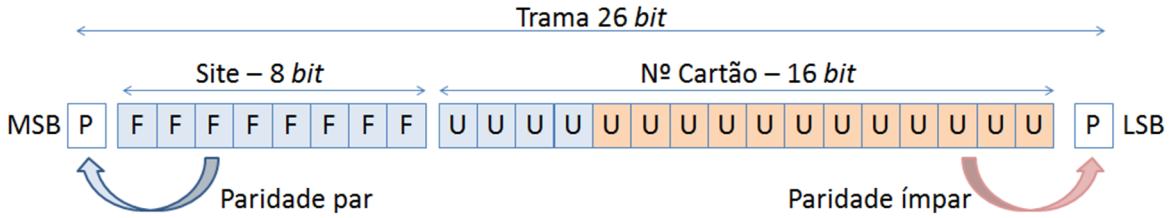


Figura 20 – Formato da trama *Wiegand* H10301.

[6] Os bits de paridade são bits de verificação. Existe paridade par e paridade ímpar.

Um bit de paridade par tem o valor "1" se a contagem do número de bits que se pretende verificar tiver um número ímpar de bits com valor "1".

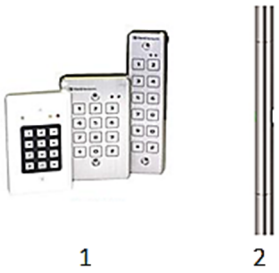
Um bit de paridade ímpar tem o valor "1" se a contagem do número de bits que se pretende verificar tiver um número par de bits com valor "1".

O uso cartões que disponibilizem a informação segundo esta norma tem a vantagem de poderem ser usados pela maior parte das unidades de controlo do mercado. A maior desvantagem prende-se com o facto de o formato permitir apenas 255 códigos de local e 65535 códigos de cartão que a nível mundial é um número pequeno e por isso quando se adquire um cartão com esta codificação não há a garantia que seja único.

Para mitigar este problema os fabricantes de cartões usam variantes, por vezes proprietárias, em que o fabricante garante a unicidade do número do cartão. São exemplo destas variantes, [20], as normas H10302 que usa trinta de sete *bits*, dos quais trinta e cinco codificam o número do cartão ou a norma H10304 também com trinta de sete *bits* mas que usa *16 bits* para código de local e 19 *bits* para número de cartão.

2.3.2. EXEMPLOS DE IMPLEMENTAÇÃO DE SISTEMAS DE CONTROLO DE ACESSOS

A temática dos sistemas de controlo de aceso é muito vasta e encontram-se vários fornecedores com diversos tipos de soluções. Provavelmente os sistemas mais simples são o controlo de aberturas de portas feitas de forma autónoma, constituindo uma topologia composta por dispositivos a funcionar isoladamente, efetuada por teclados ou por leitores de cartões, estes sistemas por não estarem ligados a nenhum sistema centralizado, são programados no local e atuam individualmente. Os modelos da série CL83 da empresa *Infocontrol* são exemplos de teclados de funcionamento autónomo, Figura 21-1. Um exemplo fora do comum de controladores autónomos é o modelo *ekey net finger scanner FSB*, Figura 21-2, da empresa eKey, que é um puxador onde está embebido um *scanner* de impressão digital que identifica o utilizador no ato de abertura da porta.



1 – Teclado de introdução de códigos; 2 – Leitor de impressão digital

Figura 21 – Sistemas de controlo de portas autónomos.

Outro exemplo de sistemas de baixa complexidade são as fechaduras eletrónicas que se usam por exemplo nos quartos de hotéis, estas fechaduras apresentam-se com formatos idênticas as fechaduras mecânicas como mostrado na Figura 22, mas abrem por presença de um cartão de proximidade ou por passagem de um cartão de banda magnética. Nestes sistemas cada fechadura tem um código associado e existe um computador central com uma base de dados de todos os códigos de fechaduras. Através de uma aplicação que programa os cartões de acesso com o código da fechadura respetiva, transforma-se os cartões em chaves dessa porta. As fechaduras são equipadas com sistemas de abertura elétrica, alimentadas por baterias com capacidade de até 150.000 aberturas [21] e permitem funcionalidades como por exemplo criação de chaves mestras para conjuntos de fechaduras [22].

Figura 22 – Fechaduras Eletrónicas, com abertura por cartão de proximidade [23].

No nível seguinte de complexidade encontram-se sistemas que usam as unidades de controlo sem ligação permanente a sistemas de gestão. As unidades de controlo são programadas através de uma aplicação existente num computador que define a lista de cartões que tem permissões para abrir as portas controladas por essas unidades. Nestas

soluções, sempre que há uma alteração na programação é necessário deslocar o computador ao local para descarregar a nova configuração. Nestas soluções as unidades de controlo funcionam de forma autónoma e por isso são usadas apenas em pequenas instalações, visto que também tem um baixo nível de funcionalidades de gestão e de supervisão.

No nível mais complexo de implementação de controlo de acessos, existe um sistema informático central de configuração e gestão que comunica com unidades de controlo, leitores biométricos e outros dispositivos. Esta topologia apesar de ser a mais dispendiosa, é a que disponibiliza mais funcionalidades, por exemplo permitem usar os leitores de abertura de portas para definir rondas de vigilância e são sistemas com maiores possibilidades de escalabilidade.

São exemplo desta arquitetura de unidades de controlo com servidores centrais, os sistemas *Honeywell* que usam a plataforma informática *Pro-Watch* [17], [18], que usam as unidades de controlo *NexWatch-Star* [24] para operar com equipamento de campo e para controlo de portas. A Figura 23 mostra o ecrã de entrada do *software Pro-Watch*. Nesta solução as unidades de controlo podem ser ligadas ao servidor através de redes de comunicação serie baseadas no protocolo RS-485, ou podem ter interfaces para comunicação sobre o protocolo TCP/IP.

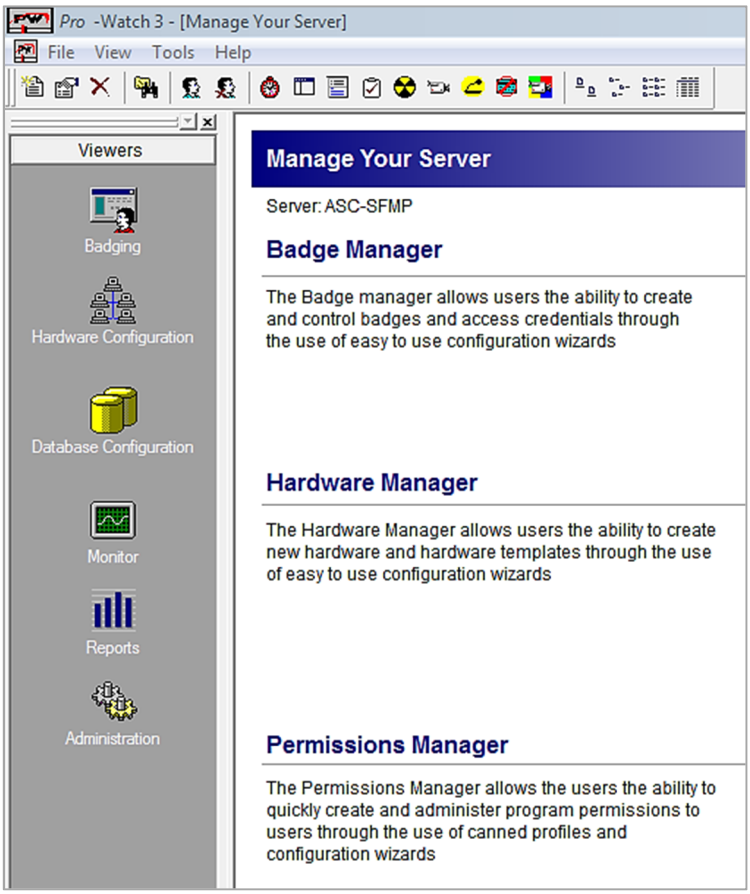


Figura 23 – *Pro-Watch*, ecrã de entrada do *software*.

Outro exemplo deste tipo de implementação é o sistema SiPass da Siemens. Na Figura 24 é apresentada a arquitetura de uma instalação tipo, nesse diagrama verifica-se a existências das unidades de controlo com vários tipos de equipamento de campo desde leitores de cartões, a teclados de introdução de códigos, dispositivos de controlo elevadores, etc, assim como sistemas mais evoluídos suportados em computadores como centrais de gestão de visitas, centrais de gestão de recursos humanos, etc. Do ponto de vista de integração verifica-se a possibilidade de interagir com sistema de videovigilância e sistemas de intrusão.

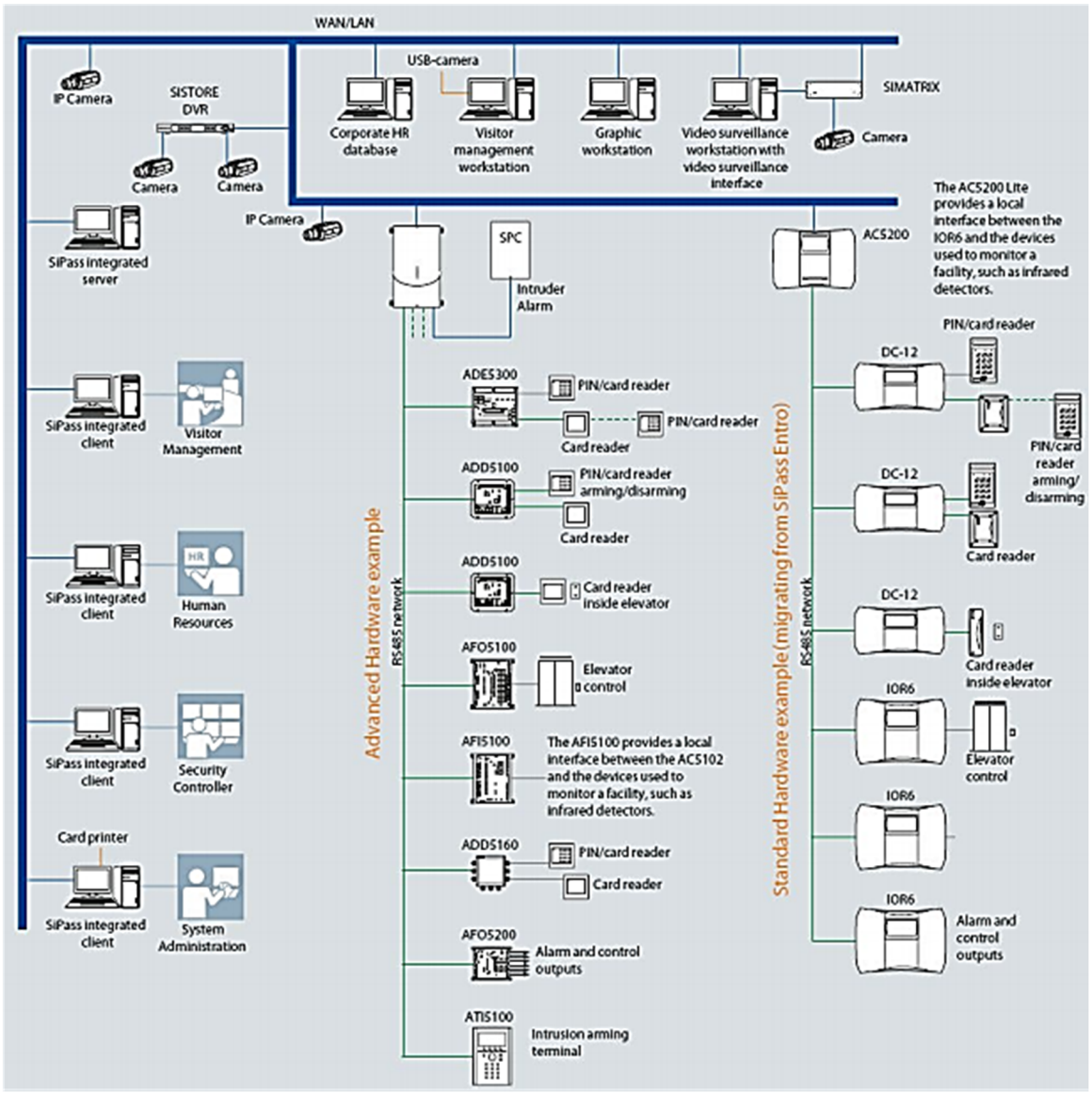


Figura 24 – Diagrama de implementação de controlo de acesso Siemens SiPass.

Relativamente aos equipamentos de campo que se ligam às unidades de controlo como leitores, fechos, sensores, unidades de controlo, são fornecidos por vários fabricantes com formas de proteção de violação, normalmente são apresentados em caixas sem abertura física ou equipados com interruptores, leitores de luminosidade ou medidores de impedância que detetam a abertura do dispositivo e/ou corte de cabos que despoletam alarmes de violação para os sistemas centrais. Por exemplo, o leitor OP10 da Honeywell [25] é fornecido numa caixa vulcanizada sem abertura e disponibiliza um sinal denominado *tamper* que é uma saída de coletor aberto, atuada por um sensor de luminosidade que vigia a proximidade do leitor à superfície onde está instalado.

Ainda dentro do âmbito de exemplos de controlo de acesso além das implementações em edifícios, existem soluções específicas destinadas a segmentos mais especializados, como são:

- Controlos de portagens: sejam eles com pagamentos eletrónicos, quer com uso de *tickets*.
- Controlos de parques de estacionamento: com os diversos mecanismos de controlo e pagamento: *ticket*, cartão cliente, via verde ou reconhecimento de matrícula,
- Controlos de fronteira: com passaportes e autenticação por elemento humano ou com passaportes eletrónicos com reconhecimento biométrico e com passagens automáticas.
- Acessos a transportes: com cartões cliente ou com bilhetes pré-pagos.
- Controlos de embarque em aeroportos: usando cartões de embarque, páginas com códigos impressos ou dispositivos eletrónicos.

Todos estes exemplos têm especificidades concretas para satisfazer as necessidades onde estão integrados, mas todos eles têm em comum o uso de um mecanismo de identificação, um leitor desse mecanismo, um sistema de validação, um obstáculo controlável e um sistema de gestão e configuração.

2.3.3. INTEGRAÇÃO DOS SISTEMAS DE CONTROLO DE ACESSOS COM OUTROS SISTEMAS

Os sistemas de segurança como mecanismos de proteção, dissuasão e alerta, apresentam-se sob diversas vertentes e normalmente associadas a implementações estanques de sistemas específicos de um determinado fabricante para uma finalidade concreta, como são exemplo as soluções proprietárias de sistemas de videovigilância, de controlo de acessos, de vigia de periferia, de deteção de intrusão, passando por sistemas de carater preventivo como são os sistemas de deteção de incendio ou deteção de gases, etc.

A interligação entre diferentes sistemas têm vantagens óbvias porque em muitos casos as suas valências são complementares, por exemplo um sistema de controlo de acessos gera um alarme de violação de uma porta controlada, o sistema de deteção de intrusão deteta uma presença não prevista num local e confirma que a violação da porta deu origem a uma intrusão real e o sistema de videovigilância permite verificar as imagens do local e grava-las para uso futuro. Neste tipo de contextos a comunicação entre os vários sistemas disponíveis é de todo desejável.

Neste sentido os fabricantes e os grandes utilizadores tem sido desenvolvido, [26], o conceito de gestão de informação de segurança: PSIM - *Physical Security Information Management*. As soluções PSIM são plataformas de *software* que integram múltiplos sistemas, apresentando uma ferramenta única de gestão das diferentes valências dos sistemas de segurança e que apresentam várias vantagens operacionais nomeadamente a concentração da informação numa única plataforma, permitindo verificar na linha temporal a sequência de acontecimentos de várias fontes diferentes. Esta capacidade de correlação de dados permite que do ponto de vista de administração se tomem decisões sobre cenários mais completos e permite definir procedimentos de atuação para cada um dos cenários com base em informação pluridisciplinar. Inclusive a plataforma pode ser configurada para orientar os operadores nas suas atuações, [26], usando procedimentos pré-estabelecidos SOPs - *Standard Operating Procedures*. Este conceito interliga sistemas com procedimentos no sentido de uma administração mais real e mais eficiente.

Neste momento, não existem normativos que regulem as interações entre sistemas e existem grandes dificuldades de implementação de verdadeiras soluções PSIM, [27], [28]. As soluções que existem atualmente estão confinadas a determinados tipos de equipamento e normalmente dependentes de um fabricante. Contudo, espera-se que no futuro a necessidade de integração global associe os envolvidos nesta temática no sentido da normalização. Nesse futuro os sistemas de controlo de acesso serão uma parte de um sistema de segurança mais lato e abrangente.

2.4. DISPOSITIVOS DE IDENTIFICAÇÃO E AUTENTICAÇÃO

Os procedimentos de identificação e autenticação, normalmente, são ações que se executam uma em complemento da outra e são uma componente fundamental dos sistemas de controlo de acessos, mas as necessidades de identificação e autenticação estão longe de ser exclusivas desses sistemas, nas bibliotecas há necessidade de identificar os livros, nas lojas é necessário identificar os produtos, nas empresas identificam-se as frotas de veículos, os edifícios, o património e os bens que se produzem,… apenas para citar alguns exemplos. Pode-se por isso, considerar que o ato de identificação é uma tarefa que é transversal à atividade humana. Esta amplitude de aplicabilidade resulta na existência de uma vasta gama de soluções e de técnicas para se efetuar a tarefa de identificação. Neste subcapítulo vão ser apresentadas as tecnologias mais usadas nas operações de identificação e autenticação dos sistemas de controlo de acessos.

Os sistemas de identificação do ponto de vista tecnológico usam soluções classificadas em cinco categorias, ilustradas na Figura 25, [29]:

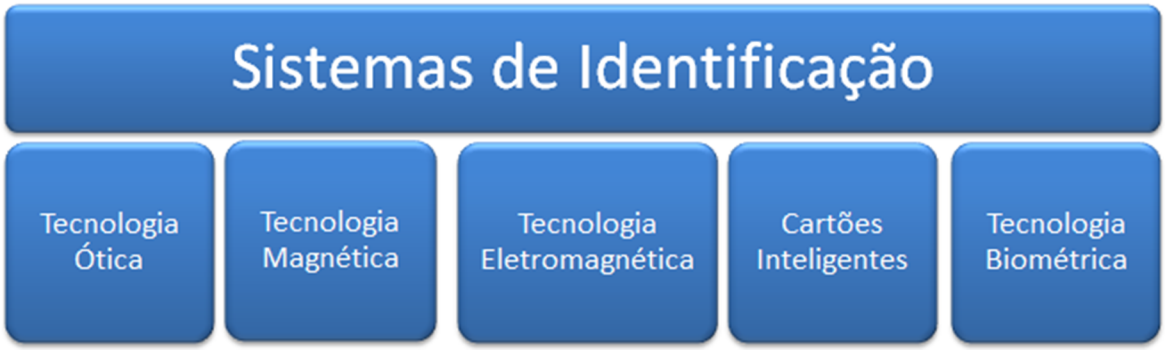


Figura 25 – Tecnologias dos sistemas de identificação.

- A tecnologia ótica efetua a identificação por análise do aspeto visual do dispositivo de identificação, neste âmbito estão incluídos os sistemas de leitura de códigos de barras, os sistemas de reconhecimento de caracteres e genericamente os sistemas de visão artificial.

- A tecnologia magnética efetua o armazenamento da informação, neste caso informação de identificação, em suporte magnético, o exemplo mais comum de

aplicação desta tecnologia são os cartões com banda magnética como os cartões bancários.

- Tecnologia eletromagnética, esta tecnologia é usada essencialmente para comunicação entre o dispositivo de identificação e o sistema de controlo de acessos usando como meio de transmissão campos eletromagnéticos. São exemplo do uso desta tecnologia os cartões de identificação ou dispositivos eletrónicos como *smartphones* que se usam na proximidade de um leitor.

- Cartões inteligentes, vulgarmente denominados *Smart Cards* são dispositivos que tem capacidades de memoria e/ou processamento, recursos que nos sistemas de controlo de acessos são usados nas respetivas ações de identificação e/ou autenticação.

- Tecnologia biométrica, neste âmbito estão todas as soluções que usam a medida de uma característica física para efetuar a identificação e autenticação das pessoas. Os exemplos mais comuns de aplicação de tecnologia biométrica em controlo de acessos são os leitores de impressão digital e os leitores de íris.

O ponto comum de todas as tecnologias é os sistemas serem constituídos por um dispositivo que contem a informação de identificação, por um dispositivo que lê essa informação e de alguma forma a transmite para o sistema de controlo de acessos.

2.4.1. IDENTIFICAÇÃO USANDO TECNOLOGIA ÓTICA

Os sistemas de tecnologia ótica mais usados na identificação de pessoas em sistemas de controlo de acessos são os sistemas de leitura de códigos de barras, como o exemplo mostrado na Figura 26. Os códigos de barras são uma representação gráfica, binária de números e letras em que codificação é feita por arranjos gráficos de sequências de áreas de cor escura e áreas de cor clara com elevado grau de contraste.

Figura 26 – Exemplo de cartão de identificação com código de barras.

Os dispositivos de leitura de tecnologia ótica, normalmente, usam feixe laser projetado sobre o código e usam sensores que medem a reflecção que o feixe provoca nas áreas de cores diferentes que constituem o código a ser lido como ilustra a Figura 27.

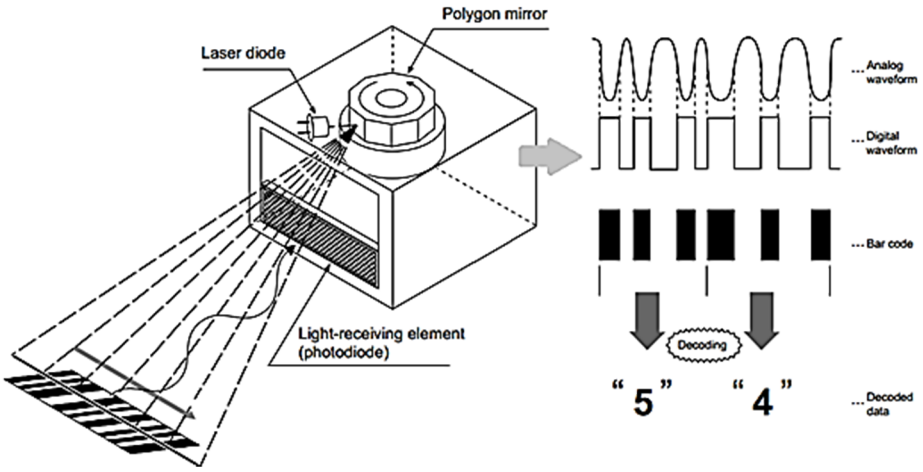


Figura 27 – Funcionamento do leitor de código de barras, [30].

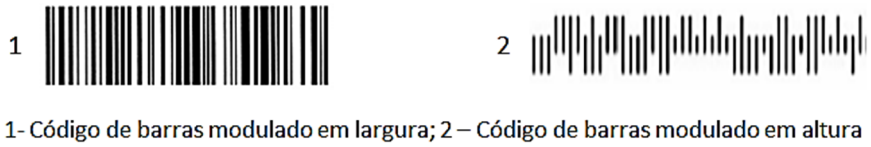
Existem essencialmente dois tipos de códigos de barras [30]: os códigos de barras lineares e os códigos de barras bidimensionais, Figura 28. Os códigos de barra lineares ou 1D, uma dimensão, contem informação apenas num sentido, os códigos bidimensionais ou 2D, duas dimensões contem informação em dois sentidos perpendiculares,

Figura 28 – Código de barras linear e bidimensional.

Os códigos de barras lineares são implementações efetuadas usando um de dois conceitos, Figura 29:

- Largura modulada: é uma representação que usa sequências de linhas de cor escura separadas por espaços de cor clara, uns e outros com largura variável que codificam a informação.

- Altura modulada: a representação da codificação é efetuada por sequências de linhas e espaços com a mesma espessura em que a informação é contida na variação da altura da linha escuras, este tipo de codificação é muito usada por exemplo em correspondência postal.



1- Código de barras modulado em largura; 2 – Código de barras modulado em altura

Figura 29 – Exemplos de diferentes modulações de códigos de barras lineares.

Existem várias normativas para codificação de códigos de barras que definem a forma de codificação, o espaço entre barras, se tem caracteres de início ou fim de código, o tipo de caracteres possíveis, se podem ser lidos nas duas direções ou em apenas numa, etc. Algumas das normas de códigos de barras são, [31]:

- UPC/EAN: usada principalmente na codificação de produtos, apenas admite algarismos.

- Code 39: foi a primeira codificação de caracteres alfanuméricos, permite códigos de comprimento variável.

- Code 128: é uma codificação que admite caracteres alfanuméricos e é caracterizada pela alta densidade de codificação.

- Interleaved 2 of 5: usada apenas para algarismos mas admite códigos de tamanho variável e permite o uso de caracteres de verificação.

Os códigos de barras bidimensionais são representações gráficas que contem informação em duas direções perpendiculares e por esse motivo permitem uma maior capacidade de

armazenamento, por exemplo o código PDF417 consegue conter até 1108 bytes numa única imagem. São exemplos de uso de códigos de barra bidimensionais aplicações de *marketing*, de envio de informação para leitura automática com por exemplo endereços de *sites*, além de informação de autenticação nos sistemas de segurança.

Os códigos de barras bidimensionais podem ser de dois tipos, [31]:

- Códigos bidimensionais empilhados que são constituídos por conjuntos de códigos lineares agrupados verticalmente em que as especificações *Code 39* e *PDF417* são exemplos, Figura 30

Figura 30 – Exemplos de códigos bidimensionais empilhados.

- Códigos bidimensionais matriciais em que a informação é representa em padrões que são lidos nas duas direções, esta codificação é a que permite maior densidade de informação armazenada, são exemplos deste formato os códigos *QR Code*, *Aztec*, *Data Matrix*, e *Maxi Code*, Figura 31.

Figura 31 – Exemplos de códigos bidimensionais matriciais.

A maior complexidade dos códigos bidimensionais impõe o uso de leitores mais sofisticados, normalmente constituídos por sensores de imagem idênticos aos usados mas camaras fotográficas, com resoluções de leitura compatíveis com o tipo e a dimensão dos códigos a ler e por processadores que executam algoritmos de análise de imagem e de reconhecimento de padrões.

As codificações apresentadas anteriormente como exemplos são codificações abertas em que a forma de ler os códigos é conhecida. Para implementações em que a privacidade da informação é importante usam-se códigos proprietários desenvolvidos para uma função específica em que o algoritmo de codificação é confidencial.

O uso dos códigos de barras é uma técnica madura com várias opções de implementação, mecanismos de deteção de erros e taxas de erro de leitura muito baixas, [32] não requere tecnologia sofisticada quer seja para suporte do código quer seja para leitura. No entanto, o facto dos códigos de barras serem facilmente copiados, por exemplo usando uma fotocopiadora ou máquina fotográfica, pode ser uma questão impeditiva para o uso desta tecnologia.

Existem sistemas de controlo de acesso com identificação por os códigos de barras, usados para acessos a realizar uma única vez, por exemplo os cartões de embarque em suporte de papel usados pelas companhias aéreas. Na Figura 32 do lado direito, está mostrado o exemplo de aplicação do projeto desenvolvido pela Brittish Airways, a Air France, a Lufthansa, a KLM e a American Airlines em colaboração com a IATA[7], [33], que visa a implementação um sistema de controlo de acessos para *auto-check-in* em aeroportos, usando códigos de barra bidimensionais apresentados em ecrãs de equipamento eletrónico, como evolução dos tradicionais cartões de embarque em formato de papel.

[7] IATA - *International Air Transport Association*, associação internacional de intervenientes do setor do transporte aéreo: aeroportos, companhias aéreas, etc, que define convenções para regulação do setor.

Figura 32 – Exemplos de códigos de barras em aplicações de acesso, [33].

De notar que cada código apresentado na Figura 31, tem especificidades diferentes que podem ser usadas num contexto ou noutro. Os exemplos mostrados Figura 32 fazem parte da mesma solução, mas um usa suporte de papel e o outro usa suporte eletrónico e por isso para na mesma solução usam-se dois formatos de código é diferentes. Em suporte papel é usado um código do tipo *QR Code*, cujos três quadrados nos extremos definem a limitação da área do código. Nos suportes digitais, os limites do ecrã podem causar problemas na leitura pelos extremos, por isso usa-se o código *Aztec* que tem a mira de alinhamento do código no centro da área.

Outro exemplo de aplicação de acessos por códigos de barras é o aeroporto de Bruxelas, [33], onde foi instalado em 2011 um sistema de controlo de acessos para entrada nas zonas de embarque do espaço *Schengen*, usando leitores de códigos de barra bidimensionais, Figura 33.

Figura 33 – Controlo de acessos com códigos de barras, no aeroporto de Bruxelas.

2.4.2. IDENTIFICAÇÃO USANDO TECNOLOGIA MAGNÉTICA

Nas décadas passadas, antes do uso comum de memórias portáteis de base de silício, usavam-se suportes com partículas magnéticas em que a orientação das partículas codificava informação para armazenamento. Estes suportes eram apresentados em vários formatos: cassetes para áudio e vídeo, disquetes em várias dimensões 5¼", 2.5", *tapes* de armazenamento com várias capacidades e formatos, etc.

No âmbito da identificação, nos anos 70 foram criados cartões com bandas magnéticas que permitem guardar até 1000 *bit* de informação [38] da mesma tecnologia que os suportes referidos anteriormente.

A capacidade de portabilidade de informação associada à facilidade de leitura e aos baixos custos de produção dos cartões, valores desde 0.40€ [36], tornou os cartões de identificação com banda magnética nos mais amplamente difundidos [35].

A norma ISO/IEC7811, usada por exemplo nos cartões bancários, nas partes 2, 6, 7 e 8 específica as propriedades da banda magnética, o método de codificação e localização de pistas de dados, Tabela 2 e Figura 34.

Tabela 2 – Características de um cartão de banda magnética segundo da norma ISO/IEC7811, [38].

Propriedade	***Track* 1**	***Track* 2**	***Track* 3**
Volume de dados	79 Caracteres	40 Caracteres	107 Caracteres
Codificação de dados	6 *bit* alfanumérico	4 *bit* BDC	4 *bit* BDC
Densidade de dados	8.3 *bit*/mm	3 *bit*/mm	8.3 *bit*/mm
Escrita	Não permitida	Não permitida	Permitida

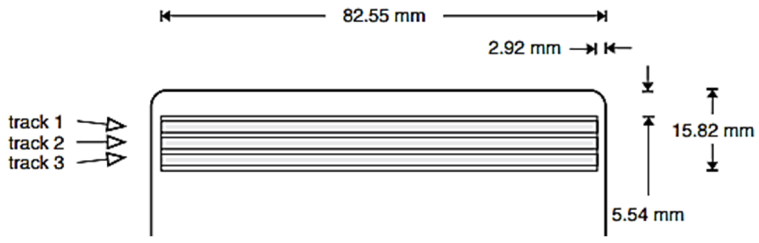


Figura 34 – Características físicas de um cartão de banda magnética, norma ISO/IEC7811, [38].

A gravação e leitura de dados dos cartões magnéticos são feitas por contacto da banda magnética no sensor magnético compatível com o tipo de cartão, [37], o contacto físico no leitor limita a duração de vida dos cartões devido ao desgaste da banda provocado pelo atrito, Figura 35.

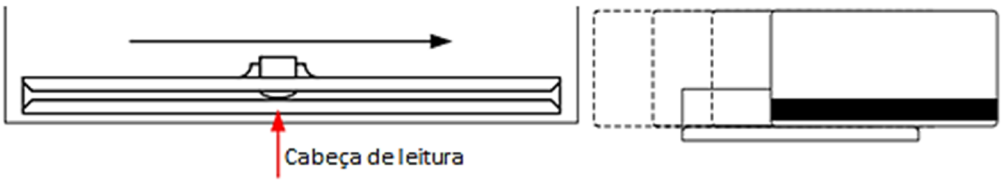


Figura 35 – Diagrama de leitura de um cartão de banda magnética, [37].

A cabeça de leitura dos leitores de cartões de banda magnética é composta por um dispositivo sensível à orientação magnética das partículas que existentes na banda. O processo de leitura gera sequências de impulsos de sinal positivo ou negativo dependendo da informação contida. A eletrónica do leitor efetua filtragem do sinal e decodifica a informação com base no arranjo temporal da posição relativa dos impulsos, Figura 36.

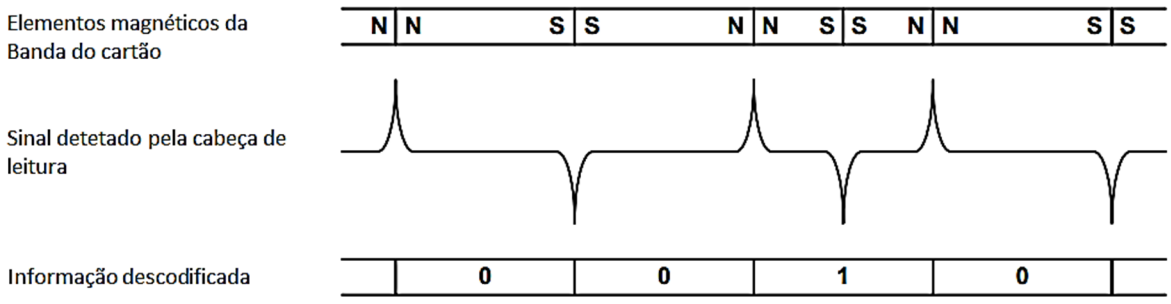


Figura 36 – Processo de leitura de um cartão de banda magnética, [37].

Alem das características intrínsecas do tipo de cartão, as bandas magnéticas podem ser classificadas, [38], em bandas de alta coercividade ou de baixa coercividade, a diferença entre os dois tipos prende-se com a energia magnética necessária para a gravação de informação na banda. Para serem gravadas as bandas de alta coercividade necessitam de

cerca de 4000Oe[8] em comparação dos 300Oe, necessários para gravar as bandas de baixa coercividade. Como as bandas de alta coercividade necessitam de maior energia para gravação são menos sensíveis a alterações da informação provocadas por campos magnéticos externos, são por isso usadas em aplicações com usos frequentes do cartão como no caso dos sistemas de controlo de acessos. Normalmente as bandas de alta coercividade apresentam cor muito escura enquanto as de baixa coercividade tem tons de castanho claro.

A maior vulnerabilidade dos cartões magnéticos é a facilidade de serem copiados e alterados, e por isso, normalmente são apenas usados em aplicações de baixo nível de segurança em que a vertente económica do custo do cartão é uma consideração importante, como por exemplo na abertura de portas de acesso a recursos dentro de edifícios de ginásios. Ou são usados em soluções multi-fator de identificação, como por exemplo as implementações que usam cartões bancários em que o cartão é usado como meio de identificação, não tem informação encriptada, e o PIN, código secreto apenas do conhecimento do utilizador, é usado como mecanismo autenticação.

A Figura 37 mostra três exemplos de leitores de cartões de banda magnética para uso em sistemas de controlo de acessos, o da esquerda associado diretamente à fechadura da porta que é um exemplo de controlo autónomo, os da direita são leitores para serem ligados a unidades de controlo para uso em sistemas distribuídos ou centralizados, o leitor do meio permite autenticação por fator múltiplo: baseada no conhecimento de PIN e baseado na

---

[8] Oe – Oersted unidade do sistema CGS para medida da intensidade de campo magnético – B, que equivale a 1000/(4π) Ampere/metro.

"A unidade recebeu esse nome em homenagem a Hans Christian Ørsted, que descobriu, em 1820, que as correntes elétricas podem criar campos magnéticos, estabelecendo assim um dos marcos iniciais do estudo do Eletromagnetismo." [47].

posse de cartão de banda magnética. Nos três leitores verifica-se a robustez do equipamento, condição importante nas implementações dos sistemas de segurança.

Figura 37 – Exemplos de leitores de cartões de banda magnética, em controlo de acessos.

2.4.3. IDENTIFICAÇÃO USANDO TECNOLOGIA ELETROMAGNÉTICA - RFID

A história da identificação usando tecnologia eletromagnética inicia-se, [39], durante a segunda guerra mundial quando se desenvolveram sistemas para distinguir as aeronaves de um país das aeronaves inimigas nas imagens de RADAR - *Radio Detecting and Ranging*. Desta necessidade surgiram as primeiras implementações de identificação por rádio frequência, RFID - *Radio Frequency IDentification*.

Atualmente a maturação tecnológica e a simplicidade de utilização faz com que a tecnologia RFID, venha a ser cada vez mais usada em aplicações de âmbitos tão diversos, como: localização de objetos [42][43], pagamento de portagens [45], identificação de animais [49], etiquetas de identificação de objetos em armazéns que podem ser lidas mesmo com os objetos dentro das caixas [50], pagamento de tarifas de transportes públicos, por exemplo bilhetes de metro, documentos de identificação, por exemplo passaportes eletrónicos, etc. Estima-se que em 2014 a tecnologia RFID seja um negócio de vinte biliões de dólares [51].

Além da simplicidade de utilização, outra grande vantagem dos dispositivos de tecnologia eletromagnética sobre os de banda magnética prende-se com a durabilidade do dispositivo,

a produção de dispositivos RFID pode ser apresentada em vários formatos e suportes: cartões, pulseiras, relógios, discos, etc, sendo este tipo de apresentação é muito mais durável que os cartões de banda magnética cujo tempo médio de uso é de três anos.

Um sistema RFID é composto por dois componentes principais, [39]:

- O leitor de RF – *Radio Frequency*, também conhecido por estação base ou interrogador. Em operação, este dispositivo emite uma radiação eletromagnética e está constantemente à escuta de um sinal de resposta.

- O identificador RF, também denominado *transponder* ou *tag* na terminologia inglesa que se adotou na linguagem corrente de outros idiomas. O *transponder* é o dispositivo que é instalado fisicamente nos objetos, após o que estes objetos passam a ter a capacidade de se autoidentificarem perante os leitores RFID através de uma resposta ao sinal emitido pelo leitor.

Ao longo do tempo, para colmatar as diversas necessidades de autoidentificação foram desenvolvidos diferentes sistemas que se podem classificar e agrupar considerando múltiplas vertentes, nomeadamente relativamente ao funcionamento dos *transponders*. Neste documento vão ser apresentadas explicações sobre as características que se consideram relevantes para a compreensão das aplicações de sistemas de controlo de acessos.

2.4.3.1. DISPOSITIVOS PASSIVOS E ATIVOS

Os *transponders* em função da sua fonte de energia podem ser classificados em ativos ou passivos.

São *transponders* do tipo ativo quando possuem fontes de energia própria, normalmente em forma de pequenas baterias, como são exemplo os dispositivos instalados nas viaturas para pagamento de portagens eletrónicas, estes dispositivos permitem leitura a maiores distâncias, mas também são mais volumosos e necessitam periodicamente da troca das baterias.

Os *transponders* são classificados como passivos quando a fonte de energia que os faz operar provem do exterior, normalmente extraído do campo eletromagnético irradiado pelo leitor. Exemplos de *transponders* passivos são os cartões usados para identificação de pessoas nos sistemas de controlo de acessos ou no controlo de assiduidade, estes *transponders* são de construção mais simples, menos volumosos, mas a distância ao leitor para se efetuar a comunicação também é mais limitada.

Por sua vez os leitores também podem ser classificados como do tipo passivo ou do tipo ativo mas com um sentido diferente da classificação dos *transponders*. Os leitores do tipo passivo podem funcionar com *transponders* ativos e neste caso com "passivo" pretende-se indicar que o leitor não emite radiação apenas lê sinais gerados por *transponders* ativos, na terminologia inglesa estes sistema denominam-se PRAT – *Passive Reader Active Tag*. Os leitores classificados como ativos, podem ser de dois tipos ARPT – *Active Reader Passive Tag*, ou ARAT – *Active Reader Active Tag*, os primeiros funcionam apenas com *transponders* do tipo passivo e os outros com *transponders* do tipo ativo, no primeiro caso o campo gerado pelo leitor é usado para alimentar o *transponder*.

2.4.3.2. RFID – CLASSIFICAÇÃO SEGUNDO A FREQUÊNCIA DE OPERAÇÃO

Considerando a frequência[9] de operação os sistemas RFID são classificados em três grupos, [29]:

- Grupo I: baixa frequência, LF, 30-300KHz;
- Grupo II: alta frequência, HF, 3-30MHz;
- Grupo III: ultra alta frequência, UHF, $\geq$ 300MHz e micro-ondas.

[9] No Anexo B , são apresentadas algumas notas sobre o espectro eletromagnético

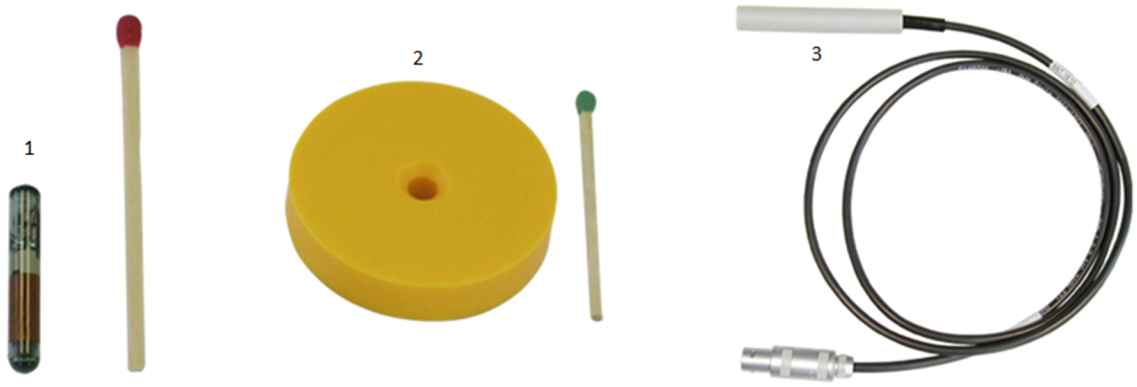
Cada um destes grupos é vocacionado para um tipo de aplicações e essa escolha influencia decisivamente a forma de construção do *transponder*, o seu aspeto final e obviamente as suas características de funcionamento.

Uma das características relevantes dos *transponders* é a distância máxima que podem estar do leitor para que haja comunicação e segundo este parâmetro podemos classificar os *transponders* em:

- Acoplamento nas imediações que se caracteriza por distâncias máximas na ordem do 1cm.
- Acoplamento de proximidade com distâncias máximas na ordem dos 5 cm.
- Acoplamento de vizinhança para distâncias até 150m.
- Acoplamento remoto para distâncias superiores da 150m

Nas aplicações com dados privados como acontece em cartões de acesso que guardam por exemplo informações biométricas, por vezes define-se como requisito o uso de *transponders* de acoplamento nas imediações, para não haver o risco do *transponder* ser acedido/copiado por dispositivos que sejam colocados fisicamente nas proximidades e acedam ao *transponder* sem o conhecimento/consentimento do portador.

As soluções de mercado aplicáveis a controlo de acesso que se enquadram no grupo I, normalmente, funcionam em frequências entre 120-134KHz e a frequência 125KHz é de uso comum. Nesta gama de frequências os dispositivos necessitam de antenas de maiores dimensões e por isso mais caras, mas são menos sensíveis a interferências do meio, porque funcionam sem problemas junto a objetos metal e não são afetados pela presença de água, são, por isso, particularmente adequados para usos industriais, [41]. As antenas dos *transponders* LF são produzidas com bobines de fio, normalmente protegidas por invólucros de vidro ou plástico, Figura 38. A taxa de transferência de dados dos dispositivos do grupo I é a menor, na ordem de 1Kbit/s e são normalmente dispositivos de acoplamento nas imediações ou de proximidade. Do ponto de vista de especificações, as normas ISO11784/5 e ISO14223 definem características dos sistemas deste grupo.



1- Transponder para uso em seres vivos; 2 – Transponder para identificação de objetos em ambientes industriais ; 3 – Antena de estação base para leitura de transponders RF

Figura 38 – Exemplos de dispositivos RFID classificados no grupo I.

Como exemplo de aplicação de identificação RFID de baixa frequência a controlo de acessos existem no mercado várias ofertas com uso de cartões a operar nesta gama de frequência e pode-se apresentar uma solução pouco convencional desenvolvida pela empresa Staywell vocacionada para animais de companhia que apenas permite o uso das portas de passagens para animais, aos que tiverem identificados com o respetivo *transponder* que pode estar instalado na coleira ou por baixo da pelo do animal, Figura 39.

Figura 39 – Exemplo de controlo de acesos a animais, *transponder* do grupo I.

Os dispositivos do grupo II, permitem taxas de transmissão superiores às do grupo I, na ordem do 10Kbit/s e permitem maiores distâncias de comunicação e por isso neste grupo existem soluções em que o *transponder* é equipado com dispositivos de memória e/ou microprocessadores. As antenas dos *transponders* deste grupo, são mais pequenas que as do grupo I sendo normalmente construídas por cerca de dez voltas de pista metálica e por

isso podem ser produzidas em formatos muito finos, por exemplo em formato de etiqueta, em suporte de papel ou plástico, Figura 40, ou em formato de cartão de identificação.

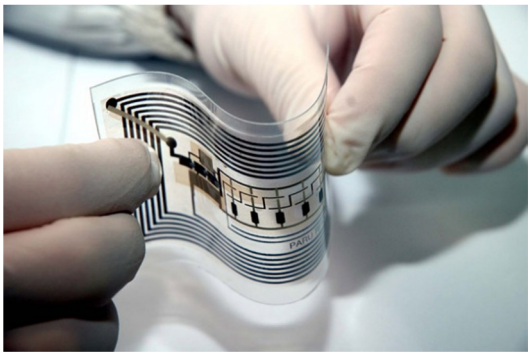


Figura 40 – *Transponder* grupo II.

Os sinais da gama de frequência do grupo II consegue atravessar a maior parte dos materiais que não sejam metálicos, [41], e os leitores tem capacidade para ler simultaneamente vários *transponders*. Nas soluções disponíveis para controlo de acessos é comum usar a frequência 13.56MHz e existem normas para desenvolvimento deste tipo de produtos: a ISO15693 define normas para meios de pagamento, a ISO/IEC14443 apresenta definições para dispositivos de identificação e autenticação como passaportes, a norma ISO18000-3 define regras para as comunicações sem fios de dispositivos de identificação como os usados nos sistemas de controlo de acessos. A Figura 41, apresenta exemplos de *transponders* do grupo II usados em sistemas e controlo de acessos.

Figura 41 – Exemplos de *transponders* RFID classificados no grupo II.

Os sistemas que usam frequências UHF como os exemplos da Figura 42, permitem as taxas de transferência mais elevadas, na ordem dos 100Kbit/s e as maiores distâncias de comunicação, dado o comprimento de onda dos sinais envolvidos, a antena do *transponder*

é do tipo dipolo, [41], e o seu grau de miniaturização permite a produção de *transponders* muito pequenos.

Uma consideração importante no uso de dispositivos do grupo III, prende-se com o facto das comunicações na banda UHF poderem estar sujeitas a licenciamento dependendo do país onde a solução vai ser instalada

Figura 42 – Exemplos de dispositivos RFID classificados no grupo III.

Um fator que é fortemente influenciado pela frequência de operação e que impacta diretamente com a seleção da tecnologia a usar para implementação das soluções, são as características espaciais de funcionamento que tem de ser consideradas dispositivo-a-dispositivo na solução definida, nomeadamente a distância máxima entre o leitor e o *transponder*, a distância mínima entre *transponders* na zona de interrogação para que consigam ser lidos, a distância mínima entre leitores e a velocidade máxima de passagem do *transponder* na zona de interrogação para poder ser lido.

Cronologicamente os primeiros *transponders* RFID usados nos sistemas de controlo de acessos operavam na frequência de 125KHz e são vulgarmente conhecidos por cartões de proximidade, os dispositivos a operar na frequência de 13.56MHz sugiram depois e são chamados genericamente de *smart cards*.

2.4.3.3. TIPO DE INFORMAÇÃO NOS *TRANSPONDERS*

Os sistemas RFID podem ser classificados com base na quantidade de informação que flui entre o leitor e o *transponder*, classificados segundo as capacidades de armazenamento do

transponder e classificados consoante o tipo processamento que o *transponder* é capaz. Nestas vertentes existem os seguintes grupos de dispositivos[29]:

- Sistemas EAS – *Electronic article surveillance system*: também conhecidos por *transponders* de 1 *bit*, são os sistemas mais simples do ponto de vista de tratamento de informação. A única função deste sistema é a deteção de um *transponder* na zona de interrogação do leitor, na realidade estes sistemas não permitem uma identificação apenas induzem perturbações do campo eletromagnético que são detetadas pelos leitores.

- *Transponders* apenas de leitura – o microchip dos *transponders* deste tipo contem informação fixa definida no processo de produção, normalmente um número. Quando o *transponder* está na zona de interrogação do leitor o *transponder* envia continuamente a informação armazenada. Nestes sistemas para haver transmissão de informação com sucesso apenas pode existir um *transponder* na zona de interrogação, porque mais de um *transponder* na zona de interrogação provoca a sobreposição de sinais e o leitor não consegue identificar nenhum dos *transponders*.

- *Transponders* com memória de leitura e escrita, nesta categoria estão classificados uma vasta gama de dispositivos que variam na capacidade, na forma de armazenamento e nos que permitem ser escritos apenas uma vez ou diversas vezes. As capacidades podem variar de 1byte a cerca 100Kbyte, estes *transponders* usam técnicas de máquinas de estados para efetuar as operações de leitura e escrita, e normalmente dispõem de mecanismos anti-colisão que permitem a operação de vários simultaneamente. Os *transponders* deste género do tipo passivos são implementados com memórias EEPROM e atualmente a ser cada vez mais frequente o uso de memórias de tecnologias flash[10]. Os *transponders* do tipo ativo

[10] Memorias flash – tipo de memória não volátil que que pode ser programada eletricamente EEPROM - *Electric Erasable and Programmable Read-Only Memory* que se distingue das memórias EEPROM

usam memórias FRAM[11], que são muito mais eficientes que as memórias EEPROM.

Os dispositivos mais sofisticados são equipados com microprocessadores que permitem a implementação de procedimentos de autenticação e transferência de informação encriptada e os *transponders* de topo de gama estão equipados com coprocessadores criptográficos apenas dedicados à implementação de algoritmos de encriptação, estes dispositivos operam quase exclusivamente à frequência de 13.56MHz.

2.4.3.4. ACOPLAMENTO ENTRE OS LEITORES E OS *TRANSPONDERS*

Nos métodos de comunicação por rádio-frequencia, a forma como se processa a transferência de informação dos *transponders* para os leitores está intimamente ligada à frequência de operação. Nos sistemas do grupo I e II, o comprimento de onda do sinal[12] - λ é normalmente muito maior que a distância entre o leitor e o *transponder*, por exemplo em dispositivos a funcionar a 125KHz temos comprimentos de onda de 2400m e para frequências de 13.56MHz $\lambda \approx$22m, assim para estes grupos a comunicação entre os

---

anteriores, porque em vez de ter de apagar todo o espaço de memória antes de se efetuar uma escrita de qualquer dimensão, pode ser escrita apenas em pequenos blocos tipicamente com a dimensão de alguns *byte*. Como as operações apagar dados são relativamente lentas, o facto só ser necessário apagar um espaço relacionado com o que se pretende escrever, faz com que esta tecnologia tenha um desempenho muito superior às EEPROM convencionais.

[11] Memórias FRAM - *Ferroelectric Random Access Memory* – Memórias de acesso aleatório cujas células apresentam capacidade ferroelétricas, isto é, tem a capacidade de manter a polarização da célula mesmo quando a alimentação deixa de estar presente. [29]

[12] Ver n oAnexo B , detalhes sobre a radiação eletromagnética.

dispositivos explica-se com a teoria eletromagnética dos campos próximos – *Near Field* em que o acoplamento é principalmente de origem magnética e os limites das distâncias de comunicação medem-se no máximo na ordem das dezenas de centímetros. Nos sistemas do grupo III, por exemplo para a frequência de operação de 2.45GHz o comprimento de onda é aproximadamente 0.12m, como neste grupo de equipamentos se efetuam comunicações a distancias na ordem de várias dezenas de metros, a distância entre os dispositivos de comunicação é muito maior que o comprimento de onda do sinal e neste caso a comunicação baseia-se na teoria eletromagnética de campos distantes – *Far Field* e o acoplamento é efetuado por campo eletromagnético. Assim, quanto à energia usada para transmissão de dados, os sistemas podem ser classificados em dois grupos como mostrado na Figura 43, [29]:

- Nos sistemas de campos próximos, para frequência LF e HF, a energia usada na comunicação é apenas magnética – B. Nestes sistemas existe um acoplamento indutivo em que a transferência de dados usa um método idêntico ao funcionamento dos transformadores em que as antenas do leitor e do *transponder* constituem os enrolamentos primário e secundário do transformador, um dos problemas desta forma de comunicação é que a potência transmitida diminui na relação inversa do cubo da distância: $\frac{1}{(Distância)^3}$

- Nos sistemas de campos distantes, para frequências UHF, a comunicação efetua-se por acoplamento eletromagnético em que a informação é transportada por ondas de rádio, que após serem emitidas tornam-se independentes da fonte que as gerou.

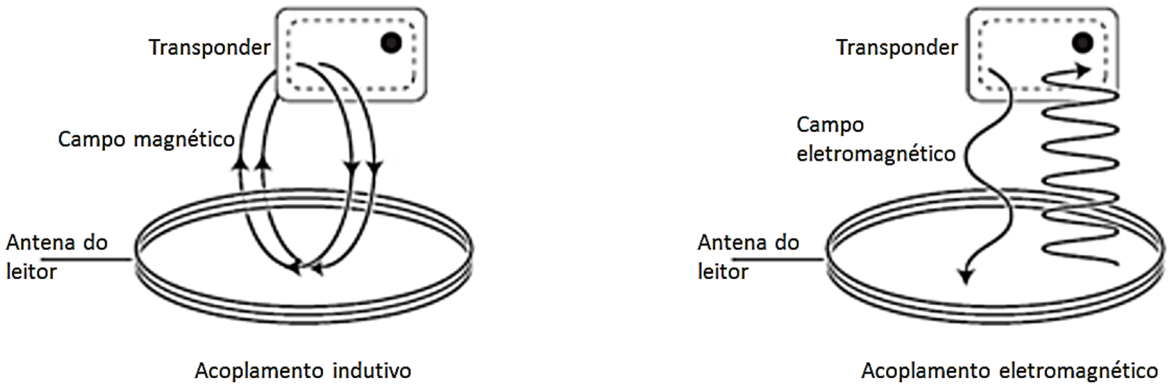


Figura 43 – Natureza da energia usada para a comunicação entre o leitor e o *transponder*, [53].

- Os *transponders* usados em sistemas de controlo de acessos em edifícios, tipicamente funcionam ou a 125KHz ou a 13.56MHz. Estes dispositivos são constituídos quatro blocos: bloco de oscilação, um circuito de retificação, um bloco de contem informação e um bloco de variação de carga. Estes *transponders*, nos vários formatos que assumem, cartões, pulseiras, discos, etc., funcionam por acoplamento indutivo que tem o seguinte princípio de funcionamento:

    - O oscilador do *transponder* é composto por uma bobine - antena e por um condensador. Quando a antena é colocada fisicamente dentro do campo gerado pelo leitor, oscilador entra em ressonância[13], gerando uma corrente que é recebida pelo circuito de retificação constituído tipicamente por um díodo e um condensador e usado para alimentar o bloco que contem informação.

    - As antenas do leitor e do *transponder* formam um sistema de duas bobines, que no conjunto funciona como um transformador em que uma variação de carga no secundário, *transponder*, induz uma variação de carga no primário, leitor. A variação de carga do lado do *transponder* é controlada pelo bloco que contem a informação a ser transmitida e a variação de carga é feita de forma a codificar a informação a transmitir.

---

[13] [53] - Um circuito elétrico ressonante é constituído por pelo menos dois armazéns de energia de natureza diferente, um indutivo e outro capacitivo, responsáveis pela oscilação de energia de uma forma de armazenamento para a outra.

Dependendo do valor dos componentes, o circuito apresenta uma frequência própria de ressonância para a qual ocorre um pico de tensão para circuitos série ou corrente para circuitos paralelos. Um circuito RLC a oscilar à frequência de ressonância as reactâncias capacitivas e as reactâncias indutivas cancelam-se mutuamente e o circuito equivalente tem uma impedância puramente resistiva.

O circuito equivalente, ideal, ressonante usado nos *transponders* que é do tipo LC paralelo cuja respetiva frequência de ressonância é determinada pela relação $f_0 = \frac{1}{2\pi\sqrt{L \cdot C}}$

- Do lado do leitor as variações de carga efetuadas pelo *transponder* são vistas com variações de tensão nos terminais do primário, antena do leitor, que são posteriormente tratadas pela eletrónica do próprio leitor como informação recebida.

Na Figura 44 é apresentado o diagrama de funcionamento da comunicação RFID usando acoplamento indutivo com variação de carga. O leitor é apresentado do lado esquerdo e o *transponder* do lado direito. No *transponder* oscilador é constituído pela antena e pelo condensador C_1, o retificador é constituído pelo díodo e pelo condensador C_2, o dispositivo de memória identificado como "Chip", pode ser uma célula de memória, ou uma célula de memória associada a processadores[14]. A modulação da carga é controlada pelo "Chip" através do transístor que controla a carga do secundário.

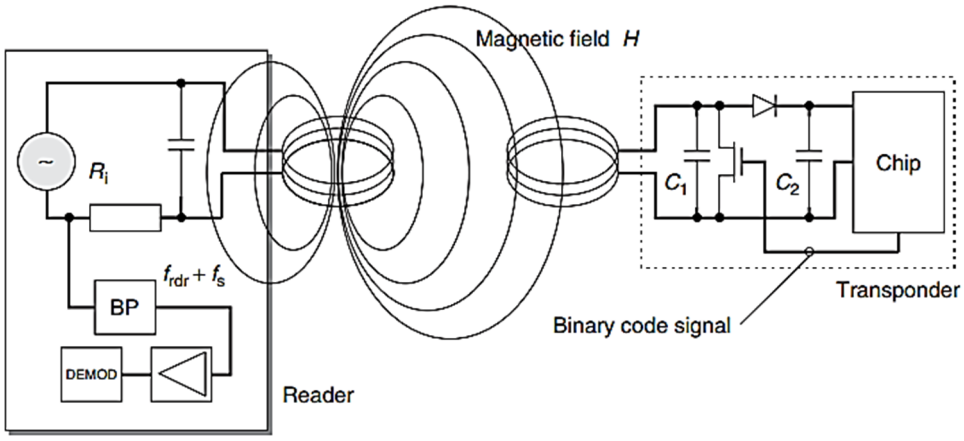


Figura 44 – Diagrama de acoplamento indutivo com variação de carga [29].

[14] Neste ponto usa-se o termo processador no plural porque existem sistemas com um processador para execução de algoritmos genéricos e um coprocessador para execução de algoritmos de criptográficos.

2.4.3.5. RESUMO DE CARACTERÍSTICAS DOS SISTEMAS RFDI

A Tabela 3 e a Figura 45, apresentam um resumo das características dos sistemas e identificação de tecnologia eletromagnética descritas nos subcapítulos anteriores abordando os aspetos mais relevantes.

Tabela 3 – Resumo das características dos sistemas RFID segundo a frequência, [29], [41].

Frequência	**LF** **120-134KHz**	**HF** **13.56MHz**	**UHF** **850-960MHz**
Distância de leitura	<0.5m	<1m	>3m
Custo	Mais caro	Menos caro	O mais barato
Penetração nos materiais	Excelente	↔	Mau
Penetração na água	Excelente	↔	Mau
Antena	Bobine: fios	Bobine: pistas	Dipolo
Taxa de transferência de dados	Baixa	↔	Alta
Leitura de múltiplos *transponders*	Baixo	Bom	Muito Bom
Aplicações	Imobilizado, Identificação industrial, Identificação animal, Segurança, etc	Produtos etiquetáveis, Proteção contra furtos, Bilhetes, Documentos de identificação, Pagamentos, Acessos, etc	Localização de veículos, Portagens, etc

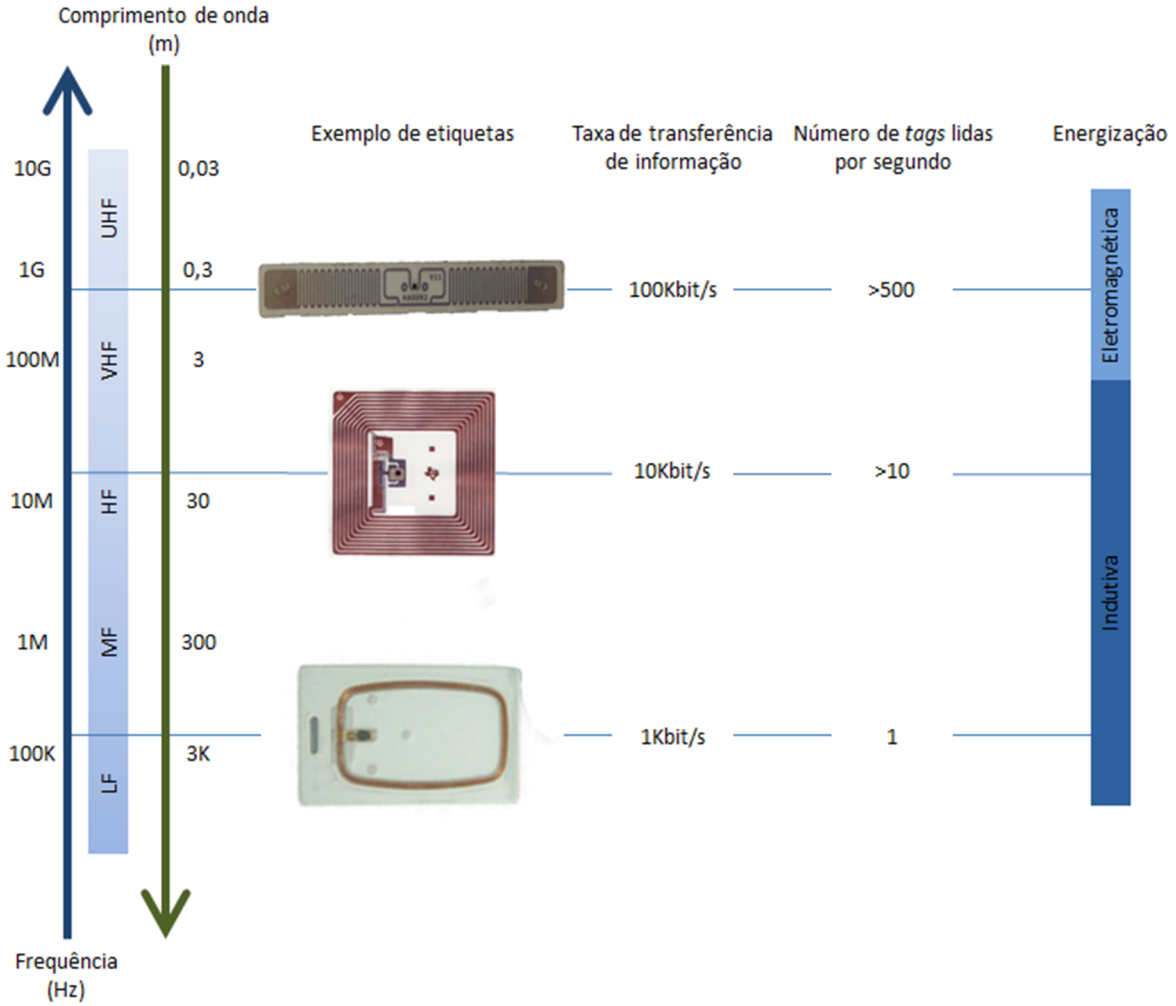


Figura 45 – Características dos *transponders* dos grupos I, II e III, adaptado de [44].

2.4.4. IDENTIFICAÇÃO COM *SMART CARDS*

Os *smart cards* são, [38], cartões de identificação e autenticação que contem na sua estrutura componentes para armazenamento, processamento e transmissão de dados. No Anexo C são mostradas as características dimensionais dos formatos de cartões existentes.

A maior vantagem do uso de *smart cards* prende-se precisamente com a capacidade de armazenamento e a capacidade de processamento. A capacidade de armazenamento aumenta a cada geração de circuitos de memória abrindo novas possibilidades de soluções e a capacidade de processamento permite executar algoritmos de tratamento de dados e

implementar mecanismos de segurança como algoritmos criptográficos sobre a informação armazenada e sobre as comunicações com o exterior. Estas funcionalidades transformaram o negócio de *smart cards* no segmento que regista maior crescimento na indústria da microeletrónica, [29].

Os diversos formatos existentes de *smart cards* são normalmente classificados em duas categorias: quanto o tipo de recursos que possuem e quanto à forma como transmitem a informação. Relativamente aos recursos os cartões podem ser do tipo cartão de memória ou cartão com memória e processador. Relativamente à forma de transmissão de dados, a comunicação pode ser efetuada através de ligação eletromagnética, como nos dispositivos RFID ou por conexão elétrica entre contactos metálicos existentes na superfície do cartão e um sistema de contactos por molas do leitor de cartões, Figura 46. A interface elétrica tem a desvantagem do desgaste provocado pelo atrito entre os contactos e o desgaste provocado pela oxidação. No Anexo D , são apresentados detalhes das características dos contactos elétricos.

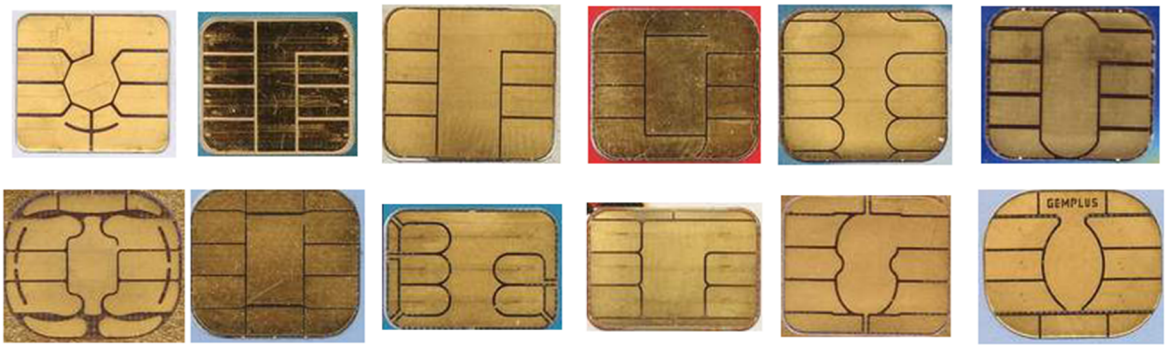


Figura 46 – Exemplos de contactos elétricos de cartões.

2.4.4.1. CARTÕES DE MEMÓRIA

Os cartões de memória caracterizam-se por tem recursos de armazenamento de dados em memórias não voláteis tipicamente EEPROM - *Electric Erasable and Programmable Read-Only Memory*. A memória do dispositivo pode conter todas as funcionalidades das memórias convencionais como, segmentação, lógica de segurança para proteção de leitura e escrita em toda a memória ou diferente de segmento para segmento, lógica de

codificação de dados, etc. A Figura 47 apresenta o diagrama de blocos dos cartões de memória com comunicação por contactos e por proximidade.

Do ponto de vista de capacidade, existem cartões para armazenar desde alguma centenas de *byte* até a alguns *Kbyte*, por exemplo, a HID, um fabricante de renome internacional na sua gama *iClass*, apresenta cartões de memória com capacidades de 256 *byte* até *4Kbyte*, [54]. Todos os cartões de memória possuem uma pequena memória ROM para conter o respetivo número de serie que também é acessível ao exterior.

Os cartões de memória com interface por contactos elétricos podem se acedidos usando por exemplo o protocolo serie I^2C - *Inter-IC*[15], [38]. Os cartões telefónicos pré-pagos são um exemplo de *smart cards* de memória com comunicação por contacto elétrico.

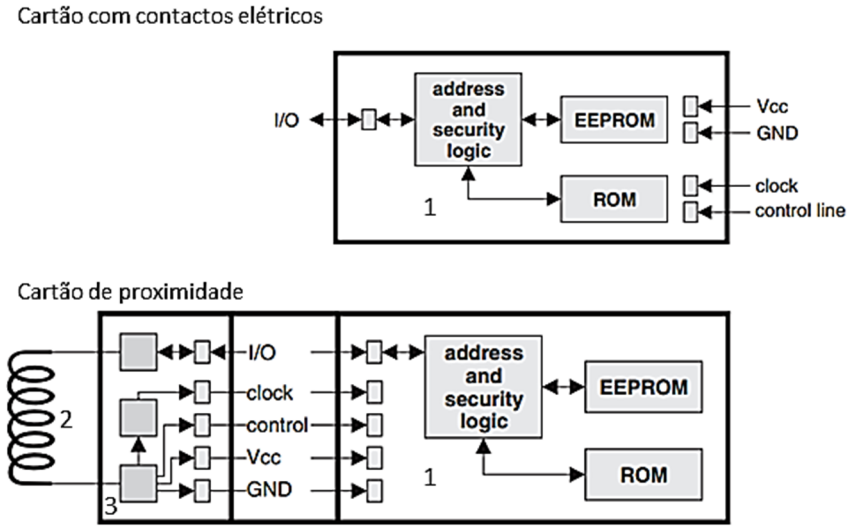


Figura 47 – Diagrama de blocos de cartões de memória, [38] [29].

Os cartões de memória que funcionam sem contactos normalmente transmitem dados usando o protocolo definido na norma ISO/IEC 14443, em distâncias até 10cm, [38]. Estes

[15] Ver detalhes do protocolo I^2C no Anexo E .

cartões alem dos recursos para implementação da memória tem uma interface de radiofrequência, semelhante aos cartões RFID, que, do sinal eletromagnético recebido pela antena retira a alimentação elétrica necessária ao funcionamento de todo o sistema, gera o sinal de relógio, gera sinais de controlo como o *reset* em caso de falta de energia e disponibiliza as funcionalidades de comunicação bidirecional, Figura 47.

Os sistemas de controlo de acessos que recorrem a cartões de memória, além da informação número do cartão que pode ser usado para o processo de identificação, podem usar informações armazenadas para o processo de autenticação como fotografia ou padrões biométricos. No entanto como os mecanismos de proteção deste tipo de cartões são algo rudimentares não são soluções a considerar quando se exigem elevados níveis de segurança.

2.4.4.2. CARTÕES COM PROCESSADOR

Os cartões com processador estão equipados com um sistema computacional completo. Contem um ou mais processadores e vários tipos de memória, como mostrado na Figura 48. No bloco de memória ROM – *Read Only Memory* está armazenado o número de serie do cartão e o SO- Sistema Operativo que é carregado na fase de produção e imutável durante a vida do cartão. Na memória EEPROM é guardada a informação relativa à aplicação em concreto, código de programação e dados, sendo a leitura e escrita na EEPROM controlada pelo sistema operativo. A memória RAM – *Random Access Memory*, é a memória volátil para uso na operação do(s) processador(es).

O aumento da capacidade de armazenamento das memórias que também se faz refletir neste tipo de cartões, conduziu a que a interface série de comunicação se transformasse numa limitação, para colmatar esta questão foram desenvolvidas novas interfaces como

interface USB para cartões ou o protocolo SWP – *Single Wire Protocol*[16] para comunicação do cartão com periféricos, Figura 48.

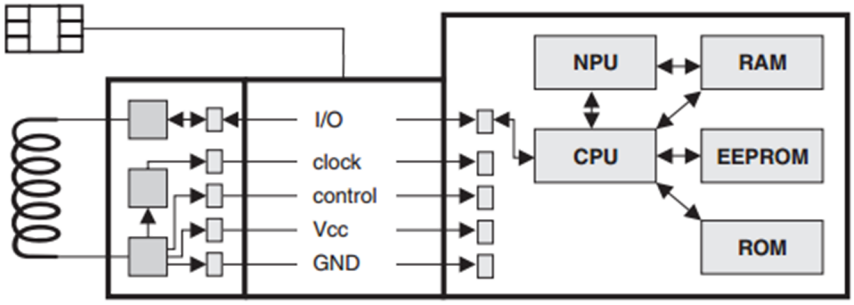


Figura 48 – Diagrama de blocos de cartões com processador, [38] [29].

As gamas mais avançadas dos cartões com processador, além do processador principal dedicado às aplicações, contêm um processador numérico, NPU - *Numeric Processing Unit* otimizado para computação de algoritmos de encriptação e desencriptação do tipo do algoritmo chave pública / chave privada em que uma chave é usada para encriptar dados e a outra para desencriptar.

Á semelhança dos cartões e memória existem cartões com processador equipados com interface elétrica, interface sem contacto ou com ambas. Neste âmbito por exemplo a HID, oferece soluções de cartões com várias tecnologias, que permitem usar o mesmo cartão em mais de uma aplicação e em mais de um sistema, por exemplo cartões com uma antena de 125KHz para identificação por número de cartão, antena de 13.56MHz para guardar informação do portador e banda magnética para compatibilizar o cartão com sistemas mais antigos. Na Figura 49, podemos ver a ilustração de um cartão multitecnologia onde existem duas antenas, uma para cada frequência, uma banda magnética e com tomada de contacto elétrico para acesso ao processador.

[16] Ver mais detalhes do *Single Wire Protocol* no Anexo E .

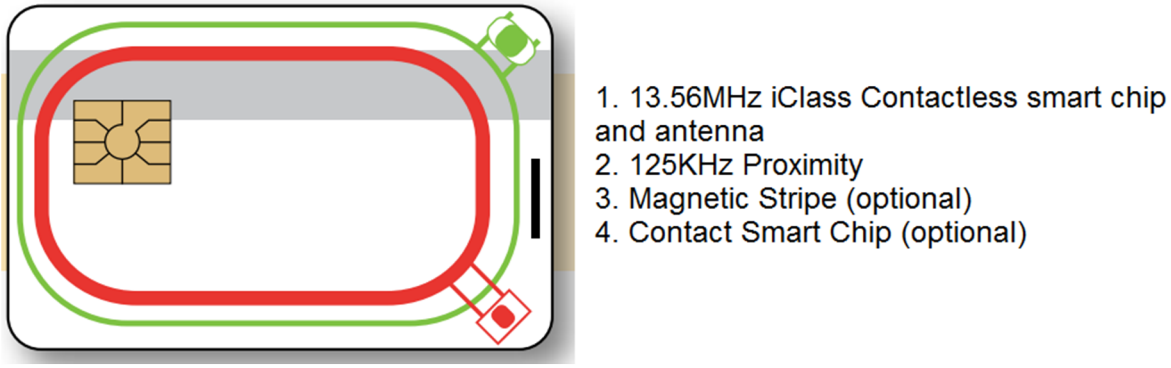


Figura 49 – Cartão HID Multitecnologia [54].

Um exemplo prático de cartão multitecnologia é o cartão de aluno do ISEP, Figura 50. Que é um cartão de proximidade para ser usado no sistema de registo de assiduidade do aluno, tem uma tomada de contactos elétricos para acesso ao processador, tem uma banda magnética de alta coercividade para uso no sistema bancário e apresenta dum código de barras para leitura ótica.

Figura 50 – Cartão do aluno ISEP.

Com estas capacidades os cartões com processador apresentam um elevado grau de adaptabilidade a várias funções, mas o seu custo também é superior ao das outras soluções já apresentadas pode variar de 4€ a 18€, por isso, a sua gama de aplicações é mais justificada quando se usam os recursos intensivamente como o caso das situações onde o nível de segurança é a maior preocupação.

Como exemplos de aplicação, os cartões GSM - *Global System for Mobile Communications* usados nos telefones móveis são um dos exemplos mais difundidos do

uso de cartões com processador com contacto elétrico. E os cartões com processador sem contacto têm grande implementação nas soluções de controlo de acessos a transportes públicos.

A Figura 51, é apresentado um diagrama das ofertas de mercado de tipos de *transponders* agrupados segundo as suas características.

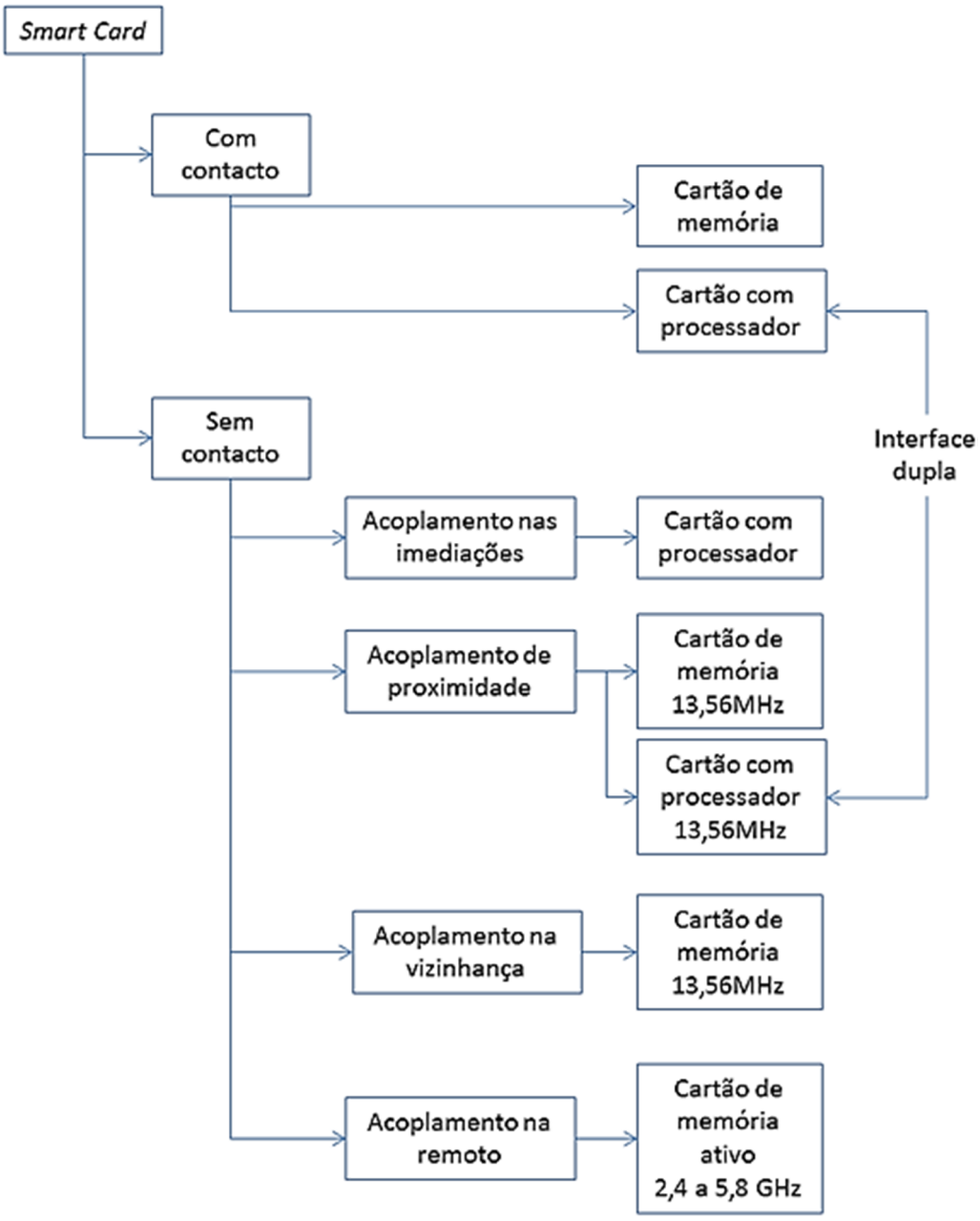


Figura 51 – Resumo das características dos sistemas eletromagnéticos de identificação.

2.4.5. IDENTIFICAÇÃO COM DISPOSITIVOS ELETRÓNICOS – NFC

Na última década, com a difusão de dispositivos eletrónicos portáteis de uso pessoal, equipados com grande nível de recursos como os *smartphones* e os *tablet,* surgiram tecnologias de identificação e autenticação usando esses dispositivos denominadas de *Near Field Communication* – NFC [55], [56]. Esta tecnologia é, [29], um conjunto de protocolos que utiliza a norma ISO/IEC14443 de dispositivos RFID para efetuar comunicação sem fios em distâncias até 20cm.

Para apoiar a tecnologia NFC inicialmente criada pela Sony e pela NXP, foi criado um fórum constituído pelas entidades envolvidas no setor, nomeadamente empresas de desenvolvimento de produtos, aplicações e serviços, empresas de comercio e organizações se fins lucrativos, contando atualmente com cerca de 130 membros, [38], que definem arquitetura da tecnologia e protocolos de comunicação e os submetem às organizações de normalização.

O protocolo de comunicação NFC é baseado no protocolo dos cartões de proximidade, com a diferença que qualquer interveniente na comunicação pode assumir o papel de *master*. Permite comunicação *half-duplex* e usa o mecanismo ouvir-antes-de-falar para evitar colisões. Quando um dispositivo assume-se como *master* escolhe, [38] a taxa de transferência de dados 106, 212 ou 424 Kbit/s. O protocolo funciona na frequência 13,56MHz, usando para transmissão o campo magnético, e admite dois modos de funcionamento, Figura 52:

- Ativo: quando os intervenientes geram a sua frequência de para transmissão de dados.

- Passivo: quando um dispositivo gera frequências de comunicação e o outro responde usando o sinal do primeiro, por acoplamento indutivo, transferindo informação por modulação de carga.

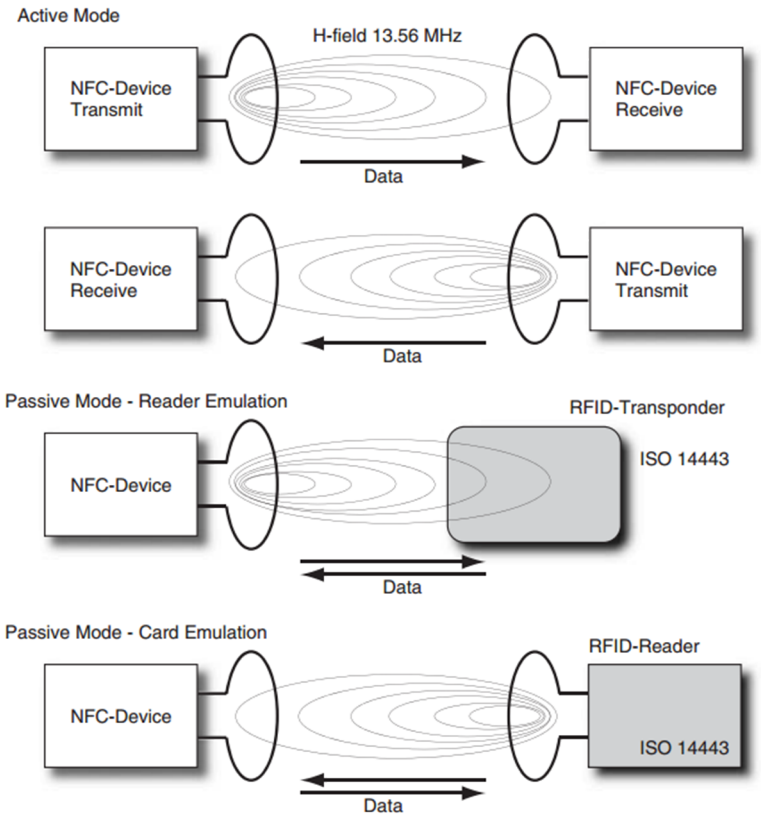


Figura 52 – Modos de operação NFC, [29].

O NFC como protocolo de comunicação é usado nos seguintes grandes grupos de aplicações:

- Serviços de leitura: nestas aplicações o utilizador usa o seu dispositivo eletrónico equipado com interface NFC para o aproximar de um *transponder* e obter a informação que o *transponder* tem guardado. Exemplos deste uso são as etiquetas inteligentes que se usam junto a peças expostas em museus cuja leitura apresenta informação sobre as peças ou redirecionam para *sites* onde essa informação existe.

- Troca de informação: estas aplicações permitem que dois dispositivos com interface NFC comuniquem entre si.

- Identificação, autenticação e pagamento: nestes casos o dispositivo NFC assume um papel passivo como se fosse um cartão de pagamento ou um cartão de acessos, permitindo que ao aproximar o dispositivo eletrónico dos pontos de transação, por

exemplo leitores de bilhetes ou leitores de controlo de acesso, execute a mesma função que um cartão físico.

As aplicações de troca de informação, de identificação, de autenticação e de pagamento, obrigatoriamente, têm de disponibilizar funcionalidades de segurança compatíveis com a solução global de pagamento ou de acessos que permitam identificações e autenticações de forma segura. Para implementar os mecanismos de identificação e autenticação seguras os dispositivos NCF normalmente utilizam uma das seguintes soluções:

- Uso permanente, no dispositivo eletrónico de um integrado de segurança, Figura 53.1.
- Uso de um cartão de memória externo com aplicação de segurança, Figura 53.2.
- Nos dispositivos de comunicação móvel usa-se o cartão (U)SIM do dispositivo para fazer a validação de segurança, Figura 53.3. Nestas aplicações, as comunicações entre a interface NFC e o cartão (U)SIM faz-se usando o protocolo SWP, ver no Anexo E mais detalhes do *Single Wire Protocol*.

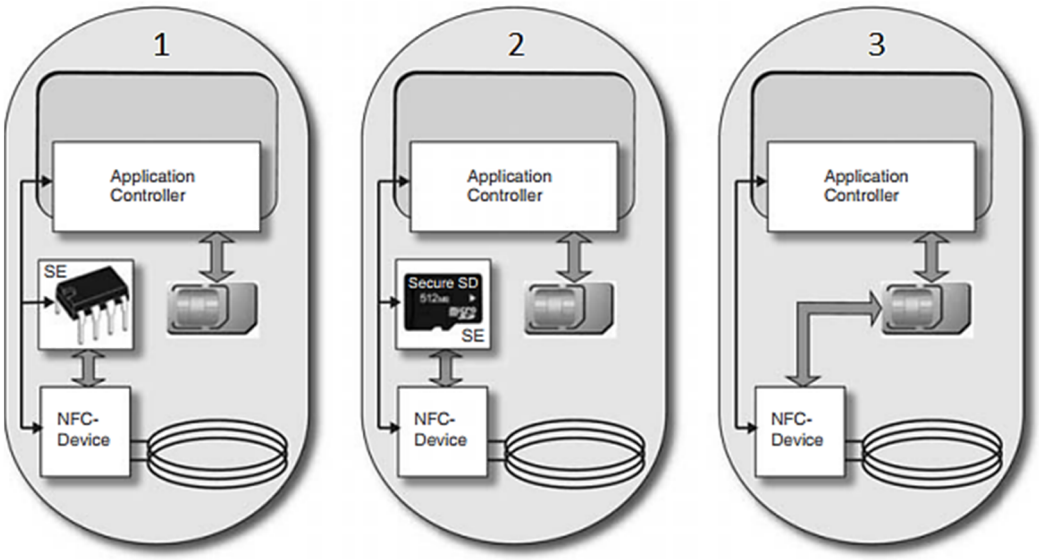


Figura 53 – Mecanismos de segurança para uso das comunicações NFC, [29].

A massificação do uso de *smart phones* tem estimulado os maiores operadores de redes móveis ao desenvolvimento de plataformas para uso do cartão USIM como elemento de segurança. A Vodafone, a Deutsche Telekom, a Telefonica, a AT&T a Verizon, a Sprint e

a Swisscom são exemplos, [58], de operadores que nos últimos anos tem desenvolvido e implementado plataformas de segurança denominadas *TSM – Trusted Servive Management*, que disponibilizam mecanismos de segurança para uso em aplicações de identificação e pagamento usando os *smartphones*.

Nos pontos seguintes são apresentados exemplos de mercado de aplicação da tecnologia NFC no ambito do controlo de acessos:

- FeliCa, sistema desenvolvido pela Sony usado essencialmente para bilhética eletrónica.

- Mifare, sistema desenvolvido pela NXP que é compatível com os cartões RFID especificado na norma ISO/IEC14443 e permite substitui-los com um dispositivo eletrónico.

- A empresa *Simons Voss* disponibiliza soluções NFC em duas vertentes, [57], Figura 54:

    - Aplicação de *software*, para os sistemas operativos iOS e Android, denominada *Mobile Key* que é uma "central de gestão" de chaves eletrónicas onde o utilizador tem uma biblioteca de códigos para usar em vários sistemas e cuja autenticação é efetuada através do cartão USIM do telemóvel, Figura 54.

    - Dispositivos como fechaduras que abrem por ação de identificação, via *transponder* RFID ou simulado por um dispositivo com interface NFC e equipado com a aplicação Mobile Key.

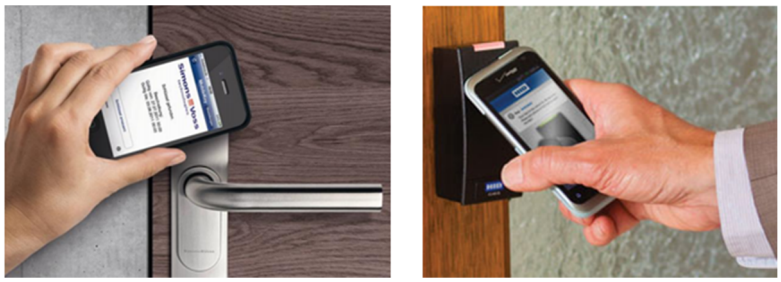


Figura 54 – Controlo de acesso usando tecnologia NFC [57].

As fechaduras apresentadas pela *Simons Voss*, além de poderem operar de forma completamente autónoma numa filosofia de controlo de acessos via NFC, também podem ser interligadas a sistemas externos para criação de estruturas baseadas no conceito PSIM - *Physical Security Information Management*, visto anteriormente. Estas potencialidades são implementadas porque as fechaduras, além da interface NFC para controlo de acessos, tem também uma interface de rede sem fios, [59] para ligar na rede denominada WaveNet, que é constituída por *routers* de controlo. Os *routers* podem enviar às portas mensagens de abertura ou mensagens de bloqueio. Cada *router* que podem comandar até 249 portas e por exemplo, em caso de alarme de incendio, o sistema central de segurança PSIM, pode comandar o *router* para instruir as portas para abrirem e poderem constituir caminho de evacuação. Noutro exemplo, em caso de alarme de intruso armado, o sistema central via *router*, pode comandar todas as portas para fecho e assim proteção das pessoas nos compartimentos, Figura 55.

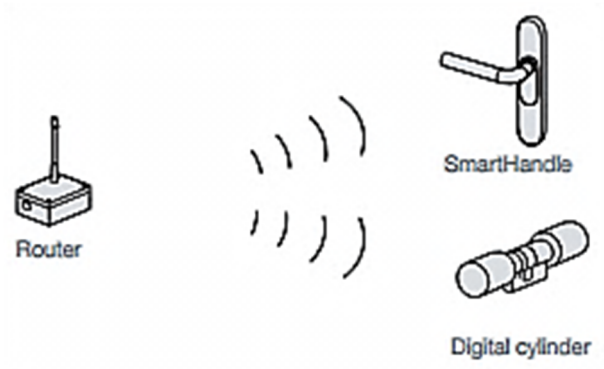


Figura 55 – Rede sem fios para controlo de bloqueadores de acesso, adaptado de [59].

2.5. IDENTIFICAÇÃO E AUTENTICAÇÃO DE PESSOAS USANDO CARACTERÍSTICAS BIOMÉTRICAS

Biometria, é uma palavra de origem grega que significa a medida da vida: bios – vida, metros – medida. Esta palavra também denomina o ramo da ciência que estuda as características métricas dos seres vivos, nomeadamente as características físicas associadas como impressão digital, padrão da íris, a forma da mão, etc. estuda as características fisiológicas relacionadas com o funcionamento do organismo como pressão arterial, batimento cardíaco, etc. e as características comportamentais que estão relacionadas com a forma como o individuo age, são exemplo destas características a forma de andar, a forma de escrever, a forma de digitar, etc. Figura 56.

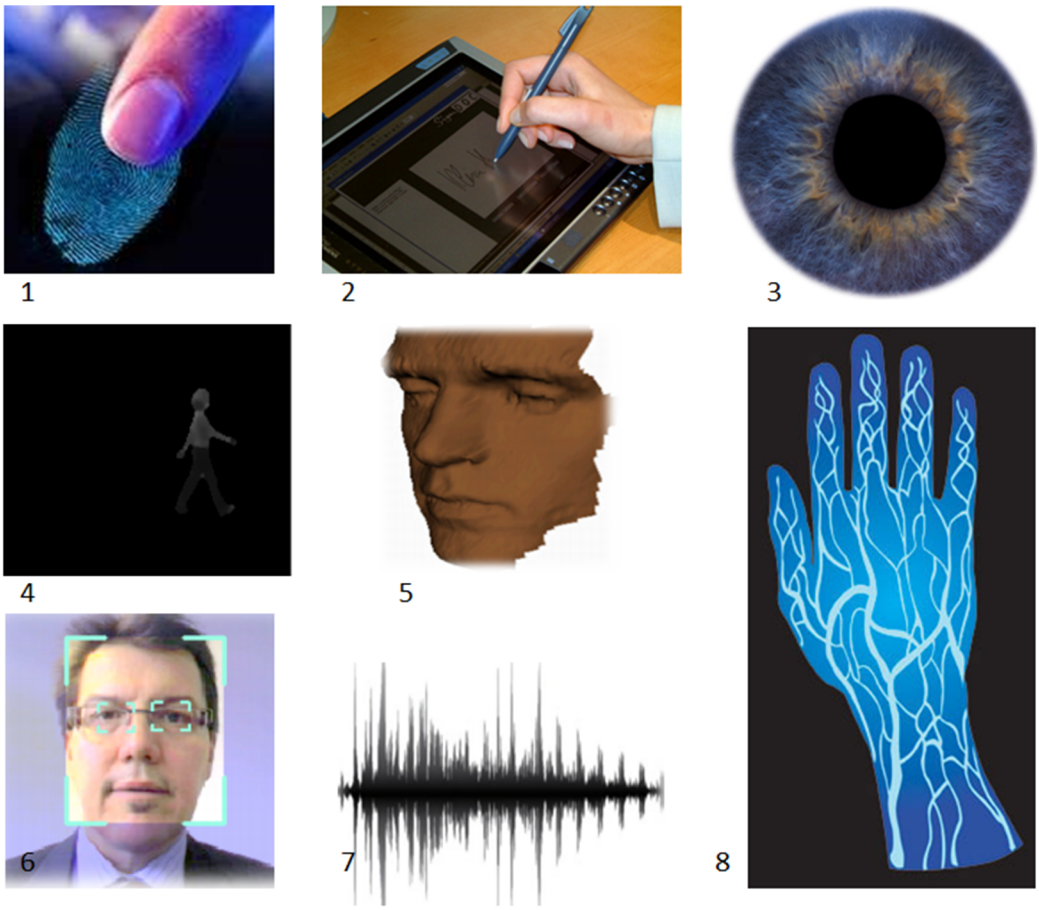


Figura 56 – Exemplos de características biométricas.

Nos últimos anos tem-se associado a palavra "biometria" a sistemas eletrónicos que usam as particularidades individuais das pessoas em processos de identificação e autenticação. Neste caso estamos a tratar, [62], de um caso particular da biometria – a antropometria, em que a identificação/autenticação antropométrica pretende distinguir as pessoas como indivíduos.

O interesse na identificação de indivíduos por análise de características antropométricas prende-se com vários fatores, nomeadamente, [60] e [63]:

- Universalidade: As características usadas para análise são comuns à grande maiorias dos indivíduos.

- Singularidade: As características usadas para análise ou tem padrões únicos para cada pessoa ou a probabilidade de existirem dois padrões iguais é muito reduzida. Por exemplo a probabilidade de encontrar duas pessoas com o mesmo padrão de impressão digital é de 1 em $1.9x10^{15}$.

- Permanência: Existem vários parâmetros biométricos cuja variação ao longo da vida do individuo é nula ou insignificante, e por isso podem ser usados como mecanismo de identificação válidos mesmo considerando períodos de tempo bastante latos.

- Mensurabilidade: Existem várias características antropométricas cujo grau de possibilidade de medida quantitativa é adequado ao uso dos processos de identificação.

- Desempenho: Existem vários parâmetros biométricos que permitem a medição pelas tecnologias existentes de forma precisa e expedita.

- Aceitabilidade: Existem vários parâmetros biométricos que podem ser medidos com o consentimento generalizado das pessoas.

- Proteção: Este fator refere-se à dificuldade de simulação perante o sistema que está na presença de uma característica específica, existem várias características difíceis de reproduzir ou simular.

Na Tabela 4 é apresentado um resumo comparativo de algumas tecnologias biométricas na vertente dos requisitos que pode constituir o ponto de partida para a seleção da tecnológica para uma implementação em particular.

Tabela 4 – Comparação das tecnologias biométricas quanto aos requisitos, adaptado de [60] e [63].

	Universalidade	Singularidade	Permanência	Mensurabilidade	Desempenho	Aceitabilidade	Proteção
Face	Alto	Baixo	Médio	Alto	Baixo	Alto	Baixo
Impressão Digital	Médio	Alto	Alto	Médio	Alto	Médio	Médio
Geometria da mão	Alto	Alto	Alto	Alto	Alto	Médio	Médio
Vascular	Médio	Médio	Médio	Médio	Médio	Médio	Alto
Íris	Alto	Alto	Alto	Médio	Alto	Baixo	Alto
Retinia	Alto	Alto	Alto	Baixo	Alto	Baixo	Alto
Assinatura	Baixo	Baixo	Baixo	Alto	Baixo	Alto	Baixo
Voz	Médio	Baixo	Baixo	Médio	Baixo	Alto	Baixo

Um sistema biométrico é na sua essência um mecanismo de reconhecimento de padrões. O sistema obtém um padrão distintivo de uma vertente antropométrica e guarda-o para posteriormente, por comparação, identificar um individuo. Um dos fatores para seleção do tipo de característica a usar prende-se com a quantidade, qualidade e reprodutibilidade das informações a extrair e estas questões estão intimamente ligadas o nível de segurança pretendido, com tecnologia a usar e por consequência o investimento necessário.

O processo de identificação biométrica nomeadamente, o processo implementado nos sistemas de controlo de acesso desenvolve-se em várias fases [60] e [64]:

- Na primeira fase do processo, o sistema não "conhece" o individuo. A pessoa a identificar, apresenta a sua característica distintiva ao dispositivo de leitura que procede à aquisição de dados, gerando uma representação digital dessa característica em formato de foto, vídeo, áudio, matrizes de valores, etc. Esta é uma fase delicada do processo porque o resultado da leitura é fortemente dependente das condições físicas e psicológicas do individuo, das condições da envolvente ambiental e da tecnologia usada para efetuar a medição. Alterações nestes fatores influenciam a aquisição da informação e pode inviabilizar o processo de identificação.

- Com a representação inicial da característica biométrica usam-se algoritmos adaptados ao tipo de leitura efetuada para extrair a informação representativa da característica, obtendo-se sequências de números chamados de "padrão". Esses algoritmos efetuam primeiro, uma avaliação da qualidade da leitura efetuada, depois realizam operações de melhoria de dados, seguindo-se das operações de segmentação que retiram dos dados iniciais apenas a informação relevante para o processo. O resultado final – o padrão, é uma forma compacta e segura de representação do traço biométrico, compacta porque normalmente ocupa menos de 1Kbyte e segura porque do padrão não é possível reverter o processo para obter o traço biométrico do individuo. Na Figura 57 é mostrado um exemplo de geração do padrão da geometria da mão, neste exemplo após a aquisição de uma imagem (de forma controlada no posicionamento da mão), são efetuadas várias medições para compor uma matriz de valores representativos da identificação do individuo.

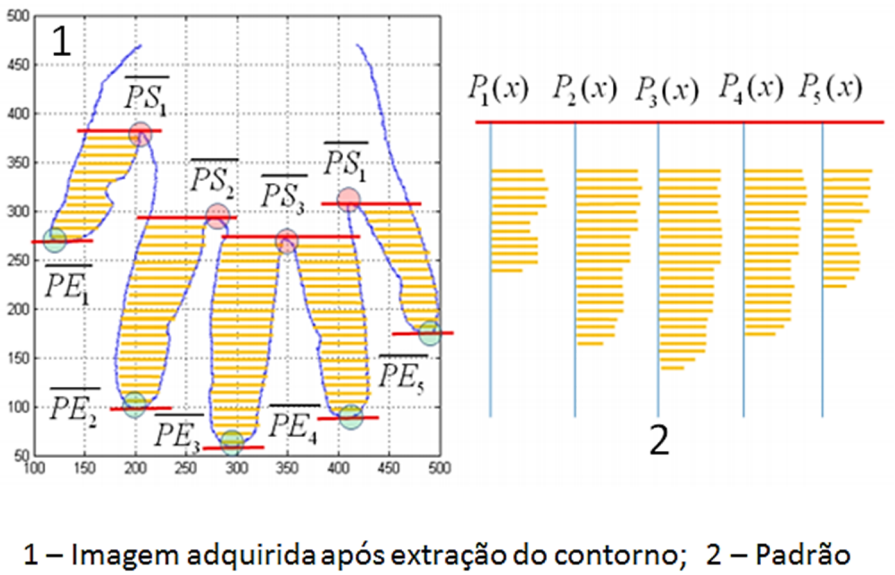


Figura 57 – Geração do padrão da geometria da mão, [65].

- A fase seguinte do processo é o armazenamento do padrão, Figura 58, esta operação apresenta alguns desafios, um deles relaciona-se com a vertente confidencial da informação porque se o padrão é apenas uma sequência de números sem significado, a associação de um padrão a uma pessoa reveste-se já de um cariz de informação privada. A outra vertente prende-se com o armazenamento da informação e neste fator entram todas as considerações dos recursos de

armazenamento como a capacidade, a segurança, a gestão, etc. Normalmente os padrões biométricos são armazenados em três suportes distintos, [60]:

- No dispositivo de leitura, nesta solução o padrão das pessoas autorizadas são armazenadas em cada dispositivo e não há associação com as pessoas, este processo funciona pelo método de identificação, em que é feita a comparação da característica apresentada no momento de leitura, com todas as existentes no leitor. Esta solução é a de implementação mais simples e garante a confidencialidade da informação, mas é pouco flexível e pouco eficiente e por isso, é apenas usada em instalações de pequena dimensão.

- Numa base de dados central, é a solução mais usada em sistemas de maior porte, mas requere repositórios de informação com características de segurança e de capacidade de armazenamento.

- Num cartão, esta forma de armazenamento de dados biométricos é de implementação simples e tem duas vantagens, a primeira é o facto de o padrão estar na posse do dono não dependendo de terceiros, a segunda vantagem é que não necessita de investimento em sistemas complexos de armazenamento, a desvantagem mais significativa desta solução é o custo do cartão, no entanto tem ainda a vantagem de poder-se usar um cartão com o nível tecnológico de encriptação adaptado à solução global. Na terminologia inglesa, este método de armazenamento é conhecido por SOC - *Storage On Card*.

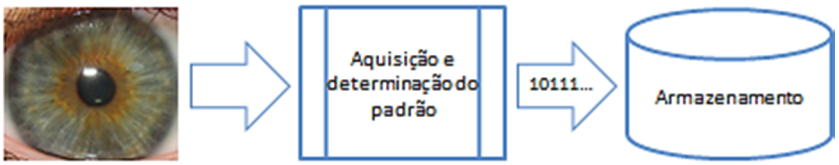


Figura 58 – Processo de armazenamento do padrão biométrico.

- A fase final do processo com identificação biométrica, Figura 59, é a fase operacional de uso. Nesta fase d e cada vez que a pessoa se apresenta perante o sistema é feita uma comparação entre o padrão lido e o padrão armazenado que é

usado como referencia. O grau de similaridade entre os padrões é comparado com o limiar de decisão de onde se obtém o resultado positivo ou negativo da identificação. Neste processo há que ter em conta os seguintes pontos:

- Duas leituras da mesma pessoa "nunca" têm resultados iguais.
- O sistema de comparação apenas indica qual o grau de similaridade dos padrões, não dizem se o sistema está na presença da pessoa ou não.
- O limiar do nível de similaridade é determinado na configuração do sistema e na verdade é uma métrica do nível de segurança que está a ser usado.

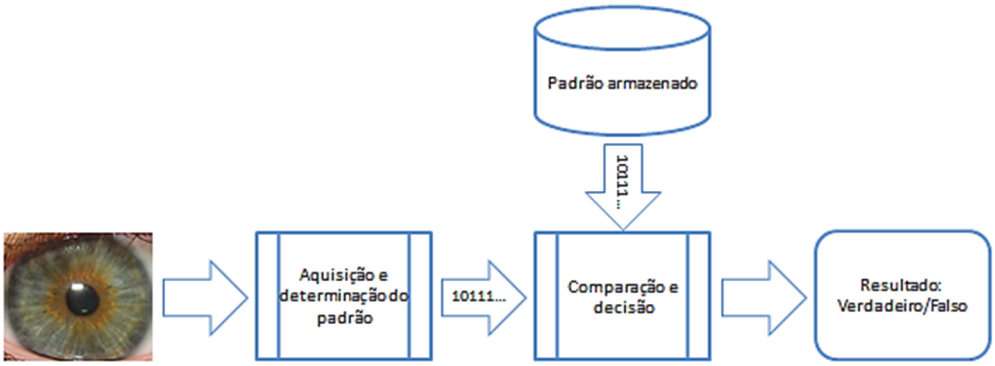


Figura 59 – Processo de identificação biométrica.

Um exemplo de uso de leituras biométricas, nos sistemas de controlo de acessos em que o padrão de comparação está na posse do portador, são os sistemas que usam identificação com cartão RFID e autenticação por biometria. Estas implementações usam normalmente, no mesmo cartão RFID duas tecnologias, uma a funcionar na frequência dos 125KHz e outra a funcionar na frequência dos 13.56MHz. A primeira transmite ao sistema a identificação da pessoa, normalmente através do número de série do cartão, a pessoa apresenta o cartão ao leitor de cartões e o sistema associa número do cartão a uma pessoa e às respetivas permissões. De seguida a pessoa apresenta a sua característica biométrica ao sistema que faz a determinação do padrão e compara-o com ao informação guardada no mesmo cartão RFID mas nos recursos que comunicam na frequência de 13.56MHz. A permissão de acesso é determinada pela comparação da operação biométrica de autenticação cruzada com as permissões obtidas pelo processo de identificação.

A escolha do uso de uma tecnologia biométrica impacta diretamente com dois fatores fundamentais: o nível de segurança pretendido e o nível de investimento possível. Para se poder comparar tecnologias e soluções dos sistemas, consideram-se vários fatores. Na vertente estritamente relacionada com a leitura e o grau de precisão os fatores mais relevantes são os seguintes, [60], [71]:

- FAR – *False Acceptance Rate*: taxa de falsos positivos, taxa que reflete a probabilidade de um utilizador não autorizado ser aceite.

$$FAR(\%) = \frac{Total\ de\ falsas\ aceitações}{Total\ de\ tentativas\ falsas} \quad (1)$$

- FRR – *False Rejection Rate*: taxa de falsas rejeições apresenta a probabilidade de um utilizador autorizado ser rejeitado.

$$FRR(\%) = \frac{Total\ de\ falsas\ rejeições}{Total\ de\ tentativas\ verdadeiras} \quad (2)$$

- EER – *Equal Error Rate*: também conhecido por CER – *Cross Over Rate*, valor em que o erro de falsos positivos é igual o erro de falsas rejeições, Figura 60.

$$ERR(\%) \equiv FAR = FRR \quad (3)$$

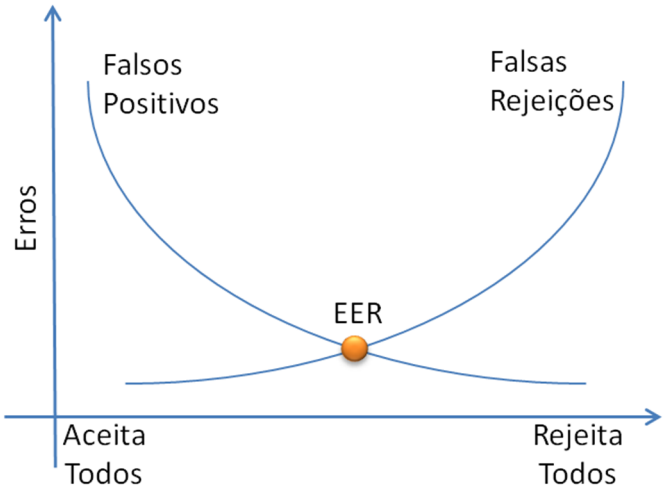


Figura 60 – Gráfico de variação de FAR e FRR.

- FTC – *Failure to Capture Rate*: representa a probabilidade de não conseguir uma aquisição quando a característica biométrica é apresentada corretamente.

- FER – *Failure to Enroll Rate*: taxa de rejeições devido à má qualidade de leitura.

- Desempenho: a avaliação do desempenho global do sistema para ser usado como parâmetro comparativo entre várias soluções. O desempenho é normalmente definido pela relação das taxas de erro consideradas importantes para o processo como por exemplo as incluídas na expressão (4).

$$Desempenho(\%) = 100 - \frac{FAR + FFR + FTC + FER}{4} \quad (4)$$

A definição do nível de limiar de decisão, para cada característica biométrica, deve ser efetuada considerando o valor EER do sistema e atribuindo uma maior ponderação na minimização dos erros FAR e detrimento do aumento das falsas rejeições, por exemplo, nos sistemas de pagamento automático e nos sistemas de controlo de acessos de alta segurança não são admitidos erros do tipo falsos positivos.

Normalmente o balanceamento dos erros FAR e FRR é analisado nas curvas ROC – *Receiver Operations Characteristic*, [71], Figura 61, que são uma representação gráfica dos erros FAR e FRR, um em cada eixo. Do ponto de vista de comparação entre sistemas a posição relativa das curvas permite selecionar o sistema com melhor compromisso, sendo que quanto mais próxima estiver a curva da origem do gráfico melhor é o sistema.

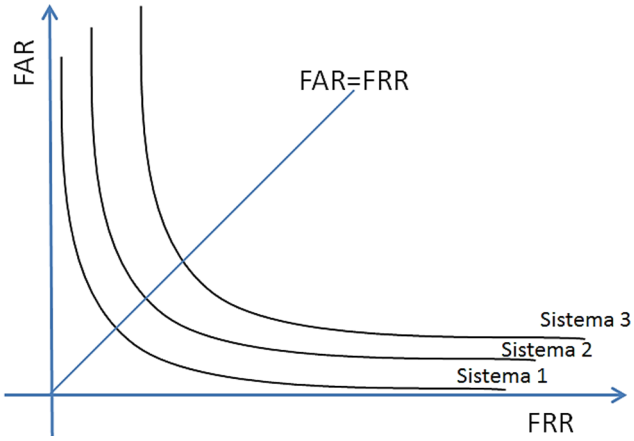


Figura 61 – Curvas ROC – *Receiver Operations Characteristic* de um sistema biométrico.

Na Tabela 5, são apresentados exemplos de taxas de erro de medidas antropométricas efetuadas na mão considerando diferentes tipos de medidas: geometria da mão, análise vascular, impressão da palma da mão. Desta informação verifica-se que quanto mais simples é o processo de medição maior é a taxa de erros, como no caso da analise de contorno da mão. Em sentido contrário quanto mais sofisticado é a mediação, menor é a

taxa de erros como no caso da medição de distância entre pontos de uma imagem de veias e artérias. Para encontrar um compromisso entre as potencialidades da medição, o custo e os resultados pode-se fazer depender o resultado final da identificação de mais de uma característica de medição mais simples, como no exemplo em que se efetua a analise do contorno da mão e da impressão digital da palma.

Tabela 5 – Comparação de tecnologias biométricas, adaptado de [65].

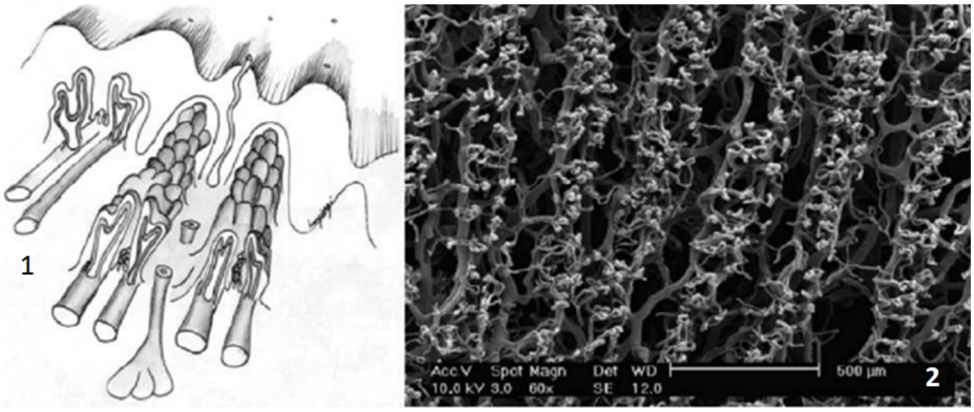
Característica	Medição	Falsos positivos (%)	Falsas Rejeições (%)
Geometria da mão	Contorno	1	6
Análise vascular	Distância entre pontos	0	0,9
Geometria da mão e impressão da palma	Contorno e impressão da palma	0	1.41

Cada tecnologia de identificação biométrica tem características diferentes que se podem usar dependendo do objetivo da aplicação. As mais comuns em sistemas de controlo de acessos são as de reconhecimento do padrão da impressão digital que do ponto de vista de implementação é a mais económica, mas que apresenta algumas limitações, e a leitura de reconhecimento do padrão de íris, que apresenta características mais interessantes do ponto de vista de segurança mas que tem uma implementação mais dispendiosa e uma aceitação por parte dos utilizadores mais difícil. O uso generalizado de dispositivos eletrónicos de uso pessoal como *smartphone* e *tables* tem conduzido a várias soluções biométricas para desbloqueio do ecrã como por exemplo o reconhecimento facial ou de ritmos de digitação. Nos subcapítulos seguintes são apresentados detalhes de implementação e características das tecnologias mais usadas.

2.5.1. RECONHECIMENTO BASEADO NA IMPRESSÃO DIGITAL

A impressão digital é um conjunto de zonas altas e baixas na superfície da pele dos dedos que se formam durante os primeiros sete meses do desenvolvimento do feto devido aos

capilares sanguíneos que existem nas zonas das mais internas da pele e que provocam saliências na parte exterior, Figura 62.



1 – Representação esquemática dos vasos sanguíneos sob a derme, que formam as impressões digitais;
2 – fotografia dos capilares de um dedo que formam a impressão digital, por Microscopia Eletrónica de Varrimento.

Figura 62 – Origem da impressão digital, [62].

Estes padrões ondulatórios da impressão digital apresentam universalidade média e unicidade e permanência alta, estes fatores aliados à facilidade de leitura elevam a impressão digital a um dos métodos biométricos mais usados nos sistemas de identificação automáticos.

As impressões digitais apresentam alguns padrões particulares, denominados "minúcias" que na Figura 63 são mostrados alguns exemplos. As minucias são zonas de fim, interceção ou cruzamento das linhas da impressão ou formatos particulares e que são usados para o processo de criação do padrão da característica biométrica. Um dos problemas das impressões digitais prende-se com sua alteração devido a ações externas sobre a pele, como cortes nos dedos ou desgaste provocado por abrasão que podem dificultar a leitura.

1 – Terminação de crista; 2- Bifurcação; 3 – Lago; 4 – Crista independente; 5 – Ponto ou ilha; 6 – Esporão; 7 - Cruzamento

Figura 63 – Minúcias da impressão digital, adaptado de [73].

Processo de obtenção do padrão por impressão digital funciona em quatro etapas, como mostrado graficamente na Figura 64. O processo inicia-se com a aquisição da imagem, seguem-se operações de filtragem e melhoramentos de imagem, depois executam-se operações de isolamento das linhas da impressão digital, a ultima etapa identifica as minucias das linhas e converte essa informação numa representação matemática.

Figura 64 – Determinação do padrão da impressão digital, adaptado de [74].

Por sua vez o processo de autenticação usando impressão digital também é efetuado em quatro etapas mostrado graficamente na Figura 65. O processo começa com a leitura da impressão apresentada, segue-se o carregamento do padrão armazenado, na etapa seguinte tenta-se efetuar um alinhamento das minucias dos dois padrões e depois determina-se o respetivo grau de similaridade.

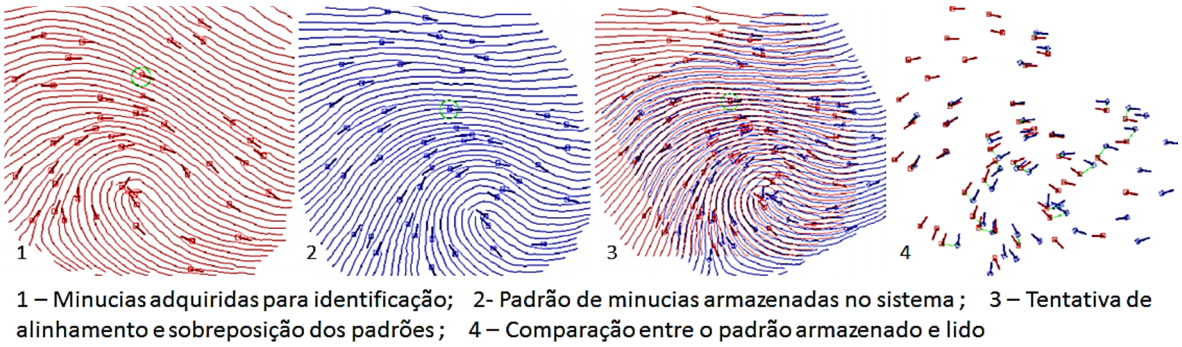


Figura 65 – Processo de autenticação usando impressão digital, adaptado de [74].

2.5.1.1. LEITORES DE IMPRESSÃO DIGITAL

Quanto ao processo de leitura os detetores de impressão digital podem são agrupados em duas categorias: sensores de toque em que o dedo é colocado sobre uma superfície e fica na

mesma posição enquanto a aquisição é efetuada. E sensores de passagem, estes sensores tem uma zona de toque estreita e o dedo tem de ser deslocado sobre essa superfície. Os leitores de impressão digital normalmente instalados nos computadores portáteis para fazer a autenticação do utilizador são do tipo de passagem, Figura 66.

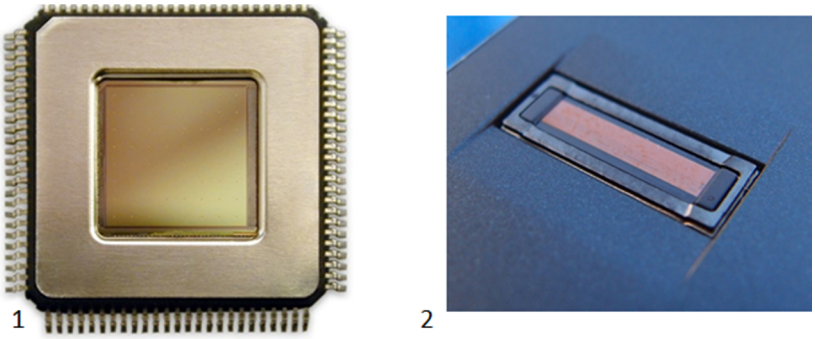


Figura 66 – Leitores de impressão digital: de toque e de passagem.

A leitura da impressão digital é uma operação bastante difundida e existem diversos métodos de a efetuar. Os leitores de impressão digital quanto à tecnologia usada para aquisição da imagem podem ser classificados em três grupos [72]: os detetores óticos, que usam a luz como instrumento de leitura, os detetores de estado sólido que usam efeitos elétricos para a obtenção da imagem da impressão digital e os detetores ultrassónicos e radiofrequência, que usam a projeção de ondas a altas frequências e constroem a imagem da impressão através da reflecção das mesmas ondas.

Detetores óticos

A captura de imagem da impressão digital usando detetores óticos recorre a uma fonte de luz e a um sensor de luminosidade. Os leitores óticos mais usado são os do tipo, [72]:

- Detetores FTIR – *Frustrated Total Internal Reflection*, Figura 67: nestes detetores a superfície de contato é a face de um prisma. O dedo é posicionado na superfície de contacto que faz com que as linhas altas da impressão digital toquem na face do prisma mas linhas baixas não tocam o que cria entre a face do prisma e a superfície

dedo uma bolsa de ar. Noutra face do prisma é injetada uma luz difusa proveniente da fonte luminosa do leitor. A luz incide na face do prisma onde está o dedo e nas bolsas de ar a luz é refletida com o mesmo angulo da incidência, nas zonas onde as linhas altas da impressão digital estão em contacto com o prisma, a luz é dispersa em direções aleatórias. Na zona frontal à terceira face do prisma está um sistema de lentes e um sensor ótico para captura da luz refletida no dedo, desta forma, a linhas baixas da impressão digital apresentam-se ao sensor luminosas e as linhas altas apresentam-se como zonas escuras.

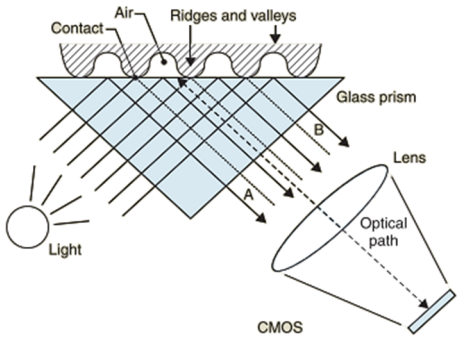


Figura 67 – Leitores de impressão digital ótico do tipo FTIR, [72].

- Detetores de imagem direta: nestes detetores a fonte luminosa e o sensor de luminosidade estão focados diretamente na superfície de contacto onde o dedo está posicionado, Figura 68. As gamas mais sofisticadas de leitores deste tipo usam mais de uma aquisição de imagem variando as condições de iluminação como comprimento de onda, orientação ou polarização, para obtenção de imagens de maior qualidade.

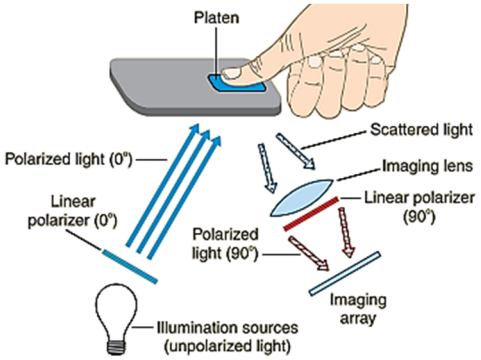


Figura 68 – Leitores de impressão digital ótico do tipo imagem direta, [72].

Os sensores óticos possibilitam obter boas imagens da impressão digital e as várias técnicas para o fazer permitem adaptar a qualidade da imagem à solução pretendida. Os maiores inconvenientes do uso desta tecnologia são: os dispositivos apresentarem maiores dimensões e o facto de alterações da pele dos dedos degradar significativamente a qualidade da imagem, além destes detetores poderem ser "enganados" como imitações da impressão digital.

Detetores ultrassónicos

Os detetores ultrassónicos usam os princípios da ultrassonografia médica para criação da imagem da impressão digital, o sistema é composto por um emissor de ondas ultrassónicas e um recetor do tipo piezoelétrico[17] para detetar as ondas que são refletidas. As gamas de ondas usadas nestes detetores não são refletidas na camada externa da pele mas na camada interna da derme, ver Figura 62. A maior vantagem deste método de leitura, reside no facto de as imagens adquiridas não serem afetadas pelas imperfeições superficiais nos dedos e serem muito mais difíceis de ludibriar com cópias das impressões digitais. No entanto são as soluções mais dispendiosas e com menor grau de miniaturização.

Detetores de estado sólido

Os detetores de estado sólido, são circuitos integrados que usam uma propriedade de um dedo vivo para gerar um sinal elétrico e criar a imagem da impressão digital. O uso destes detetores resulta em leitores de dimensões mais compactas, de menor consumo de energia e normalmente de menor custo que os leitores dos outros tipos. No entanto, são mais

[17] Efeito piezoelétrico – característica de alguns materiais, nomeadamente cristais e cerâmicas, de gerarem uma tensão elétrica quando são sujeitos a pressões mecânicas.

sensíveis aos pequenos movimentos do dedo que surgem durante a leitura, as superfícies de contacto são normalmente de menor duração que o vidro usado nos leitores óticos e podem ser afetados por descargas elétricas provenientes dos dedos que vão ler.

Os tipos de sensores de estado sólido para leitura da imagem da impressão digital são, [72], [74].

- Sensor capacitivo: Estes sensores são constituídos por um circuito integrado com uma matriz de placas de micro-condensadores, quando o dedo é colocado em contacto com o circuito integrado comporta-se com a segunda placa dos condensadores, Figura 69, criando pequenas cargas elétricas nas placas da matriz cuja quantidade é dependente da distância da superfície do dedo à superfície do circuito integrado. As linhas altas e baixas da impressão digital induzem diferentes quantidades de carga nas placas dos micro-condensadores, que são usadas para a criação da imagem da impressão digital. Por exemplo os sensores de estado sólido capacitivos da empresa Veridicom, são matrizes de 300x300 condensadores separados por 50 micron.

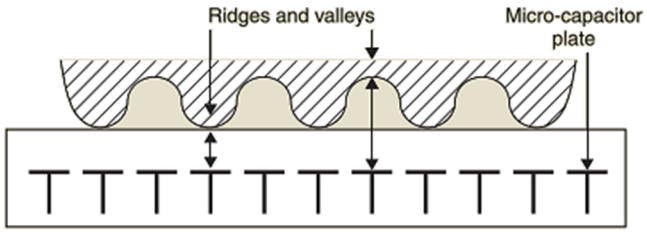


Figura 69 – Leitores de impressão digital de estado sólido capacitivo, [72].

- Sensor de pressão: nos sensores de pressão a zona de contacto do dedo com o circuito integrado sensor é constituída por matrizes de material piezoelétrico, os diferentes níveis de pressão criados pelas linhas altas e baixas da impressão digitam levam à geração de níveis de tensão diferentes em diferentes locais e esta informação é usada para a construção da imagem.

- Sensor de temperatura: nestes sensores a matriz existente na superfície de contato é sensível à temperatura e a construção da imagem da impressão digital é efetuada

com base no diferencial de temperatura dos respetivos sensores, esses diferencias representam a variação das linhas altas para as linhas baixas da impressão digital.

Nos últimos tempos, o desenvolvimento dos leitores de impressão digital teve um novo impulso para satisfazer necessidades controlo de acessos lógicos em dispositivos portáteis como *smartphones* e que tem dado origem a registos de novas patentes, por exemplo, a Apple na versão 5S do iPhone, integrou um leitor de impressão digital do tipo capacitivo no botão de comando do dispositivo, Figura 70. O leitor tem 170 mícron[18] de espessura e resolução de 500dpi[19], [70].

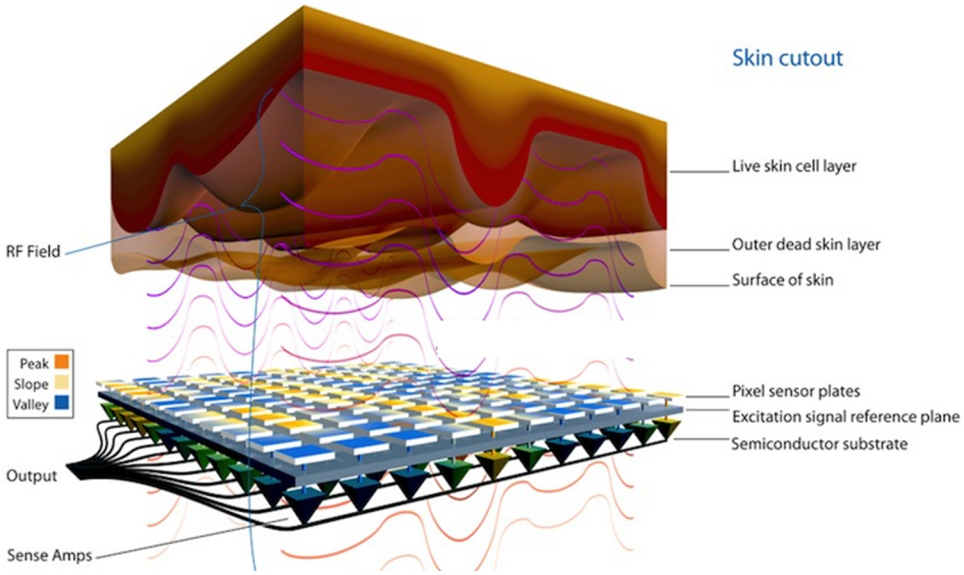


Figura 70 – Sensor de impressão digital capacitivo usado no iPhone 5S.

[18] Mícron = $1x10^{-6}$, relativo à unidade de medida linear micrómetro - μm.

[19] dpi - *Dots per inch*, medida linear do número de pontos por polegada que constituem uma imagem. A densidade de pontos da imagem define a sua resolução e está associada diretamente com à qualidade. Quando se trata de imagens em ecrãs a grandeza de medida, chama-se ppi - *pixels per inch*, relativa aos pontos que constituem a imagem no ecrã denominados de *pixéis*.

2.5.1.2. EXEMPLOS DE DETETORES DE IMPRESSÃO DIGITAL USADOS EM CONTROLO DE ACESSOS

O mercado de sistemas de controlo de acessos disponibilizam diversas soluções de identificação e/ou autenticação com leitura de impressão digital. O exemplo apresentado na imagem da esquerda da Figura 71, mostra um leitor de impressão digital integrado na fechadura em que o conjunto funciona de forma autónoma. Na imagem do meio o leitor de impressão digital está integrado no puxador da porta, este sensor está ligado a uma unidade de controlo que gere os acesso da porta, o fabricante, [75], para esta solução apresenta como taxa de erro de falsos positivos FAR = $1x10^{-6}$ e de falsas rejeições de FFR = $1,4x10^{-2}$.

Figura 71 – Exemplos de leitores de impressão digital em sistemas de controlo de acessos.

O exemplo apresentado na direita da Figura 71 mostra um leitor típico de impressão digital usado em sistemas de controlo de acesso de maior escala, estes leitores são normalmente instalados suporte fixos como paredes e tem dispositivos de vigia de alterações da instalação física. Este tipo de leitores operam em conjunto com unidades de controlo e devido à necessidade de compatibilização com diversos fabricantes os leitores normalmente possuem várias formas de comunicar com a unidade de controlo, o exemplo apresentado na figura é o RKLB575 da HID, este leitor possui interfaces, [76], *Wiegand*, RS232, RS485 e USB.

Normalmente os leitores que têm sensores de impressão digital permitem autenticação multi-fator, no exemplo apresentado do RKLB575, permite fazer autenticação usando

combinações dos seguintes métodos: impressão digital, código PIN e cartão de proximidade, 13.56MHz. O sensor de impressão digital deste leitor captura imagens de 18x22mm e com resolução de 500dpi. Na captura da impressão digital o fabricante fornece os valores característicos mostrados na Tabela 6

Tabela 6 – Parâmetros característicos do leitor de impressão digital HID RKLB575, [76].

Parâmetro	Valor
Taxa de falsos positivos - FAR	< 0.01%
Taxa de rejeições - FRR	< 0.01%
Tempo de leitura de cartão	< 0.5S
Tempo de captura de impressão digital	< 2S, típico 1S
Tempo de verificação de uma impressão digital	< 1S

2.5.2. RECONHECIMENTO BASEADO EM PADRÕES DOS OLHOS

Os olhos humanos são órgãos internos com um nível de proteção elevado (ao contrário das impressões digitais) e de fácil observação do exterior que contem características de singularidade que podem ser usadas para identificação de indivíduos. A Figura 72 mostra um diagrama dos constituintes de um olho.

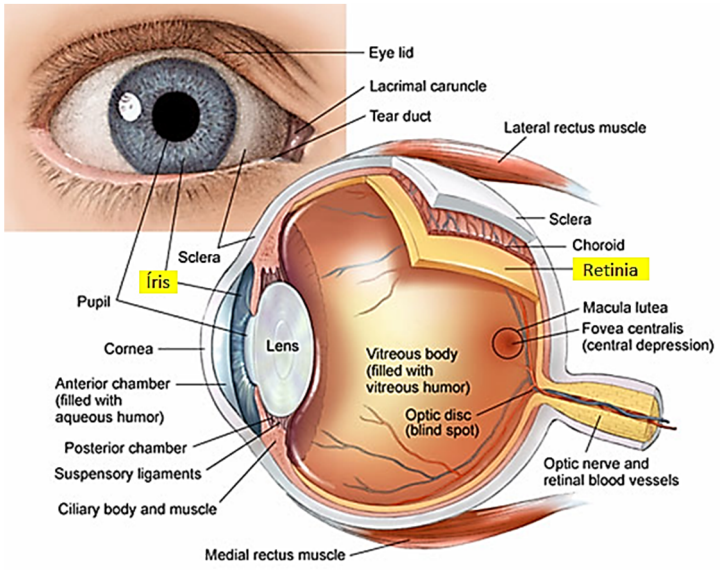


Figura 72 – Diagrama do olho humano, adaptado de [77].

A técnica biométrica mais usual que tem por base as características do olho faz reconhecimento do padrão da íris. A íris é uma estrutura circular existente na face externa do olho, que fica entre a parte branca - esclerótica e o círculo preto – pupila. A íris tem por função controlar o diâmetro da pupila e por conseguinte a quantidade de luz que entra no olho e é composta por camadas que contem células pigmentadas, vasos sanguíneos e músculos criando sulcos e estrias visíveis do exterior, Figura 73.

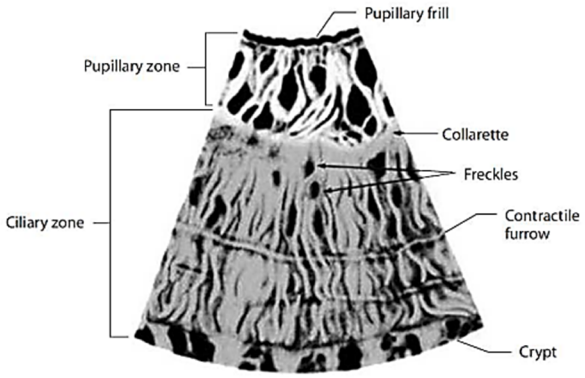


Figura 73 – Detalhes da íris [62].

O padrão visível da íris é independente da sua cor e tal como o padrão da impressão digital forma-se aleatoriamente durante a gestação e ficam definidos nos primeiros anos tornando-se muito estáveis ao longo da vida do indivíduo, [67]. As poucas situações que podem interferir com o padrão a íris são operações cirúrgicas, neste caso provoca-se uma descontinuidade na aceitação do padrão de referência antigo e tem de ser fazer feita uma nova aquisição para armazenamento, do ponto de vista de segurança, estes casos não são preocupantes porque vão provocar rejeições a pessoas que tem acesso e o nível de segurança não é afetado.

Enquanto nos leitores de impressões digitais existem soluções que usam diversas técnicas de aquisição de imagem, os leitores de íris usam apenas a técnica: a captura fotográfica. Esta característica dos sistemas pode levantar a possibilidade de serem ludibriados com fotos de íris de alta resolução. Um forma de contornar esta limitação é o sensor alem de capturar a imagem da íris verificar a oscilação do diâmetro da íris à taxa de 0.5Hz que é uma característica típica do olho vivo, [62].

O processo normal de reconhecimento de íris é composto por cinco fases, [66], [72]:

- A aquisição da imagem.
- A localização e isolamento da imagem da íris, retirando todos os elementos estranhos como pestanas, pálpebras, etc. Figura 74.

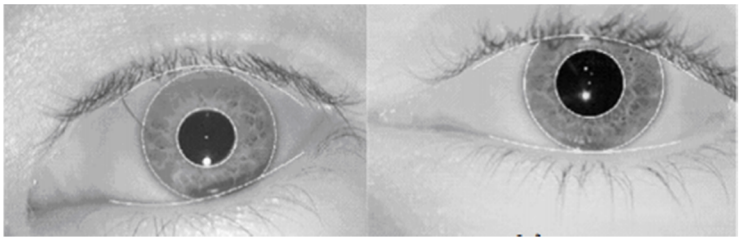


Figura 74 – Localização da imagem da íris, [69].

- Normalização: este processo transforma a imagem da íris do formato de círculo furado, numa imagem retangular em dimensões cartesianas, Figura 75.2.
- Codificação: o processo de codificação depende do sistema usado mas tipicamente executa conjuntos de filtros sobre a imagem normalizada para obter uma sequência binária do código da imagem da íris. Na Figura 75.3 é mostrado uma representação gráfica do código de uma íris.
- Verificação: finalmente duas representações binárias, a armazenada e a adquirida, podem ser comparadas para determinar o grau de similaridade.

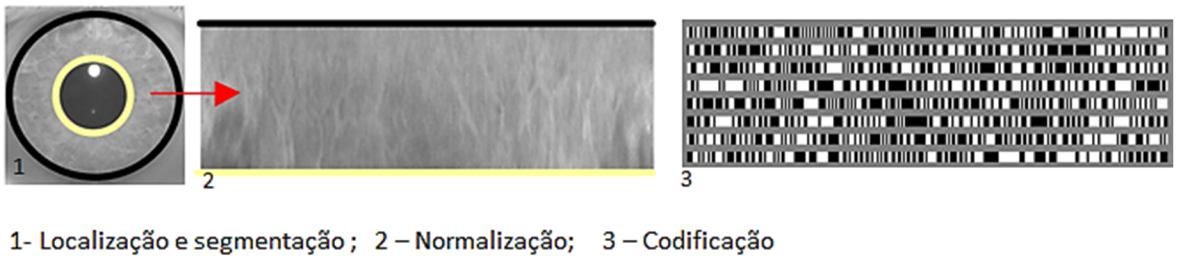


Figura 75 – Fases do tratamento da imagem da íris, adaptado de [66], [68], [69].

Quando à distância de leitura existem três tipos de leitores de íris, [72], Figura 76:

- Os leitores de curta distância em que para efetuar a aquisição é necessário posicionar o intercílio num ponto fixo para colocar os olhos a distâncias de 2 a 5cm das camaras de aquisição.

- Os leitores de média distância, que permitem leituras de 50 a 100cm. Neste caso o dispositivo de leitura tem um ou dois espelhos. E no processo de leitura o utilizador usa os espelhos para olhar para os seus próprios olhos fazendo desta fora o alinhamento com as camaras de aquisição. Este é o tipo de leitores normalmente usado nos sistemas de controlo de acessos.

- Leitores de grandes distâncias, normalmente até dois metros, estes leitores são os menos difundidos. São normalmente constituídos por conjuntos de várias camaras, umas com lentes de grande angular, para fazer a localização da face e dos olhos que controlam outro conjunto de camaras equipadas com lentes *zoom* para fazer a aquisição a íris. Os sistemas mais complexos deste grupo permitem fazer aquisição da íris mesmo com pessoas em movimento.

1- Leitor de curta distancia; 2 – Leitor de média distancia; 3 – Leitor de longa distancia

Figura 76 – Exemplos de leitores de íris.

A norma ISO/IEC19794-6 define os parâmetros mínimos que os dispositivos de captura de imagens de íris devem ter, em que os mais importantes são, [72]:

- Iluminação: os equipamentos de aquisição têm de usar uma fonte de iluminação do olho com uma luz de comprimento de onda perto do infravermelho 700 a 900nm, [72], nesta gama de frequência, a melanina, molécula responsável pela cor dos olhos tem um grau de reflecção praticamente constante, permitindo criar uma imagem do padrão da íris muito mais rica e independente da cor.

- Contraste: a imagem capturada tem de possuir no mínimo uma separação de 70 níveis de cinzento entre a íris e a esclerótica e de 50 níveis de cinzento entre a íris e a pupila.

- Área: a imagem tem de mostrar pelo menos 70% da área da íris.
- Reflexos: os reflexos na imagem devem ser evitados, quer usando filtros na aquisição da imagem que controlando a direção da luz ambiente.
- Tamanho a pupila: O tamanho da pupila deve ser menor que 7mm, tamanhos superiores degradam a qualidade da imagem da íris. Para evitar este problema pode-se adicionar luz visível na fonte de iluminação do sistema para obrigar à contração da pupila e à dilatação da íris.
- A relação sinal ruido deve ser inferior a 40dB.
- Resolução ótica: a resolução ótica deve ser no mínimo de 16.7pixeis/mm.

2.5.2.1. EXEMPLOS DE LEITORES DE ÍRIS USADOS EM CONTROLO DE ACESSOS

Os sistemas de controlo de acessos que requerem níveis de segurança mais elevados justificam o investimento na aquisição de sistemas de autenticação que podem usar o padrão da íris. Os detetores usados nestas soluções são normalmente os de média distância que para o fim que se pretende são um compromisso entre o conforto dos utilizadores e as exigências (custo) das implementações.

1- Panasonic: ET330; 2 – LG: IrisAccess 4000; 3 – IrisID: iCAM7000

Figura 77 – Exemplos de leitores de impressão íris em sistemas de controlo de acessos.

Os fabricantes deste tipo de equipamentos apresentam dispositivos com diversas variantes, para aumentar a possibilidade de integração nos sistemas de controlo de acessos. Na Figura 77, são apresentados alguns desses exemplos. O modelo ET330 da Panasonic é um leitor de íris equipado com duas interfaces de protocolo *Wiegand*, uma de entrada e outra de saída, o objetivo destas interfaces é o leitor poder ser integrado em unidades de controlo de recursos limitados mas que em conjunto com o leitor de íris ofereçam elevados níveis de segurança.

Os exemplos das marcas LG e IrisID são exemplos de leitores que já tem incorporado capacidade de autenticação por múltiplo factor usando cartões RFID ou teclados para introdução de PIN. A possibilidade que estes modelos apresentam de ler cartões RFID, permite o modo de funcionamento em que o padrão de referência da íris armazenado esteja no cartão do utilizador e não no sistema central. Normalmente os leitores desta gama têm interfaces TCP/IP que ligam a sistemas centralizados para se efetuarem operação de configuração, de alarmística ou de verificação de estado de funcionamento.

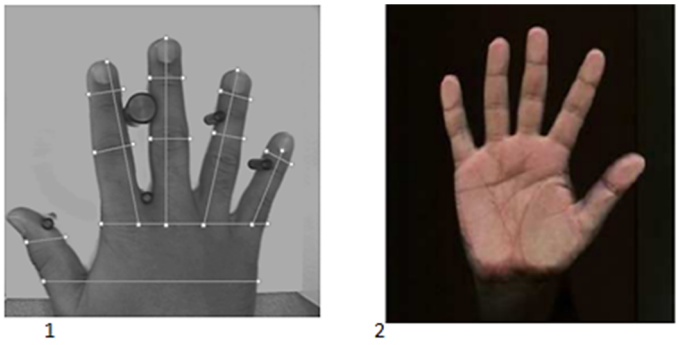
2.5.3. RECONHECIMENTO BASEADO NOUTRAS TECNOLOGIAS BIOMÉTRICAS

Nos sistemas de controlo de acessos as tecnologias biométricas mais comuns são as que usam o padrão da impressão digital e da íris. No entanto dependendo de fatores relacionados com a implementação pretendida ou dos níveis de segurança exigidos, usam-se outras características antropométricas para reconhecimento de indivíduos. Este subcapítulo faz uma breve descrição de algumas dessas técnicas

Geometria da mão

Dentro das técnicas que analisam características físicas, a geometria da mão permite um conjunto de métricas que aumentando a complexidade do sistema de leitura possibilita adequar o nível de erros de decisão às necessidades da implementação, mantendo o grau de aceitabilidade dos utilizadores. As técnicas de reconhecimento que usam a geometria da mão baseiam o resultado da identificação na da medida de um ou mais dos seguintes

parâmetros, [61]: contorno da mão, proporções da estrutura da mão, comprimento, largura, distâncias características da pele, etc.



1- Aquisição da geometria da mão usando posicionadores;
2 – Aquisição da geometria da mão livre

Figura 78 – Exemplos de formas de aquisição da geometria da mão, adaptado de [65].

A análise da geometria da mão é um método que é independente da limpeza da pele e do nível de humidade da mão. O leitor pode estar equipado com um conjunto de pontos fixos, para o correto posicionamento da mão, Figura 78, e de uma fonte de luz para que a imagem seja criada sob condições controladas. A Figura 57 mostra um exemplo de definição do padrão da geometria da mão baseado em medições de distâncias.

Para que o sistema não seja iludido por imagens ou modelos de mão, alguns leitores, [61] requerem que os dedos se movam numa das fases de leitura ou estão equipados com dispositivos de medição da temperatura ou condutividade da pele.

Os maiores problemas desta técnica prendem-se com a falta de precisão na medição, o custo dos dispositivos e o decorrer da idade dos indivíduos que pode conduzir a alterações de características e a possíveis falsas rejeições.

Reconhecimento facial

O reconhecimento facial é um método biométrico, onde se assiste o maior aumento de utilizações [61] por dois fatores fundamentais, o alvo não necessita de estar em contacto com o leitor e por que o leitor não necessita de requisitos especiais de *hardware*, as

imagens geradas pelas camaras de vídeo de outros sistemas como *webcam* ou camaras de segurança podem ser usadas nos processos de reconhecimento facial.

O reconhecimento facial pode ser efetuado identificando pontos característicos da face como início/fim do olho, pontos característicos da boca, do nariz, etc. O padrão para reconhecimento biométrico é conseguido por medição relativa entre os vários pontos.

O método de análise da imagem pode ser efetuado usando dois abordagens diferentes: 2D e 3D, [80]. A análise 2D usa uma fotografia em que a face olha diretamente de frente para a camara, Figura 79. Na abordagem 3D, é adquirida mais de uma imagem que além do estudo bidimensional, efetua também uma análise considerando a profundidade da imagem 3D, permitindo englobar as curvas do rosto no padrão de reconhecimento o que leva a resultados mais precisos. No entanto, [61], a face não possui tantas características universais e únicas como a impressão digital ou a íris e por isso o nível de confiança do reconhecimento facial é ligeiramente inferior aos outros métodos.

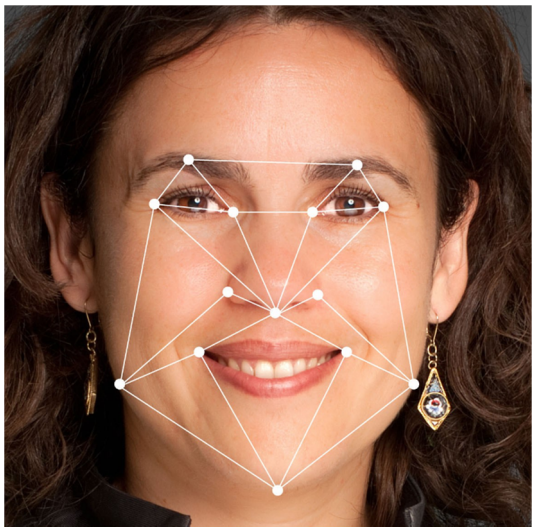


Figura 79 – Exemplos de medições usadas no reconhecimento facial, imagem obtida em [79].

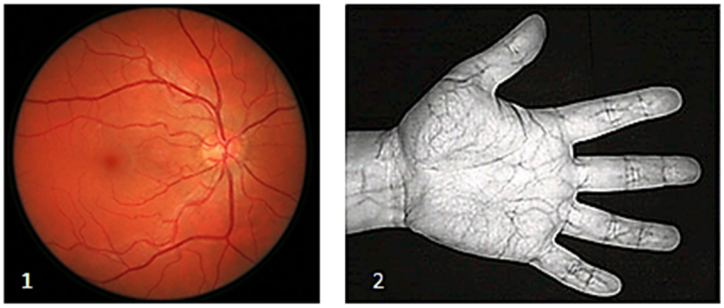
O reconhecimento facial é uma tecnologia que pode ser usada para reconhecimento à distância e existe uma elevada aceitação pública, um dos maiores problemas prende-se com as condições de luminosidade e com o facto das alterações provocadas pela idade poderem dificultar o reconhecimento.

Reconhecimento de padrões vasculares

O padrão formado pelo conjunto de veias e artérias é uma característica do corpo humano de elevado grau de aleatoriedade, [60], de universalidade e de estabilidade. É uma característica protegida do meio exterior, de difícil falsificação e que permite operações de reconhecimento de elevada precisão, por exemplo o terminal de leitura FingerVein da empresa Kimaldi apresenta taxa de falsos rejeições menores que 0.01%, taxas de falsos positivos de 0.0001% e taxas de falha de leitura de 0.03%, [81].

A leitura do padrão vascular é efetuada expondo a parte do corpo a ler a um fonte de luz na gama do espectro próximo do infravermelho, a esta frequência os vasos sanguíneos absorvem toda a radiação e os tecidos envolventes refletem quase toda a radiação tornando-se praticamente translúcidos, permitindo obter imagens em que as veias e artérias apresentam-se em tons escuros.

Nos sistemas de controlo de acessos para identificação usam-se normalmente os padrões da mão, de um único dedo, ou da retinia. Os dedos são zonas da mão com uma grande concentração de terminações nervosas, que nos conferem a sensibilidade que temos, e por isso são zonas de grande densidade de capilares sanguíneos. A retinia por sua vez, Figura 72, é a secção esférica da parte interior do olho onde estão as células fotossensíveis que formam a imagem transportada pela luz que entra pela pupila. Esta zona do olho é fortemente irrigada por sangue que é conduzido por uma complexa e aleatória rede de veias e artérias. Estas partes do corpo contem padrões de vasos sanguíneos muito elaborados e altamente distintivos dos indivíduos e por isso são usados nos reconhecimentos biométricos.



1- Padrão vascular da retinia; 2 – Padrão vascular da mão

Figura 80 – Padrão vascular da retinia e da mão, [82].

Do ponto de vista de aceitabilidade a mão ou o dedo é mais facilmente usado, visto que os equipamentos de medição do olho são algo aparatosos e induzem a sensações de desconforto. A maior desvantagem do uso desta tecnologia é o custo elevado dos leitores.

Reconhecimento em múltiplas análises

Uma forma de melhorar os indicadores característicos de uma técnica é efetuar o reconhecimento com base de múltiplas análises. Este método consiste em usar a mesma característica antropométrica e analisá-la sobre mais de uma vertente. Com este conceito pode-se efetuar implementando várias formas nomeadamente, Figura 81, [83]:

- Característica única / Vários sensores: a mesma característica é medida por mais de um sensor de tecnologia diferente, que geram dados iniciais diferentes. Atualmente está-se a usar esta técnica na aquisição da impressão digital para o Cartão do Cidadão.

- Característica única / Vários tratamentos: neste método usa-se uma aquisição inicial que é tratada de mais de uma forma diferente, por exemplo numa impressão digital pode-se fazer a localização das minucias e uma análise à textura.

- Característica única / Várias aquisições: nesta situação uma característica é medida com mais de uma variante, por exemplo, no reconhecimento facial fazer uma aquisição frontal e outra lateral.

- Característica única / Várias unidades: neste caso são usadas várias aquisições de elementos diferentes, por exemplo no reconhecimento por iris fazer a aquisição dos dois olhos

- Varias características: no limite pode-se combinar mais de uma característica biométrica num único ato de reconhecimento.

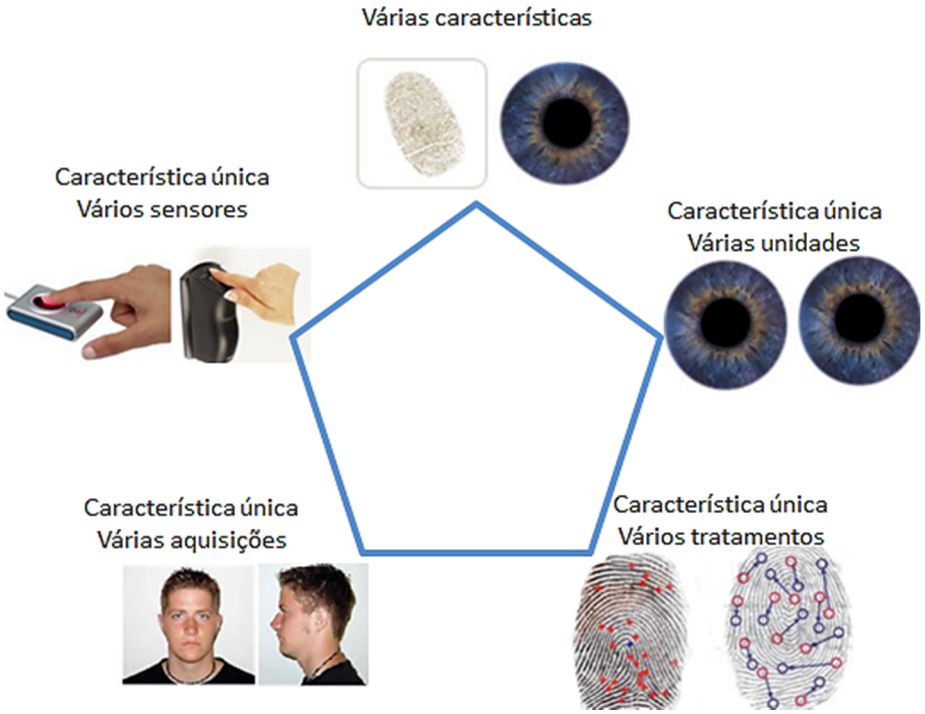


Figura 81 – Reconhecimento por múltiplas análises, adaptado de, [83].

2.5.4. COMPARAÇÃO DE TÉCNICAS DE RECONHECIMENTO BASEADO EM BIOMETRIA

Como em todas as áreas de conhecimento para satisfazer uma necessidade não há uma solução ideal, e também não existem uma técnica de biométrica ideal. Cada uma das metodologias apresenta pontos fortes e limitações a ter em conta. Na Tabela 7 é apresentado um resumo das características mais importantes das tecnologias descritas. Os adjetivos usados na tabela como "baixo" ou "caro" referem-se à comparação entre as soluções que constam na tabela.

Tabela 7 – Comparação entre tecnologia de reconhecimento biométrico.

Impressão digital	
Pontos fortes	**Considerações**
• É a tecnologia mais usada, comercializada desde 1970. • Fiabilidade comprovada.	• Problemas de leitura por danos na pele • Problemas com estado de limpeza da pele • Estabilidade afetada pela idade

• Possibilidade de verificação em vários dedos. • Taxa de falsos positivos: muito baixa. • Taxa de falsas rejeições: muito baixa. • Preço baixo.	• Possibilidade de ludibriado com modelos simples. • Possibilidade de dedos artificiais ativarem impressões latentes.
Íris	
Pontos fortes	**Considerações**
• Fiabilidade comprovada, comercializada desde 1997. • Não necessita de contacto físico com o leitor. • Estabilidade elevada ao longo da vida.	• Uso mais complexo e mais demorado. • Leitores caros. • Pode ser afetado pelo uso de óculos. • Pode apresentar taxas superiores de falsas rejeições. • Pode suscitar alguma resistência de uso.
Geometria da mão	
Pontos fortes	**Considerações**
• É a tecnologia mais usada, comercializada desde 1970. • Fiabilidade comprovada. • Preço médio.	• Estabilidade afetada pela idade. • Possibilidade de ludibriado com modelos simples.
Reconhecimento facial	
Pontos fortes	**Considerações**
• Pode operar sem intervenção do utilizador. • Capaz de fazer identificação à distância • Não necessita de contacto físico com o leitor. • Preço baixo de *hardware*.	• Estabilidade afetada pela idade. • Possibilidade de ludibriado com modelos simples. • Condições de luminosidade e angulo de captura. • Alterações correntes podem afetar a capacidade de identificação, como por exemplo o uso de óculos.

3. PROJETO

No capítulo 1, definiram-se os objetivos do trabalho e as fases que conduzem à sua implementação. Neste capítulo vai ser descrita a execução das duas primeiras fases: o inventário das funcionalidades e o desenvolvimento do projeto que responde a essas necessidades.

3.1. INVENTÁRIO DE FUNCIONALIDADES

A fase de inventário das funcionalidades que se pretendem para a nova plataforma credenciação tem duas vertentes, uma relacionada com a inventariação das funcionalidades dos sistemas, métodos e processos que estão atualmente a ser usadas e se pretendem manter para o futuro. A outra vertente prende-se com a compilação de novas funcionalidades desejáveis pelos administradores e utilizadores de todo o ecossistema de controlo de acessos e credenciação existente no ASC.

3.1.1. INVENTÁRIO DE FUNCIONALIDADES EXISTENTES

O método usado para obter um inventário das funcionalidades existentes, foi recorrer a entrevistas dos colaboradores da ANA que desenvolvem serviços relacionados com o controlo de acesos e a credenciação.

A compilação das funcionalidades foi efetuada por agrupamento em blocos conceptuais que abordam cada uma das facetas em uso. E esses blocos, foram representados através de diagramas. Por exemplo, existe uma atividade para criação de cartões permanentes, como base na informação obtida nas entrevistas efetuadas, desenhou-se um diagrama que descreve a atividade, que descreve o fluxo de informação e a sequência de ações executadas. No Anexo G são apresentadas informações sobre técnicas de representação e técnicas de modelação de conceitos.

Nos próximos subcapítulos apresentam-se os blocos conceptuais identificados que representam as atividades e os intervenientes relacionados com a credenciação no Aeroporto Francisco Sá Carnerio.

3.1.1.1. PROCESSO DE CREDENCIAÇÃO PERMANENTE OU TEMPORÁRIO

Na ANA, o pedido de credenciação permanente de pessoas efetua-se sempre via portal "Cartão do Aeroporto", Figura 82.

Figura 82 – Portal "Cartão do Aeroporto".

Para poder efetuar pedidos de credenciação, as entidades solicitadoras tem de estar registadas no portal, ter definido um representante da empresa e ter definido um meio de pagamento dos custos associados à credenciação. O processo de registo da entidade, Figura 83, passa pela oficialização do pedido e apresentação de documentação ao Gabinete de Segurança, que cria a entidade no portal e no SGO. Após o registo no sistema o representante da empresa pode efetuar pedidos de cartões, para as pessoas com vínculo laboral a essa empresa.

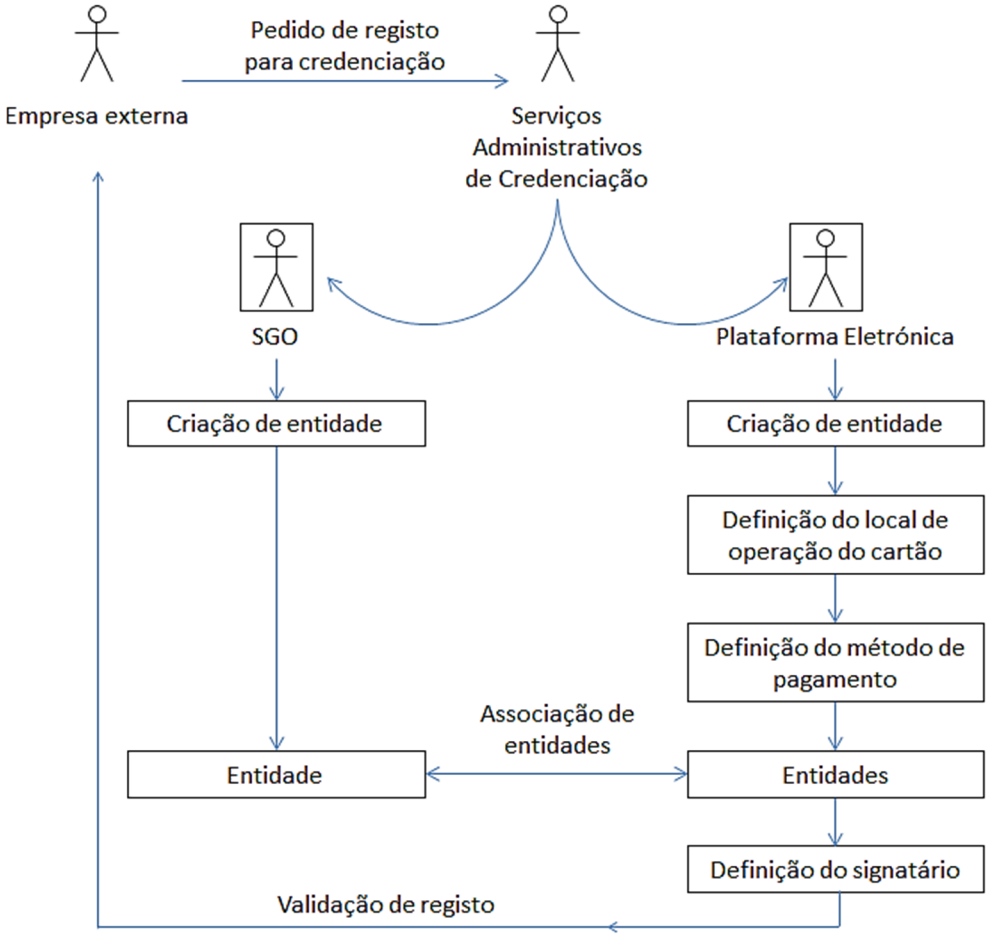


Figura 83 – Processo de credenciação permanente: Entidade.

O pedido de credenciação permanente ou temporário de pessoas é efetuado no portal "Cartão do Aeroporto" pelo representante da respetiva empresa e desenvolve-se em várias fases, como representado no diagrama da Figura 84. Depois do preenchimento dos dados

solicitados e da introdução no sistema das cópias dos documentos exigidos, o pedido é validado pelo colaborador da ANA que é o supervisor dos serviços que a empresa externa vai prestar e atesta a necessidade da pessoa em causa ser detentor de um cartão de acesso.

Depois das verificações das informações introduzidas e do pagamento dos valores em causa, o registo criminal da pessoa é verificado pela PSP – Polícia de Segurança Pública e no caso de cidadãos estrangeiros, os dados são também analisados pelo SEF – Serviço de Estrangeiros e Fronteiras.

Quando todas as condições para emissão do cartão estiverem reunidas, o Gabinete de Segurança e a Empresa que efetuou o pedido são informadas que o cartão está pronto para emissão.

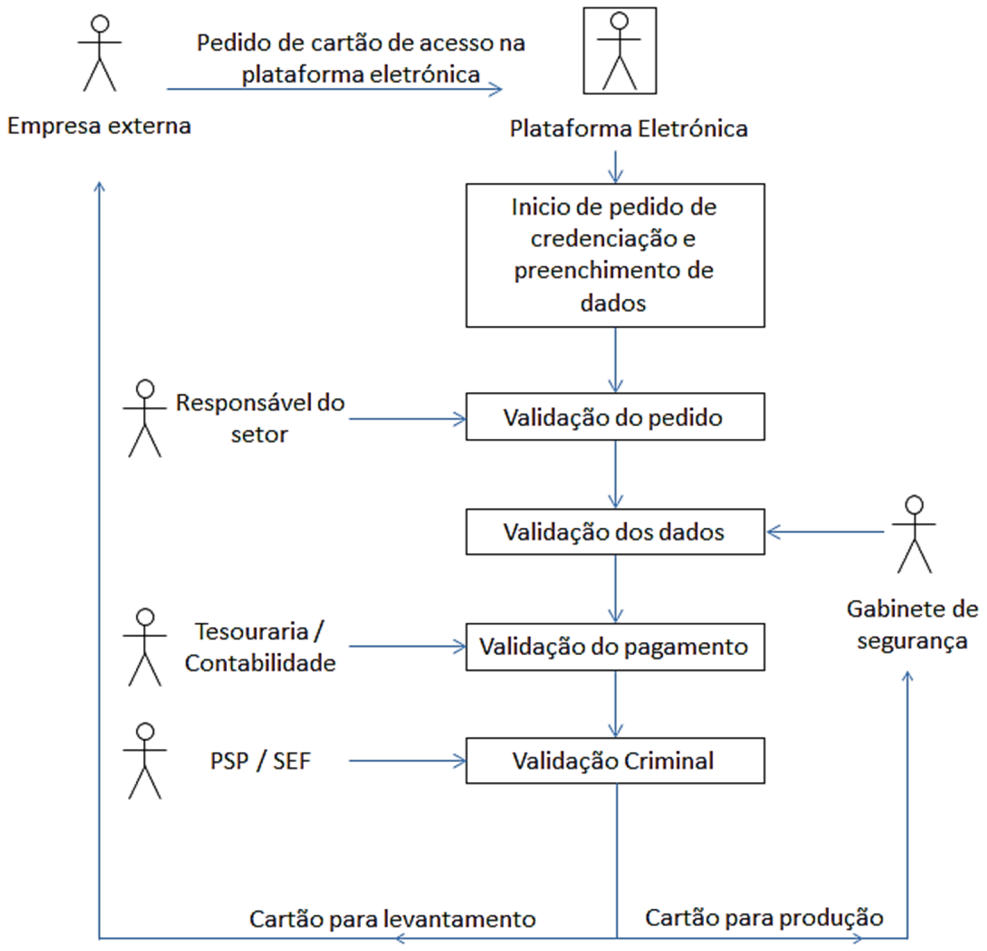


Figura 84 – Processo de credenciação permanente ou temporária: Plataforma eletrónica.

A emissão do cartão é efetuada quando o futuro portador se deslocar ao Gabinete de Segurança para se efetuar a aquisição da sua foto. Nesse momento, procede-se à criação do cartão no sistema SGO, no Pro-Watch e procede-se à produção do cartão físico, como mostrado na Figura 85.

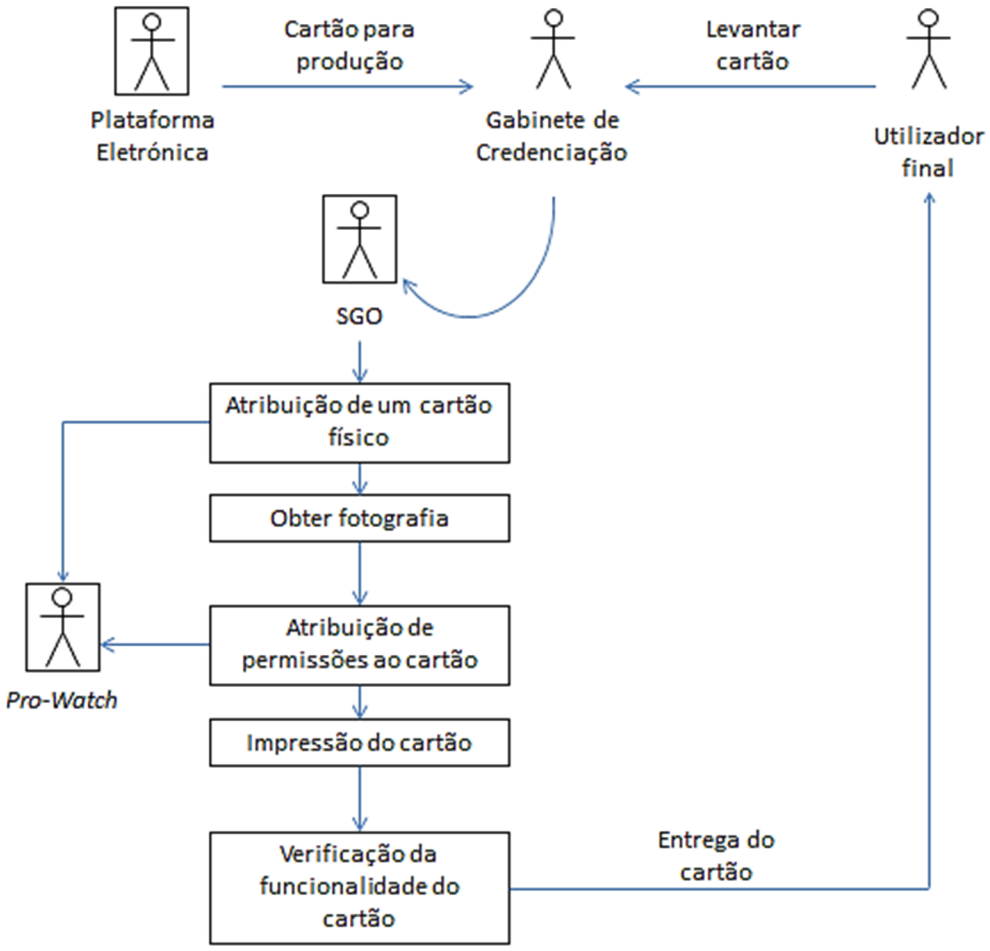


Figura 85 – Processo de credenciação permanente ou temporária.

Os cartões permanentes ou temporários emitidos, são cartões de proximidade RFID em suporte do tipo ID-1 da marca HID. Os cartões comunicam através de uma variante proprietária do protocolo *Wiegand*. Dado este tipo de protocolo usado, os cartões são adquiridos em lotes, com indicação específica do cliente ANA e o fabricante garante que o número de cada cartão definido no processo de fabrico é único a nível mundial. Os cartões usados são equipados com duas tecnologias de comunicação:

- Uma tecnologia a funcionar na frequência dos 125KHz que transmite a informação do número de série do cartão.

- Outra tecnologia a funcionar na frequência dos 13.56MHz que permite aceder a dois segmentos de memória de 4Kbyte cada um. Inicialmente este espaço de memória era usado para armazenamento de informação biométrica, mas atualmente os pontos de controlo por registo biométrico estão descontinuados e este recurso do cartão deixou de ser usado.

Os cartões RFID são usados para identificação do portador em áreas de acesso condicionado por comparação com a fotografia, usados para identificação eletrónica nos postos de controlo das portarias, são usados para abertura de portas integradas no SACA e são usados no registo de assiduidade dos colaboradores ANA. O subcapítulo 1.1 descreve os detalhes deste tipo de cartões.

3.1.1.2. PROCESSO DE CREDENCIAÇÃO PONTUAL

A credenciação pontual visa suprir necessidades de duração muito limitada, esta credenciação é sempre efetuada pela Polícia de Segurança Pública e pode ser emitida para dois fins: visitas acompanhadas por pessoas autorizadas ou entradas de passageiros para reclamação de bagagem perdida.

O pedido de credenciação para visitas, é efetuado pelo colaborador da ANA responsável pelo acompanhamento da visita, Figura 86, e formalizado por *e-mail* com o envio de um documento de identificação do visitante. Este pedido dá origem a um cartão em suporte de cartolina como descritos no subcapítulo 1.1

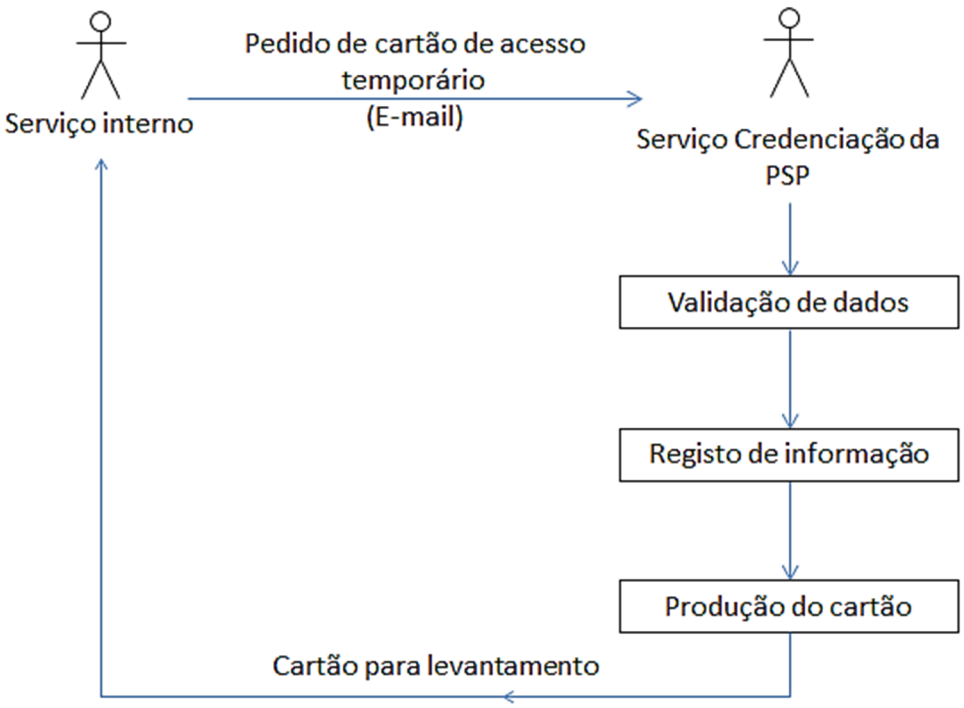


Figura 86 – Processo de credenciação pontual.

O pedido para acesso a áreas reservadas, para reclamação de bagagem é efetuado por passageiros detentores de um bilhete de avião, que comprovem que usaram o serviço de chegadas e é efetuado diretamente no balção de credenciação da PSP no Aeroporto, Figura 87. Este pedido, dá origem a um cartão de acesso autocolante que tem de ser apresentado nos postos de rastreio, juntamente com a identificação do passageiro: bilhete de identidade, cartão do cidadão ou passaporte e que permite o acesso, sem acompanhamento, às zonas de reclamação de bagagem.

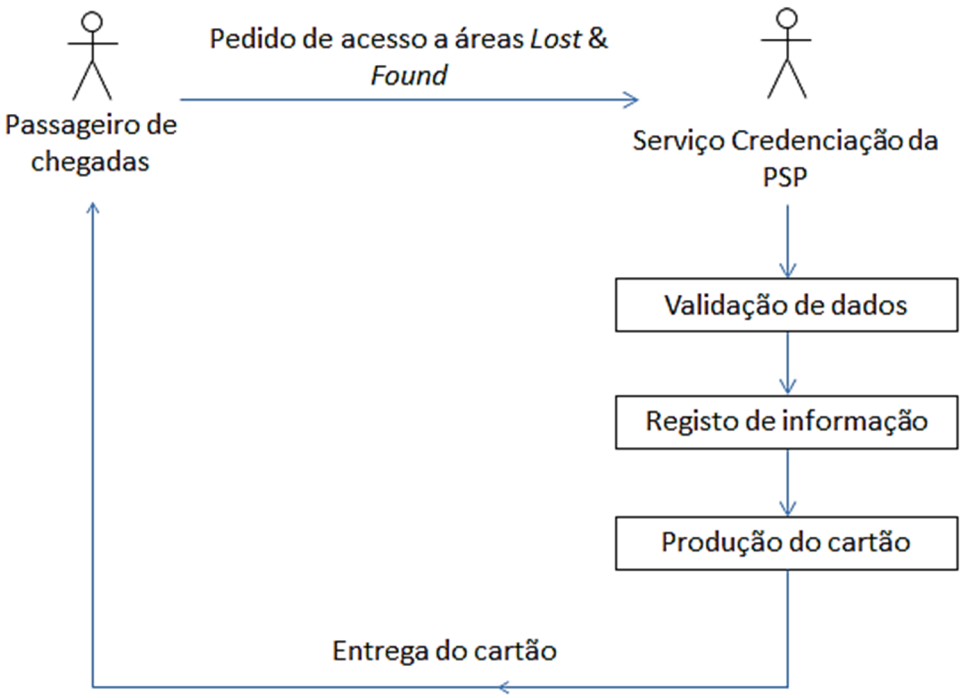


Figura 87 – Processo de credenciação pontual *Lost&Found*.

3.1.1.3. PROCESSO CREDENCIAÇÃO DE VIATURAS

A credenciação de viaturas, faz-se usando um modelo em formato de papel, que efetua o pedido de credenciação e apresenta os dados relevantes. A verificação de documentos é efetuada pelo Gabinete de Segurança que regista a informação no SGO e efetua alguns procedimentos de verificação nomeadamente inspeção da condição das viaturas, aval do serviço interno responsável pela viatura ou pelo serviço que a viatura vai dar apoio e verificação dos respetivos pagamentos, Figura 88. Este processo dá sempre origem a um dístico de acesso permanente ou acesso temporário para fixação de forma visível no vidro da frente da viatura e no caso das identificações permanentes dá origem, também, a um conjunto de números de identificação para serem fixados no exterior lateral das viaturas.

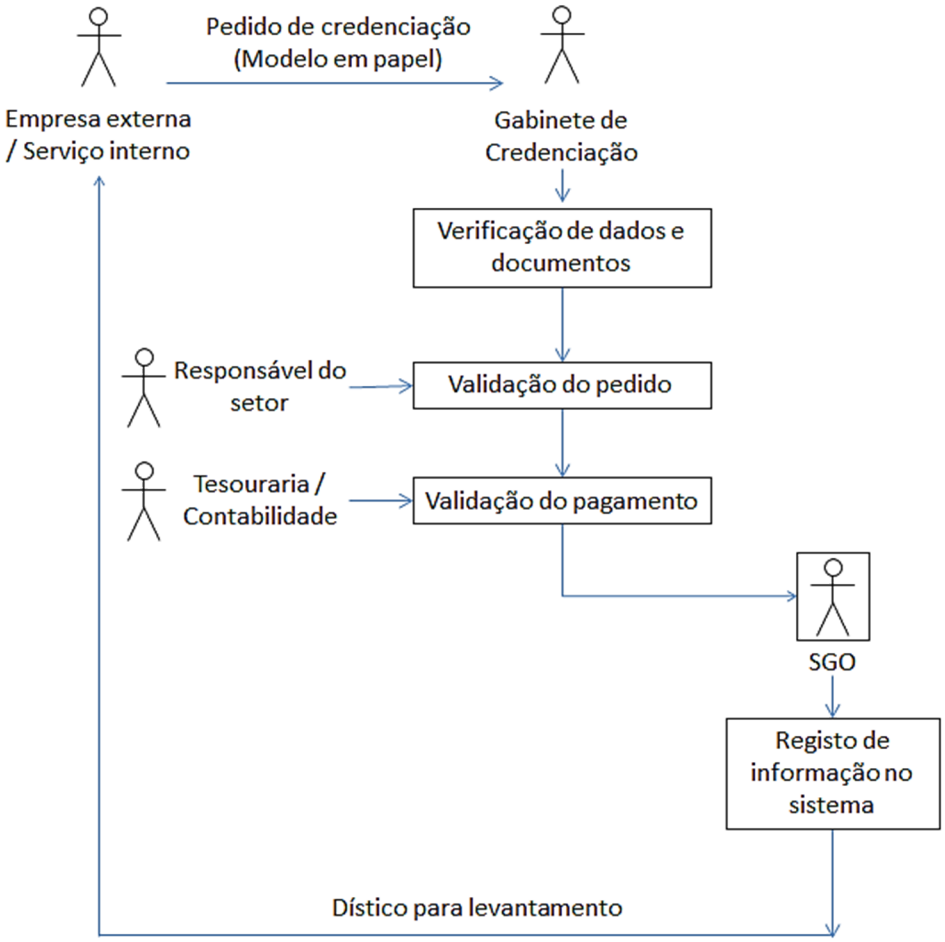


Figura 88 – Processo de credenciação de viaturas.

3.1.1.4. PROCESSO DE LICENÇAS DE CONDUÇÃO

Neste momento as licenças de condução são apresentadas num cartão impresso, como mostrado na Figura 89.

Figura 89 – Exemplo de licença de condução.

O pedido de licenças de condução de viaturas dentro do perímetro do aeroporto, é efetuado num modelo de suporte em papel e dirigido a um departamento do Serviço de Operações Aeroportuárias.

Para ser detentor da licença de condução o preponente tem de possuir licença de condução emitida pelo país de origem da pessoa para o tipo de viatura em causa e é submetido a um exame teórico-prático de condução em áreas reservadas, a uma formação e a avaliação sobre o uso de extintores contra incêndios, Figura 90. Após reunidas as condições necessárias a licença de condução em áreas restritas é emitida com prazo de validade igual à do cartão de acesso.

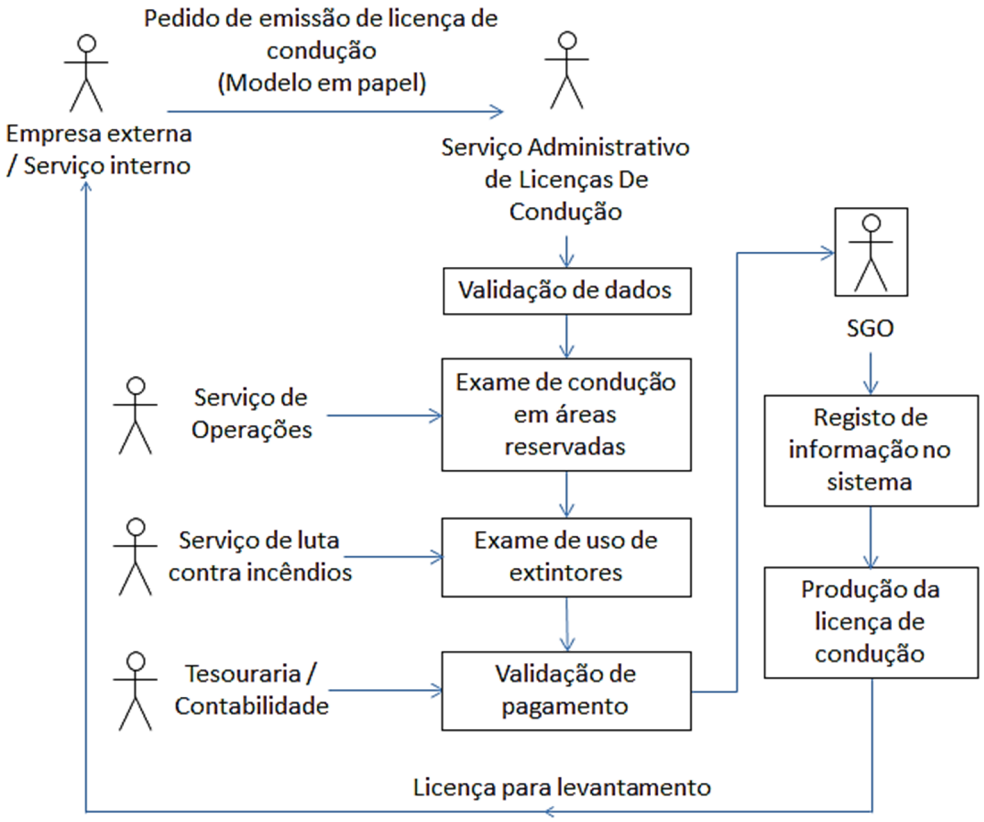


Figura 90 – Processo de licenças de condução.

3.1.1.5. ACESSO A ÁREAS RESTRITAS ATRAVÉS DE PORTARIAS

Na posse do cartão permanente ou temporário, quando uma pessoa se apresenta num posto de rastreio - portaria, apresenta o seu cartão no leitor RFID que está associado à aplicação GIC-C, este sistema lê o número do cartão e verifica na sua base de dados se a pessoa tem permissões de acesso e se for o caso se tem permissões de condução, apresentando o resultado da pesquisa ao vigilante que controla a passagem na portaria, como representado na Figura 91.

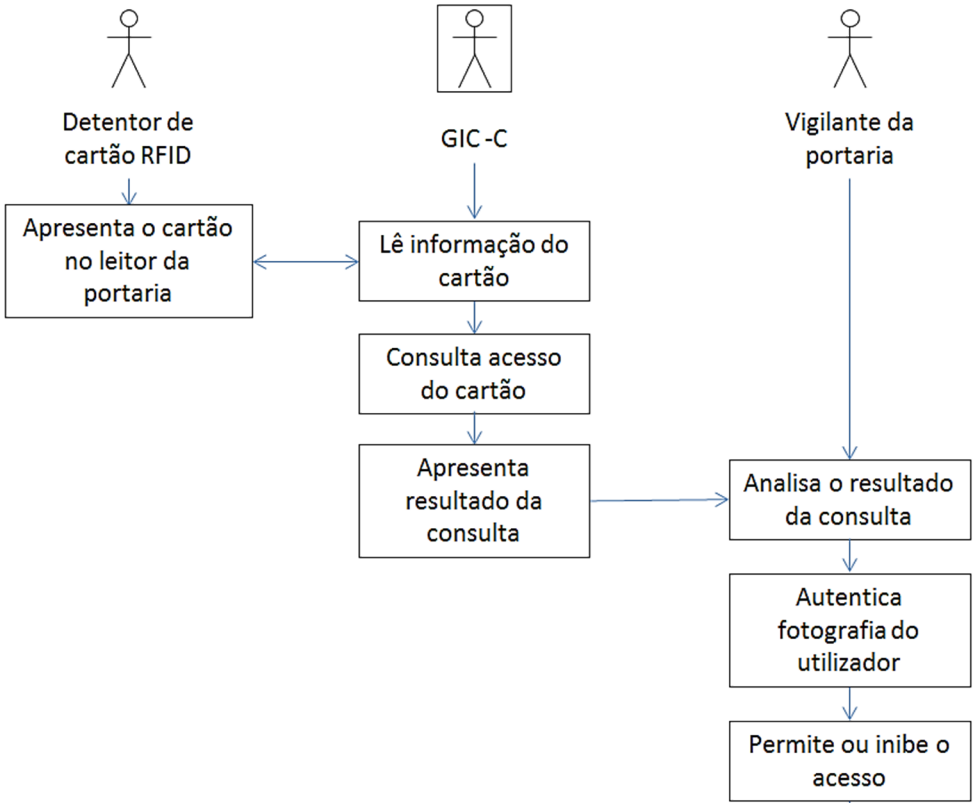


Figura 91 – Diagrama de acesso a áreas restritas através de portarias.

3.1.1.6. ABERTURA DE PORTAS DE CONTROLO AUTOMÁTICO

O sistema de controlo de portas automáticas, funciona com unidades de controlo de acessos distribuídas pelos espaços que gerem as aberturas dependendo dos acessos que o portador tem. Este processo de controlo funciona com a apresentação do cartão RFID ao leitor da porta e a consequente abertura porta ou permanência no estado fechado, Figura 92.

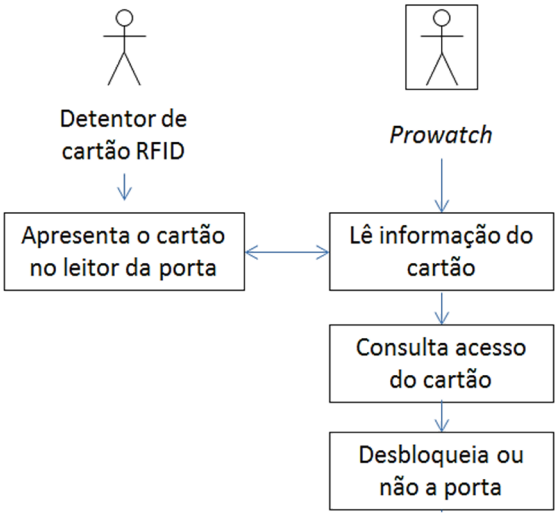


Figura 92 – Diagrama de abertura de portas de controlo automático.

3.1.1.7. REGISTO DE ASSIDUIDADE

O registo de assiduidade dos colaboradores ANA, faz-se aproximando o respetivo cartão permanente de leitores dedicados a essa função de registo. Os serviços de controlo de assiduidade e processamento de salários usam a informação obtida por leitores de cartão RFID para efetuarem as suas funções.

Sempre que o utilizador recebe um novo cartão, os serviços de credenciação do aeroporto, informam os serviços centrais da nova atribuição para poderem associar o novo cartão ao colaborador, como representado na Figura 93.

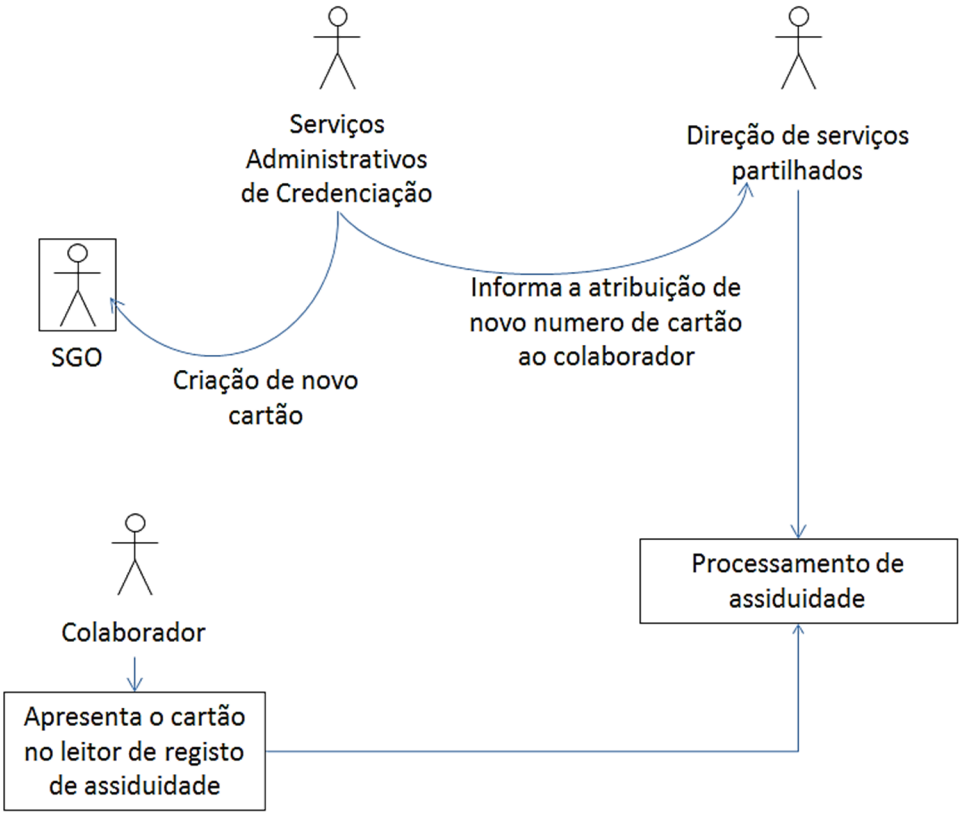


Figura 93 – Diagrama de registo de assiduidade.

3.1.2. INVENTÁRIO DE NOVAS FUNCIONALIDADES

Á semelhança do inventário das funcionalidades existentes, o método usado para elencar a lista de novas funcionalidades pretendidas foram as entrevistas com as pessoas que operam os diversos processos de credenciação das quais se obteve a seguinte compilação:

- Possibilidade de incluir ficheiros aos diversos registos de pessoas ou viaturas.
- Possibilidade de executar na plataforma, de forma transparente, as atividades implementadas de atribuição de acesso que são executadas no *Pro-Watch*.
- Possibilidade de usar fotografias fornecidas em ficheiros, para o serem colocadas nos cartões.
- Integração dos processos efetuados em papel relativos às licenças de condução.

- Integrar na plataforma as funcionalidades das credenciações pontuais. E desenvolvimento de ferramentas de pesquisa.

- Possibilidade de introduzir listas de ferramentas, associadas a pessoas que possam ser consultadas nos diversos postos de controlo.

- Possibilidade de introduzir listas de artigos proibidos, associadas a cartões que possam ser consultadas nos diversos postos de controlo.

- Integração dos processos efetuados em papel relativos a credenciação de viaturas.

3.2. SELEÇÃO DE FERRAMENTAS DE DESENVOLVIMENTO

O desenvolvimento da implementação da plataforma pretendida é efetuado por dois grandes grupos ferramentas, um que engloba o registo e manutenção de informação, funções garantidas por sistemas de gestão de bases de dados. O outro grupo é o das ferramentas de desenvolvimento de aplicações.

As equipas de desenvolvimento de *software* da ANA, usam genericamente ferramentas *Microsoft*®. Dada a continuidade de serviço que se pretende para o sistema a ser implementado e considerando:

- A experiencia de desenvolvimento dos colaboradores;

- As licenças de *software* existentes;

- A oferta de soluções integradas de ferramentas de desenvolvimento e

- As perspetivas de evolução e suporte das ferramentas de desenvolvimento do fornecedor *Microsoft*.

A escolha das ferramentas a usar recaiu sobre a plataforma *SQL Server*, na versão 2008 – R2 como suporte de informação. E recaiu sobre o *Visual Studio*, na versão 2013 como ferramenta de desenvolvimento.

O *SQL Server*, [86], é um sistema de gestão de bases de dados segundo o modelo relacional. Esta ferramenta, usa como linguagens de interrogação o ANSI SQL e uma variante denominada T-SQL: *Transact-SQL* que contem um conjunto estendido de funções que permitem, por exemplo, definição de blocos de código programável e declaração de variáveis. Alem das funções de gestão de base de dados o *SQL Server* disponibiliza vários recursos importantes nomeadamente:

- *Database Mirroring* – funcionalidade que cria uma replica da base de dados num *SQL Server* executado noutro servidor, pronta a entrar em funcionamento quando a primeira instancia falha, criando uma solução global de alta disponibilidade.

- *Snapshot database*- possibilidade de criar, num determinado momento, uma vista estática da base de dados. Essa vista fica disponível, somente para leitura, mas pode ser tratada com uma base de dados independente. A operação inversa também é possível, isto é reverter toda uma base de dados para uma vista criada no passado por *shapshot*. Esta funcionalidade é particularmente útil quando se fazem testes sobre uma base de dados e depois, há a necessidade de repor o estado inicial.

- *Data Partitionig*- Esta funcionalidade permite que a pesquisa em tabelas de grandes dimensões seja feita apenas numa parte da tabela em vez da tabela toda através de um critério baseado na avaliação dos valores de uma coluna. Esta funcionalidade é útil, por exemplo quando se pesquisa numa tabela com um campo de data e se pretende obter dados relativos aos registos mais recentes otimizando o desempenho da procura.

- *Common Language Runtime* – CLR- Integração do *SQL Server* com o *Visual Studio* de forma a desenvolver código de funções de utilizador no *SQL Server*, usando uma das linguagens de programação do *Visual Studio*, como o *Visual C#* ou o *Visual Basic*.

- *Reporting Services* – É uma plataforma de geração de relatórios que podem ser guardados em diversos formatos, nomeadamente pdf, xlsx, csv, etc.

O *Visual Studio*, [87], é uma plataforma que contém um conjunto de programas de desenvolvimento de aplicações de *software* em várias linguagens de programação, entre as

quais: *Visual Basic*, *C, C++, C#, J#* e *ASP.NET* usada para o desenvolvimento de aplicações *Web*. As linguagens de programação são executadas sobre uma base denominada *.Net Framework* que disponibiliza um ambiente de execução independente da linguagem, suportado por um conjunto de bibliotecas comuns a toda plataforma. Das possibilidades disponíveis, selecionou-se a linguagem *Visual Basic* como base para o desenvolvimento das aplicações deste projeto.

O Visual Basic disponibiliza vários recursos gráficos de desenvolvimento, nomeadamente ambientes de desenho de aplicações, interação com sistemas de bases de dados adaptando os modelos relacionais ao paradigma de programação por objetos, criação de classes usando a modelação UML – *Unified Modeling Language*, ferramentas de análise de código – *debugers*, mecanismos de gestão de erros, etc.

3.3. APLICAÇÕES EXTERNAS

A plataforma que se pretende desenvolver vai ter duas interligações com aplicações externas, como mostrado na Figura 94. Por um lado, vai interligar com o portal "Cartão do Aeroporto" para fluxo de informação processual. A interface com o portal vai ser implementada pela disponibilização por parte da plataforma de serviços de troca de informação. Por outro lado a plataforma vai interagir com o sistema *Pro-Watch*, para que a atribuição de permissões de abertura de portas seja feita na interface da plataforma a.

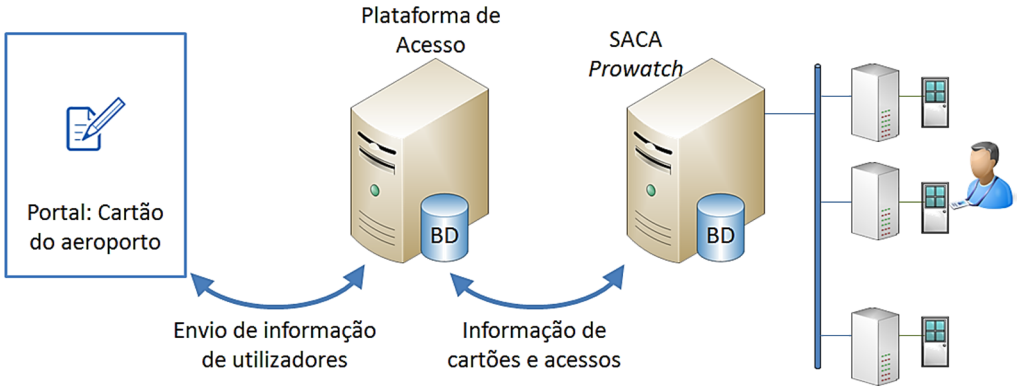


Figura 94 – Interligações da plataforma com aplicações externas.

3.3.1. PORTAL "CARTÃO DO AEROPORTO"

O portal "Cartão do Aeroporto", é uma aplicação *web*, corporativa que disponibiliza para o exterior da empresa um serviço *on-line* de pedidos de cartões de indentificação para cada área geografica onde a ANA tem instalações.

No caso de ASC, o portal interliga-se aos sitemas locais de credeciação de duas formas:

- Uma comunicação bidirecional entre o portal e o SGO para troca de informação relativa a portadores de cartões e das entidades que representam.
- Uma comunicação unidirecional no sentido SGO-portal, para pesquisa de informação relacionada com os cartões de uma pessoa.

A plataforma a desenvolver tem de manter estas funcionalidades, com o minimo de impacto possivel na implementação do portal.

3.3.2. SISTEMA AUTOMÁTICO DE CONTROLO DE ACESSOS

O Sistema Automático de Controlo de Acessos – SACA, é um sistema para abertura automática de portas através da apresentação de um cartão RFID. O sistema instalado no ASC é composto por servidores e unidades de controlo de acessos. Nos servidores são executados os serviços e as funcionalidades do sistema. As unidades de controlo de acesoss interligam o equipamento de campo para controlo de portas, como leitores de cartões, trincos, sensores, etc. O lado direito da Figura 94 apresenta a topologia de implementação do SACA no ASC em que os servidores comunicam com as unidades de controlo por rede *Ethernet*, numa VLAN exclusiva para este sistema.

3.3.2.1. UNIDADE DE CONTROLO DE ACESSO

As Unidades de Controlo de Acesso – UCA, que operam as portas do ASC, fisicamente são constituídas por, Figura 95:

- Uma unidade microprocessada que faz a interligação com o servidor e executa as operações de decisão de acesso sobre as portas que lhe estão atribuídas. O dipositivo instalado, usado para esta função é o modelo StarII da marca NexSentry.

- Cada UCA está equipada com dois módulos de controlo de equipamento de campo do modelo Wiro 4/8/4 da marca NexSentry. Os Wiro 4/8/4 permitem a ligação de:

    - Quatro leitores de cartões que usem o protocolo *Wiegand*.

    - Oito entradas digitais.

    - Quatro saídas por relé.

- A alimentação elétrica das UCAs é garantida por uma fonte de alimentação $230V_{AC}$-$24V_{DC}$ que também efetua o carregamento de um conjunto de baterias que suportam o sistema e todos os dispositivos de campo quando há falhas de energia da rede.

- As ligações elétricas dos equipamentos de campo são efetuadas em réguas de ligadores e as saídas dos WIRO são transmitidas para o exterior através de uma bateria de relés, isto para evitar que picos de corrente provocados por motores de desbloqueio de portas ou bobines de trincos elétricos possam destruir as placas WIRO.

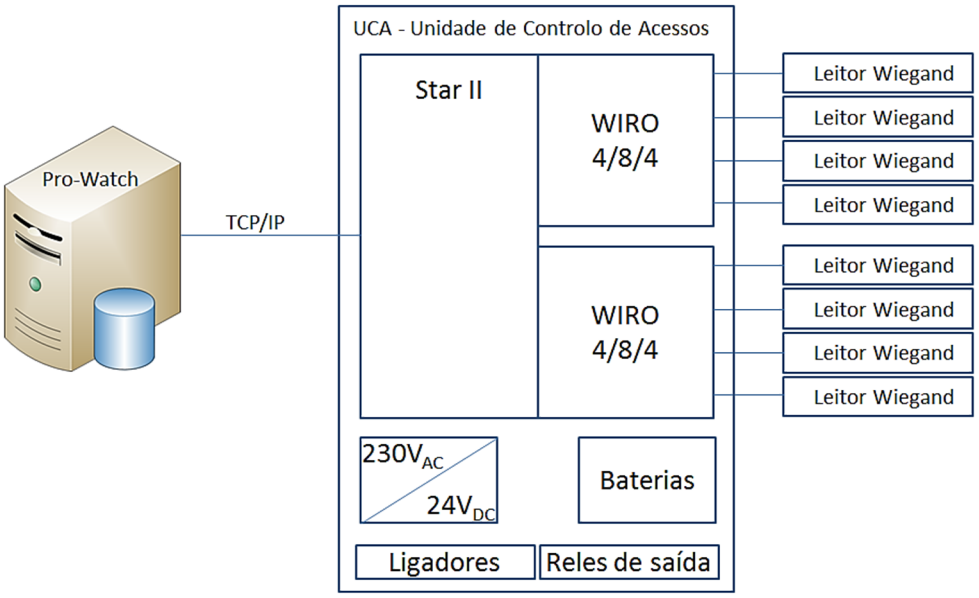


Figura 95 – Diagrama da UCA - Unidade de Controlo de Acessos.

As UCAs têm três tipos de ligações com o exterior: leitores de cartões, saídas por relé e entradas digitais. Em sede de configuração de UCAs, é possível efetuar combinações que usam um ou mais elementos de cada tipo para a implementação da topologia adequada ao caso concreto, por exemplo:

- Porta com um leitor de cartões numa das faces e botão de pressão na outra face, botão, esse, ligado a uma entrada digital do WIRO;
- Porta com um leitor de cartões numa das faces e abertura mecânica do outro lado;
- Porta com dois leitores de cartões, um em cada lado da porta;
- A Figura 96 mostra o diagrama elétrico de um WIRO, de notar que cada entrada de leitor de cartões usa seis ligações: duas para alimentação elétrica, PWR e GND, duas ligações para comunicação *Wiegand*, Data0 e Data1 e duas ligações para controlo de sinalizadores luminosos, LED0 e LED1.

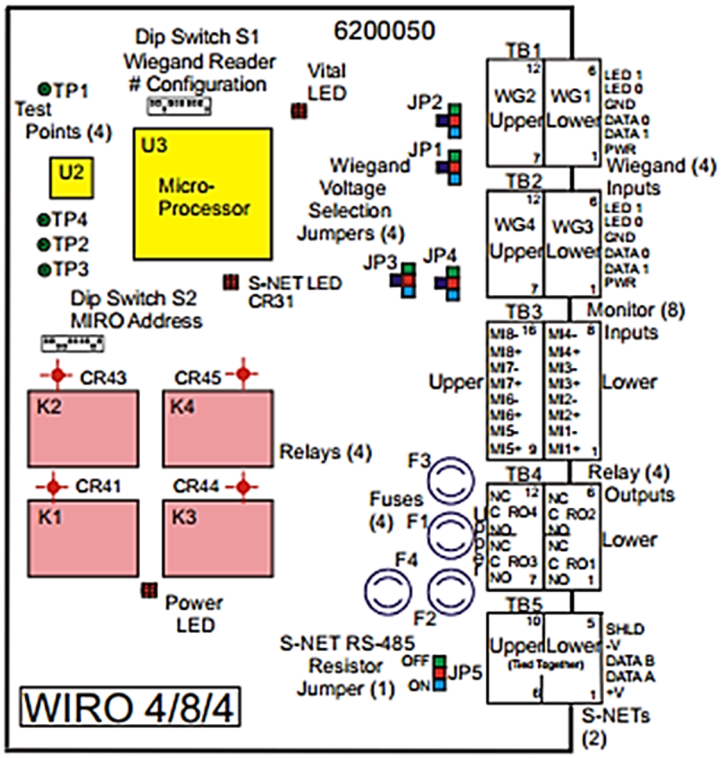


Figura 96 – Diagrama de uma placa WIRO, [89].

No ASC além do controlo de portas, estas UCAs, também são usadas para controlo de barreiras de parques de estacionamento e controlo do piso a que se pode aceder através de elevadores.

3.3.2.2. *PRO-WATCH*

O *Pro-Watch*, [88] é uma marca registada propriedade da empresa Honeywell Inc., e denomina uma plataforma de *hardware* e *software* para implementação de sistemas de controlo de acessos. No ASC o *Pro-Watch* está instalado nos servidores de controlo de acessos e é usado em duas vertentes: na configuração de equipamento e na atribuição de acessos.

- Na vertente da configuração de equipamento, o *Pro-Watch*, estabelece ligação com as UCAs e define todos os parâmetros da sua configuração nomeadamente: o tipo de cartões que são lidos, associação de leitores de cartões, das entradas digitais e

das saídas por relé a portas físicas, define os tempos para ativação de saídas, define os tipos de ações ou alarmes para as entradas, etc. A Figura 97, apresenta o ecrã do *Pro*-Watch para configuração dos recursos das UCAs onde se podem ver a associação dos leitores, contactos e comando às respetivas portas.

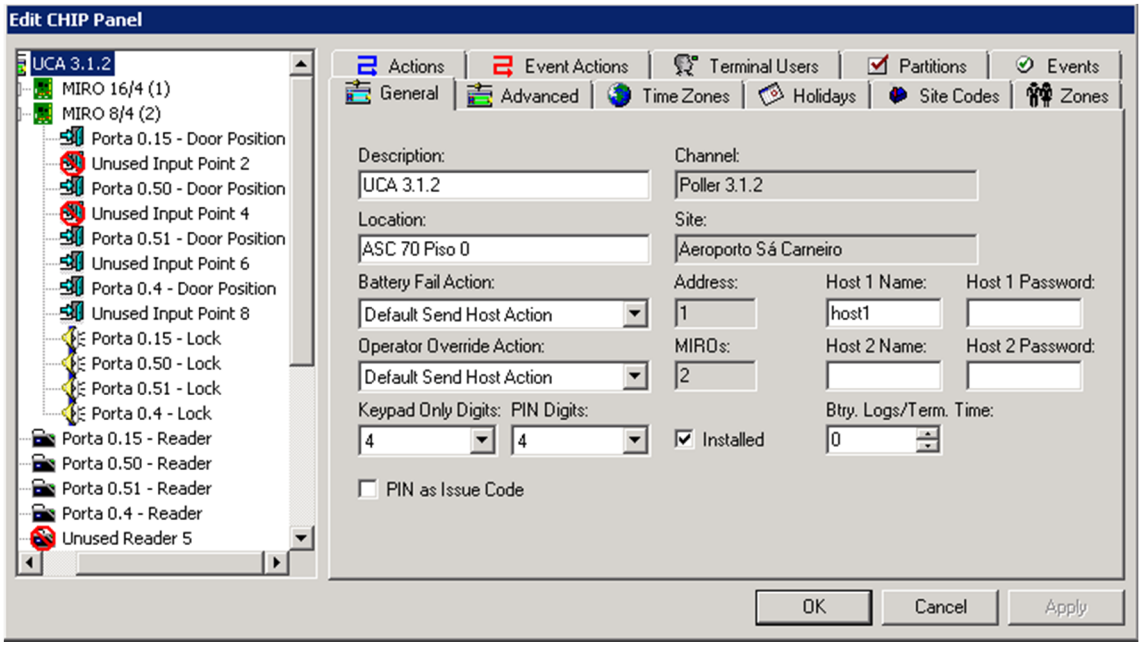


Figura 97 – *Pro-Watch* écran de configuração de UCA.

Da configuração do equipamento de campo, resultam na plataforma de *software*, [88], as entidades *Logical Device* que representam genericamente as portas físicas com os seus vários equipamentos.

Na Figura 98 apresenta-se o ecrã de configuração de uma porta e dos elementos que a compõe, neste exemplo está-se a definir uma porta com dois leitores de cartões, bloqueio por trinco elétrico e um sensor de posição da porta. No processo de configuração, estes recursos são relacionados com entradas e saídas físicas do WIRO, onde posteriormente são efetuadas as ligações elétricas.

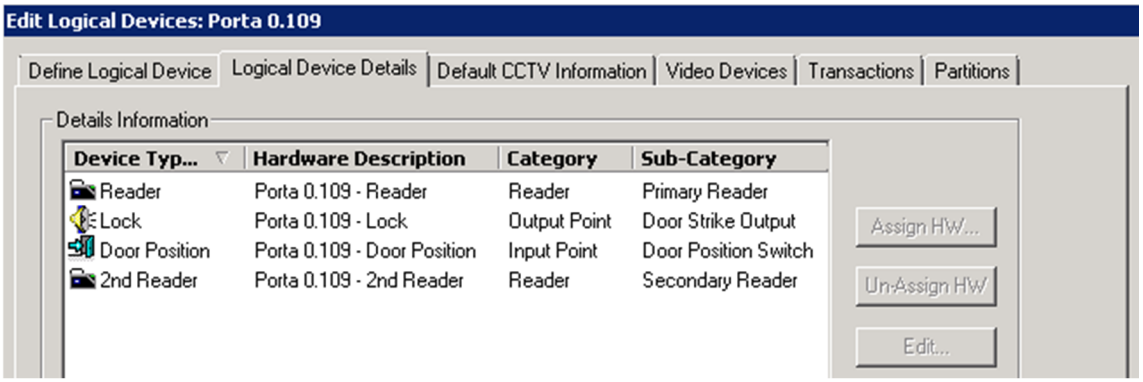


Figura 98 – *Pro-Watch* écran de configuração de porta – *Logical Devices*.

A vertente da atribuição de acessos do *Pro-Watch* relaciona as permissões das pessoas com as portas – *Logial devices* a que podem aceder. No *Pro-Watch* existem duas entidades que podem ser usadas para organização da atribuição de permissões de acesso são elas os *Clearance code* e as *Company*.

Os *Clearance code* são nomes que se atribuem a agrupamentos de portas, podem-se criar tantos quantos necessários e cada porta pode ser associada a mais de um *Clearance code*. Os *Clearance code* são usados para portas que tem algum tipo de relacionamento, por exemplo se um corredor está segmentado por quatro portas e as pessoas que tem acesso ao corredor podem abrir todas as portas, cria-se um *Clearance code* para esse conjunto de portas e no momento de atribuição de acessos em vez de indicar quatro portas apenas se referência o respetivo *Clearance code*.

As *Company* são entidades que no sistema representam empresas, ou denominações de grupos de pessoas, esta entidade é usada para agrupar pessoas com o mesmo perfil de permissão de acessos.

A atribuição de acessos pode ser feita de várias formas usando as entidades *Logial devices*, *Clearance code* e *Company*:

- Podemos associar portas individuais - *Logial devices* individualmente a pessoas.
- Podemos associar os *Logial devices* a grupos de pessoas - *Company*.
- Podemos associar grupos de portas - *Clearance code* individualmente a pessoas.

- Podemos associar grupos de portas - *Clearance code* a grupos de pessoas - *Company.*

Normalmente as instalações equipadas com o *Pro-Watch*, como no caso do Aeroporto Francisco Sá Carneiro, têm realidades de permissões de acesso complexas e por isso a forma da atribuição de permissões faz-se com duas abordagens complementares: uma geral e uma específica. Numa primeira fase, agrupam-se as pessoas em perfis de acesso, normalmente associados à função que desempenham e agrupam-se as portas em caminhos sequenciais, por exemplo agrupam-se as portas das mangas de desembarque e os seus corredores de acesso ou agrupam-se as portas em grupos funcionais como por exemplo as portas de todas as salas de quadros elétricos. Na fase seguinte, aplica-se a abordagem geral de atribuição de acessos, associando grupos de portas a grupos de pessoas. Na última fase de atribuição de acessos, faz-se a afinação das permissões atribuídas na abordagem geral, com a abordagem específica, e nesse caso atribuem-se portas individuais a pessoas individualmente.

A execução do processo de identificação de pessoas *Pro-Watch*, é efetuada através do número de série do cartão RFID que está na posse da pessoa. Cada pessoa pode ter associado vários cartões, mas um número de cartão só pode estar associado a uma pessoa. Os cartões associados a uma pessoa podem estar em vários estados: perdido, expirado, ativo, etc. mas em cada momento apenas um cartão pode estar no estado ativo.

O acesso de passagem de uma pessoa num porta é concedido pelas atribuições de *Logial devices*, *Clearance codes* e *Companies* a essa pessoa, e a leitura do numero do cartão RFID, num dos leitores das UCAs é o processo de identificação dessa pessoa perante o sistema SACA.

3.3.2.3. ANÁLISE DA BASE DE DADOS DO *PRO-WATCH*

A plataforma a desenvolver com este projeto, pretende integrar todas as funcionalidades operacionais existentes nos vários sistemas de credenciação e controlo de acessos, nomeadamente as existentes no *Pro-Watch*. Isto implica que a plataforma vai ter de disponibilizar na sua interface, as funções de atribuição de cartões e acessos às pessoas e replica-los para o *Pro-Watch*.

A informação usada pelo *Pro-Watch*, está guardada numa base de dados *SQL Server* denominada PWNT. Como qualquer aplicação proprietária, não existe disponível qualquer tipo de informação sobre como a base de dados está concebida.

Abrindo a base de dados do *Pro-Watch* no *SLQ Server Management Studio*, verifica-se que é composta por cerca de quinhentas tabelas, por cerca de quinhentas tabelas de visualização e cerca de trezentas rotinas de programação, Figura 99.

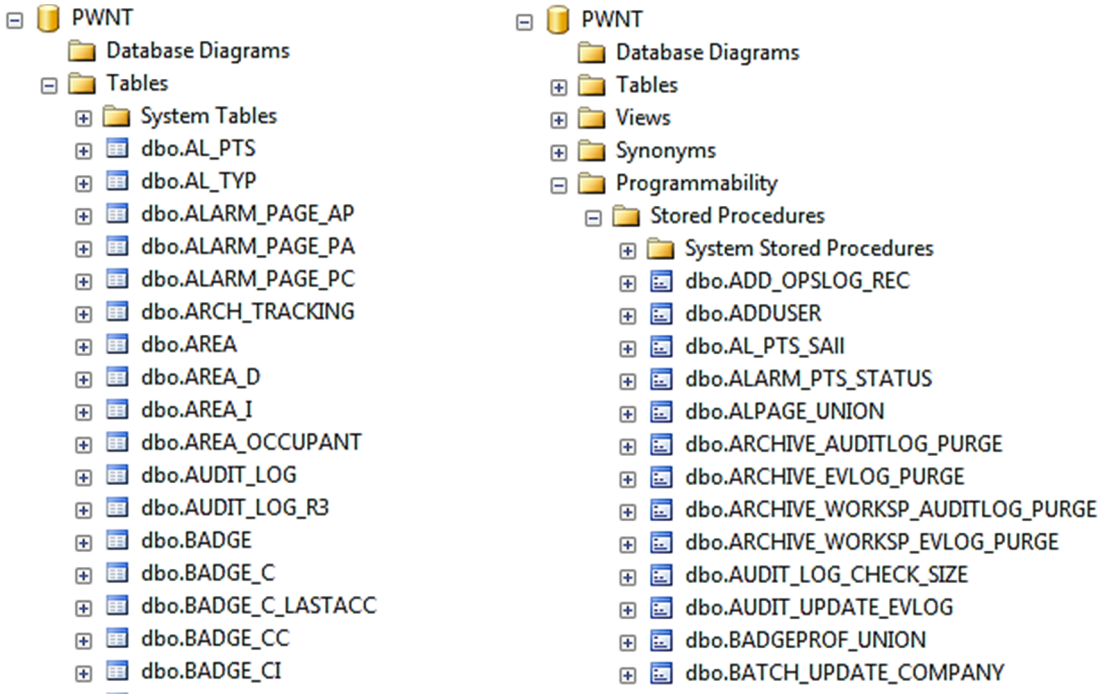


Figura 99 – Exemplo do conteúdo da base de dados do *Pro-Watch*.

No cenário em que há necessidade de atribuir cartões e permissões às pessoas usando a plataforma a desenvolver e replicar essa informação para o *Pro-Watch*, vai ser necessário

adquirir conhecimentos mínimos da estrutura da base de dados PWNT, para aplicar ao SACA, as ações efetuadas na plataforma. A obtenção deste conhecimento vai ser efetuada por análise das tabelas existentes, executando as seguintes ações:

- Na primeira fase efetua-se num servidor de testes, uma instalação de raiz do *Pro-Watch* que inclui a criação da respetiva base de dados.

- De seguida, cria-se uma cópia de segurança da base de dados "vazia", para se poder repor o estado inical da base de dados.

- Na segunda fase procede-se à análise de informação, primeiro, executando pequenos passos na interface gráfica do *Pro-Watch* e analisado as repercussões que essas ações provocam nas tabelas da base de dados. A análise efetua-se aplicando a seguinte metodologia:

    a. Contar o número de registos de todas as tabelas da base de dados PWNT e registar essa informação. Ver informações sobre este tipo de operação no Anexo H

    b. Efetuar testes em pequenos passos, introduzindo informação no *Pro-Watch*, sobre um assunto restrito: pessoa, cartão, *Clearance code*, *Company*, etc.

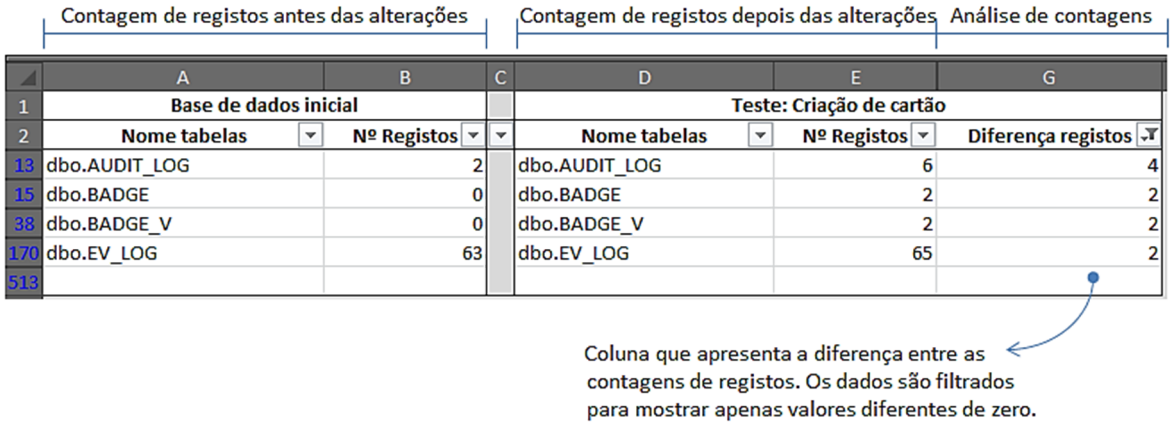
    c. Voltar a contar o número de registos de todas as tabelas da base de dados PWNT e comparar com a contagem anterior, como mostrado na Figura 100.

    d. Analisar as tabelas da base de dados que tem número de registos diferentes nas duas contagens, tentando:

        i. Encontrar os campos que possuem a informação introduzida na interface gráfica.

        ii. Encontrar as tabelas que possam estar a fazer o suporte nas relações entre tabelas e tentando encontrar as ligações através dos campos chave.

e. Voltar ao primeiro ponto e usar estes passos de forma sistemática numa aproximação tentativa-e-erro, até ser conseguir definir a forma de implementar todas as necessidades.



	A	B	C	D	E	G
1	Base de dados inicial			Teste: Criação de cartão		
2	Nome tabelas	Nº Registos		Nome tabelas	Nº Registos	Diferença registos
13	dbo.AUDIT_LOG	2		dbo.AUDIT_LOG	6	4
15	dbo.BADGE	0		dbo.BADGE	2	2
38	dbo.BADGE_V	0		dbo.BADGE_V	2	2
170	dbo.EV_LOG	63		dbo.EV_LOG	65	2
513						

Figura 100 – Comparação do número de registos na PWNT antes e depois de alterações.

A aplicação desta metodologia e a análise efetuada, do ponto de vista de suporte de informação, conduziu ao conhecimento das tabelas apresentadas na Tabela 8.

Do ponto de vista conceptual concluiu-se que a base de dados PWNT não trabalha com entidades do tipo "Pessoa" porque a implementação PWNT tem como entidade central, o conceito de cartão – *badge*, a informação do portador do cartão está dentro do registo do próprio cartão.

Tabela 8 – Base de dados do *Pro-Watch*: tabelas de interesse.

Tabela	**Informação contida**
Badge_C	Tabela que contém a informação primária relativa ao cartão como: Número do cartão. Companhia a que pertence. Datas de emissão e validade. PIN. Ultima porta acedida. Ultimo alarme gerado. Vários campos de marcadores. Etc. Esta tabela também contém a informação usada na associação entre os cartões e as *company*.
Badge_V	Contém informação do portador do cartão:

	Nome do utilizador. Vários campos para conter a morada. Campos para conter informação de contacto como número de telefone. Fotografia do portador. Etc.
Badge_Status	Contém informação sobre o estado dos cartões, a base de dados em produtivo tem apenas um registo com o valor "Active".
Badge_fields	Tabela que contém lista de campos que estão a ser usados na tabela Badge_C. Nesta tabela definem-se: o nome dos campos e o tipo informação que cada campo pode conter e por exemplo se o campo é auto-incrementável, qual o ultimo valor atribuído, se o valor tem de ser único na coluna, se admite valores nulos, etc.
Badge_CC	Tabela que faz a associação entre os cartões e o *clearence codes* que lhe estão atribuídos.
Clear	Tabela onde estão definidos os *clearence codes*.
Clear_L	Tabela que faz a associação entre os *clearence codes* e os *logical devices*.
Logical_Dev	Tabela que tem a lista dos *logical devices* configurados.
Company	Tabela que tem a lista das *company*.
Autit_Log	Tabela onde estão guardados os registos de alteração de informação nas outras tabelas indicando que tabela e que campos foram alterados.
EV_LOG	Tabela onde são registados os eventos que ocorrem no sistema, como apresentação de cartões nos leitores, ativação de saídas, mudança de estado das entradas, alarmes, etc.
Panel	Tabela com informação relativa às UCAs.
Output	Informação da configuração das saídas das UCAs.
Input	Informação da configuração das entradas digitais das UCAs.
Reader	Informação da configuração dos leitores de cartões das UCAs.

A Figura 101 mostra a relação entre as tabelas de interesse da base de dados do *Pro-Watch*.

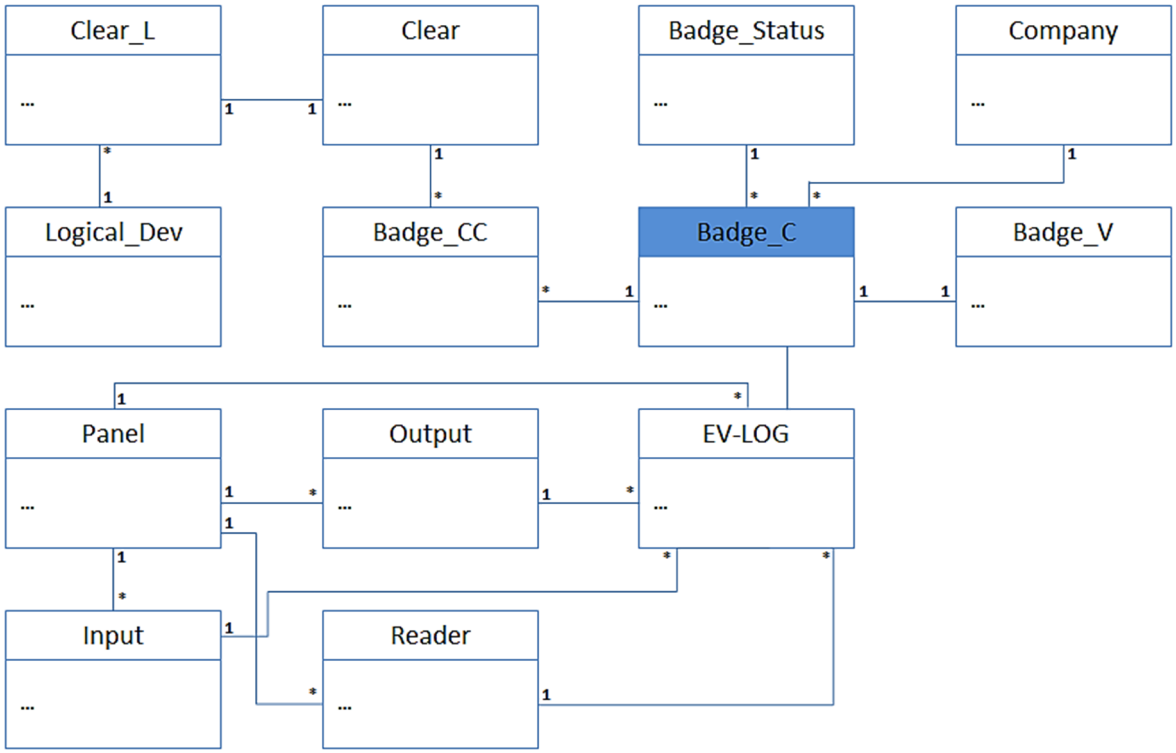


Figura 101 – Base de dados do *Pro-Watch*: relações entre tabelas relacionadas com acessos.

3.4. MÓDULOS APLICACIONAIS DA PLATAFORMA

Do conhecimento adquirido na análise e inventário das funcionalidades existentes nos vários sistemas e processos de credenciação no ASC, conclui-se que existem cinco grandes grupos de atividades:

- Credenciação de pessoas com cartões permanentes ou temporários cuja diferença reside apenas no período de validade.
- Credenciação de viaturas, permanente ou temporária
- Gestão de licenças de condução.
- Credenciação pontual, para visitas e passageiros.
- Controlo de acessos em portarias.

A execução deste conjunto de atividades tem de ser englobada na plataforma a desenvolver. Do ponto de vista da hierarquia de recursos humanos, dos serviços que executam atualmente funções, as três primeiras atividades: credenciação de pessoas e viaturas e gestão licenças de condução, apesar de estarem a ser efetuadas por dois gabinetes distintos, hierarquicamente, os dois estão sobre a alçada do mesmo serviço. A função de credenciação pontual é efetuada pela Policia de Segurança Publica. E a função de controlo de portarias é efetuada por agentes de vigilância de uma empresa de segurança. Desta análise, definiu-se que a plataforma a desenvolver vai ser constituída por três aplicações e uma base de dados para repositório de toda a informação.

Assim, a plataforma de credenciação vai ser constituída por uma aplicação – módulo *Cred*, que contempla as três primeiras atividades, uma aplicação – módulo *Pontu*, para ser usada no balção da PSP nas atividades de credenciação pontual e uma aplicação – módulo *Port*, que suporta o controlo de acessos para rastreio nas portarias, como mostrado na Figura 102.

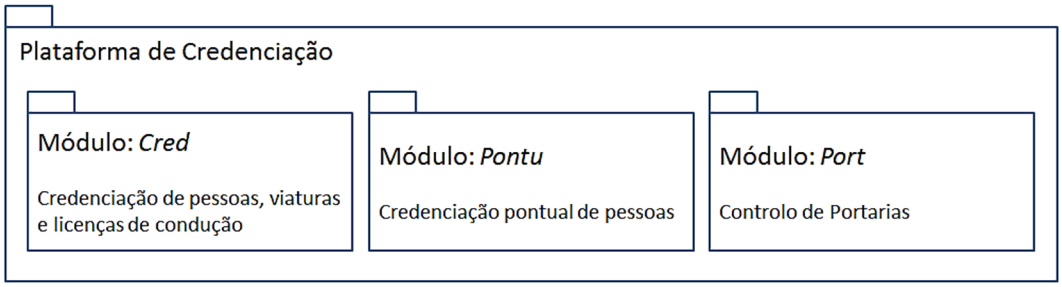


Figura 102 – Módulos aplicacionais da plataforma de credenciação.

Do ponto de vista de perfis de pessoas foram definidos dois grupos de perfis:

- Portador de cartão de credenciação: dentro deste tipo existe o perfil vazio que não tem privilégios associados e existe o perfil de portador de cartão de acessos.

- Perfis associados ao de desempenho de funções de utilização da plataforma de credenciação, por exemplo perfil de Administrador que possui todos os privilégios dentro da plataforma ou o exemplo do perfil de Vigilante relativo às pessoas que podem operar na aplicação *Port*.

Do ponto de vista de perfis de utilizador, a plataforma de credenciação pode ser usada por seis tipos de utilizadores, como mostrado na Figura 103.

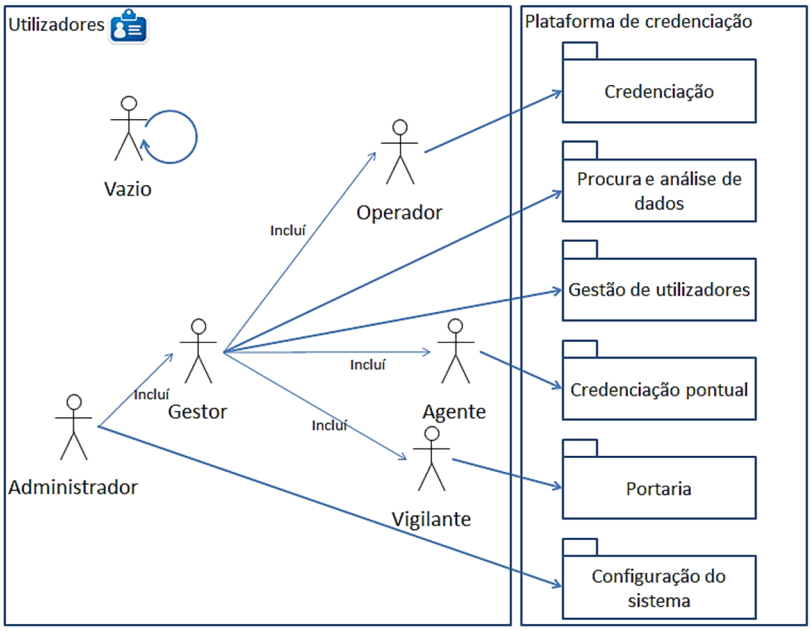


Figura 103 – Perfis utilizados na plataforma de credenciação.

- Vazio: Perfil sem qualquer tipo de permissões, caracterização usada para classificar pessoas que deixaram de desempenhar funções ativas no sistema.

- Portador: Perfil de portador de cartão de credenciação, que pode ter ou não permissões de acesso no sistema de controlo de acessos.

- Vigilante: elemento que opera nas portarias e acede às funcionalidades operacionais do módulo *Port*.

- Agente: agente da Polícia de Segurança Pública, que acede às funcionalidades operacionais do módulo *Pontu*.

- Operador: utilizador que acede às funcionalidades operacionais de um ou mais elementos: credenciação de pessoas, viaturas ou licenças de condução, do módulo *Cred*.

- Gestor: utilizador que acede todas as funcionalidades operacionais de todos os módulos, acede às funcionalidades análise de informação e acede às funcionalidades de gestão de utilizadores.

- Administrador: utilizador que acede a todas as funcionalidades de todos os módulos, acede às funcionalidades de parametrização e configuração do sistema, e tem permissões de acesso à base de dados.

A Figura 104, mostra a hierarquia de privilégios de acesso às funcionalidades da plataforma de credenciação.

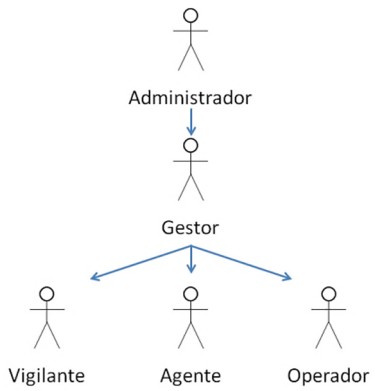


Figura 104 – Hierarquia de perfis de utilizador da plataforma de credenciação.

3.4.1. CASOS DE USO

O projeto de implementação da plataforma de credenciação, requere a representação do sistema a ser implementado nas suas diferentes perspetivas nomeadamente a nível lógico apresentando os diferentes componentes que constituem o sistema e as respetivas relações. A nível processual que descreve a forma como as atividades de desenrolam e encadeiam. A nível desenvolvimento que apresenta os blocos de implementação, as suas interfaces e as respetivas interligações. E a nível físico que descrevem os recursos necessários a implementação das diferentes soluções.

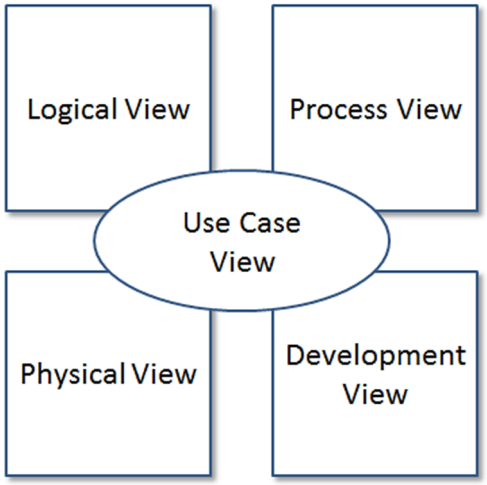


Figura 105 – Modelação do sistema através de casos de uso, adaptado de [96].

Os diagramas de caso de uso são uma forma resumida de abordar estas quatro perspetivas quer sobre a ótica do sistema quer sobre a ótica das funcionalidades do utilizador, Figura 105. No projeto de implementação de *software* que agora é iniciado vai-se utilizar os diagramas de caso de uso para fazer a representação geral do sistema a implementar e sempre que for necessário um maior nível de detalhe usam-se diagramas específicos para representações mais detalhadas. No Anexo G são apresentadas mais informações sobre técnicas de modelação e representação de aspetos de implementação de soluções de *software*.

3.4.1.1. CREDENCIAÇÃO DE PESSOAS

O módulo de credenciação – *Cred* contém as funcionalidades de credenciação permanente e temporária de pessoas, as funcionalidades de credenciação de viaturas e as funcionalidades de gestão de licenças de condução, Figura 106.

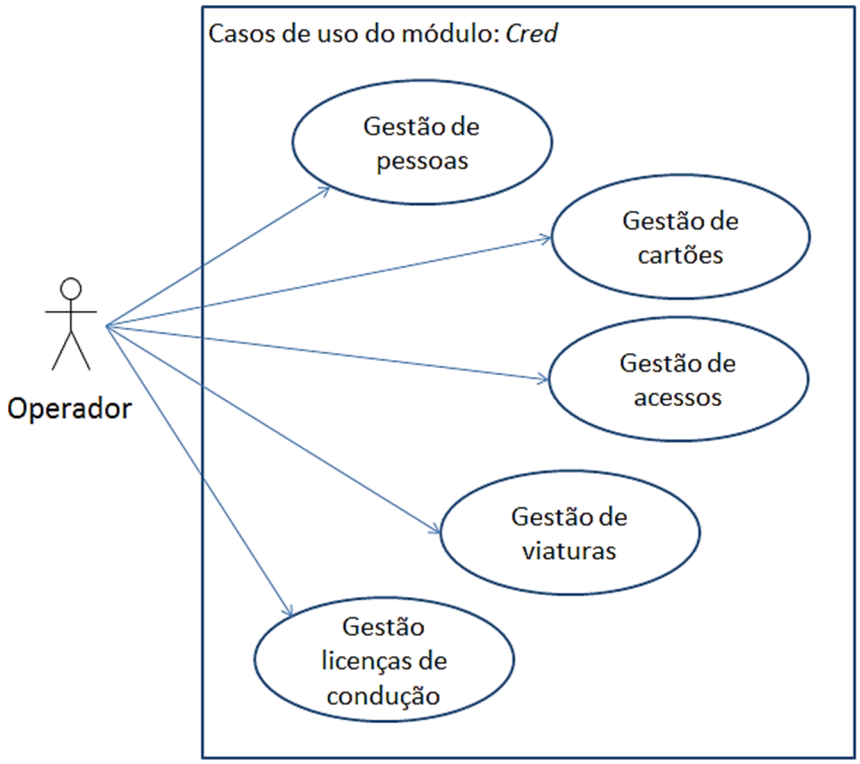


Figura 106 – Casos de uso do módulo *Cred*.

O caso de uso de gestão de pessoas, comporta as funcionalidades relacionadas com todos os aspetos que cada pessoa tem com o sistema, nomeadamente relacionados com o registo de dados pessoais e anexação de documentos, registos associados à atividade desenvolvida no aeroporto, informações relativas aos acessos atribuídos e aos cartões de que a pessoa é detentora, informações relativas à licença de condução e informações relativas às infrações cometidas e as respetivas penalidades.

O caso de uso de gestão de pessoas é apresentado de forma mais detalhada na Figura 107 e a seguir são apresentados os detalhes de desenvolvimento dos restantes casos de uso do módulo – *Cred*.

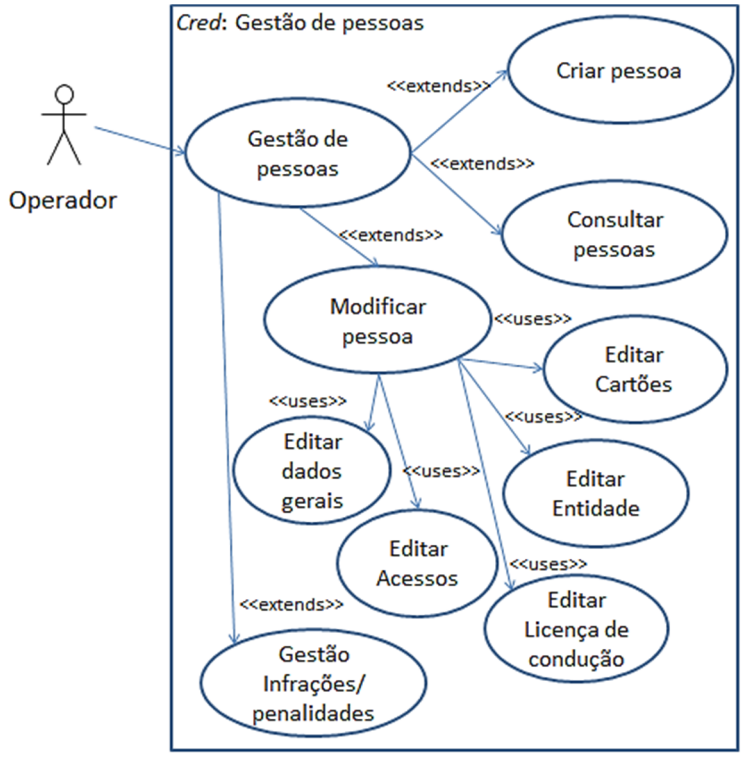


Figura 107 – Casos de uso do módulo *Cred* – Gestão de pessoas.

Caso de uso:	**Gestão de pessoas: Criar pessoa**
Objetivo:	Introduzir no sistema uma pessoa, para que lhe sejam concedidos privilégios no âmbito da credenciação.
Atores:	Pessoas com o perfil mínimo de "Operador".
Pré-Condição:	A pessoa em causa, apresentar a documentação exigida nos processos de credenciação em vigor.
Descrição:	O caso de uso começa quando o operador acede ao formulário de criação de pessoas no sistema. O operador preenche os campos obrigatórios do formulário e executa a função de criação. - Se a informação introduzida pelo operador for válida o sistema executa a operação de criação. - Se a informação não for válida o sistema apresenta uma mensagem a informar dos problemas encontrados e não executa a operação.
Notas:	Este caso de uso só deve ser realizado em condições excepcionais, porque o processo de criação de pessoas dever ser efetuado pelo portal "Cartão do Aeroporto".
Caso de uso:	**Gestão de pessoas: Editar dados gerais**
Objetivo:	Efetuar operações relacionadas com informação geral da pessoa.
Atores:	Pessoas com o perfil mínimo de "Operador".

Pré-Condição: A pessoa estar registada no sistema.

Descrição: Cenário principal

O caso de uso começa quando o operador acede ao formulário de edição de dados gerais, no qual pode definir ou alterar a morada, meios de contacto e adicionar anexos. O operador preenche e/ou altera os campos obrigatórios do formulário e executa a função de guardar informação.

- Se a informação introduzida pelo operador for válida o sistema executa a operação.

- Se a informação não for válida o sistema apresenta uma mensagem a informar dos problemas encontrados e não executa a operação.

Caso de uso: **Gestão de cartões**, Figura 108

Objetivo: Efetuar operações relacionadas com os cartões atribuídos as pessoas.

Atores: Pessoas com o perfil mínimo de "Operador".

Pré-Condição: A pessoa estar registada no sistema.

Descrição: Cenário principal

O caso de uso começa quando o operador acede ao formulário de edição de cartões, no qual pode associar um novo cartão à pessoa ou mudar o seu estado de ativo para não ativo. O operador preenche e/ou altera os campos obrigatórios do formulário e executa a função de guardar informação.

- Se a informação introduzida pelo operador for válida o sistema executa a operação.

- Se a informação não for válida o sistema apresenta uma mensagem a informar dos problemas encontrados e não executa a operação.

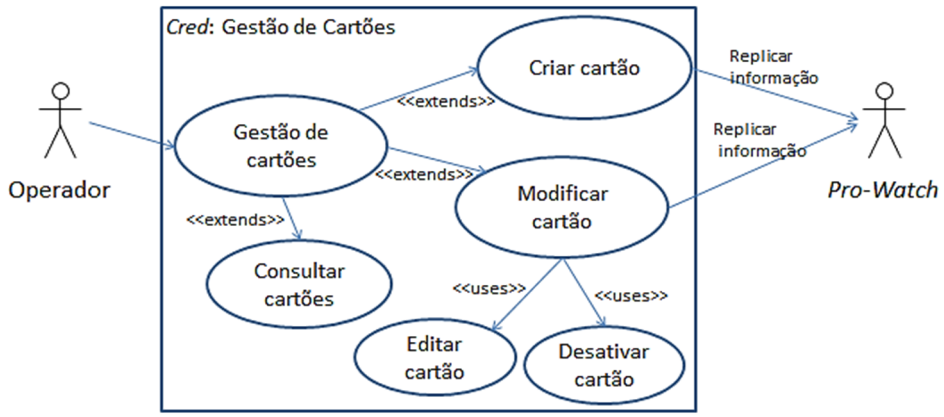


Figura 108 – Casos de uso do módulo *Cred* – Gestão de cartões.

Caso de uso: **Gestão de Acessos,** Figura 109

Objetivo: Efetuar operações relacionadas com permissões dos acessos a áreas reservadas.

Atores: Pessoas com o perfil mínimo de "Operador".

Pré-Condição: A pessoa estar registada no sistema.

Descrição: Cenário principal

O caso de uso começa quando o operador acede ao formulário de edição de acessos, no qual pode definir ou alterar as áreas e/ou as portas que uma pessoa tem acesso ou pode abrir. O operador preenche e/ou altera os campos obrigatórios do formulário e executa a função de guardar informação.

- Se a informação introduzida pelo operador for válida o sistema executa a operação.

- Se a informação não for válida o sistema apresenta uma mensagem a informar dos problemas encontrados e não executa a operação.

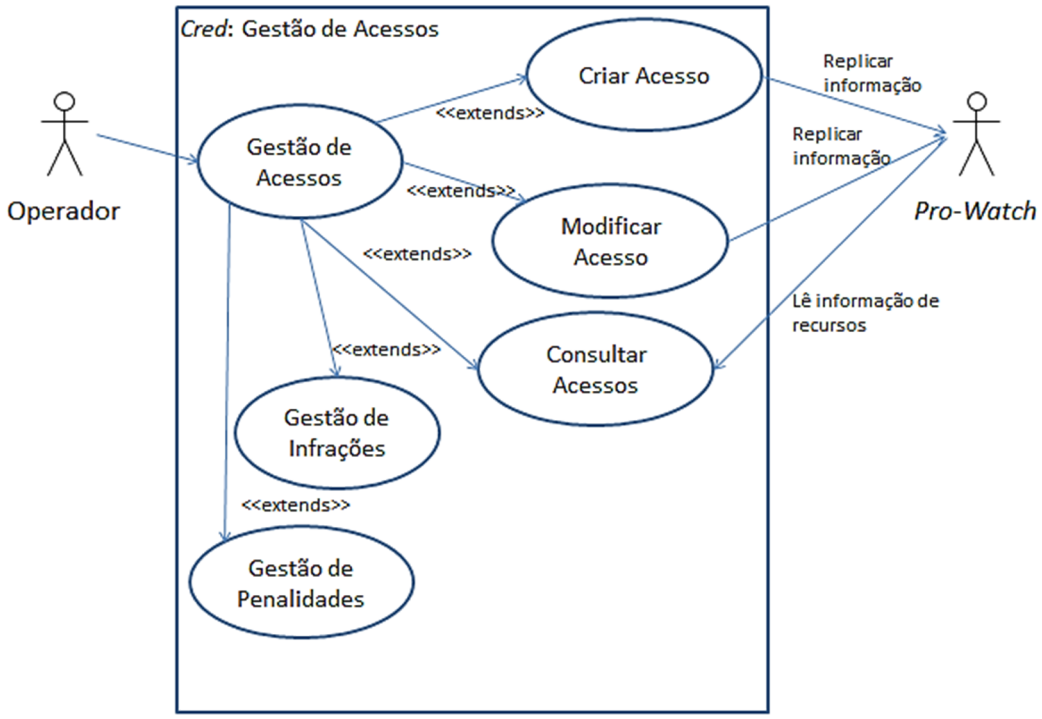


Figura 109 – Casos de uso do módulo *Cred* – Gestão de acessos.

Caso de uso: **Gestão de pessoas: Editar entidade**

Objetivo: Efetuar operações relacionadas com as funções laborais das pessoas.

Atores: Pessoas com o perfil mínimo de "Operador".

Pré-Condição: A pessoa estar registada no sistema.

Descrição: Cenário principal

O caso de uso começa quando o operador acede ao formulário de edição de Entidade, no qual pode definir ou alterar a informação relativa à entidade laboral que a pessoa pertence, qual a sua função e o serviço que presta. O operador preenche e/ou altera os campos obrigatórios do formulário e executa a função de guardar informação.

- Se a informação introduzida pelo operador for válida o sistema executa a operação.

- Se a informação não for válida o sistema apresenta uma mensagem a informar dos problemas encontrados e não executa a operação.

Caso de uso: **Gestão de Licença de condução**, Figura 110

Objetivo: Efetuar operações relacionadas com a licença de condução de veículos em áreas reservadas.

Atores: Pessoas com o perfil mínimo de "Operador".

Pré-Condição: A pessoa estar registada no sistema.

Descrição: Cenário principal

O caso de uso começa quando o operador acede ao formulário de edição de licença de condução, no qual pode definir ou alterar a informação relativa à licença de condução da pessoa, podendo efetuar a atribuição de uma licença, anexar documentos e ativar ou desativar a licença. O operador preenche e/ou altera os campos obrigatórios do formulário e executa a função de guardar informação.

- Se a informação introduzida pelo operador for válida o sistema executa a operação.

- Se a informação não for válida o sistema apresenta uma mensagem a informar dos problemas encontrados e não executa a operação.

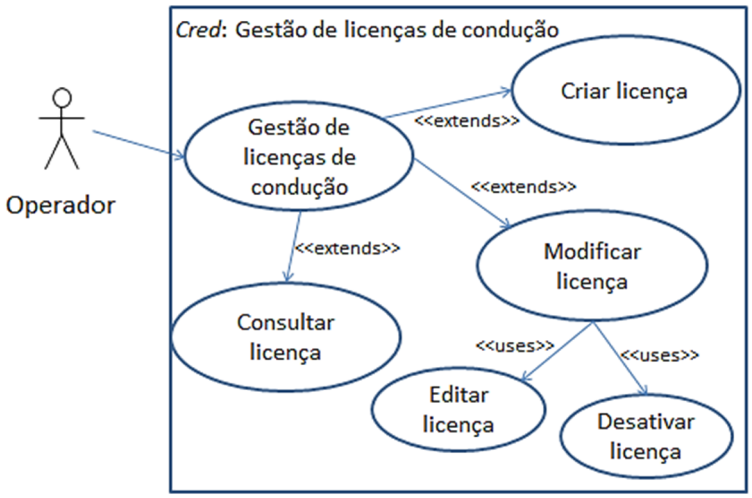


Figura 110 – Casos de uso do módulo *Cred* – Gestão de licenças de condução.

Caso de uso: **Gestão de pessoas: Gestão de infrações**

Objetivo: Efetuar operações relacionadas a gestão de infrações.

Atores: Pessoas com o perfil mínimo de "Operador".

Pré-Condição: A pessoa estar registada no sistema.

Descrição: Cenário principal

O caso de uso começa quando o operador acede ao formulário de gestão de infrações, no qual pode introduzir o registo de infrações relacionadas com acessos ou com a licença de condução e permite aplicar penalidades. O operador preenche e/ou altera os campos obrigatórios do formulário e executa a função de guardar informação.

- Se a informação introduzida pelo operador for válida o sistema executa a operação.

- Se a informação não for válida o sistema apresenta uma mensagem a informar dos problemas encontrados e não executa a operação.

3.4.1.2. CREDENCIAÇÃO DE VIATURAS

O caso de uso de gestão de viaturas, comporta as funcionalidades relacionadas com todos os aspetos de registo de viaturas que acedem a áreas restritas, nomeadamente relacionados com o registo de dados de identificação das viaturas, anexação de documentos, registos associados aos dísticos de identificação e informações relativas aos acessos, Figura 111.

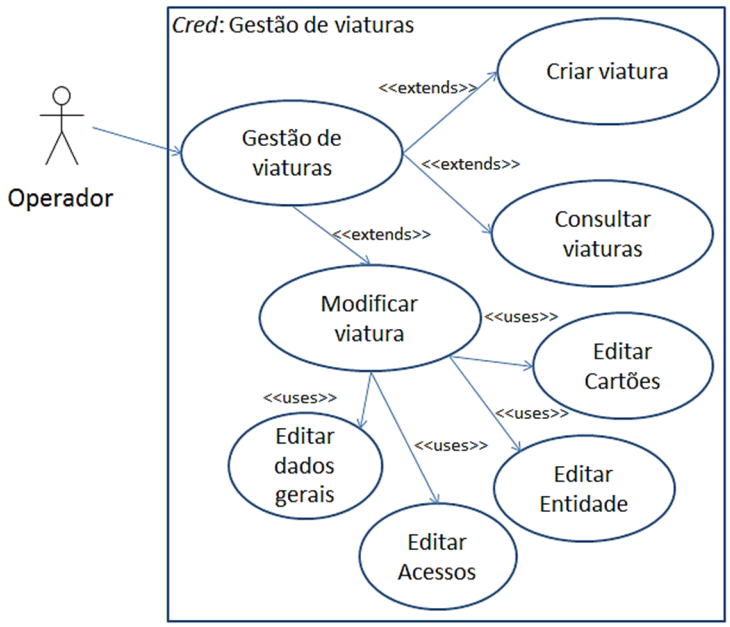


Figura 111 – Casos de uso do módulo *Cred* – Gestão de viaturas.

Caso de uso: **Gestão de viaturas**

Objetivo: Efetuar operações relacionadas com a gestão de viaturas.

Atores:	Pessoas com o perfil mínimo de “Operador”.
Pré-Condição:	A entidade detentora da viatura estar registada no sistema.
Descrição:	Cenário principal O caso de uso começa quando o operador acede ao formulário de gestão de viaturas, no qual pode introduzir novas viaturas, editar a informação existente. O operador preenche e/ou altera os campos obrigatórios do formulário e executa a função de guardar informação. - Se a informação introduzida pelo operador for válida o sistema executa a operação. - Se a informação não for válida o sistema apresenta uma mensagem a informar dos problemas encontrados e não executa a operação.

3.4.1.3. GESTÃO DE LICENÇAS DE CONDUÇÃO

O caso de uso de gestão de licenças de condução, comporta as funcionalidades relacionadas com todos os aspetos da licença de condução de viaturas nas zonas do lado ar do ASC, nomeadamente a atribuição de novas licenças, a edição da informação existente e a gestão de infrações e penalidades, Figura 112.

Caso de uso:	**Gestão de licenças de condução**
Objetivo:	Efetuar operações relacionadas a gestão de licenças de condução.
Atores:	Pessoas com o perfil mínimo de “Operador”.
Pré-Condição:	A pessoa a quem se vai atribuir a licença estar registada pelo menos com perfil de portador.
Descrição:	O caso de uso começa quando o operador acede ao formulário de gestão de licenças, no qual pode atribuir licenças a utilizadores já registados, editar a informação existente e fazer o registo de infrações. O operador preenche e/ou altera os campos obrigatórios do formulário e executa a função de guardar informação. - Se a informação introduzida pelo operador for válida o sistema executa a operação. Se a informação não for válida o sistema apresenta uma mensagem a informar dos problemas encontrados e não executa a operação.

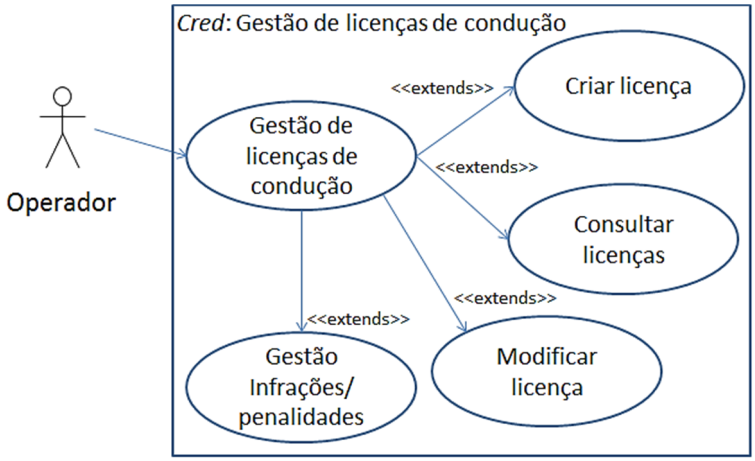


Figura 112 – Casos de uso do módulo *Cred* – Gestão de licenças de condução.

3.4.1.4. GESTÃO DE UTILIZADORES DO SISTEMA

O caso de uso de gestão de utilizadores permite executar todas as funcionalidades relativas à criação de utilizadores com permissões de uso dos módulos que constituem a plataforma de credenciação, nomeadamente criação de utilizadores e atribuição de perfis, Figura 113.

Caso de uso:	**Autenticação de utilizador: *Login***
Objetivo:	Reconhecer o utilizador com permissões e perfil para usar o sistema.
Atores:	Administrador, Gestor, Operador, Agente e Vigilante.
Pré-Condição:	O utilizador estar registado no sistema com um dos perfis de operação e ser detentor da respetiva "Palavra-chave" de acesso ao sistema.
Descrição:	Cenário principal O caso de uso começa quando o utilizador acede ao formulário de acesso ao sistema. O utilizador identifica-se preenchendo os campos "Número de utilizador" e "Palavra-chave", ou identificando-se através do seu cartão RFID. O utilizador confirma a sua identificação através do botão "Login". - Se a identificação do utilizador for válida o sistema apresenta o ambiente de trabalho relativo ao perfil do utilizador. - Se a identificação não for válida o sistema apresenta uma mensagem a informar que as credenciais não são validas e volta ao início.
Pós-Condição:	Apresentação da interface com a configuração associada ao perfil do utilizador.

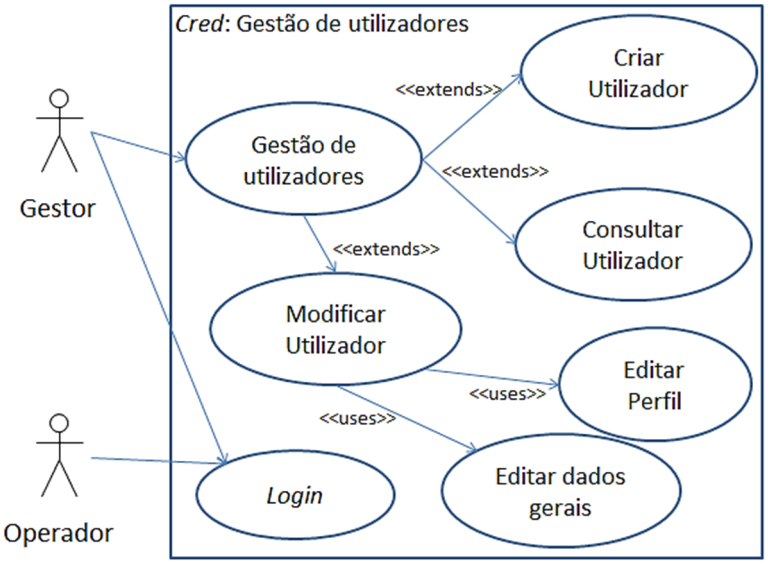


Figura 113 – Casos de uso do módulo *Cred* – Gestão de utilizadores.

Caso de uso:	**Gestão de utilizadores de módulos da plataforma**
Objetivo:	Efetuar operações relacionadas a gestão de utilizadores dos módulos que constituem a plataforma.
Atores:	Pessoas com o perfil mínimo de "Gestor".
Pré-Condição:	O utilizador estar registado no sistema no mínimo com o perfil de utilizador.
Descrição:	Cenário principal O caso de uso começa quando o gestor acede ao formulário de gestão de utilizadores, no qual pode criar utilizadores e atribuir e/ou alterar perfis a utilizadores já registados. O gestor preenche e/ou altera os campos obrigatórios do formulário e executa a função de guardar informação. - Se a informação introduzida pelo gestor for válida o sistema executa a operação. Se a informação não for válida o sistema apresenta uma mensagem a informar dos problemas encontrados e não executa a operação.

3.4.1.5. CONFIGURAÇÃO DO SISTEMA

O caso de uso de configuração do sistema representa a execução das funcionalidades relativas à configuração da instalação dos diversos módulos, nomeadamente as configurações de indicação da localização do servidor da base de dados, configurações de

acesso de portarias e configurações relativas às portas COM usadas nas portarias para leitura de cartões, Figura 114.

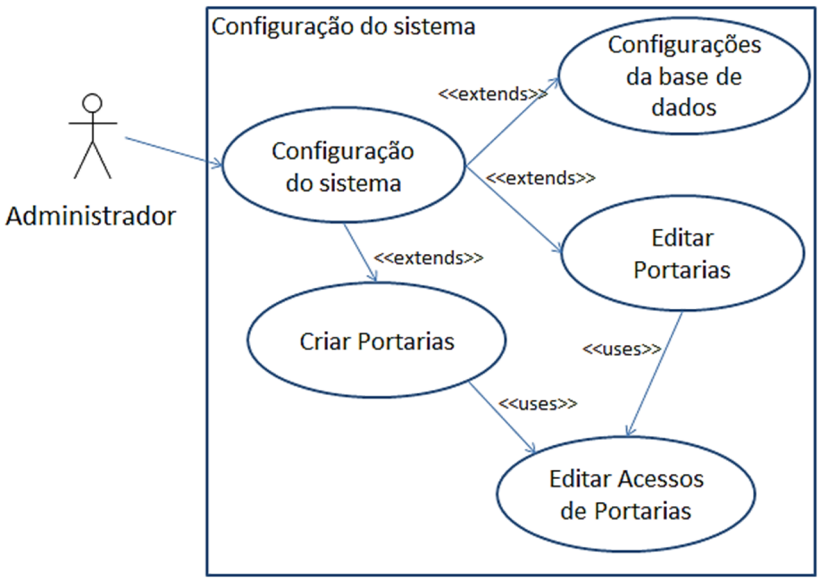


Figura 114 – Casos de uso – Configuração do sistema.

Caso de uso:	**Configuração do sistema**
Objetivo:	Operações de configuração da instalação dos módulos da plataforma.
Atores:	Administrador.
Descrição:	Cenário principal O caso de uso começa quando o Administrador acede ao formulário de gestão do respetivo módulo o administrador preenche e/ou altera os campos obrigatórios do formulário e executa a função de guardar informação. - Se a informação introduzida pelo gestor for válida o sistema executa a operação. Se a informação não for válida o sistema apresenta uma mensagem a informar dos problemas encontrados e não executa a operação.

3.4.1.6. MÓDULO DE CREDENCIAÇÃO PONTUAL

O módulo de credenciação pontual – *Pontu*, é a aplicação da plataforma usada pelos agentes da PSP para credenciação de visitantes ou passageiros de chegadas. A função deste módulo é o registo de informação das pessoas a credenciar e a emissão do respetivo cartão

de acesso. A Figura 115 apresenta do diagrama de sequência das ações desenvolvidas pelo *Pontu*. A Figura 116 mostra o caso de uso desta aplicação.

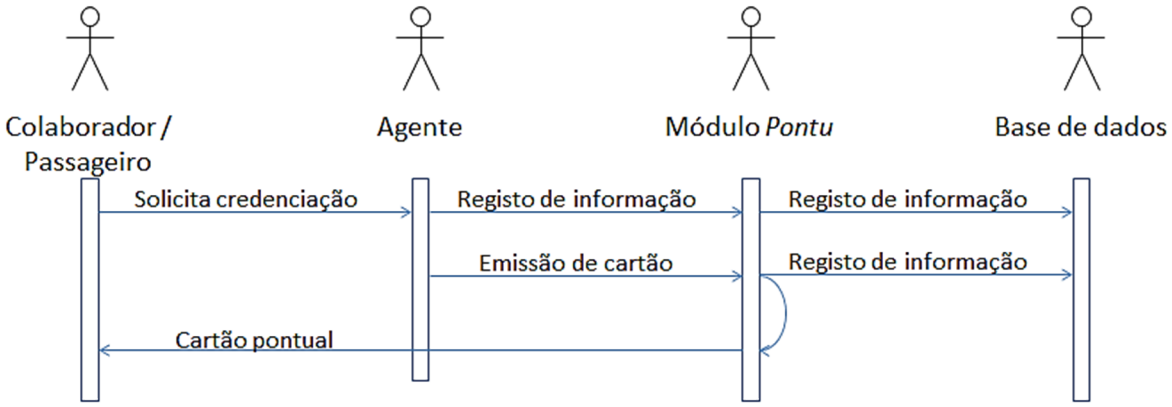


Figura 115 – Diagrama de sequência do módulo *Pontu*.

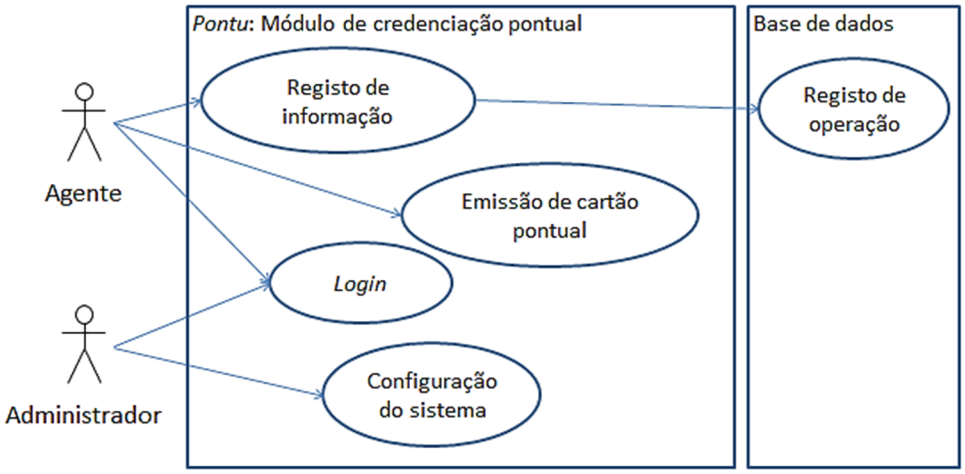


Figura 116 – Casos de uso do módulo *Pontu*.

Caso de uso:	**Credenciação pontual**
Objetivo:	Efetuar operações relacionadas com a credenciação de visitantes.
Atores:	Agente.
Pré-Condição:	---
Descrição:	Cenário principal O caso de uso começa quando o agente acede ao formulário de credenciação pontual e preenche os campos obrigatórios do formulário, executa a função de guardar informação e emissão de cartão de acesso pontual.

- Se a informação introduzida pelo gestor for válida o sistema executa a operação. Se a informação não for válida o sistema apresenta uma mensagem a informar dos problemas encontrados e não executa a operação

3.4.1.7. MÓDULO DE PORTARIAS

O módulo *Port* é a aplicação da plataforma para ser instalada no computador presente nas portarias e que será usado pelo vigilante da portaria que faz o rastreio de passagem. Esta aplicação vai receber do leitor de cartões o número do cartão lido, vai consultar a base de dados central sobre se o utilizador tem permissões de entrada na zona que a portaria controla e apresenta ao vigilante o resultado da consulta. A Figura 117 apresenta a sequência de operações de funcionamento do módulo *Port* e na Figura 118 são apresentados os casos de uso deste módulo.

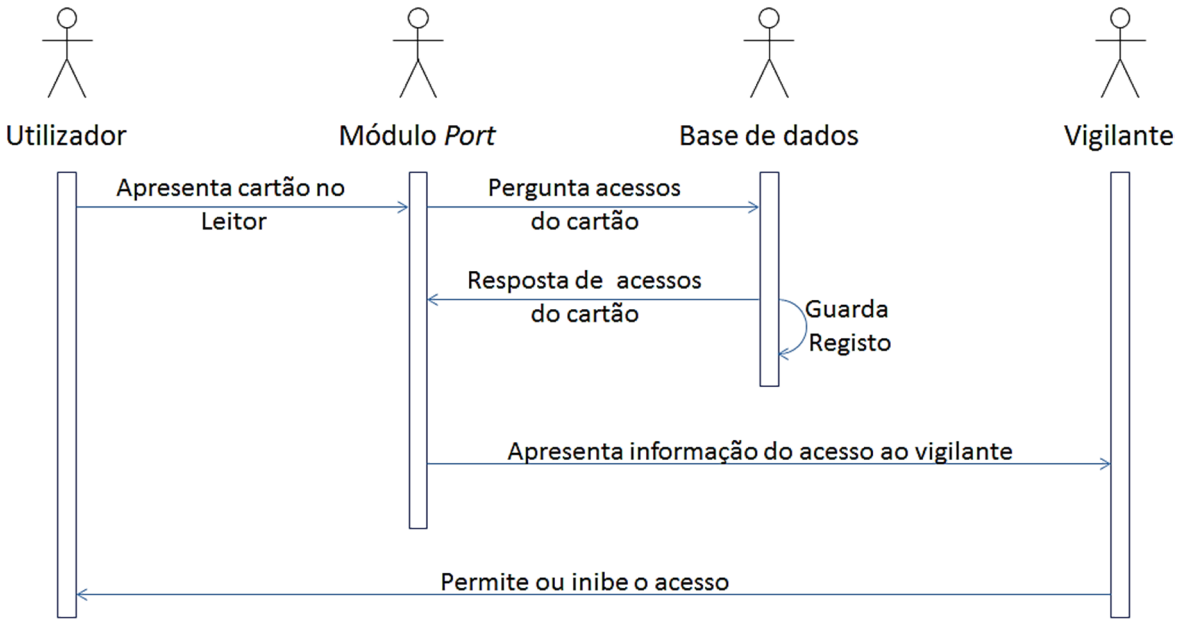


Figura 117 – Diagrama de sequência do módulo *Port*.

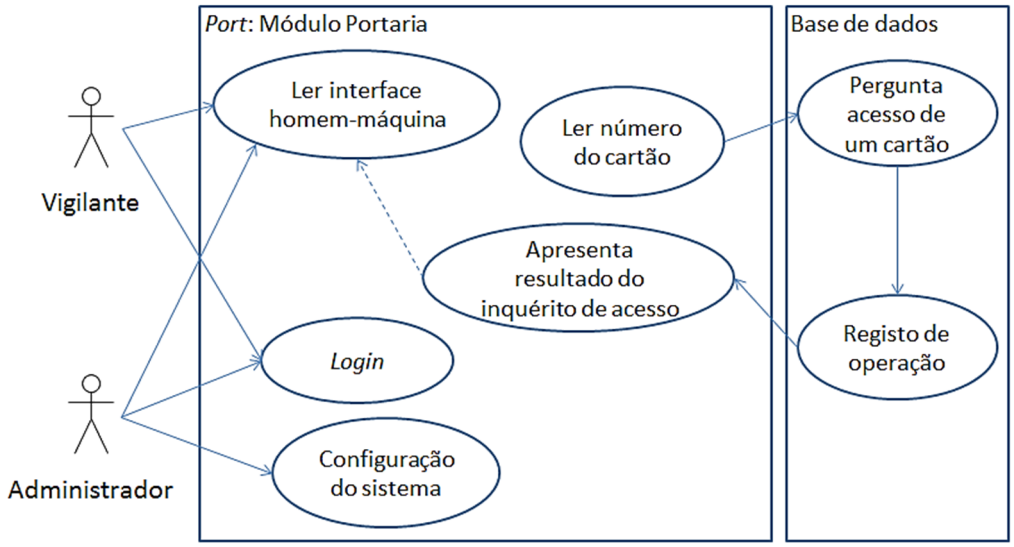


Figura 118 – Casos de uso do módulo *Port*.

Caso de uso:	**Lê o cartão de acesso**
Objetivo:	Lê o número do cartão RFID e pergunta à base de dados se o portador do cartão tem acesso à área que a portaria controla
Atores:	Vigilante
Pré-Condição:	Ter sido apresentado um cartão no leitor.
Descrição:	Cenário principal O caso de uso começa quando a aplicação recebe um número de cartão enviado pelo leitor de cartões e faz uma consulta à base de dados sobre o acesso do cartão.
Pós-Condição:	--
Caso de uso:	**Apresentar resultado do inquérito de acesso**
Objetivo:	Apresenta o resultado de uma consulta à base de dados sobre a permissão de acesso de um cartão numa portaria específica.
Atores:	Vigilante
Pré-Condição:	-
Descrição:	Cenário principal O caso de uso começa quando a aplicação recebe o resultado da consulta do nível de acesso de um cartão. Face a esse resultado, na interface homem-máquina apresenta: A identificação do utilizador: nome, número do cartão e foto. Acessos válidos para o utilizador. Sinalização indicando se o utilizador pode ou não passar a portaria em causa.

3.4.2. DIAGRAMA DE CLASSES

Para suportar a implementação dos casos de uso foi desenvolvido um modelo de objetos, representado em diagramas de classes, onde se definem as entidades, os conceitos e as respetivas interdependências. Na Tabela 9, estão listadas as entidades identificadas da análise dos casos de uso.

Tabela 9 – Classes identificadas para a implementação da plataforma.

Classe	Descrição	Relações
Pessoa	Para conter a informação e as atividades relacionadas com as pessoas envolvidas no sistema.	Em cada momento cada pessoa tem um perfil ativo. Cada pessoa tem zero ou mais acessos atribuídos. Cada pessoa tem um ou mais cartões associados, sendo que em cada momento apenas um está ativo.
Acesso	Para conter a informação sobre permissões de acessos a áreas restritas.	São atribuídos a pessoas e a portarias.
Cartão	Para conter a informação e as atividades relacionadas com os cartões físicos RFID.	Associado a pessoas e a viaturas.
Licença de condução	Para conter a informação e as atividades relacionadas com as licenças de condução em áreas reservadas.	Associado a pessoas.
Entidade	Para conter a informações sobre empresas que operam no Aeroporto.	Associado a pessoas e a viaturas.
Morada	Para conter informação sobre endereço postal.	Associado a pessoas e a entidades.
Infração	Para conter a informação e as atividades relacionadas infrações.	Associado a pessoas.
Perfil	Identificação de tipo de utilizador sobre os recursos da plataforma.	Associado a pessoas.
Portaria	Para conter as atividades relacionadas com portarias.	Cada portaria tem um ou mais acessos.
Registo	Para conter a informação relativa ao registo das operações e eventos que ocorrem sobre o sistema.	Relações com todas as outras entidades.

Na Figura 119 é apresentado o diagrama de classes que modela as entidades geridas pela plataforma nas diferentes aplicações.

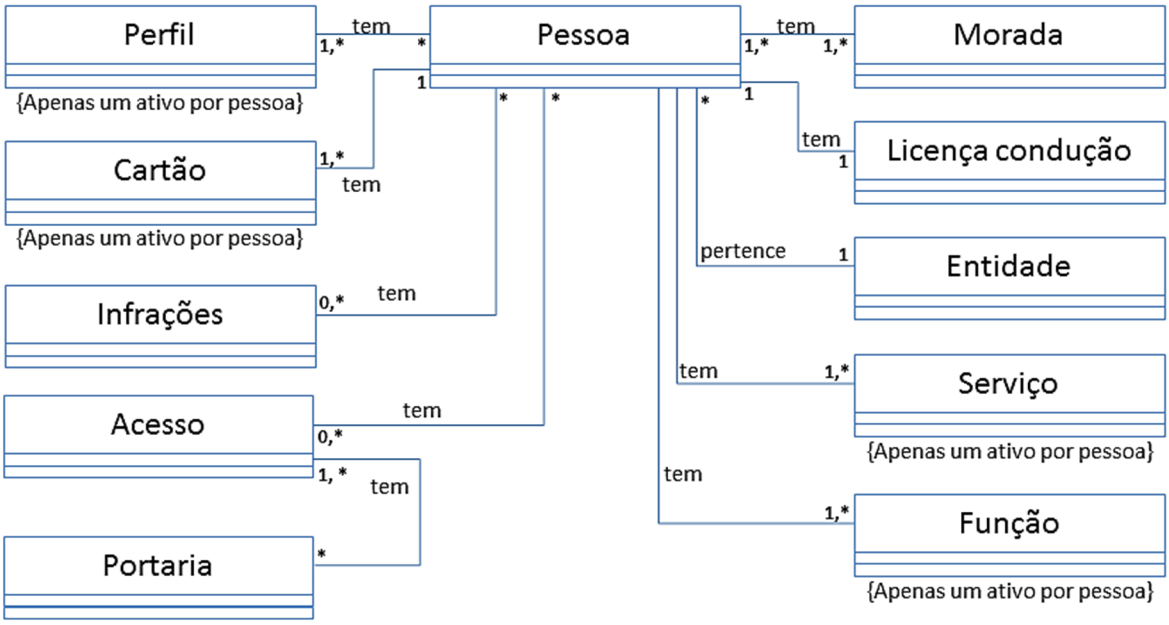


Figura 119 – Diagrama de classes – Relações com pessoas e portarias.

Os requisitos dos sistemas de segurança exigem que todas as operações e eventos ocorridos sejam registados. Estes registos têm de contemplar todas as operações de criação, alteração e associação de dados, todos os eventos de *login, logout* e todas passagens nas portarias. Para gerir esta informação foi definida uma classe *Registo*.

A necessidade de registo de todos acontecimentos no sistema induz a existência de registos de diversos tipos associados a diferentes situações. Este facto implica diferentes especificidades para cada tipo de registo. A diferenciação das especificidades de registo vai ser implementada com uma estrutura do tipo generalização-especialização. Do lado a generalização temos a entidade *Registo* que representa as operações e os eventos que ocorrem. Do lado da especialização surgem entidades que guardam a informação relativa a um tipo específico de registo.

Por exemplo, a apresentação de um cartão numa portaria envolve a entidade *Cartão* que é lido, envolve a entidade *Pessoa* detentora do cartão e envolve a entidade *Pessoa* que está de vigia na portaria, envolve à entidade *Portaria* e envolve a entidade *Acesso* associada à entidade *Pessoa* e à *Portaria*. A este evento está associado a um tipo especificao de registo. Outro exemplo, no evento de atribuição de um cartão a uma pessoa estão envolvidas: a entidade *Pessoa* portadora do cartão, a entidade *Pessoa* que efetua a operação de atribuição do cartão e a entidade *Cartão*. Sendo que este evento, também está associado a um tipo de registo especifico mas diferente do exemplo anterior.

A Figura 120 apresenta o diagrama de classes da estrutura generalização-especialização para registo de operações. De notar que, no diagrama foi introduzida uma classe chamada *Tipo de Registo*, para conter um dicionário de classificação de tipos de registos, este recurso vai ser útil posteriormente, quando se pretender implementar pesquisas e análises aos dados guardados.

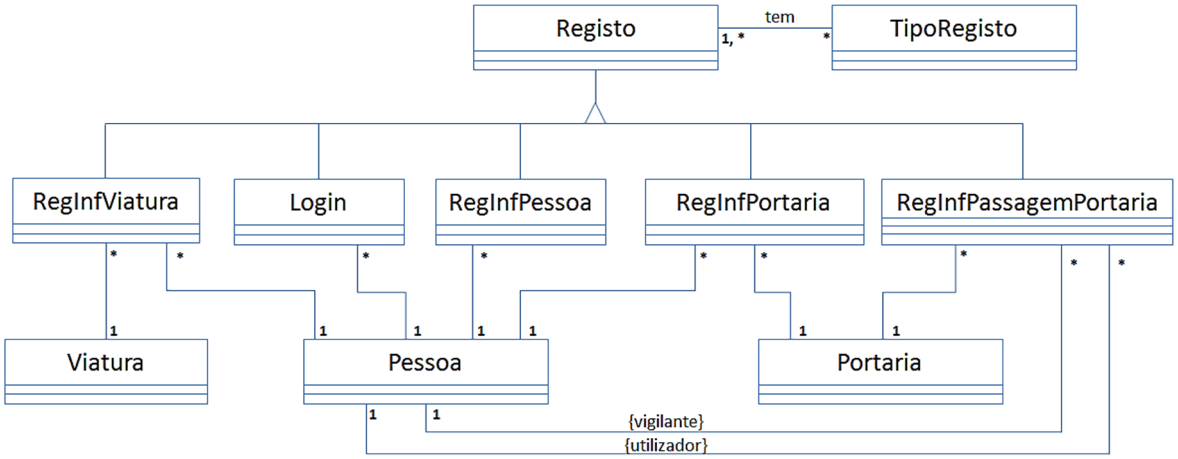


Figura 120 – Diagrama de classes de registo de eventos.

Os registos de operações ou eventos são sempre efetuados sobre as instâncias de classes que tem representação física como as Pessoas, as Viaturas e as Portarias, nas correspondentes *RegInfPessoa*, *RegInfViatura*, e *RegInfPortaria*. A passagem de uma pessoa numa portaria resulta num registo na entidade *RegInfPassagemPortaria*. Os eventos

de alteração: do perfil, dos acessos, da morada, de cartão, da licença de condução de infrações de uma pessoa, resultam num registo na entidade *RefInfPessoa*. A Tabela 10 apresenta o resumo descritivo das classes envolvidas no registo de acontecimentos que ocorrem na plataforma.

Tabela 10 – Classes para registo de eventos do sistema.

Classe	Descrição	Relações
Registo	Classe com os conteúdos gerais da relação generalização-especialização de registo de eventos.	Classes de especialização.
Tipo de Registo	Dicionário de classificação de tipos de registo, para classificação dos acontecimentos.	Registo.
RegInfPessoa	Para gerir a informação relativa a eventos relacionados com as Pessoas, nomeadamente: criação e alteração de dados.	Pessoa, Acessos, Cartão, Infrações, Licença de condução, Entidade, Viatura, Perfil.
RegInfViatura	Para gerir a informação relativa a eventos relacionados com as Viaturas, nomeadamente: criação e alteração de dados.	Viatura, Entidade.
RegInfPortaria	Para gerir a informação relativa a eventos relacionados com as Portarias, nomeadamente: criação e alteração de dados.	Pessoa, Acessos.
RegInfPassagemPortaria	Para gerir a informação relativa a eventos relacionados com as apresentações de cartões em portarias.	Pessoa, Portaria, Acessos e Cartão.

3.4.3. MODELO DE DADOS

No subcapítulo 3.2 – Seleção de ferramentas de desenvolvimento, foi definido que a ferramenta a ser usada para implementação da base de dados é o *SQL Server*, as bases de dados permitidas nessa plataforma caracterizam-se serem aplicações do paradigma relacional. A definição do modelo relacional de armazenamento de dados a ser desenvolvida para este projeto, foi efetuada considerando o modelo de classes definido na secção anterior em conjunto com as descrições dos casos de uso.

A transposição dos diagramas de classes para o diagrama relacional, requere algumas considerações, nomeadamente:

- As relações entre as entidades *Pessoa* e *Portaria* com a entidade *Acesso* são relações muitos-para-muitos, este facto na implementação relacional exige o uso de uma tabela extra que contenha a relação entre uma pessoa ou portaria e um acesso específico. Esta tabela extra também via ser usada para guardar a informação das datas em que relação é válida.
- Associado às relações entidades *Pessoa* e *Portaria* com a entidade *Acesso*, vai ser usado uma propriedade denominada *Ativo*, para se fazer inibições temporárias de acessos, sem ter de fechar e criar novas instâncias.
- Em cada momento, cada pessoa tem um perfil atribuído, que no diagrama de classes está definida como uma relação um-para-muitos. No entanto, ao longo do tempo, o perfil atribuído à pessoa pode ser alterado. Para manter o registo histórico destas ocorrências, a relação um-para-muitos vai ser transformada numa relação muitos-para-muitos em que, em cada instante, apenas um perfil está no estado *Ativo*, esta implementação também vai ser efetuada à custa de uma tabela extra.
- Os cartões RFID são intransmissíveis, desta forma a relação entre a entidade *Pessoa* e a entidade *Cartão* é sempre uma relação um-para-muitos e por isso as informações de datas de atribuição e de validade assim como se o cartão está ativo ou não, são guardadas na tabela do cartão.
- A estrutura generalização-especialização, do lado da informação geral tem a data do evento que o despoletou e o tipo de evento, do lado da especialização tem as informações particulares do tipo de evento. Por questões de simplicidade de implementação do modelo de dados no paradigma relacional e de construção dos *queries SQL*, vai-se implementar uma tabela para cada caso de especialização que também inclui a informação contida na generalização.
- Para o funcionamento do sistema existem listas de caracter estático que caracterizam determinadas particularidades do sistema, com por exemplo a lista de perfis de utilizador. As tabelas que contem este tipo de informação vão ser denominadas tabelas-dicionário, para transmitir a ideia de lista de itens de caracter pouco alterável.

Os diagramas apresentados nas Figura 121 à Figura 130, apresentam as diferentes vertentes do modelo relacional para ser implementado no *SQL Server*.

A Figura 121, mostra o diagrama de tabelas relacionadas com as informações associadas às pessoas que vão ser registadas no sistema. Além dos dados pessoais como nome, filiação, etc. são registados os meios de contacto, a morada, assim como as informações relacionadas com a função que a pessoa desempenha. O campo de identificação da pessoa "Id" vai conter o número do documento que identifica a pessoa, que pode ser do tipo passaporte, cartão do cidadão ou bilhete de identidade, esta lista de possibilidades está definida na tabela "TipoDocumentoID". Associado às pessoas podem ser introduzidos, na tabela "Anexo", ficheiros relativos ao processo pessoal.

A tabela "Tipo de pagamento" contém a lista de possibilidades de tipo de pagamentos associados ao processo de credenciação das pessoas. As tabelas "Serviço" e "Funções", são listas – dicionários de serviços e funções relacionadas com a vertente laboral da pessoa.

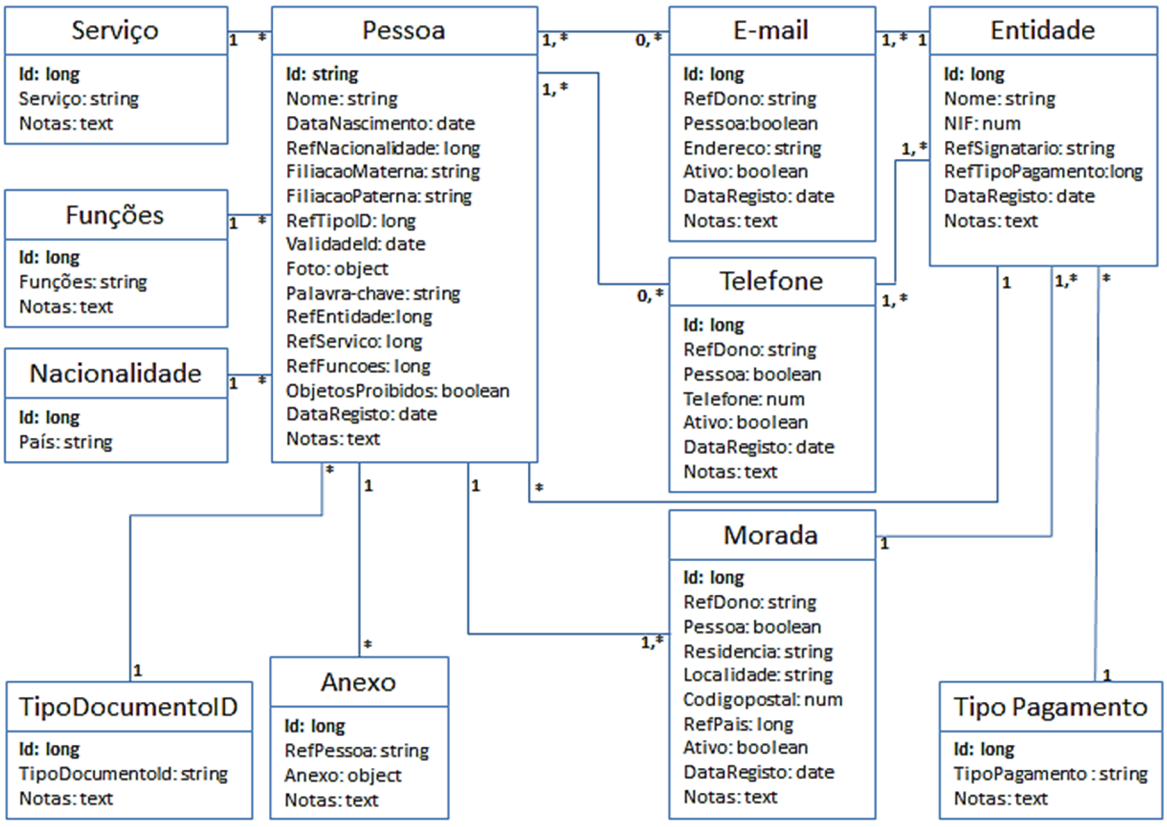


Figura 121 – Diagrama relacional: Dados pessoais.

A Figura 122 apresenta a estrutura de dados que implementa a associação de pessoas, a perfis e a cartões. Em cada momento, as pessoas tem um perfil ativo, mas ao longo do tempo podem usar diferentes perfis, esta relação muitos-para-muitos é implementada pela tabela "Validade", que para cada pessoa, em cada momento, apenas tem um perfil ativo, mas mantem o histórico dos perfis anteriormente atribuídos. A tabela "Cartão" faz a associação de cartões a pessoas, com a mesma restrição do perfil de utilizador: em cada momento apenas um cartão tem o estado válido por pessoa, como cada cartão só pode ser atribuído a uma pessoa, neste caso temos uma relação um-para-muitos, que é implementada usando a tabela "Cartão".

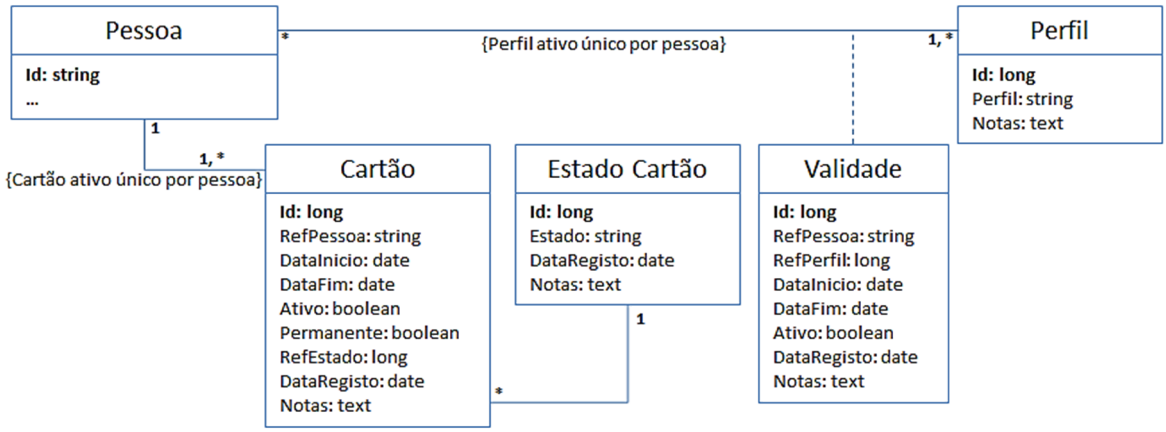


Figura 122 – Diagrama relacional: Atribuição de cartões e perfis de utilizador.

A ANA adquire cartões de identificação em lotes com números exclusivos. Para que haja um controlo na atribuição de novos cartões a pessoas, a base de dados contem a informação dos lotes de cartões ativos e apenas permite registo de cartões que pertençam a esses lotes, Figura 123.

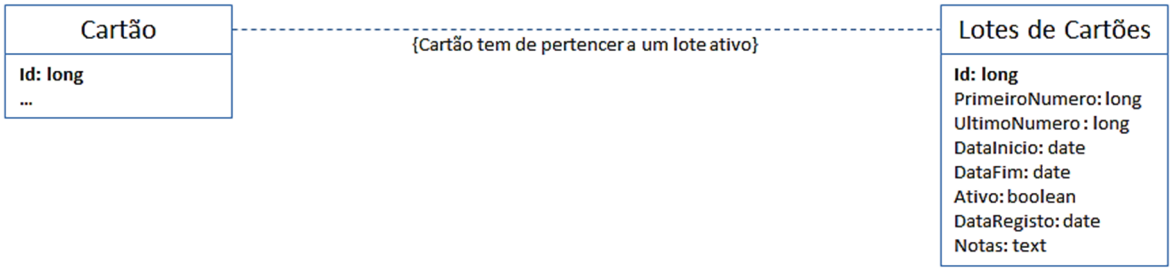


Figura 123 – Diagrama relacional: Lotes de cartões.

A Figura 124 descreve a estrutura de dados que suporta a funcionalidade de atribuição de acessos a pessoas e a portarias. Em cada momento cada pessoa e portaria podem ter atribuído vários acessos e os acessos ativos podem variar ao longo do tempo, estas relações muitos-para-muitos entre as entidades *Pessoa* e *Portaria* com a entidade *Acessos*, é implementada com as tabelas "Validade Acesso Pessoa" e "Validade Acesso Portaria", respetivamente. As tabelas de validade do acesso definem o período de tempo em que o acesso está atribuído e esta informação em conjunto com o valor do campo "Ativo", permitem manter o histórico das atribuições.

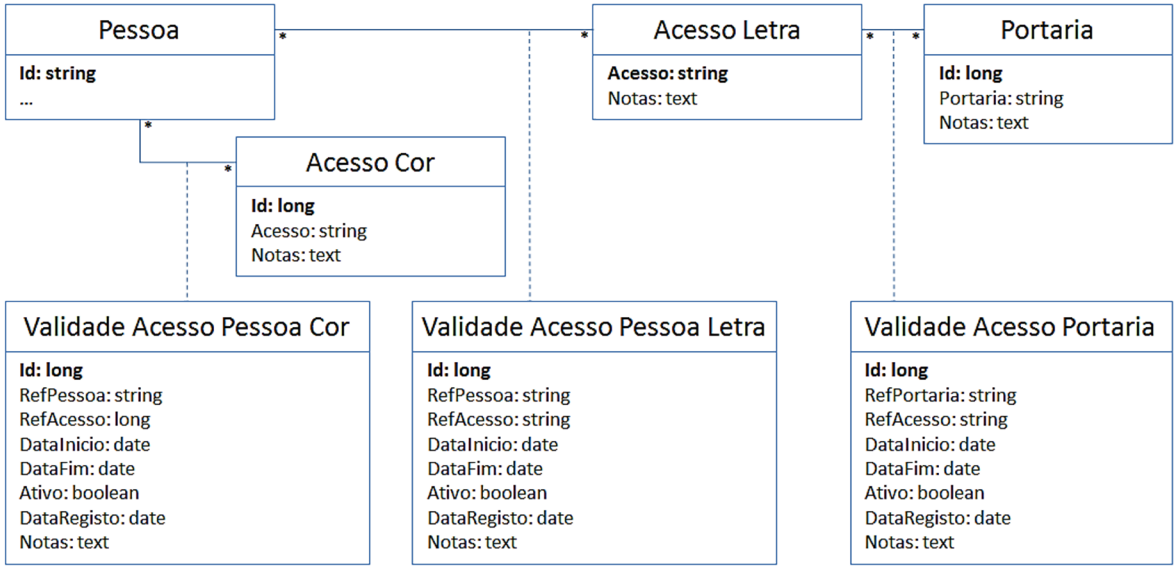


Figura 124 – Diagrama relacional: Acessos.

No uso das suas credenciações de acesso, as pessoas podem incorrer em incumprimentos como deixar portas abertas, dar acesso de passagem a terceiros, etc. Os incumprimentos, quando detetados dão origem a penalidades que podem ir de simples advertências, até à retirada do cartão. O diagrama mostrado na Figura 125 apresenta a forma como o registo da informação sobre infrações e penalidades é implementada. As tabelas "Tipo Infração" e "Tipo Penalidade", são tabelas-dicionário que refletem as listas de infrações e penalidade vigentes nos procedimentos em vigor.

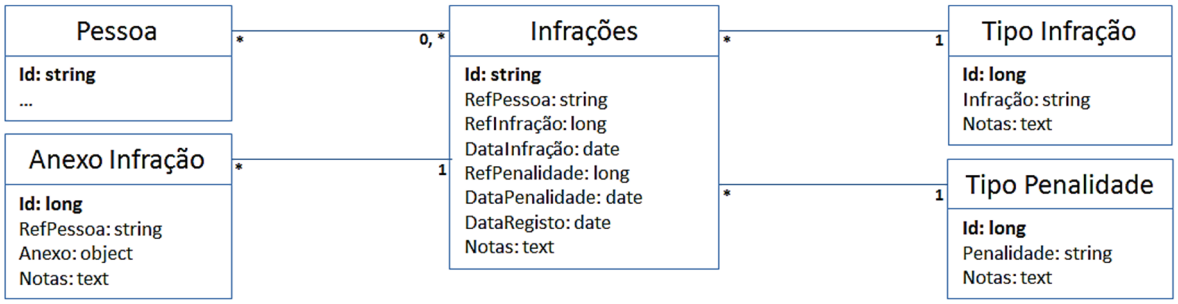


Figura 125 – Diagrama relacional: Infrações penalidades.

A Figura 126, mostra implementação do registo de dados relacionado com as licenças de condução. Esta estrutura além da informação sobre a licença, faz também o armazenamento de informação sobre as renovações da licença e sobre as infrações e penalidades no âmbito da condução no lado ar.

A tabela "Tipo Carta Condução", faz o registo do tipo de carta de condução emitida no país de origem do portador. A tabela "Tipo de Licença Condução", determina o tipo licença de condução e consequentemente o tipo de viaturas que o portador pode manobrar nas áreas restritas.

Esta estrutura também permite guardar documentos associados à licença, funcionalidade particularmente útil nos processos de renovação.

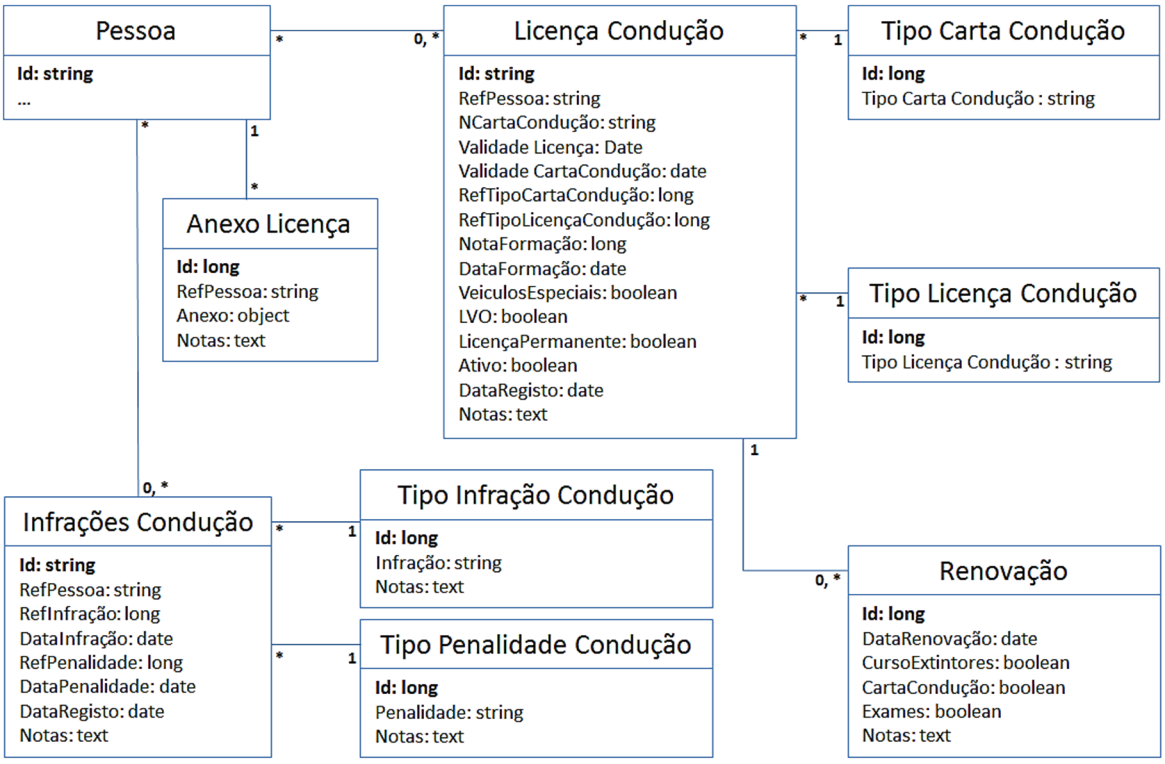


Figura 126 – Diagrama relacional: Licença de condução

A estrutura de suporte da informação relativa a viaturas é apresentada na Figura 127. A informação guardada é relativa às características do veículo, à entidade que o possui, ao serviço que presta e á zona onde normalmente está estacionada.

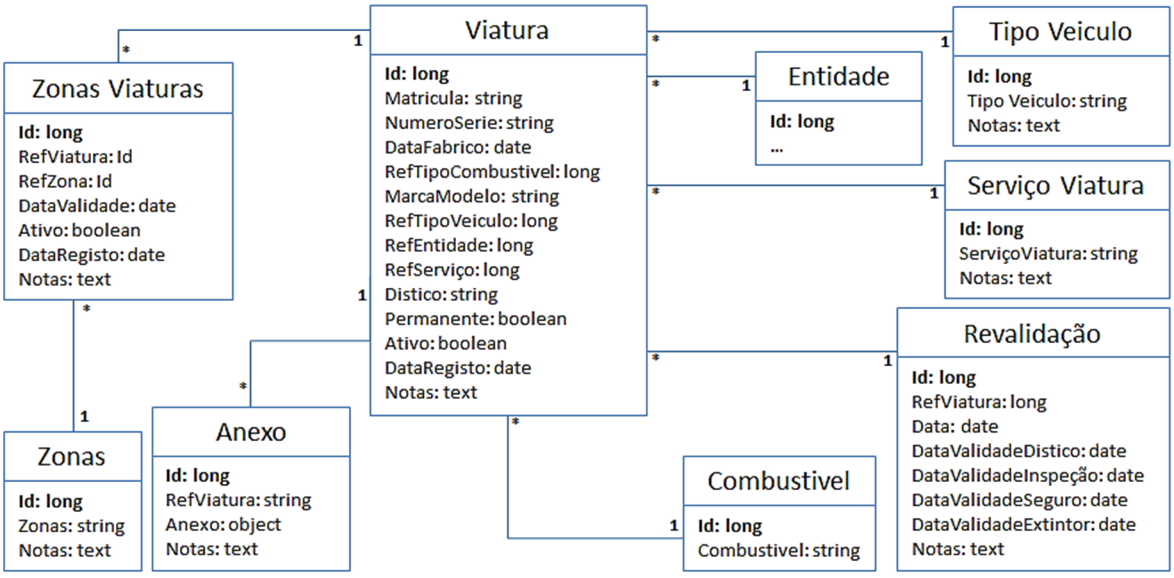


Figura 127 – Diagrama relacional: Viaturas

Como explicado, os casos de uso dos cartões pontuais têm duas vertentes: uma para visitas e outra para passageiros. A Figura 128 mostra o digrama de implementação de tabelas para armazenamento de informação relativa aos cartões pontuais associados a visitas. Esta estrutura guarda informação sobre a pessoa que solicita a credenciação, o visitante, a entidade que representa, o motivo da visita, as portarias em que está autorizado a passar para zonas restritas e as áreas a que tem acesso.

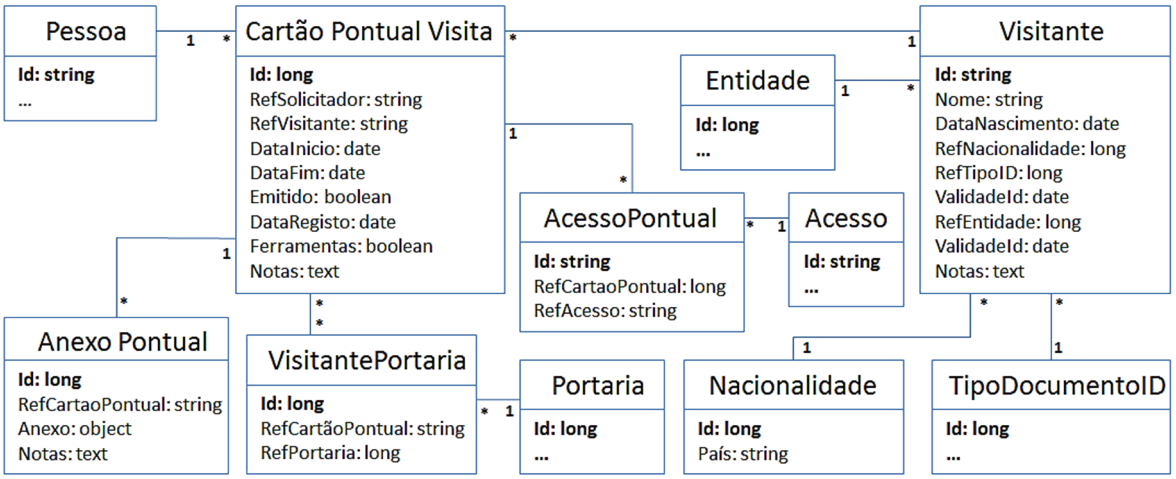


Figura 128 – Diagrama relacional: Cartões pontuais, visitas

Na credenciação pontual para passageiros, Figura 129, faz-se o registo do passageiro, do voo que deu origem à credenciação e a validade do cartão.

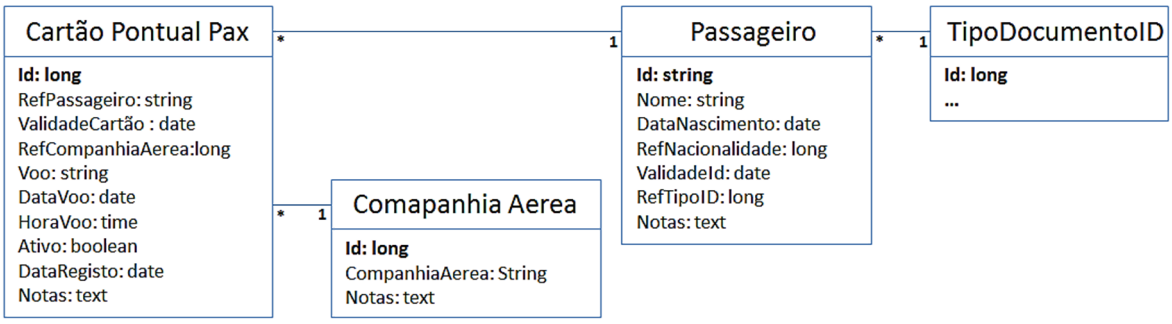


Figura 129 – Diagrama relacional: Cartões pontuais, passageiros

Para efeitos de auditoria e análise de eventos, a plataforma a implementar vai fazer o registo de todas as atividades que ocorrem no sistema, nomeadamente relacionadas com alteração de dados de pessoas, viatura e portarias e também efetua registos de entrada e

saída dos utilizadores no sistema assim como o registo de passagens nas portarias. Este registo é efetuado nas tabelas apresentadas no diagrama da Figura 130.

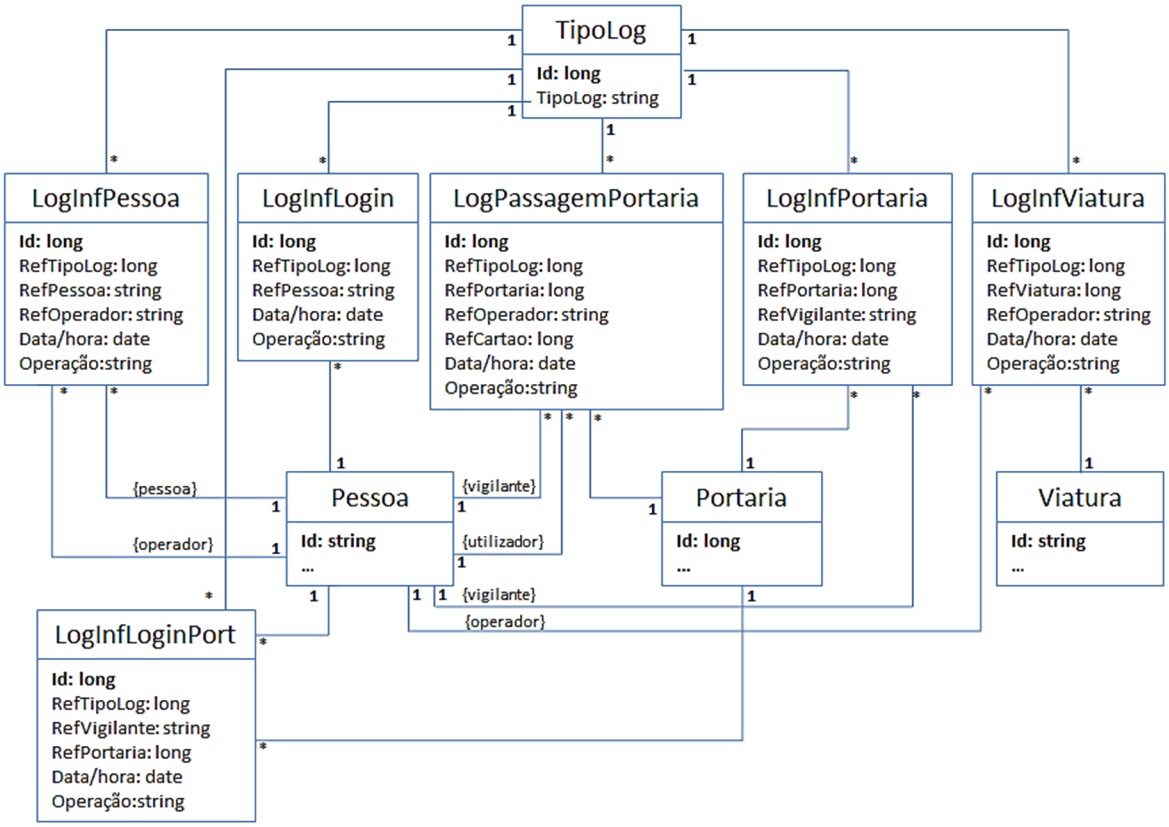


Figura 130 – Diagrama relacional: Registos

A implementação do modelo relacional descrito, envolve a construção, no *SQL Server*, de uma estrutura de dados, constituída pelas tabelas e respetivas relações descritas nos diagramas apresentados. A Tabela 11, mostra o exemplo da definição de uma tabela de dados a implementar. Esta forma de apresentação descreve a funcionalidade da tabela, a definição dos campos que a constituem, o seu significado, assim como os campos são usados como chave. No Anexo I são apresentadas as estruturas de todas as tabelas a serem criadas na base de dados para implementar o modelo relacional de suporte à plataforma, nesse anexo, usa-se a metodologia descritiva apresentada para a Tabela 11.

Tabela 11 – Tabela de relação entre a entidade Pessoa e a entidade Perfil – *tblPessoa-Perfil*.

tblPessoa-Perfil		Para conter as atribuições de perfis às pessoas.	
Campos	RefPessoa	Referencia à tabela *Pessoas*.	Chave primária
	RefPerfil	Referencia à tabela *Perfil*.	
	DataInicio	Data de início da atribuição do perfil.	
	DataFim	Data validade da atribuição do perfil.	
	Ativo	Marcador para indicar se este perfil está a ser usado. Por cada Pessoa, apenas um perfil está ativo em cada momento.	
	Criação	Data de criação do registo	
	Notas	Campo de texto livre para informações complementares.	

Resumindo, as tabelas a serem implementadas e apresentadas no Anexo I podem ser classificadas relativamente à sua finalidade, da seguinte forma:

- Tabelas que contem informação relativa a entidades apresentadas nos modelos de classes, como por exemplo *tblPessoa*, *tblCartao*, *tblPortaria*, que são relativas respetivamente às entidades *Pessoa*, *Cartão* e *Portaria*.

- Tabelas de relação usadas para suportar a implementação do modelo relacional. São exemplo destas tabelas as que fazem a associação muitos-para-muitos das entidades, por exemplo a *tblValidadePerfil*, que liga as entidades do tipo *Pessoa* aos perfis que lhe estão atribuídos.

- Tabelas dicionário, estas tabelas apresentam listas de informação que é pouco alterável ao longo do tempo, por exemplo a tabela *tblDicDocumentoID*, contem listas de tipos de documentos de identificação como passaporte, cartão do cidadão e bilhete de identidade.

- Tabelas de registo, são as tabelas usadas para guarda informação sobre o eventos do sistema, por exemplo a tabela *tblLogPassagemPortaria*, guarda informação da apresentação de cartões de acesso nas portarias, consultado esta tabela pode-se analisar quem passou, onde, quando e qual o vigilante que estava de serviço

3.4.3.1. BASE DE DADOS PORTARIAS

Fisicamente, as portarias são espaços em zonas mais ou menos remotas que estão equipadas com um computador de assistência às atividades que lá decorrem. A aplicação *Port* da plataforma de credenciação vai ser instalada nesses computadores. Por questões de segurança e numa tentativa de impermeabilizar o acesso à plataforma de credenciação através dos pontos de rede de comunicação que estão instalados nas portarias, vai-se criar uma base de dados tampão.

A base de dados tampão contém uma cópia da informação exitente na base de dados "principal". Sendo essa informação apenas a estritamente necessária para o funcionamento da aplicação *Port*.

Desta forma, o ponto de rede de comunicações que liga ao computador da portaria, apenas dá acesso à base de dados tampão e não à base de dados onde está toda a informação da plataforma, Figura 131.

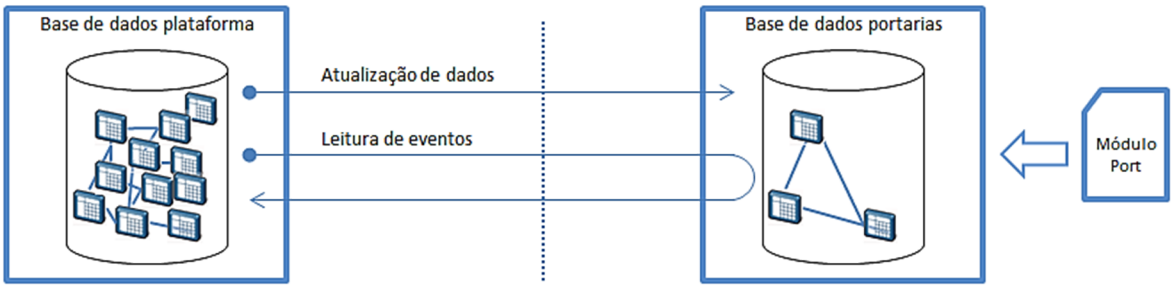


Figura 131 – Base de dados das portarias

A base de dados para uso das portarias vai conter três grupos de informações:

- Informação sobre as portarias e as respetivas permissões de acessos.
- Informação sobre as pessoas e as respetivas permissões de acessos.
- Informação relativa ao registo dos eventos associados às portarias.

As tabelas com informação sobre as pessoas e sobre as portarias são atualizadas pela base de dados da plataforma sempre que a informação é alterada por uso da aplicação *Cred*. O

sincronismo entre as bases de dados é efetuado por funções despoletadas pelos eventos de alteração de dados da base de dados *Credenciação*.

Neste cenário, é sempre a base de dados da plataforma que toma a iniciativa da troca de informação entre as duas bases de dados. Na base de dados *Portaria*, não existe qualquer informação sobre o servidor onde está a ser executada a base de dados da plataforma. Na Figura 132 é mostrado o modelo relacional da base de dados Portarias e no Anexo I k), são apresentados os detalhes de implementação das tabelas.

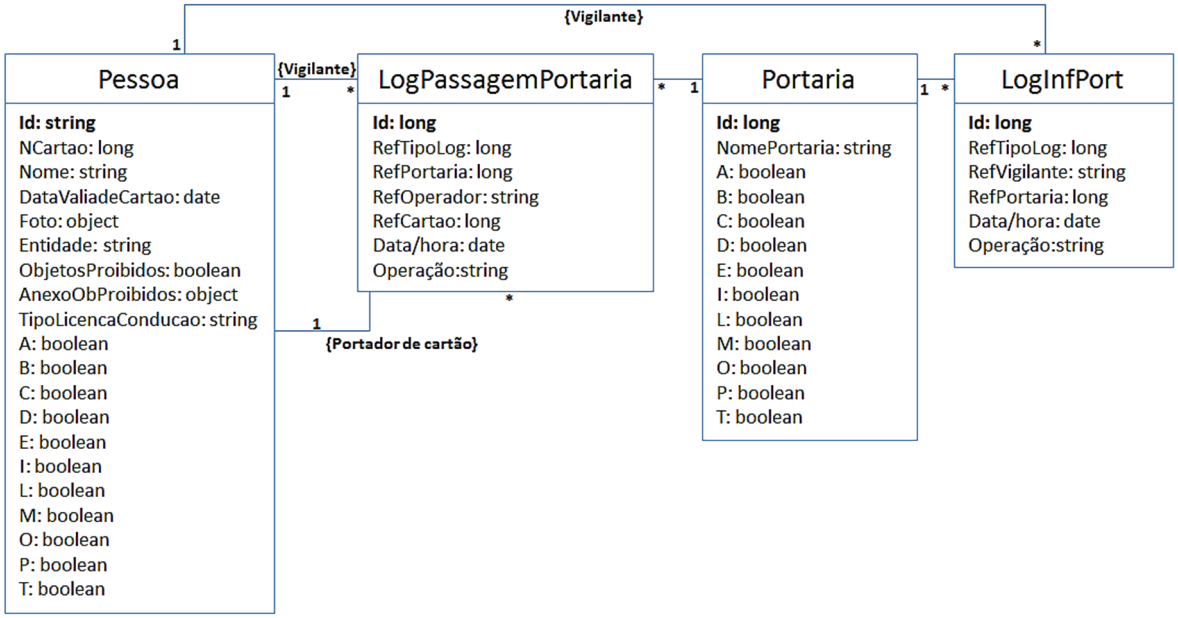


Figura 132 – Modelo relacional da base de dados das portarias

3.4.4. INTERFACE COM A BASE DE DADOS

A implementação do modelo relacional das bases de dados tem componentes que não estão diretamente associados como a definição de entidades, mas que são intrínsecos à própria implementação do modelo e que incorrem num acréscimo significativo de complexidade como se pode verificar nos digramas apresentados nas Figura 121 à Figura 130.

Do ponto de vista conceptual, as aplicações que usam a informação guardada na base de dados, sejam elas as aplicações que constituem a plataforma de credenciação, sejam aplicações externas como a plataforma "Cartão do aeroporto", sejam aplicações a desenvolver no futuro, são blocos completamente separados da base de dados e com fases e ciclos de desenvolvimento completamente independentes. Por outro lado, a informação guardada na base de dados que se está a desenvolver, tem valor em si mesmo, independentemente das aplicações que a usam.

Assim, considerando a arquitetura de implementação da base de dados *versus* aplicações externas, não é uma boa solução "obrigar" que as aplicações, para usarem a informação guardada, tenham de conhecer a complexidade de relacional da base de dados, Figura 133.

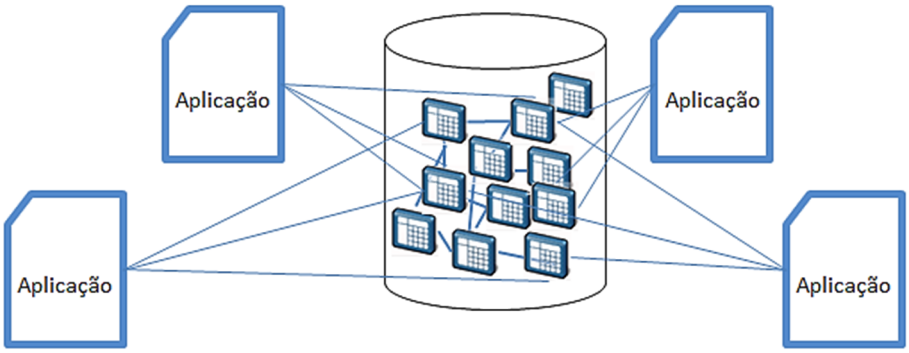


Figura 133 – Informação da base de dados acedida diretamente pelas aplicações.

Para resolver esta questão, vão ser criadas em cada uma das bases de dados, camadas de encapsulamento da complexidade da implementação, permitindo que as aplicações de *software* não tenham de conhecer os detalhes da base de dados e possam aceder e

manipular a informação através de interfaces associadas às entidades definidas nos diversos modelos, Figura 134.

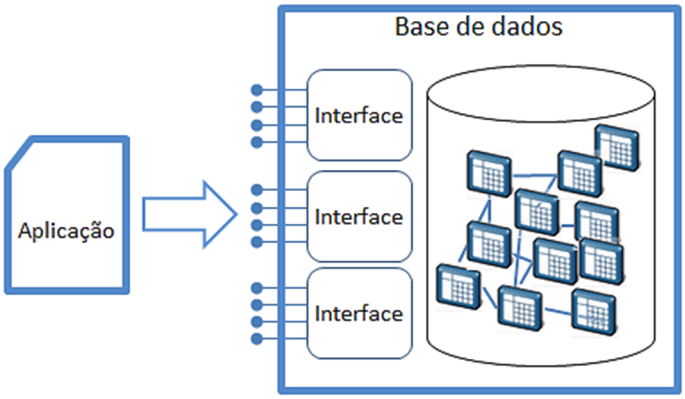


Figura 134 – Informação da base de dados acessível através de interface.

Do ponto de vista das aplicações, quando pretenderem aceder ou efetuarem operações sobre a informação apenas tem de conhecer a interface que executa a funcionalidade pretendida e conhecer os respetivos parâmetros de entrada e saída.

Apesar do custo de desenvolvimento da camada de encapsulamento, esta forma de implementação, além de ocultar os detalhes da estrutura de dados apresenta também as seguintes vantagens:

- A gestão da informação é efetuada pela base de dados, garantindo uma maior robustez e tornando-a mais impermeável as consequências das ações das aplicações externas.

- Permite criar mecanismos de validação dos parâmetros de entrada da interface, para proteção da coerência da informação. Por exemplo não permitir a introdução de datas anteriores a outras que se consideram limites.

- Permite introduzir mecanismos de registo de operações de dados de forma transparente para as aplicações, por exemplo quando uma aplicação executa uma interface que cria ou altera dados, a própria interface faz o registo adequado nas tabelas de histórico a relatar o acontecimento.

- Em caso de alterações de funcionalidades de aplicações ou no caso de necessidade de uso da informação por outras aplicações o acesso aos dados é feito de forma mais transparente e mais fácil.

- A necessidade de novas funcionalidades apenas envolve o desenvolvimento de uma nova interface, mantendo as implementações já efetuadas, inalteradas.

A implementação das interfaces vai ser efetuada usando dois recursos do *SQL Server*: *views* e *stored procedures*. Dependendo do objetivo final da interface ou se usa um ou se usa o outro:

- Quando pretende apresentar informação agregada, apenas para consulta, sobe determinado aspeto das entidades definidas nos diagramas de classes, mesmo que a informação esteja armazenada em diversas tabelas, implementam-se interfaces baseada em *views*.

- Quando se pretende efetuar consultas que decorrem de perguntas- *queries*, a várias tabelas condicionadas por um ou mais requisitos, implementam-se interfaces baseadas em *stored procedures*.

- Quando se pretende efetuar criação de registos ou alteração de informação armazenada, implementam-se também, interfaces baseadas em *stored procedures*.

As interfaces a desenvolver, baseadas em *views* são relativamente estáticas, não admitem parâmetros de entrada para filtragem de informação e por isso são apenas usadas em casos muitos particulares de informação que varia pouco, como é o caso das listas de dados armazenadas nas tabelas-dicionário.

As interfaces baseadas em *stored procedures* dão suporte à pesquisa e manipulação de informação, estas implementações tem, normalmente, definidos parâmetros de entrada necessários a execução das funcionalidades e podem tem um ou mais parâmetros de saída dependendo da funcionalidade da rotina. Nos *stored procedures* a implementar vão se aplicar as seguintes orientações relativas aos parâmetros de saída:

- Sempre que um *stored procedure* é executado é devolvida informação a caracterizar o sucesso da execução da rotina. Este parâmetro de saída tem sempre o

nome “Execucao” e a tabela *tblDicErros* apresenta uma lista de códigos de caracterização do sucesso da execução ou de erros que possam ter sido detetados. Por exemplo: o “0” representa execução com sucesso e o “-10” representa “Erro – Data de início posterior à data de fim”. O Anexo K apresenta a lista de todos os códigos de erro que podem ser devolvidos por *stored procedures* no parâmetro “Execução”.

- Os *stored procedure* sempre que são executados com sucesso podem também devolver outro tipo de informação relativa à sua funcionalidade. Esta informação devolvida poder ser uma variável simples como um número ou um conjunto de caracteres ou pode ser uma quantidade de informação numa estrutura mais complexa como por exemplo o resultado da filtragem da informação de uma tabela ou um objeto a conter um ficheiro.

Os *stored procedure* vão ser sempre implementados em três blocos de código de programação:

- O bloco inicial de código, faz a validação dos parâmetros de entrada. Qualquer valor fora da gama esperada implica que a funcionalidade da rotina não seja executada.

- O segundo bloco de código que constitui o *stored procedure*, executa a funcionalidade da rotina, quando executado sem erros é devolvido no parâmetro “Execucao” o valor “0” e nos restantes parâmetros de saídas é devolvida a respetiva informação.

- No último bloco de código faz-se o registo nas tabelas de histórico, da execução do evento no sistema.

A Tabela 12 apresenta um exemplo de aplicação de uma interface usando *stored procedure*. Neste caso, para criação de um registo de informação sobre de lotes de cartões, as aplicações externas usam a interface *CriaLoteCartões* indicando as informações que caracterizam o lote e recebem de volta no parâmetro “Execução” um código indicando se a operação foi executada com sucesso ou uma combinação com os erros detetados: -10, -11, -20.

Tabela 12 – Interface para criação de lotes de cartões.

CriaLoteCartões – Para criar um registo de lote de cartões de acesso. *Stored procedure*			
in	PrimeiroNumero		Primeiro número do lote
in	UltimoNumero		Ultimo número do lote
in	DataInicio		Data de início de uso do lote
in	DataFim		Data de fim de uso do lote
in	Ativo		Marcador que informa se o lote está em uso
out	Execucao	-10	Erro – Data de início posterior à data de fim
		-11	Erro – Data de fim anterior à data atual
		-20	Erro – Número de início maior que o número de fim
		0	Registo efetuado com sucesso

No Anexo L apresenta-se a lista de todas as interfaces a implementar na base de dados para criação da camada de encapsulamento das bases de dados no âmbito deste projeto.

3.4.4.1. INTERFACE COM O PWNT

O resultado de conhecimento adquirido sobre a base de dados PWNT vai ser vertido na criação de *stored procedures* na base de dados do PWNT, que no conjunto vão constituir a interface para ser usada pela plataforma a desenvolver e que disponibiliza no PWNT a execução das necessidades da plataforma.

A interface a implementar na PWNT tem o mesmo princípio de funcionamento e o mesmo princípio de implementação da interface a ser criada na base de dados da plataforma, Figura 135.

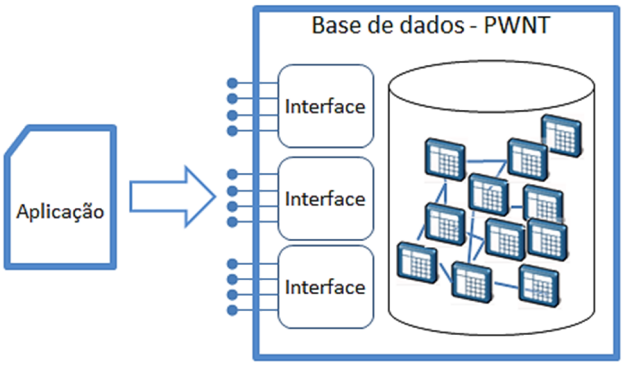


Figura 135 – Informação da base de dados acessível através de interface.

Na Tabela 13 estão listadas as interfaces que constituem as necessidades de interação da plataforma com a PWNT, no Anexo L k) são apresentados os detalhes dos parâmetros de entrada e de saída dos *stored procedures* que constituem a interface.

Tabela 13 – Interfaces a criar no *Pro-Watch*.

CriaCartão	Cria um cartão no PWTN.
PessoaCompanhia	Associa uma pessoa a uma companhia.
ListaCartaoPessoa	Lista os cartões de uma pessoa e a sua informação.
ListaCC-Cp	Lista os *Clearance code* que estão atribuídos a uma *Company*.
ListaLD-CC	Lista os *Logical devices* que estão atribuídos a um *Clearance code*.
ListaPassagensPessoa	Lista as passagens de uma pessoa pelas portas.
ListaPassagensPorta	Lista as passagens por uma porta.

A implementação e teste dos *stored procedures* a efetuar na base de dados PWNT, pode necessitar de operações de voltar-a-trás para que se possam repetir as análises, refazer os testes ou repor o último estado estável da base de dados, assim, antes de executar qualquer alteração efetuam-se cópias de segurança que permitem repor a situação anterior. No limite, pode-se repor a cópia de segurança do primeiro estado da base de dados após instalação do *Pro-Watch*.

A fase final da implementação da interface de encapsulamento da base de dados do *Pro-Watch*, compreende a execução de testes das funções desenvolvidas e da aferição do impacto das alterações introduzidas. Os testes são primeiramente executados em servidores de laboratório com apenas uma unidade de controlo instalada, quando se considera que os resultados são estáveis, os testes são repetidos no sistema produtivo usando inicialmente o

cartão e os dados relativos ao autor deste trabalho e seguidamente usando os dados de um grupo restrito de pessoas, atentas a qualquer comportamento diferente do esperado.

3.4.4.2. INTERFACE COM O PORTAL CARTÃO DO AEROPORTO

O fluxo de informação entre a plataforma de credenciação e a plataforma do Cartão do Aeroporto vai ser efetuado, via base de dados da plataforma, usando a metodologia de interfaces de encapsulamento de funcionalidades apresentadas na secção anterior. Para se implementar essa troca de informação vão-se implementar as interfaces mostradas na Tabela 14 e detalhadas no Anexo L l).

Tabela 14 – Interfaces com a plataforma Cartão do Aeroporto.

Interface	Descrição
NovaPessoa	Envia informação para a base de dados da plataforma de credenciação para a criação de uma nova pessoa.
NovaEntidade	Envia informação para a base de dados da plataforma de credenciação para a criação de uma nova entidade.
AtribuicaoCartao	Envia informação para a base de dados da plataforma de credenciação para a criação ou renovação de um cartão de uma pessoa
ListaCartoesPessoa	Devolve a informação sobre os cartões de uma pessoa
ListaPessoasEntidade	Devolve informação sobre todas as pessoas associadas a uma entidade
ListaErros	Devolve informação sobre os erros despoletados pela interface da plataforma de credenciação

3.5. INTERFACE HOMEM-MÁQUINA DAS APLICAÇÕES

Conforme definido na secção 3.4, a plataforma de credenciação é composta por três módulos aplicacionais: *Cred*, *Pontu* e *Port*, Figura 102, cada uma destas aplicações têm funcionalidades diferentes, vão operar em locais diferentes e ser usadas por utilizadores diferentes. Esta secção apresenta a linhas de guia com os requisitos a considerar na implementação da interface homem-máquina das aplicações.

As aplicações vão ser desenvolvidas usando a linguagem de programação *Visual Basic*, a interface das aplicações com os utilizadores vai usar o ambiente gráfico normalmente presente nos programas executados no sistema operativo *Windows*. As interfaces a desenvolver vão conter quatro grupos de espaços, Figura 136:

1. Zona de menus: nesta área serão implementados os menus que dão acesso às funcionalidades relativas às entidades definidas nos diagramas de classes ou a temas específicos como por exemplo ecrãs de configuração. Estes menus são apresentados ou escondidos, dependendo do perfil do utilizador que estiver autenticado na entrada da aplicação: operador, gestor ou administrador.

2. Corpo da central: nesta área são apresentados os objetos que agrupam as funcionalidades do tema que foi selecionado nos menus.

3. Zona de separadores: é uma área do interior do corpo central usada quando a entidade ou tema que se está a trabalhar está associada a várias funcionalidades ou casos de usos, usam-se objetos com separadores para agrupar e organizar essas funcionalidades.

4. Zona de estado: nesta área apresentam-se informações relativas ao funcionamento da aplicação.

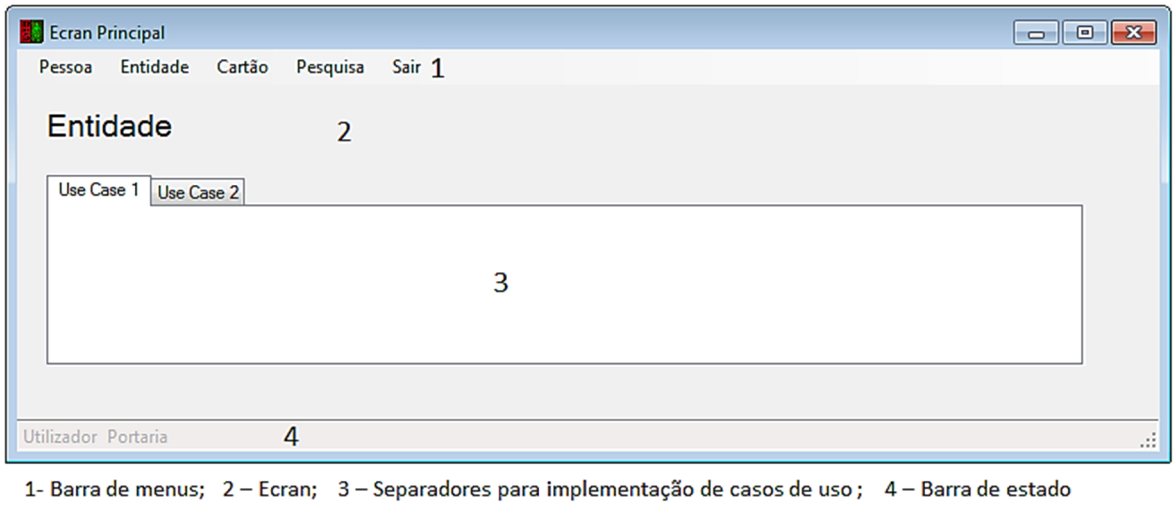


Figura 136 – Apresentação geral da interface gráfica das aplicações.

3.5.1. MÓDULO CRED

O módulo *Cred*, é a aplicação que vai ser instalada no gabinete de credenciação e onde se definem as configurações do sistema, onde se efetuam as credenciações de pessoas e viaturas, onde se emitem os cartões de acesso perante, os cartões de acesso temporário e as licenças de condução. A Tabela 15, apresenta a lista dos menus a criar nesta aplicação e os casos de uso que são tratados em cada janela disponibilizada pelos menus.

Tabela 15 – Interface homem-máquina: módulo Cred

Menu	Casos de uso
Pessoa	Gestão de pessoa Gestão de acessos de pessoas Gestão de cartões associados a pessoas Gestão de licenças de condução Gestão de infrações e penalidades Gestão de cartões
Entidade	Gestão de entidades
Viaturas	Gestão de viaturas
Portaria	Gestão de portarias Gestão de acessos de portarias
Acessos	Informação importada do *Pro-Watch*
Pesquisa	Ferramentas de seleção e filtragem de informação
Dicionários	Gestão de listas de itens

Na área de apresentação de estado da aplicação *Cred*, é mostrado o nome do utilizador que está autenticado na aplicação e um indicador do estado da ligação de comunicação ao servidor de base de dados e ao servidor *Pro-Watch*.

3.5.2. MÓDULO PONTU

O módulo *Pontu* é a aplicação que vai ser instalada no balção de credenciação pontual da PSP para emissão de cartões pontuais de visita e de passageiros. A Tabela 16 apresenta os menus a incluir na aplicação e os casos de uso tratados.

Tabela 16 – Interface homem-máquina: módulo Pontu

Menu	Casos de uso
Credenciação pontual	Gestão de informação de visitantes Gestão de cartões pontuais
Credenciação de passageiros	Gestão de informação de passageiros Gestão de cartões de passageiros
Dicionários	Gestão de listas de itens

Na área de apresentação de estado da aplicação *Pontu*, é mostrado o nome do agente que está autenticado na aplicação e um indicador do estado da ligação de comunicação ao servidor de base de dados.

3.5.3. MÓDULO PORT

O módulo *Port* é a aplicação que dá assistência ao controlo de acessos nas portarias. Em funcionamento normal – vigilante credenciado, a aplicação não apresenta menus e encontra-se em estado de espera de leitura de cartão, Tabela 17. Na área apresentação de estado, a aplicação *Port* mostrada o nome do vigilante que está autenticado na aplicação, o nome da portaria, um indicador que sinaliza o estado da ligação de comunicação ao servidor da base de dados e um indicador que sinaliza o estado de ligação ao leitor de cartões. Quando é um utilizador com perfil de Administrador que está registado, a aplicação mostra um menu de configuração que permite definir os recursos do sistema.

Tabela 17 – Interface homem-máquina: módulo Port

Menu	Casos de uso
Leitura e validação de cartões	Leitura e validação de cartões
Configuração	Configuração da portaria Configuração dos recursos para acesso ao leitor de cartões

3.6. MÓDULO DE *HARDWARE* PARA LEITURA DE CARTÕES

Na secção 3.4.1.7 definiu-se o caso de uso da aplicação controlo de acessos que vai ser instalada nas portarias, esse caso de uso estabelece a necessidade de desenvolvimento de um módulo de *hardware* que tem como função de fazer a leitura do número do cartão de identificação da pessoa e envia-lo para o computador onde está a ser executada a aplicação de *software* que avalia se a pessoa tem permissões de passagem. Este módulo vai ser composto por um leitor de cartões e por um conversor de formato de dados *Wiegand*/RS232, Figura 137.

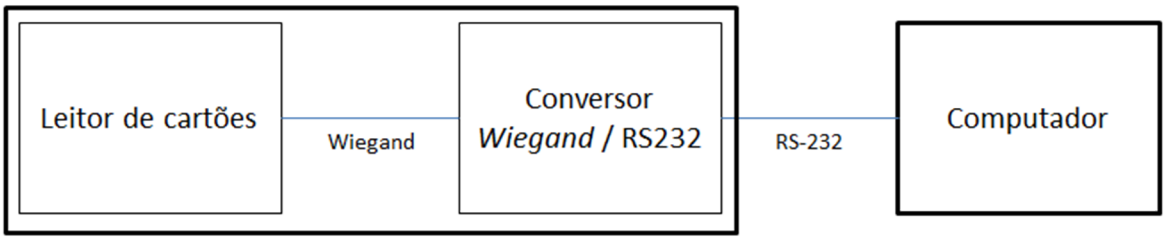


Figura 137 – Módulo de *hardware* a instalar na portaria.

3.6.1. SELEÇÃO DOS COMPONENTES DO MÓDULO DE *HARDWARE*

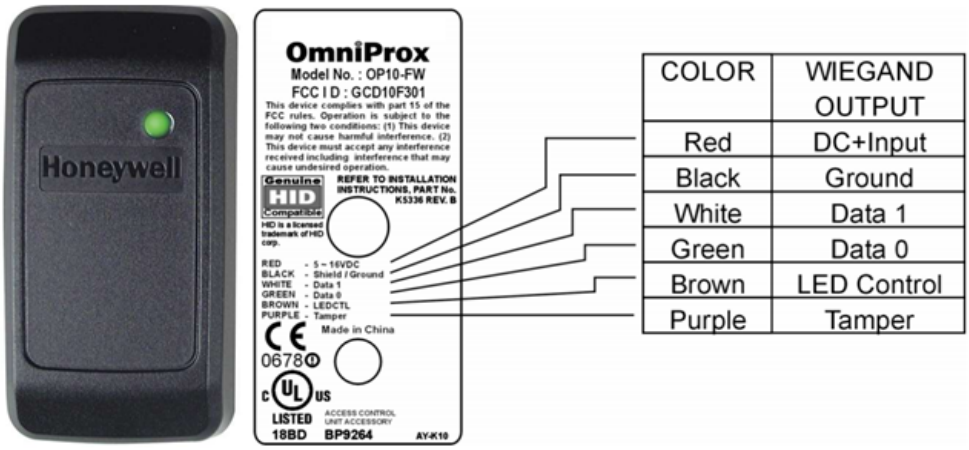
Os utilizadores do sistema de credenciação no ASC usam cartões RFID do fabricante HID com resposta de dados na frequência de 125KHz onde está codificado um número identificador do cartão. Este número do cartão vai ser usado para identificar a pessoa que se apresenta na portaria e para verificar se a pessoa tem acesso de passagem.

3.6.1.1. LEITOR DE CARTÕES

Os edifícios do ASC estão equipados com um sistema de controlo de acessos que usa o leitor de cartões *HONEYWELL* OP-10. Por conveniência de gestão de peças de armazém

vai-se usar o mesmo leitor no módulo de *hardware* da aplicação *Port*. As principais características do OP-10 são as apresentadas na Tabela 18.

Tabela 18 – Características do leitor de cartões OP10, [103] [104].



Tensão de alimentação:	5.0 – 16.0 V_{DC}
Consumo quando alimentado a 5V:	35mA
Tipo de comunicação:	Wiegand
Comprimento máximo do cabo:	150m
Sinalizadores:	Led tricolor e *Buzzer*
Distância de leitura do cartão:	7,6cm

O OP10 funciona da seguinte forma [104]:

- Quando o leitor é alimentado, o *led* e o *buzzer* são ativados três vezes sinalizando o início de operação.
- Em estado de repouso o *led* está ligado e apresenta a cor vermelha.
- Quando é apresentado um cartão que responde na frequência dos 125KHz, o *led* pisca na cor verde e o *buzzer* emite um som breve. Após a leitura do cartão o número identificador é transmitido para o exterior pela interface *Wiegand* e o leitor volta ao estado de repouso.

Na Figura 138 são apresentadas as formas de onda dos sinais de dados *Wiegand* após a leitura do número de um cartão pelo leitor OP-10, nesta figura verificamos a existência dos

37 *bits* do número do cartão lido em que os "0" são sinalizados pela linha Data0 no nível baixo e os "1" são sinalizados pela linha Data1 no nível baixo, como descrito na secção 2.3.1.1.

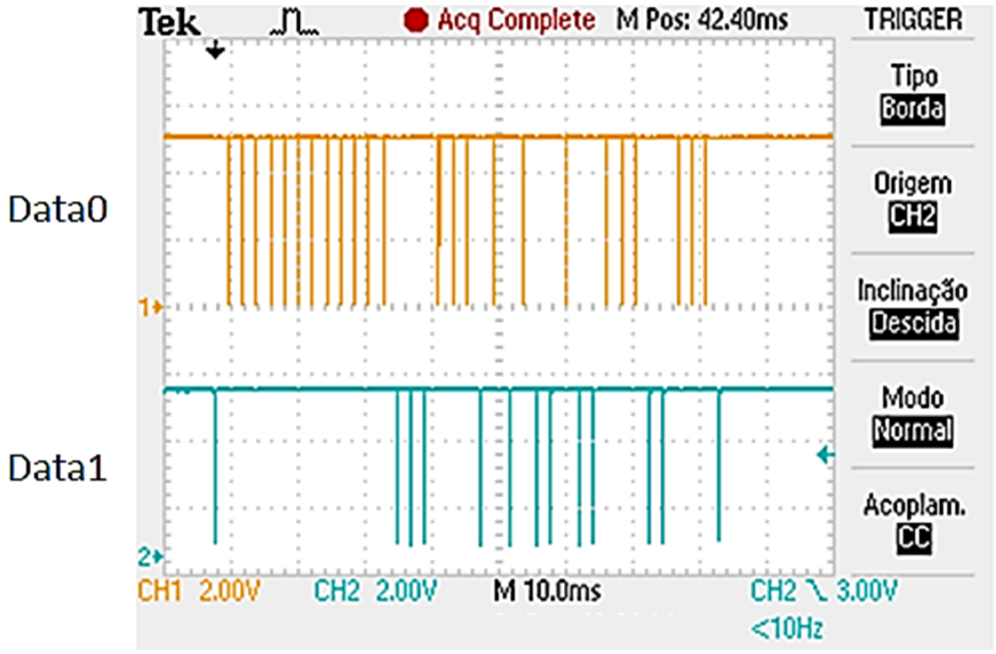


Figura 138 – Forma de onda *Wiegand* da leitura de um cartão de acesso.

O leitor OP10 vem equipado com uma saída denominada *tamper* que é controlada por um sensor de luminosidade que coloca o sinal *tamper* no estado lógico baixo ou alto dependendo da medida efetuada. Esta funcionalidade é usada para monitorização do estado da instalação do leitor, por exemplo se o leitor estiver instalado numa superfície opaca, o sinal *tamper* está no nível lógico baixo, baixa luminosidade, se o leitor for deslocado da superfície a luminosidade aumenta e do sinal muda de estado, indicando essa alteração da instalação.

3.6.1.2. CONVERSOR *WIEGAND*/RS-232

Os computadores não têm portas *Wiegand* que possam receber diretamente a informação enviada pelo leitor de cartões, por isso é necessário intercalar entre o leitor e o computador

onde está a ser executada a aplicação *Port*, um conversor do formato *Wiegand* para um formato reconhecido pelo computador, por exemplo RS-232.

O conversor que está disponível para o projeto é o modelo W2RS232 fabricado pela ETConcept [105]. O W2RS232 é um conversor bidirecional cujas principais características são apresentadas na Tabela 19.

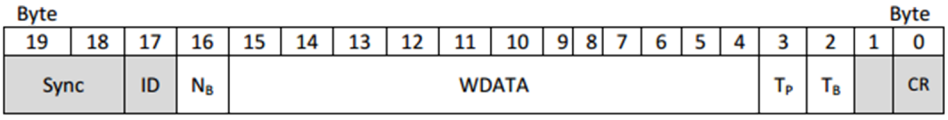
Tabela 19 – Características do conversor Wiegand-RS232, [105]

Tensão de alimentação:	7.0 – 16.0 V_{DC}
Consumo:	30mA
Ficha de alimentação:	CTF fémea de dois contactos
Interface RS232	
Ficha:	DB9 macho
Distancia máxima:	50m
Modo de operação:	Bidirecional sem controlo de fluxo
Débito de caracteres:	9600 bps
Interface *Wiegand*	
Ficha:	CTF fémea de sete contactos
Formato *Wiegand* suportado:	De 6 a 96 bits
Período de espera entre tramas:	Mínimo 30mS
Largura do pulso *Wiegand*:	Mínimo 50µS, máximo 200 µS
Período de bit *Wiegand*:	1mS, 2mS
Portos genéricos de entrada/saída:	2

O conversor quando ligado a um equipamento de saída *Wiegand* como é o caso do leitor de cartões, autoconfigura-se para o modo de funcionamento "Input". Neste modo de funcionamento, o conversor recebe tramas *Wiegand*, converte-as para tramas de formato RS-232 e transmite-as pela porta serie.

O modo de funcionamento "Input" é sinalizado pelo *led* IN ativo. Quando o conversor está a aguardar a receção de tramas o *led* RDY – *Ready* está aceso. Quando o conversor recebe uma trama *Wiegand* e processa a trama, a ação é sinalizada pelo led ACK

A Figura 139 mostra o formato da trama de dados que é enviada pela porta série do conversor.



Byte 19 - 18	**Sync** : Padrão de sincronismo. Valor: 55_h 55_h
Byte 17	**ID** : Identificador do comando. Valor: 01_h
Byte 16	**N_B:** Dimensão da trama Wiegand (incluindo os bits de paridade). Valor: (6 - 96)
Byte 15-4	**WDATA :** Dados Wiegand (Incluindo os bits de paridade).
Byte 3	**TP:** Largura dos pulsos Wiegand **(Não implementado nesta versão)**
Byte 2	**TB :** Duração dos bits Wiegand Bit **(Não implementado nesta versão)**
Byte 1	**Reservado para utilização futura**
Byte 0	**CR:** Caracter terminador (*Carriage Return*) Valor = $0D_h$

Figura 139 – Formato da trama série do conversor W2RS232, [105].

A cada leitura de cartão, o conversor irá gerar uma trama RS-232 com os valores mostrados na Tabela 20. A trama é iniciada por dois *bytes* de sincronismo de valor 55_H, seguidos do valor 01_H que indica que a informação contida na trama é correspondente à informação recebida pela interface *Wiegand* no modo de funcionamento "Input" do conversor. O *byte* seguinte da trama, indica a quantidade de *bytes* de dados recebidos na interface *Wiegand* e que são enviados na trama RS-232, na leitura dos cartões RFID este

byte apresenta sempre o valor fixo igual $25_H = 37_D$, porque o número de serie do cartão é representado em trinta e sete *bits*. De seguida, a trama contem os treze *bytes* do bloco de dados e é finalizada por três *bytes* de valor fixo $FF000D_H$.

Tabela 20 – Formato da trama série RS-232 na leitura de cartões RFID.

Bytes	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Dados	55_H	55_H	01_H	25_H	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	FF_H	00_H	$0D_H$

Tamper

O sinal de *tamper* do leitor de cartões é ligado eletricamente, à entrada TMPR do conversor *Wiegand*-RS-232. Quando o sinal de *tamper* do leitor de cartões muda de estado, o conversor deteta essa alteração de estado na sua entrada *tamper* e envia para a porta série uma trama identificada com o valor 02_H no byte "Id", Figura 140, seguido da representação do estado da linha: 00_H quando a linha está no estado lógico baixo e FF_H quando a linha está no estado lógico alto. Desta forma o módulo *Port* consegue detetar alterações na instalação do leitor e despoletar alertas.

Byte					Byte
5	4	3	2	1	0
Sync		ID	B_T	-	CR

Byte 5 - 4	**Sync** : Padrão de sincronismo. Valor = $55_h 55_h$
Byte 3	**ID** : Identificador do comando. Valor: 02_h
Byte 2	**B_T:** Valor do sinal de TAMPER Valor= 00_h Nível do TMPR é 0 FF_h Nível do TMPR é 1
Byte 1	**Reservado para utilização futura**
Byte 0	**CR:** Caracter terminador(*Carriage Return*) Valor = $0D_h$

Figura 140 – Trama série sinal de *tamper*, [105].

3.6.2. PROJETO DO MÓDULO DE *HARDWARE*

O esquema elétrico do módulo de *hardware* usado para a leitura do cartão RFID está apresentado na Figura 141 e é composto por:

- Leitor OP10,
- Conversor W2RS232,
- Transformador de energia elétrica,
- Cabo série RS-232 cruzado.

Do ponto de vista de alimentações elétricas o leitor usa tensões de 5.0 a 16.0 V_{DC} consumindo 35mA e o conversor pode ser alimentado com tensões de 7.0 a 16.0 V_{DC} e apresenta consumos de 30mA. Por questões de disposição da instalação física o leitor e o conversor vão ser instalados nas proximidades do computador e por isso vão ficar perto um do outro. Assim, a alimentação elétrica do leitor e do conversor vai ser efetuada pelo mesmo transformador $230V_{AC}$-$12V_{DC}$ de 500mA.

O transformador é ligado à ficha CTF de dois pinos do conversor e o leitor é alimentado pela ligação feita na ficha CTF de sete pinos como mostrado na Figura 141.

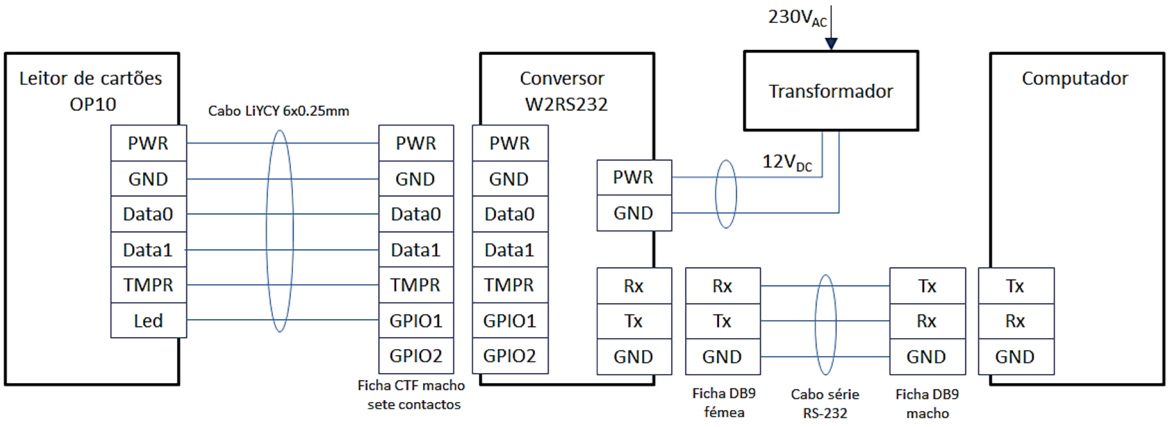


Figura 141 – Esquema de ligações do módulo de *hardware*.

Considerando todos os componentes, estima-se que o valor do módulo de *hardware* definido neste projeto tenha um custo de materiais da ordem dos 287€, como determinado na Tabela 21.

Tabela 21 – Módulo de *hardware*: lista de componentes e preço unitário[20]

Componente			**Qt.**	**Preço Un. (€)**	**Total (€)**
Descrição	**Marca**	**Ref.:**			
Leitor de cartões	HONEYWELL	OP-10	1un	170,00	170,00
Conversor Wiegand RS232	ETConcept	W2RS232	1un	100,00	100,00
Transformador $230V_{AC}$-$12V_{DC}$, 500mA	--	--	1un	5,00	5,00
Cabo LiYCY 6x0.25mm	--	--	3m	0,60	1,80
Cabo Null-Modem 3m para comunicação RS-232	--	--	1un	5,00	5,00
Diversos: solda, parafusos, manga termo retrátil, ligadores de gel e todos os consumíveis necessários à instalação.			Vg	5,00	5,00
				Total:	**286,80€**

3.6.2.1. TESTES DE *HARDWARE*

Depois da montagem do *hardware* os testes de verificação de funcionamento consistem em ligar o módulo a um computador, monitorizar a porta série do computador com uma aplicação de escuta, apresentar cartões de séries diferentes no leitor e verificar que o número obtido pela leitura corresponde ao identificador do cartão. O critério de aceitação do teste é a boa execução da leitura do número de identificação de pelo menos cinco cartões. No Anexo J é aprestado um formulário de apoio à realização dos testes de *hardware*.

[20] Preços meramente indicativos tomando como referencia os valores dos artigos dos *sites* dos fabricantes e do *site* RS-Amidata.

4. PROTÓTIPO FUNCIONAL

O objetivo deste trabalho é a implementação de um protótipo funcional que responda às necessidades identificadas. Como foi descrito no capítulo 3 - Projeto, o protótipo, do ponto de vista global tem duas componentes, uma de *hardware*, constituída por um sistema que faz a aquisição do número de um cartão de identificação e outra de *software* constituída por três aplicações e duas bases de dados.

Este capítulo faz uma apresentação da implementação do protótipo e apresenta descrições sobre como as soluções foram construidas e descrições de detalhes técnicos específicos.

4.1. CONSTRUÇÃO DO PROTÓTIPO FUNCIONAL DE *HARDWARE*

No seguimento do preconizado na secção 3.6.2, o módulo de *hardware* é composto por um leitor de cartões RFID e um conversor. O leitor faz aquisição do número de identificação do cartão de acesso e transmite para o exterior, pelas suas linhas de dados a informação lida. O conversor recebe a informação do formato *Wiegand* e converte-a numa trama de *bytes* no formato RS-232. E esta trama é enviada para uma porta série do computador que analisa a informação, Figura 142.

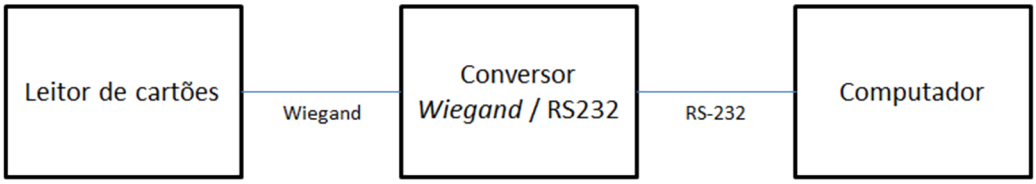


Figura 142 – Diagrama de blocos do protótipo funcional de *hardware*.

Na Tabela 21 apresenta-se a lista de equipamento a usar na construção do protótipo. A construção do protótipo *hardware* consiste na montagem dos componentes selecionados, que resulta num sistema como o mostrado na Figura 143 que é uma implementação direta do diagrama de ligações apresentado na Figura 141.

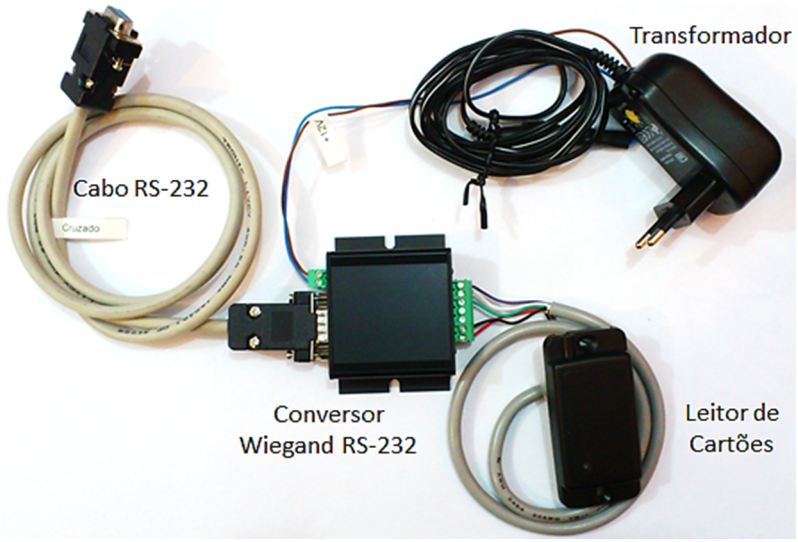


Figura 143 – Fotografia do protótipo funcional de *hardware*.

Numa montagem definitiva, com vista a uma instalação segundo as melhores práticas de execução, deve-se ter em conta os seguintes pontos:

- O leitor OP10 é fornecido com um cabo de 20cm sem terminação. Este cabo entra no leitor através de um furo vulcanizado e sem acesso ao interior. A instalação do leitor deve ser efetuada sobre um caixa de derivação com parafusos inacessíveis pelo exterior quando o leitor está instalado e onde se faz prolongamento do cabo do leitor até à ficha CTF de sete contactos do conversor. Por questões de proteção contra humidades e oxidações, recomenda-se que a ligação entre os cabos seja feita com ligadores banhados a gel.

- Uma escolha adequada de cabo de dados para usar no prolongamento do cabo de leitor é o cabo LiYCY, devido às suas características de proteção eletromagnética e de resistência mecânica [106]. Pode-se usar por exemplo o LiYCY com seis condutores de 0.25mm de secção. Recomenda-se a ligação da malha do cabo ao ponto elétrico de terra fazendo o isolamento elétrico da malha dos outros condutores com manga termo retrátil.

- A conexão do conversor ao computador é realizada com um cabo denominado Null-Modem, terminado em fichas DB9 macho e fémea. Se este cabo for executado manualmente, deve-se considerar que apenas estão ligados os pinos 5 de cada ficha, linha de referência, e os pinos 2 aos pinos 3 da outra ficha que resulta na ligação do pino de sinal de receção- Rx de uma ficha ao pino de sinal de transmissão- Tx da outra ficha – cabo cruzado como apresentado na Figura 141.

4.1.1. EXECUÇÃO DOS TESTES DE *HARDWARE*

Para aferição da funcionalidade, o protótipo foi ligado a um computador e usou-se uma aplicação de monitorização de porta série para inspecionar a informação recebida do leitor de cartões.

A validação do protótipo funcional de *hardware* foi efetuada executando os testes definidos no Anexo J - Bateria de testes de *Hardware*, segundo o procedimento definido no capítulo de projeto.

4.2. DESENVOLVIMENTO DO PROTÓTIPO FUNCIONAL DE *SOFTWARE*

Na implementação do protótipo além da componente de *hardware*, existe também uma componente de *software* composta pelos seguintes módulos distintos:

- Base de dados que implementa os recursos de armazenamento de informação que vai ser denominada *Credenciação.*
- Base de dados tampão para conter a informação necessária ao funcionamento das portarias, denominada *Portarias*.
- Aplicação CRED, para fazer gestão das funcionalidades de credenciação.
- Aplicação PORT, para fazer o controlo de acessos a portarias.
- Aplicação PONTU, para fazer a emissão e gestão de cartões pontuais e de passageiros.

4.2.1. IMPLEMENTAÇÃO DE BASES DE DADOS

No *SQL Server*, ferramenta selecionada para suporte de dados, a implementação das bases de dados começa com a sua criação. Este processo é efetuado usando o menu "*New Database*" como mostrado Figura 144.

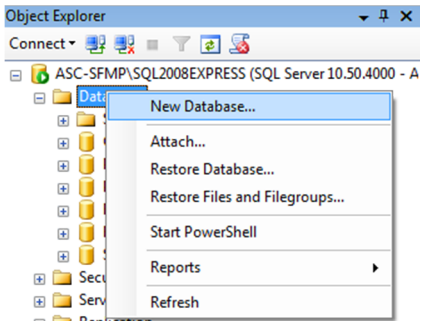


Figura 144 – Criação da base de dados.

O passo seguinte na criação das bases de dados é a respetiva parametrização, Figura 145 e Figura 146. Nas bases de dados *Credenciação* e *Portarias*, optou-se por definir os ficheiros de base de dados e registo com crescimento sem restrições em incrementos de 10% e os outros parâmetros foram deixados com os valores definidos por omissão, em particular os parâmetros descritos na Tabela 22.

Tabela 22 – Principais parâmetros de configuração das bases de dados.

Parâmetro	Função	Valor
Auto Creat Statistics	Armazenamento de dados estatísticos nas tabelas de sistema sobre a operação das bases de dados.	True
Auto Shrink	Indica se o motor SQL pode efetuar automaticamente operações de eliminação de espaço em disco que não seja utilizado.	False
Database Read-Only	Indica o estado de leitura ou leitura e escrita da base de dados	False
Maximun number of concurrent connections	Determina o número máximo de ligações às base de dados o valor "0" indica ligações ilimitadas.	0
Windows Authentication mode	Esta propriedade com o valor True, indica que o processo autenticação de utilizadores para acesso ao SQL *Server* é efetuado pelas contas de utilizador do sistema operativo. Esta opção tem a vantagem de não ser necessário a criação e gestão de contas dentro do servidor SQL.	True
Recursive Triggers Enabled	Indica se *triggers* podem ser despoletados por outros *triggers*. Ver Anexo H d).	False

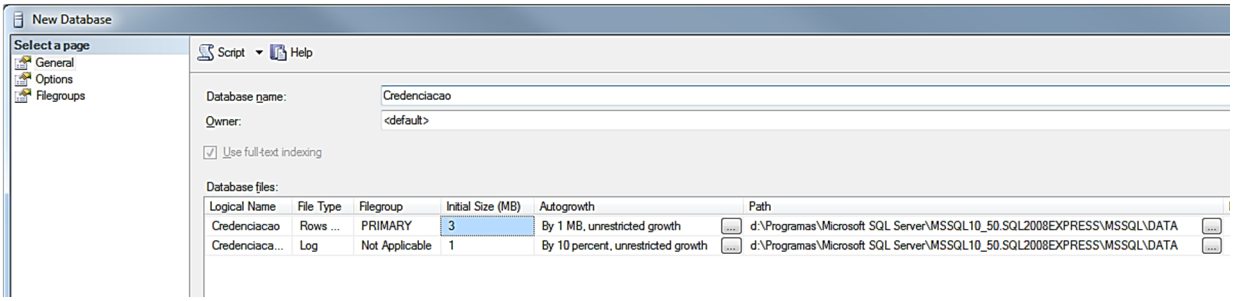


Figura 145 – Ficheiros da criação da base de dados.

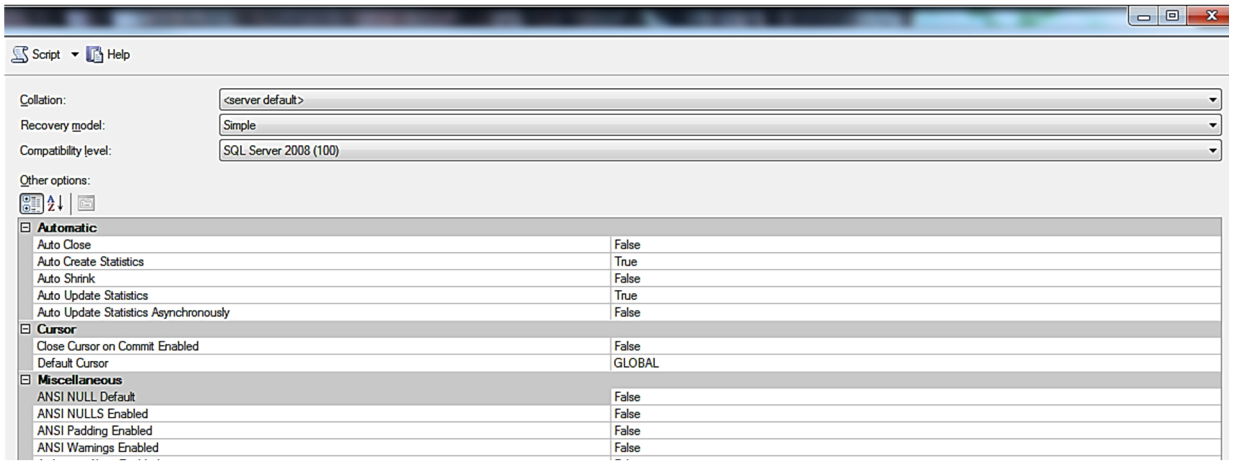


Figura 146 – Parâmetros da criação da base de dados.

4.2.1.1. ESTRUTURA DE TABELAS

Na secção 3.4.3 é apresentado o modelo de dados a ser suportado pelas bases de dados, no Anexo I é apresentada a lista de todas as tabelas que implementam o modelo de dados. A criação das tabelas no servidor, foi efetuada usando os recursos gráficos do *SQL Server*, transpondo as definições apresentadas no projeto para as tabelas na base de dados como mostrado na Figura 147.

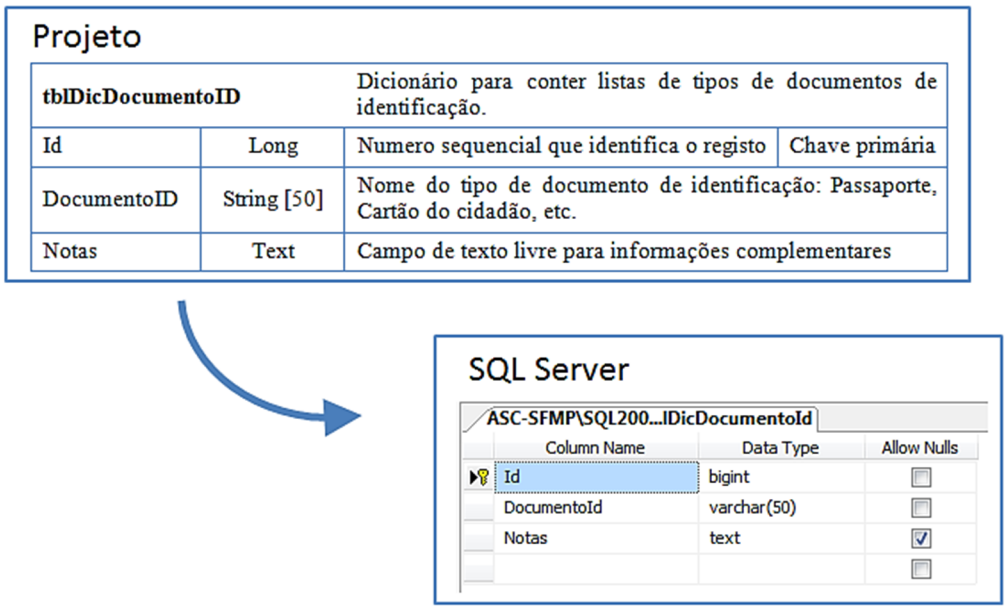


Figura 147 – Implementação de tabelas na base de dados.

Definiu-se, que a denominação das tabelas começa sempre com o prefixo "tbl", seguido do nome da entidade definida no modelo de dados. Esta designação pode ainda estar seguida de uma descrição complementar, resultando na sintaxe: `tbl<Entidade>[Descrição]`.

Por exemplo a `tblPessoa` contém a informação fundamental da entidade pessoa, como Nome, foto, etc., e a `tblPessoaAnexo` contém informação dos ficheiros anexos da entidade pessoa.

4.2.1.2. INTERFACE DA BASE DE DADOS COM APLICAÇÕES EXTERIORES

Como definido no capítulo de projeto, as aplicações externas não vão ter acesso direto à estrutura da base de dados para que não tenham de conhecer a complexidade do modelo relacional, nem se pretende que as aplicações possam manipular informação diretamente sobre as tabelas. Por isso, foi definido criar na base de dados uma camada "por cima" da estrutura de armazenamento, que funciona como interface entre as aplicações e a informação contida na base de dados. Na solução desenvolvida para este projeto, a interface entre as aplicações e a base de dados foi implementada usando os recursos *views* e *stored procedures* do *SQL Server*.

Construção da Interface com *Views*

Os *views* são entidades no formato de tabela, apenas para leitura, que contem informação agregada ou filtrada sobre determinado ponto de vista.

Por exemplo, foi criado o *view* vPortariaAcesso que contêm para cada entidade *Portaria* toda a informação relativa aos respetivos acessos. No modelo relacional esta informação está guardada em três tabelas porque a relação entre a tabela `tblPortaria` e a tabela `tblAcesso` é do tipo muitos-para-muitos o que obriga ao uso de uma terceira tabela para manter a relação entre as duas. Com o uso da *view* `vPortariaAcesso` as aplicações

externas não necessitam de conhecer os detalhes da implementação e obtém a informação das três tabelas diretamente, como mostrado na Figura 148.

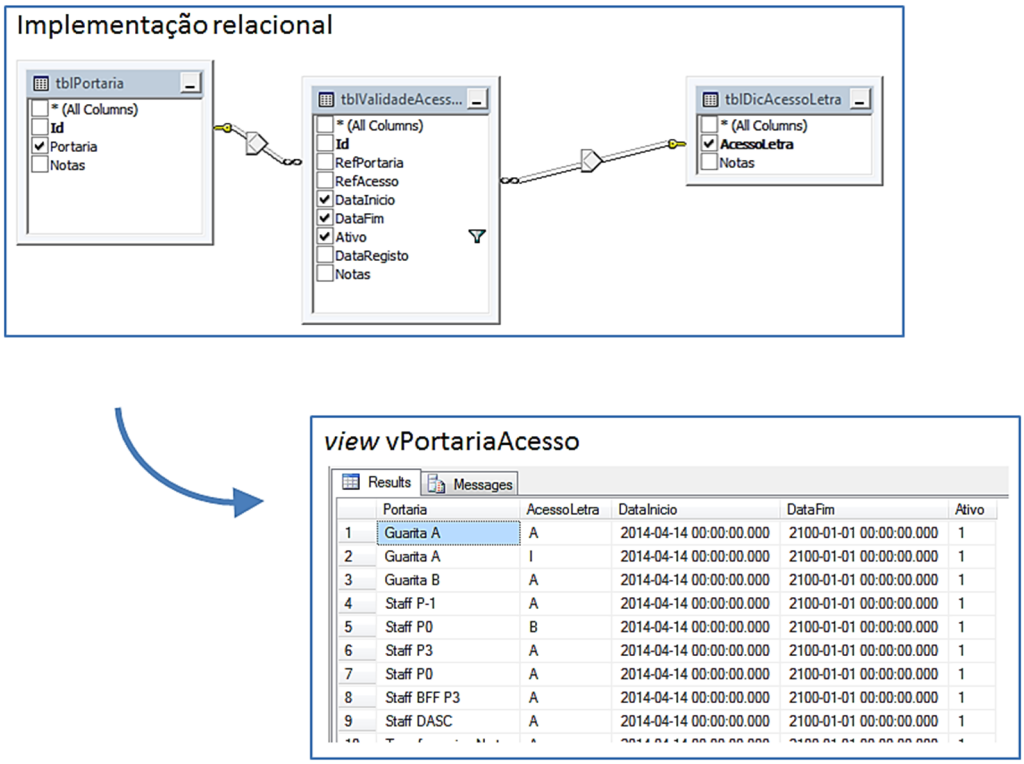
Usando esta lógica foram criadas os *views* necessários para responder às necessidades de implementação das aplicações. Os *views* criados estão listados no Anexo L .

Para denominação dos *views* definiu-se o prefixo "v", seguido de uma discrição da funcionalidade combinada com nome da principal entidade definida no modelo de dados sobre a qual recai a informação do *view*, desta definição resulta a sintaxe: `v<Entidade/Descrição>.`

Exemplos:

- `vPassageiro-IdNome` – *view* que contém a informação do número do documento de identificação de passageiros e o respetivo nome.

- `vPortariaAcesso` – *view* que contém a informação sobre acessos relativos a entidade *Portaria*.



	Portaria	AcessoLetra	DataInicio	DataFim	Ativo
1	Guarita A	A	2014-04-14 00:00:00.000	2100-01-01 00:00:00.000	1
2	Guarita A	I	2014-04-14 00:00:00.000	2100-01-01 00:00:00.000	1
3	Guarita B	A	2014-04-14 00:00:00.000	2100-01-01 00:00:00.000	1
4	Staff P-1	A	2014-04-14 00:00:00.000	2100-01-01 00:00:00.000	1
5	Staff P0	B	2014-04-14 00:00:00.000	2100-01-01 00:00:00.000	1
6	Staff P3	A	2014-04-14 00:00:00.000	2100-01-01 00:00:00.000	1
7	Staff P0	A	2014-04-14 00:00:00.000	2100-01-01 00:00:00.000	1
8	Staff BFF P3	A	2014-04-14 00:00:00.000	2100-01-01 00:00:00.000	1
9	Staff DASC	A	2014-04-14 00:00:00.000	2100-01-01 00:00:00.000	1

Figura 148 – *Views* como interface do modelo relacional.

Construção da Interface com *Stored Procedures*

No *SQL Server* os *stored procedures* são funções desenvolvidas numa linguagem de programação denominada *Transact SQL*, [92], que permitem executar operações sobre as estruturas de dados e sobre a informação armazenada. Estas funções podem receber parâmetros de entrada e devolver resultados em vários formatos, sejam eles um valor elementar ou conjuntos de valores, por exemplo, campos agrupados em linhas.

Na implementação do protótipo foram criados os *stored procedures* necessários para satisfazerem as necessidades das aplicações externas, desde *stored procedures* para fazer perguntas à base de dados até *stored procedures* que efetuam operações alteração e criação de dados. No Anexo L encontram-se as especificações dos *stored procedure* implementados para servir a plataforma.

Por exemplo, foi criado o *stored procedure* `spPessoa-DevolveFoto` implementado na base de dados *Credenciação*, é uma função que aceita como parâmetro de entrada o identificador de uma pessoa e retorna um objeto do tipo imagem que contem a foto da pessoa identificada. O *stored procedure* `spPessoa-GuardaFoto` é uma função que recebe como parâmetro de entrada a identificação de uma pessoa e um objeto do tipo imagem com a foto dessa pessoa e guarda-a na estrutura de dados. O *stored procedure* `spCartaoAcedePortaria`, recebe como parâmetros de entrada o número de identificação de um cartão de acesso, a identificação de uma portaria e a identificação do vigilante de serviço e verifica se a pessoa identificada pelo cartão, tem acesso à portaria, devolvendo um valor booleano que representa a permissão de passagem.

Como definido no capítulo de projeto, ponto de vista de implementação, o código que constitui os *stored procedures* criados, é composto por três blocos de instruções. No primeiro bloco é feita a validação dos parâmetros de entrada, uma tentativa de execução com parâmetros de entrada inválidos, por exemplo data de início posterior a data de fim, faz com que as funcionalidades do *stored procedure* não sejam efetuadas e é devolvido um código de erro. O segundo bloco é constituído pelas instruções que executam a funcionalidade do *stored procedure*. No último bloco da implementação estão as instruções

que efetuam as operações de registo de execução do *stored procedure* nas tabelas usadas para efeitos de registo histórico.

O trecho Código 1 mostra um exemplo de implementação de um *stored procedure*. O *stored procedure* em causa, denominado `spPortaria-Cria` tem como objetivo criar um novo registo de Portarias. Neste exemplo, o código das linhas 29 a 39, é relativo à existência do próprio *stored procedure*, inicialmente indicando que se pretende operar sobre a base de dados *Credenciação* e de seguida instruído que se essa base de dados já tiver um *stored procedure* com este nome, apaga-o para se poder criar o procedimento atual.

Código 1 – Exemplo Código *Transct SQL* para implementar *Stored Procedures*.

```
1  -- ================================================================================================
2  --
3  --  _____ _                          _  _____                          _
4  -- / ____| |                        | | |  __ \                        | |
5  --| (___ | |_ ___  _ __ ___  __| | | |__) | __ ___   ___ ___  __| |_   _ _ __ ___  ___
6  -- \___ \| __/ _ \| '__/ _ \/ _` | |  ___/ '__/ _ \ / __/ _ \/ _` | | | | '__/ _ \/ __|
7  -- ____) | || (_) | | |  __/ (_| | | |   | | | (_) | (_|  __/ (_| | |_| | | |  __/\__ \
8  --|_____/ \__\___/|_|  \___|\__,_| |_|   |_|  \___/ \___\___|\__,_|\__,_|_|  \___||___/
9  --
10 --
11 -- ================================================================================================
12 -- spPortaria-Cria
13 -- ================================================================================================
14 -- Author:  Sérgio Martins
15 -- Create date: 2014-03-27
16 -- Description: Cria um novo registo de portaria
17 --
18 -- Parametros de entrada
19 --  NomePortaria - Nome da portaria que se pretende criar
20 --  Notas        - Informações complementares sobre a portaria a criar
21 --  RefOperador  - Identificador do operador que está a criar a portaria
22 --
23 -- Parametros de saida
24 --    0 - Sucesso
25 --   -2 - operador inválido
26 -- -500 - Erro de servidor de dados
27 --
28 -- ================================================================================================
29 SET ANSI_NULLS ON
30 GO
31 SET QUOTED_IDENTIFIER ON
32 GO
33 USE [Credenciacao]
34 GO
35 -- ================================================================================
36 IF  EXISTS (SELECT * FROM sys.objects WHERE object_id = OBJECT_ID(N'[dbo].[spPortaria-Cria]')
               AND type in (N'P', N'PC'))
37 DROP PROCEDURE [dbo].[spPortaria-Cria]
38 GO
39 -- ================================================================================
40 CREATE PROCEDURE dbo.[spPortaria-Cria]
41  -- parametros de entrada
42  @NomePortaria varchar(50),
43  @Notas text= NULL,
44  @RefOperador varchar(15),
```

```
45
46    -- parametros de saída
47    @Resultado bigint OUTPUT
48  AS
49  BEGIN
50    SET NOCOUNT ON;
51    ------------------------------------------------------------------------------------
52    -- variaveis temporarias
53    ------------------------------------------------------------------------------------
54    DECLARE @Temp int
55    --para conter o texto de log
56    DECLARE @Texto varchar(2000)
57
58    ------------------------------------------------------------------------------------
59    --VERIFICAÇÃO DA VALIDADE DOS PARAMETROS DE ENTRADA
60    ------------------------------------------------------------------------------------
61    SET @Temp =  (SELECT COUNT(id) AS Contador FROM dbo.tblPessoa WHERE (id = @RefOperador))
62    IF(@Temp = 0)
63    BEGIN
64      --Utilizador inválido
65      SET @Resultado = -2
66      RETURN
67    END
68
69    ------------------------------------------------------------------------------------
70    --GUARDAR A INFORMAÇÃO
71    ------------------------------------------------------------------------------------
72    BEGIN TRY
73    INSERT INTO [Credenciacao].[dbo].[tblPortaria]
74               ([Portaria],[Notas])
75    VALUES     (@NomePortaria, @Notas)
76
77      ------------------------------------------------------------------------------------
78      -- REGISTO DA OPERAÇÃO
79      ------------------------------------------------------------------------------------
80      -- Procura do id da portaria
81      SET @Temp =  (SELECT [Id] FROM [Credenciacao].[dbo].[tblPortaria] WHERE [Portaria] =
                     @NomePortaria)
82      -- Construção do texto de log
83      SET @Texto = 'O operador: ' + CONVERT(VARCHAR(15),@RefOperador) + ' criou a portaria ' +
                     CONVERT(VARCHAR(15),@Temp) + ' com o nome: ' + @NomePortaria
84      -- Registo de log
85      INSERT INTO [Credenciacao].[dbo].[tblLogInfPortaria]
86                          ([RefTipoLog],[RefPortaria],[RefOperador],[DataHora],[Operacao])
87           VALUES         (48, @Temp, @RefOperador, Current_Timestamp, @Texto)
88      -- Devolve sucesso
89      SET @Resultado = 0
90    END TRY
91
92    BEGIN CATCH
93      --Devolde Erro de gravação de dados
94      SET @Resultado = -500
95    END CATCH;
96
97  END
98  GO
```

Nas linhas 42 a 47 definem-se os parâmetros de entrada, os parâmetros de saídas e os respetivos tipos de variáveis. A partir da linha 49 começa o código propriamente dito.

Nas linhas 61 a 67 está implementado o bloco de código que faz a verificação da validade dos parâmetros de entrada. Neste exemplo apenas é necessário verificar a validade da variável de entrada que identifica o operador que está a efetuar a operação. Se o operador não estiver registado na base de dados, a execução do *stored procedure* termina neste

bloco e é devolvido na variável de saída o código de erro -2. Este código de erro, tal como está definido no Anexo K traduz-se por "Erro – Pessoa desconhecida".

Estando verificada a validade dos parâmetros de entrada, entra-se no segundo bloco de código que implementa a funcionalidade do *stored procedure*. Neste exemplo cria-se um novo registo da entidade portaria, linhas 73 a 75.

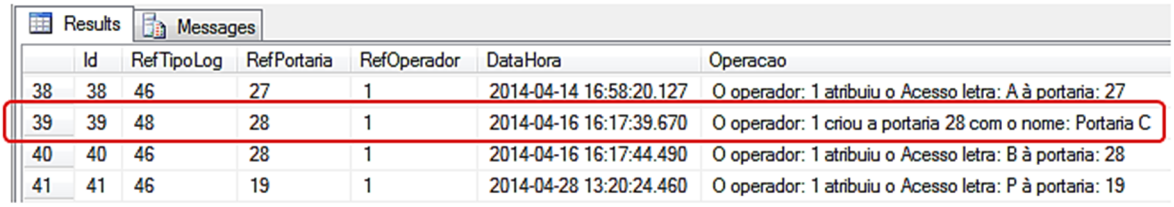
O último bloco de instruções faz o registo da execução desta operação na tabela de registo correspondente, linhas 81 a 89. Neste código há que notar as seguintes questões:

- Como estamos a operar sobre a entidade *Portaria*, a tabela de registo de eventos é a `tblLogInfPortaria`. No entanto, cada entidade tem uma tabela de registo de eventos própria como descrito nas listas apresentadas no modelo apresentado na Figura 130.

- No registo de eventos, o primeiro campo de registo é o `RefTipoLog`. Este campo é um valor numérico que está associado à informação do dicionário `tblDicTipoLog` que classifica o tipo de operação efetuada. Cada *stored procedure* de alteração de dados usa um código deste dicionário para facilitar o processo de procura de registos de eventos. Neste exemplo, o valor 48 corresponde à descrição "Cria Portaria". Esta informação é útil para, por exemplo, se pretendermos saber quem e quando criou registos da entidade *Portaria*, acedemos à `tblLogInfPortaria` e filtramos a sua informação de forma a obtermos apenas os registos cujo campo `RefTipoLog` tenha valor 48, desta forma obtemos a informação de quem criou e quando foram criadas as várias portarias.

- Genericamente, nos registos de eventos, além do tipo de operação, indica-se sempre as pessoas envolvidas sejam operadores, vigilantes ou portadores de cartões de acesso e regista-se a data-hora da ocorrência.

Na Figura 149, é apresentado um extrato do conteúdo da tabela `tblLogInfPortaria`, onde se pode ver, o resultado do registo da execução do *stored procedure* `spPortaria-Cria.`



Results | Messages

	Id	RefTipoLog	RefPortaria	RefOperador	DataHora	Operacao
38	38	46	27	1	2014-04-14 16:58:20.127	O operador: 1 atribuiu o Acesso letra: A à portaria: 27
39	39	48	28	1	2014-04-16 16:17:39.670	O operador: 1 criou a portaria 28 com o nome: Portaria C
40	40	46	28	1	2014-04-16 16:17:44.490	O operador: 1 atribuiu o Acesso letra: B à portaria: 28
41	41	46	19	1	2014-04-28 13:20:24.460	O operador: 1 atribuiu o Acesso letra: P à portaria: 19

Figura 149 – Exemplo do registo da execução do `spPortaria-Cria`.

Um detalhe de implementação que é importante referir no código dos *stored procedures* é o uso da estrutura *try-catch*, usada para encapsular as instruções de alteração de informação. A estrutura *try-catch*, contem dois blocos de instruções: o bloco *try* e o bloco *catch*. No bloco *try* colocam-se as instruções que fazem alteração de informação e que podem constituir uma fonte de geração de erros, por exemplo por violação de consistência de dados. As instruções deste bloco são executadas em conjunto, como se fosse uma única instrução atómica. Qualquer erro em qualquer instrução, inviabiliza, anula e volta atrás a execução de todas as instruções do bloco *try*. As instruções contidas no bloco *catch*, só são executadas se ocorrer um erro no bloco *try*, estas instruções implementam operações de tratamento de erros. Nos *stored procedures* desenvolvidos para implementação do projeto usaram-se estruturas *try-catch* sempre se altera dados nas tabelas.

Na implementação do projeto foram criados os *stored procedures* nas bases de dados *Credenciação*, *Portarias* e *PWNT* - base de dados do *Pro-Watch*, necessários à satisfação das necessidades das aplicações.

A denominação dos *stored procedures* nas bases de dados *Credenciação* e *Portarias* começa sempre com o prefixo "sp". Na base de dados *PWNT* o prefixo usado é "aasp", para que na ordenação alfabética dos *stored procedures*, os desenvolvidos no âmbito deste projeto não se misturem com os outros e sejam sempre mostrados no topo da lista. Na denominação, o prefixo é seguido do nome da principal entidade definida no modelo de dados e sobre a qual recai a funcionalidade do *stored procedure.* No nome pode ser incluída uma discrição da funcionalidade, que resulta na sintaxe:

- `sp<Entidade>[Descrição]` ou,
- `aasp<Entidade>[Descrição]`

Exemplos:

- `spPessoa-Cria` - *stored procedure* da base de dados *Credenciação* que cria um elemento da entidade Pessoa.

- `spCartaoAcedePortaria` - *stored procedure* da base de dados *Portarias* que devolve um valor lógico correspondente à permissão de passagem do portador de um cartão por um portaria especifica.

- `aaspCompany-ClearenceCode` - *stored procedures* na base de dados PWNT que devolve todos os *clearence code* de uma entidade *company*.

4.2.1.3. SINCRONISMO DE BASES DE DADOS *CREDENCIAÇÃO-PORTARIAS*

A base de dados *Portarias* contém uma cópia de parte da informação existente na base de dados *Credenciação*. Sendo esta informação apenas a estritamente necessária ao funcionamento da aplicação PORT. O motivo da existência desta base de dados é a criação de uma "parede" de segurança, para que os pontos de rede de dados onde os computadores que executam a aplicação PORT, não tenham acesso direto à base de dados principal.

No entanto, para ao correto funcionamento do sistema como um todo, é necessário garantir que em cada momento a informação da base de dados *Portaria* é igual à informação da base de dados *Credenciação*. Assim, sempre que houver alteração da informação na base de dados *Credenciação* a informação na base de dados *Portaria* tem de ser atualizada. O processo para implementar este sincronismo é usar o recurso *trigger* do *SQL Server*. Ver detalhes sobre *triggers* no Anexo H d).

O trecho Código 2, mostra um exemplo de implementação de *triggers* usados no sincronismo das bases de dados. Neste caso o *trigger* é executado após a alteração de registos na `tblPortaria,` sendo importante realçar os seguintes pontos:

- Na linha 34 faz-se a criação do *trigger* na base de dados *Credenciação*.

- A instrução `AFTER UPDATE`, linha 35, indica o tipo de evento que faz despoletar a execução do código. Esta instrução, neste exemplo concreto, indica que o código do *trigger* é executado após a alteração de campos da tabela a que o *trigger* está associado.

- Na linha 40, define-se que a tabela alvo da atualização está na base de dados *Portarias*. Este mecanismo é o processo de implementação do sincronismo entre as bases de dados.

- A informação de alteração de registos que faz despoletar os *triggers*, está contida nas entidades *inserted* nos casos de inserção ou alteração de registos e na entidade *deleted* nos casos de remoção de registos. Por isso nas linhas 41 e 42 do exemplo, para sincronizar a informação inserida na tabela da base de dados *Credenciação* na tabela da base de dados *Portarias*, usa-se a referência a *inserted* que contem essa informação.

- No caso concreto dos *triggers* usados neste projeto, não é necessário fazer validação de dados porque essa informação já foi verificada no *stored procedure* que provocou a execução do *trigger*.

Código 2 – Exemplo Código *Transct SQL* para implementar *Triggers*.

```
1  -- ================================================================================
2  --      _______    _
3  --     |__   __|  (_)
4  --        | |_ __ _  __ _  __ _  ___ _ __
5  --        | || '__| |/ _` |/ _` |/ _ \ '__|
6  --        | || |  | | (_| | (_| |  __/ |
7  --        |_||_|  |_|\__, |\__, |\___|_|
8  --                    __/ | __/ |
9  --                   |___/ |___/
10 --
11 -- ================================================================================
12 -- trgPortariaAtualiza
13 -- ================================================================================
14 --
15 -- Autor:     Sérgio Martins
16 -- Data:      2014-04-13
17 -- Descrição: Sincroniza a alteração de informação de portarias, entre as base de
18 --              dados Credenciação e Portarias
19 --
20 -- ================================================================================
21
22 SET ANSI_NULLS ON
23 GO
24 SET QUOTED_IDENTIFIER ON
25 GO
26 USE [Credenciacao]
```

```
GO
-- ================================================================================
IF  EXISTS (SELECT * FROM sys.objects WHERE object_id =
            OBJECT_ID(N'[dbo].[trgPortariaAtualiza]') )
 DROP TRIGGER [dbo].[trgPortariaAtualiza]
GO
-- ================================================================================

CREATE TRIGGER trgPortariaAtualiza ON [Credenciacao].[dbo].[tblPortaria]
AFTER UPDATE
AS
BEGIN
SET NOCOUNT ON;

 UPDATE [Portarias].[dbo].[tblPortaria]
  SET [Nome] = (SELECT DISTINCT i.Portaria FROM inserted i )
  WHERE [Id]=  (SELECT DISTINCT i.Id FROM inserted i )

END
GO
```

A Tabela 23, apresenta a lista de *triggers*, que foram criados para responder à alteração de dados nas tabelas da base de dados *Credenciação* e que garantem o sincronismo de informação entre as bases de dados *Credenciação* e *Portarias*.

Tabela 23 – Lista de *triggers* implementados para sincronismo *Credenciação-Portarias*.

BD Credenciação	*Trigger*	*Insert*	*Update*	*Delete*	BD Portarias
tblLicencaConducao	trgLicencaConducaoApaga			X	tblPessoa
	trgLicencaConducaoCria	X			
tblPessoa	trgPessoaAtualiza	X	X		tblPessoa
	trgPessoaCria	X			
tblPortaria	trgPortariaApaga			X	tblPortaria
	trgPortariaAtualiza		X		
	trgPortariaCria	X			
tblValidadeAcessoPessoaCor	trgPessoaAcessoCorAltera	X	X		tblPessoa
	trgPessoaAcessoCorApaga			X	
tblValidadeAcessoPessoaLetra	trgPessoaAcessoLetraAltera	X	X		tblPessoa
	trgPessoaAcessoLetraApaga			X	
tblValidadeAcessoPortaria	trgPortariaAcessoAltera		X		tblPortaria
	trgPortariaAcessoCria	X			
tblValidadePerfil	trgPerfilAtualiza	X	X		tblPessoa

4.2.2. IMPLEMENTAÇÃO DA APLICAÇÃO CRED

O módulo CRED, é a aplicação que onde se faz a administração do ambiente controlo de acessos de pessoas e viaturas. Nesta aplicação criam-se os utilizadores de cartões de acesso, definem-se as respetivas permissões, emitem-se cartões de acesso permanente e temporários, criam-se as portarias, etc. Este subcapítulo, é construído por duas partes, na primeira apresentam-se as funcionalidades desenvolvidas para a aplicação do ponto de vista de operação, a segunda parte foca-se nos detalhes técnicos de implementação das soluções criadas.

4.2.2.1. DESCRIÇÃO FUNCIONAL DA APLICAÇÃO CRED

No início da execução da aplicação é apresentado o ecrã de registo de entrada onde os utilizadores identificam-se com o seu número e introduzem a palavra-chave, Figura 150. Quando o ecrã é lançado, o campo de utilizador é preenchido com o número de identificação da última pessoa que usou a aplicação, se for a mesma pessoa a tentar entrar apenas tem de introduzir a palavra-chave.

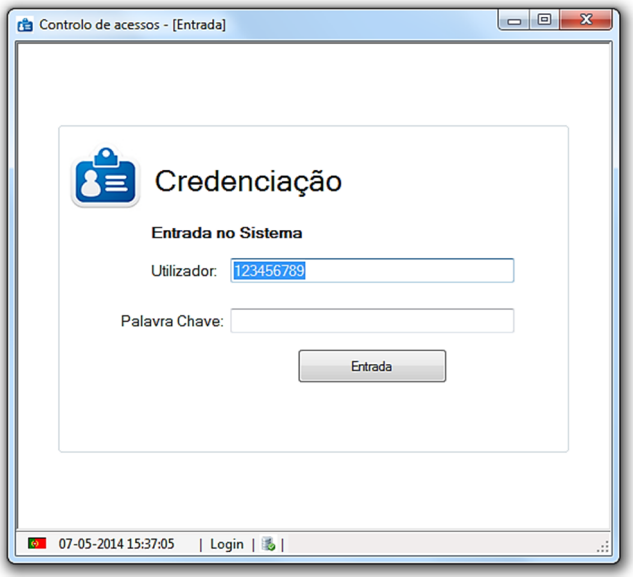


Figura 150 – CRED – Ecrã de *login*.

Após a validação da palavra-chave é apresentado o ecrã de início de trabalhos, Figura 151, onde o utilizador seleciona as funcionalidades pretendidas. Neste ecrã existem três áreas:

- Barra de menus que dá acesso aos ecrãs de operação:
    - Menu Pessoas - funcionalidades relacionadas com as pessoas: registo de dados, atribuição de acessos, de perfis, licenças de condução, emissão de cartões, etc.
    - Menu Portarias – Funcionalidades relacionadas com portarias.
    - Menu Entidades – Funcionalidades relacionadas com as empresas que operam no Aeroporto.
    - Menu Acessos – Funcionalidades relacionadas com o sistema automático de controlo de acesso Pro-Watch.
    - Menu Viaturas – Funcionalidades de registo de viaturas.
    - Menu Dicionários – Listas de consulta de parâmetros pré-definidos como: lista de companhias aéreas, listas de tipo de documentos de identificação, etc.
    - Menu Pesquisa – Funcionalidades de pesquisa sobre eventos relacionados com pessoas e portarias.

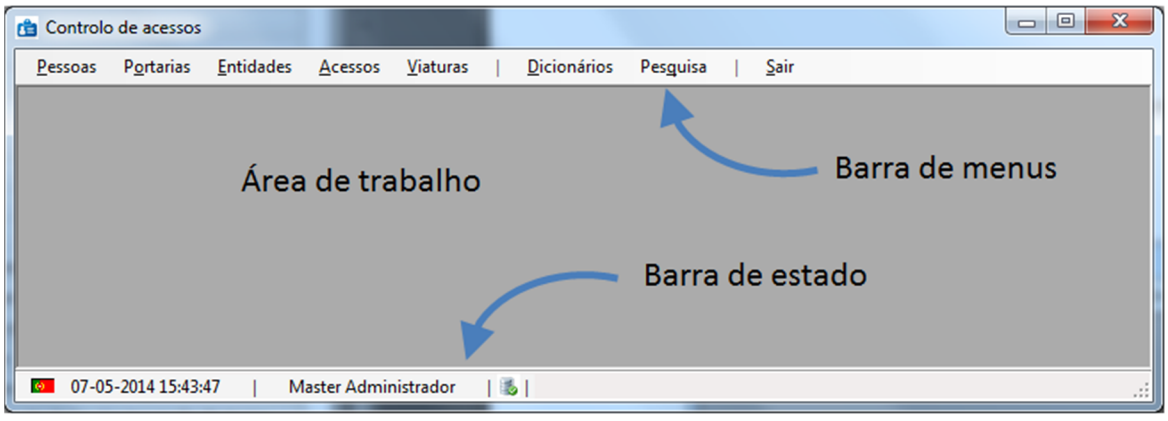


Figura 151 – CRED – Ecrã principal.

- Barra de estado, apresenta informação sobre a execução do sistema como a data/hora, o nome da pessoa que está a operar com o sistema e mostra uma imagem com a indicação do estado da ligação da aplicação CRED com a base de dados *Credenciação*.

Ecrã Pessoas

O ecrã Pessoas permite efetuar todas as operações sobre as pessoas registadas no sistema. Quando se entra no ecrã são apresentados inicialmente três objetos como mostrado na Figura 152:

- Uma caixa de texto que permite introduzir o número de identificação de uma pessoa já registada no sistema.
- Uma caixa de texto que permite introduzir o nome de uma pessoa já registada no sistema.
- Um botão que permite criar uma nova pessoa no sistema

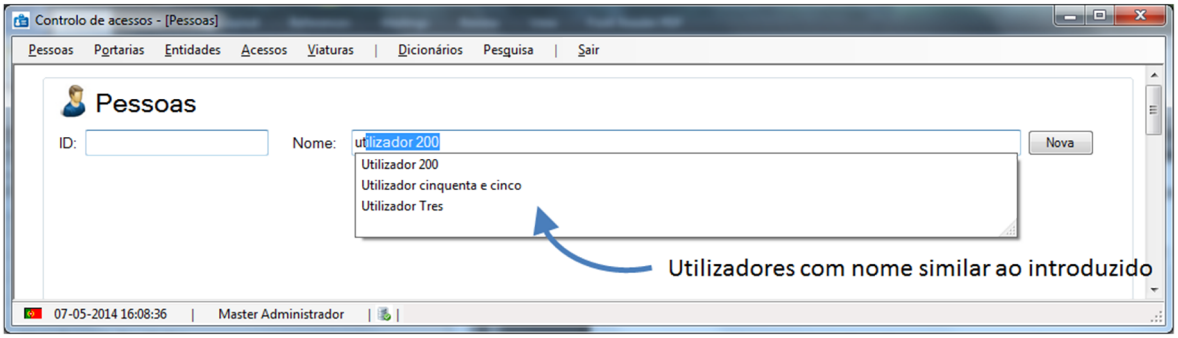


Figura 152 – CRED – Entrada no ecrã Pessoas.

As caixas de texto permitem introduzir dados de identificação de pessoas já existentes no sistema para se aceder à respetiva informação. Pode-se introduzir a identificação da pessoa, numa caixa ou na outra. Em cada uma delas à medida que se introduzem caracteres, é mostrada uma lista de pessoas já registadas que correspondem aos caracteres introduzidos,

Figura 152. Após a introdução da identificação da pessoa, é mostrado no ecrã a respetiva informação, como apresentado na Figura 153.

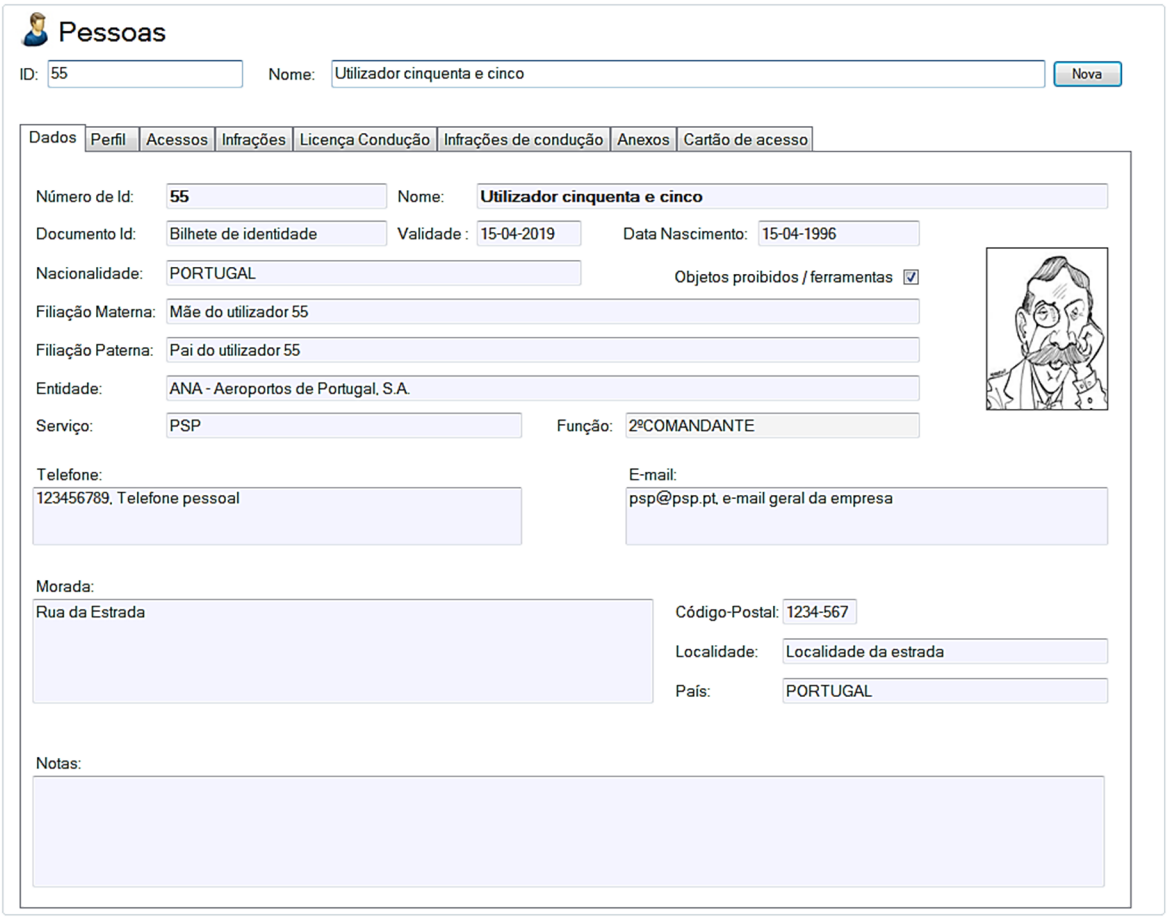


Figura 153 – CRED – Ecrã Pessoas.

A informação relativa às pessoas está organizada em páginas de tabuladores com a seguinte distribuição:

- Dados – página que contem informação geral de identificação da pessoa, meios de contacto, entidade que representa, foto, etc.
- Perfil – Listas de perfis que a pessoa assume. Em cada momento apenas um perfil está ativo.
- Acessos – Lista de códigos de acesso a áreas representado por letras e/ou cores e com o respetivo período de validade. Em cada momento apenas um código de cores está ativo e no máximo cinco códigos de letra.

- Infrações – Lista de infrações relacionadas com a vertente de segurança.
- Licenças de condução – Informação relativa à licença de condução de viaturas em áreas reservadas.
- Infrações de condução - Lista de infrações relacionadas com a vertente de condução de viaturas em áreas reservadas.
- Anexos - Lista de ficheiros anexos ao processo
- Cartão de acessos – Lista de cartões de acesso e funcionalidades de criação e emissão de cartões assim como associação de cartões a permissões de acesso no sistema *Pro-Watch*. Em cada momento apenas um cartão tem o estado ativo.

No Anexo M são apresentados todos os ecrãs da aplicação CRED.

As informações apresentadas nas páginas de tabuladores, só disponibilizam informação para leitura. Quando for necessário alterar essa informação pressiona-se com o botão direito do rato sobre o campo que se pretende alterar e é mostrado um menu com opções de operação dependentes do campo selecionado.

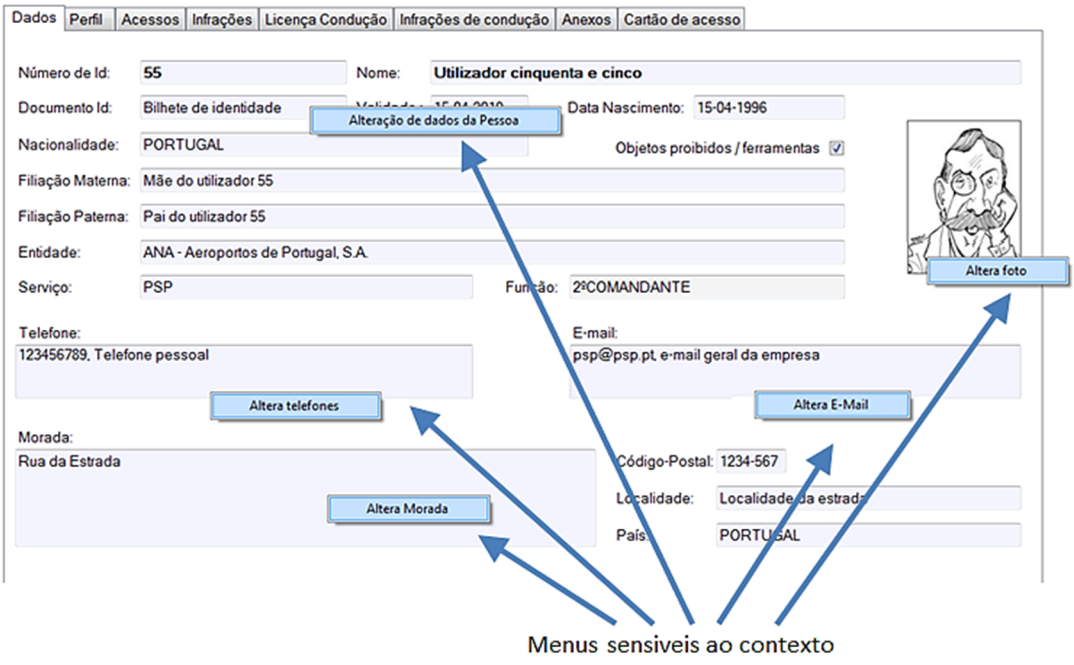
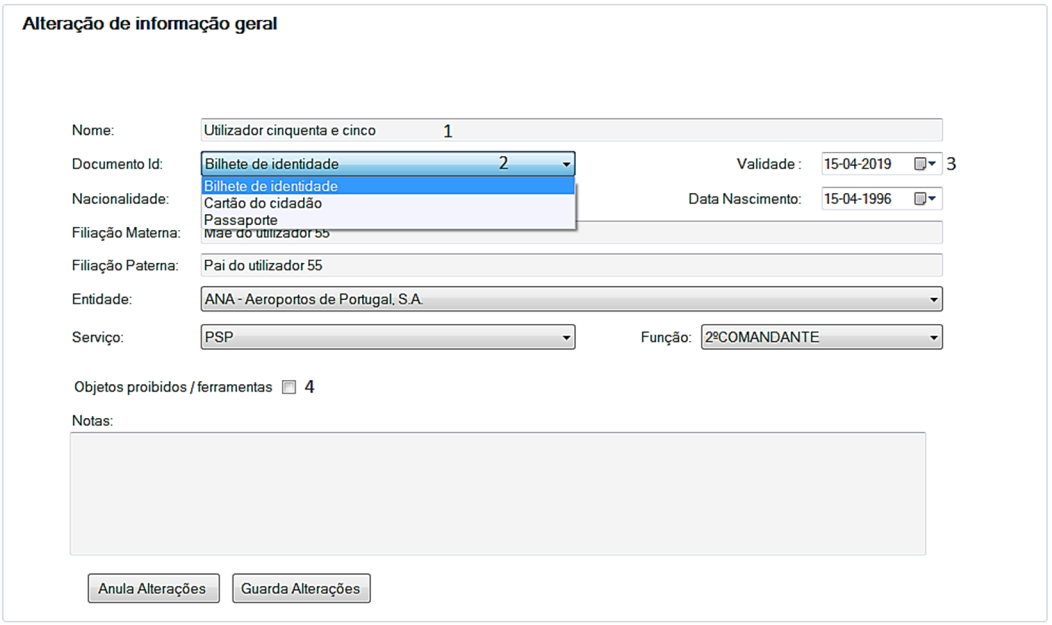


Figura 154 – CRED – Menus sensíveis ao contexto.

A ativação de um menu para alteração de dados apresenta um ecrã que pode assumir dois formatos: ou são apresentados vários campos de introdução de informação, como mostrado na Figura 155 ou é apresentada uma de tabela como mostrado na Figura 156. No caso das caixas de introdução existem campos de escrita livre, campos de escolha de valores definidos em dicionários, campos de seleção e campos de datas.



1 – Campo de escrita livre; 2 – Campo de escolha de valores definidos em dicionários; 3 – Campo de datas; 4 – Campo de selecção

Figura 155 – Alteração de dados com caixas de introdução.

No caso da introdução de dados em tabelas, também existem menus sensíveis ao contexto para executar operações específicas, Figura 156.

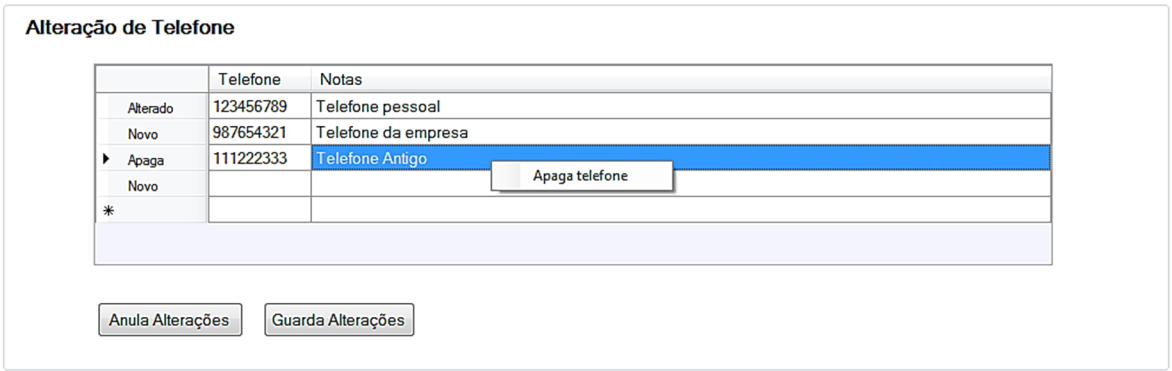


Figura 156 – Alteração de dados em tabela.

Nos dois mecanismos de introdução/alteração de dados, a informação apenas é guardada se o botão "Guardar Alterações" for pressionado. O uso do botão "Anula Alterações" volta ao ecrã anterior e qualquer alteração de dados efetuada é descartada.

Por exemplo, na página onde se mostra a lista de ficheiros anexos do processo da pessoa, Figura 157, o menu sensível ao contexto contem opções para adicionar anexos, para remover anexos e tem uma opção para abrir o anexo selecionado, isto é, executa a aplicação que lê o ficheiro correspondente e apresenta o ficheiro nessa aplicação.

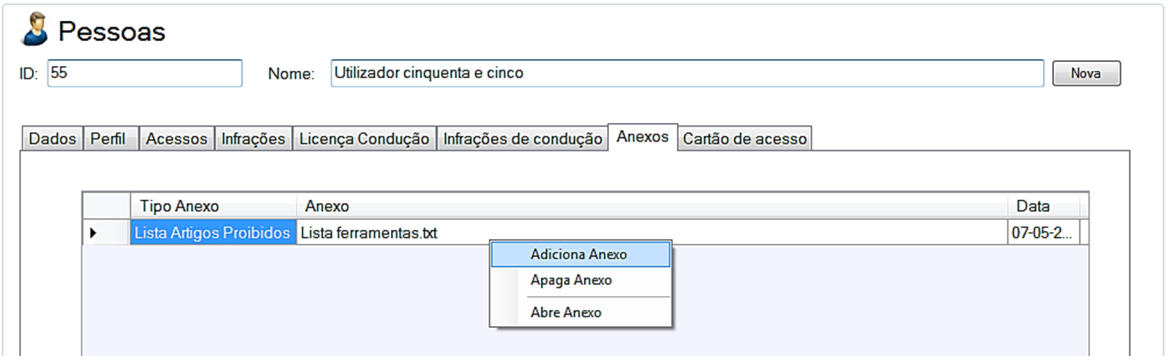


Figura 157 – Anexos do processo de uma pessoa.

Aquisição de fotografia

Uma exceção aos dois métodos de introdução de dados apresentados anteriormente é o mecanismo de introdução da fotografia da pessoa. Para alterar a foto de uma pessoa, pressiona-se com o botão direito do rato sobre a foto e seleciona-se a opção "Altera foto". Este procedimento dá aceso ao ecrã mostrado na Figura 158 onde é apresentada a foto atual, se existir, e ficam disponíveis os botões de operação.

Figura 158 – Alteração da foto da pessoa.

A introdução de uma nova foto pode ser feita usando um de dois métodos: ou se usa o botão "Foto de ficheiro" para selecionar um ficheiro já existente com a foto, ou se faz a aquisição da imagem da pessoa através de uma camara ligada ao computador onde a aplicação está a ser executada, pressionando o botão "Aquisição de foto". Com esta opção, é apresentado um novo ecrã, Figura 159. O processo de aquisição da foto inicia-se com o posicionamento da pessoa na imagem da camara, quando se tiver a composição pretendida, adquire-se a imagem, pressionando o botão "Terminar Aquisição".

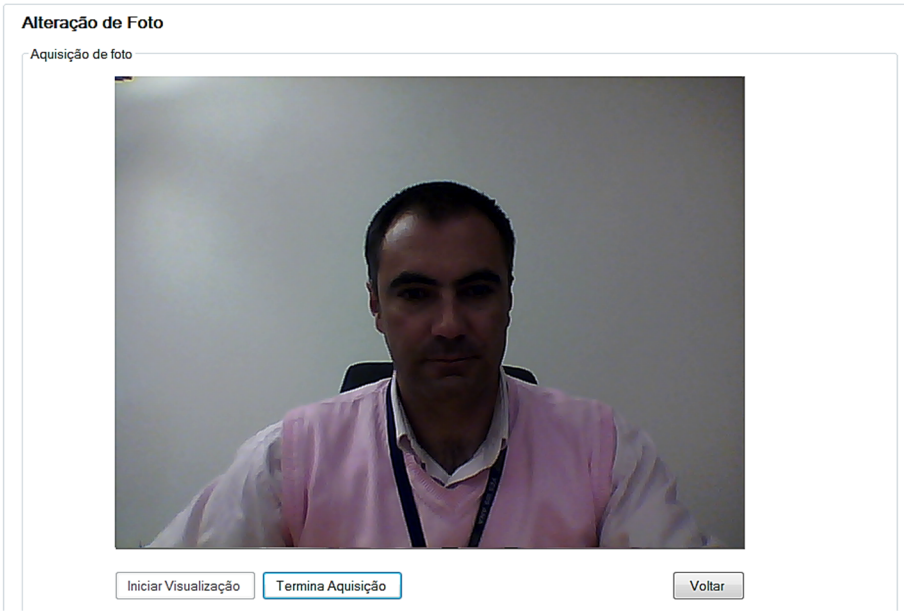


Figura 159 – Aquisição de foto.

A imagem adquirida pela camara, está então pronta para ser tratada. O operador, ao passar o rato sobre a foto com o botão esquerdo pressionado, faz aparecer um retângulo a verde que permite efetuar o enquadramento da foto pretendida, ao soltar o botão do rato, fixa-se a imagem para constituir a nova foto, Figura 160.

Ao pressionar o botão "Guarda", a nova foto é guardada na base de dados e passa a ser a foto que é impressa com nos cartões de acessos.

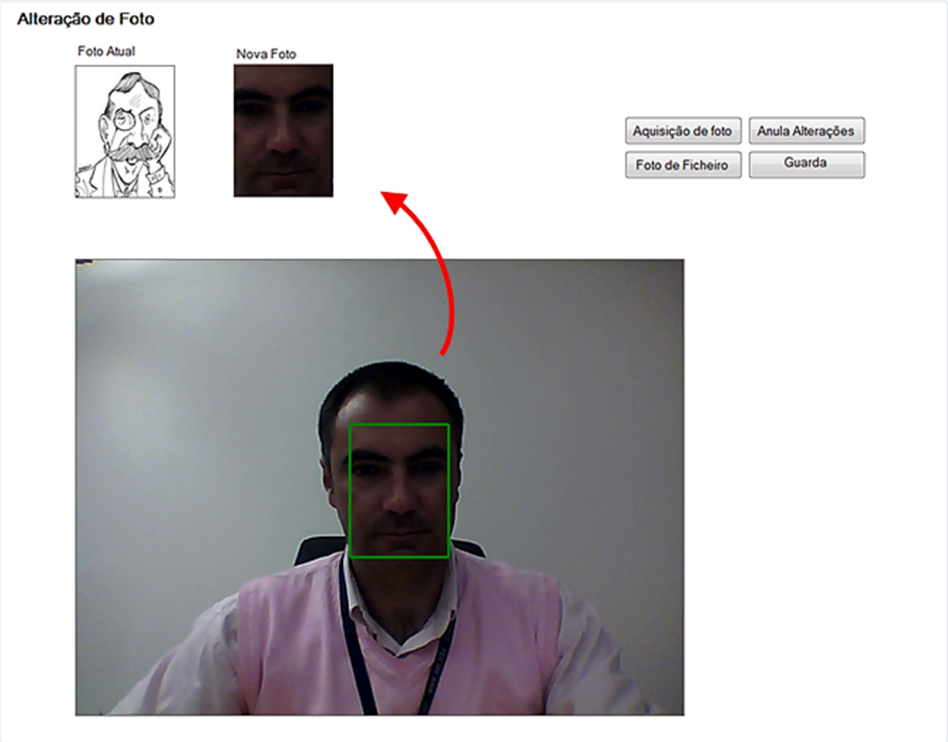


Figura 160 – Enquadramento da foto.

Impressão de cartões

A última página do ecrã "Pessoas", apresenta uma lista dos cartões de acesso que foram atribuídos à pessoa selecionada. Na definição de informação do cartão, no campo "Pro-Watch Company" faz a associação do cartão às permissões de acesso de portas controladas pelo sistema *Pro-Watch*.

Em cada momento, apenas um cartão está no estado "Ativo". O botão "Impressão de cartão", Figura 161, para o cartão ativo, dentro das datas de validade, dá acesso à funcionalidades de impressão.

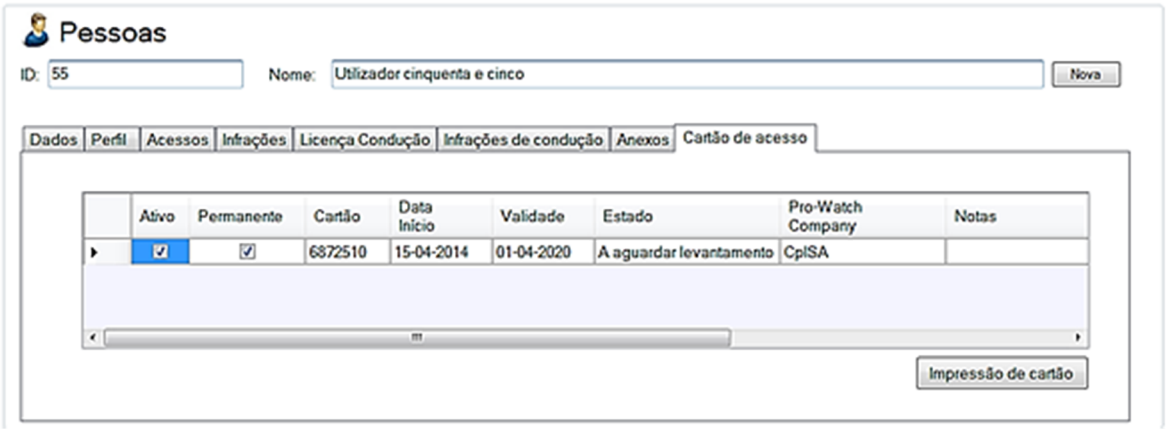


Figura 161 – Lista de cartões de acessos.

Usando o botão de impressão de cartão, é apresentado um ecrã com o aspeto final do cartão de acesso, contendo a informação da pessoa: nome, foto, acessos, data de validade, etc., Figura 162.

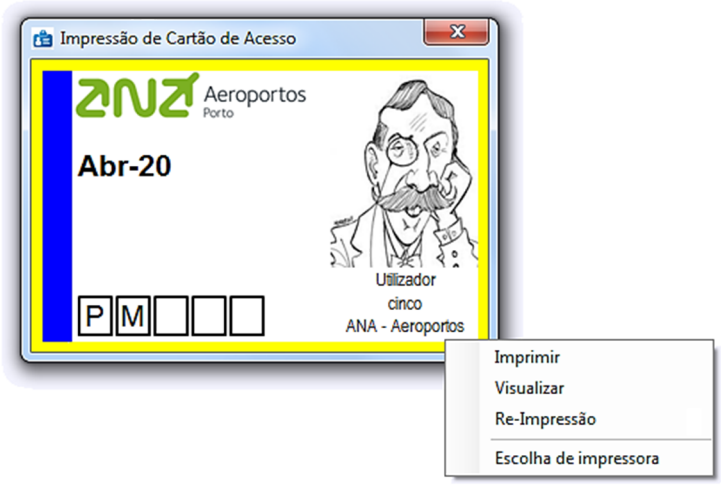


Figura 162 – Impressão de um cartão de acesso.

Quando se pressiona o botão esquerdo do rato sobre o cartão ou se usa a tecla "Menu" do teclado, é apresentado o menu da Figura 162 que tem quatro operações:

- Imprimir – que envia o cartão para ser impresso na impressora definida.
- Visualizar – que mostra uma previsão do resultado da impressão do cartão, usando os *drives* da impressora definida.

- Reimpressão – que permite imprimir a data de validade até mais duas vezes, possibilitando da reutilização do cartão.

- Escolha de impressora – mostra um ecrã para selecionar a impressora na qual o cartão vai ser impresso.

Quando um cartão está no estado "Emitido", no menu do cartão, o item "Imprimir" não está visível e apenas é possível efetuar reimpressões. A reimpressão é uma reutilização de um cartão já emitido efetuando o prolongamento da data de validade. Esta operação pode ser feita até duas vezes, em que, em cada reimpressão do cartão corta-se a data de validade atual e imprime-se a nova data, Figura 163.

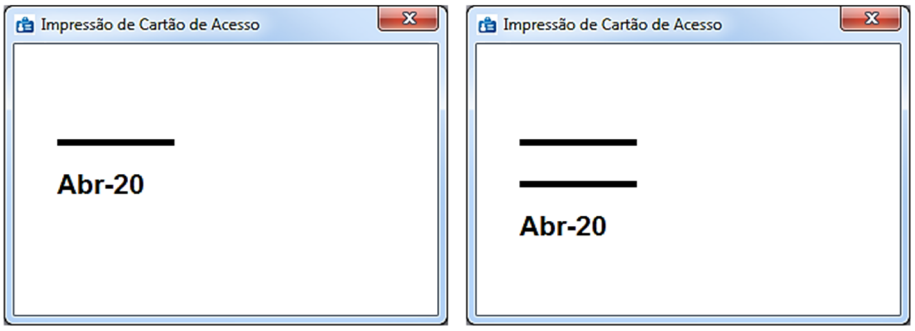


Figura 163 – Reimpressão de um cartão de acesso.

Ecrã Portarias

De forma similar ao ecrã "Pessoas" o ecrã "Portarias", inicia-se com uma caixa de seleção de portarias existentes, que dá acesso à respetiva informação e com o botão que permite criar uma nova portaria, Figura 164

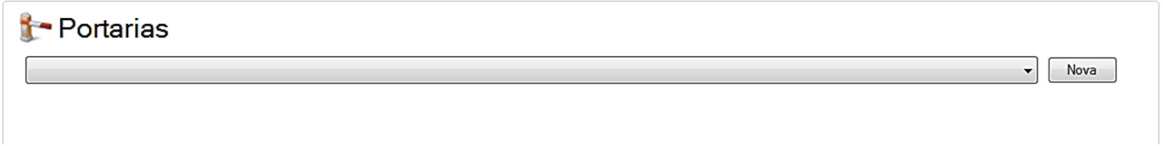


Figura 164 – Ecrã portarias.

Selecionando uma portaria na caixa de escolha, é apresentada a lista de acessos permitidos através dessa portaria. A lista de acesso pode ser alterada como habitualmente pressionando na lista com o botão direito do rato, Figura 165.

Figura 165 – Ecrã portarias - Acessos.

Ecrã Entidades

O ecrã "Entidades" permite fazer a gestão das empresas a que as pessoas que vão ser portadoras de cartões de acesso representam. O modo de funcionamento deste é ecrã é similar aos outros já apresentados, após a apresentação do ecrã podemos selecionar uma entidade já existente através do seu número de identificação fiscal ou através do nome, e pode-se também proceder à criação de uma nova entidade, Figura 166.

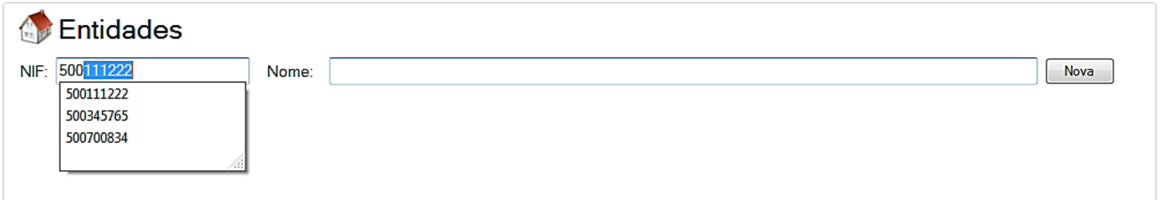


Figura 166 – Ecrã entidades.

Após a seleção de uma entidade, são apresentados os seus dados gerais que podem ser alterados através das funcionalidades disponíveis nos menus sensíveis ao contexto, Figura 167.

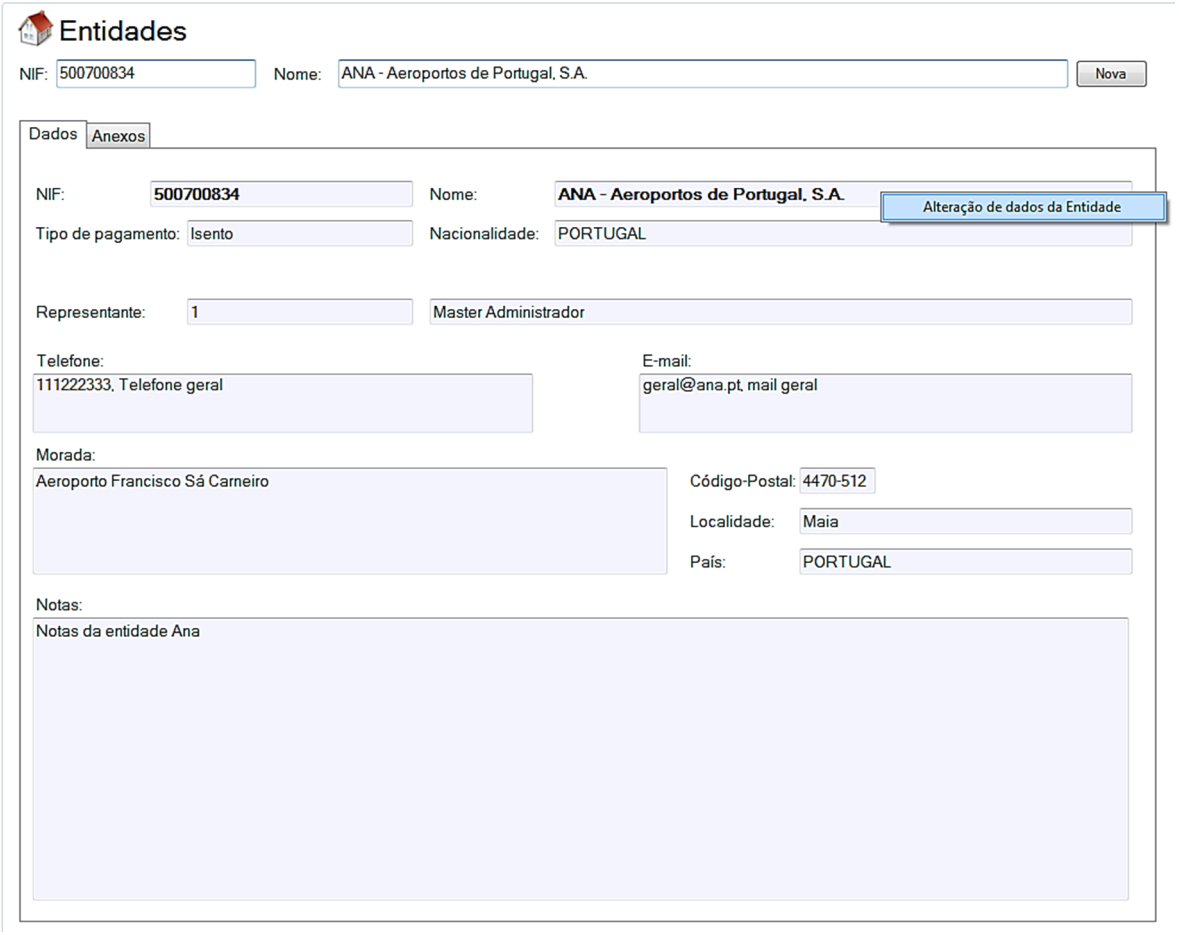


Figura 167 – Ecrã entidades – dados gerais.

Ecrã Acessos

Conforme descrito na secção 3.3.2, as portas - *Logical Devices*, controladas pelo sistema SACA instalado no Aeroporto Francisco Sá Carneiro, podem ser agrupadas em conjuntos denominados *Clearence Codes*. As pessoas, identificadas por cartões de acesso – *Badges,* são organizadas em grupos, normalmente com as mesmas permissões de acessos, denominadas *Companies*. A definição de permissões de acesso é feita pela associação de *Clearence Codes* com *Companies*.

O ecrã acessos da aplicação CRED, apresenta a informação relevante do sistema *Pro-Watch* que controla as permissões de abertura de portas do SACA. Este ecrã é apresentado em páginas de tabuladores principais e páginas de tabuladores secundárias. Existe uma

página principal para cada entidade: *Logical Device*, *Clearence Code*, *Company*, *Badge*. E existe uma página secundária para cada entidade diferente da página principal selecionada, como se pode verificar na Figura 168.

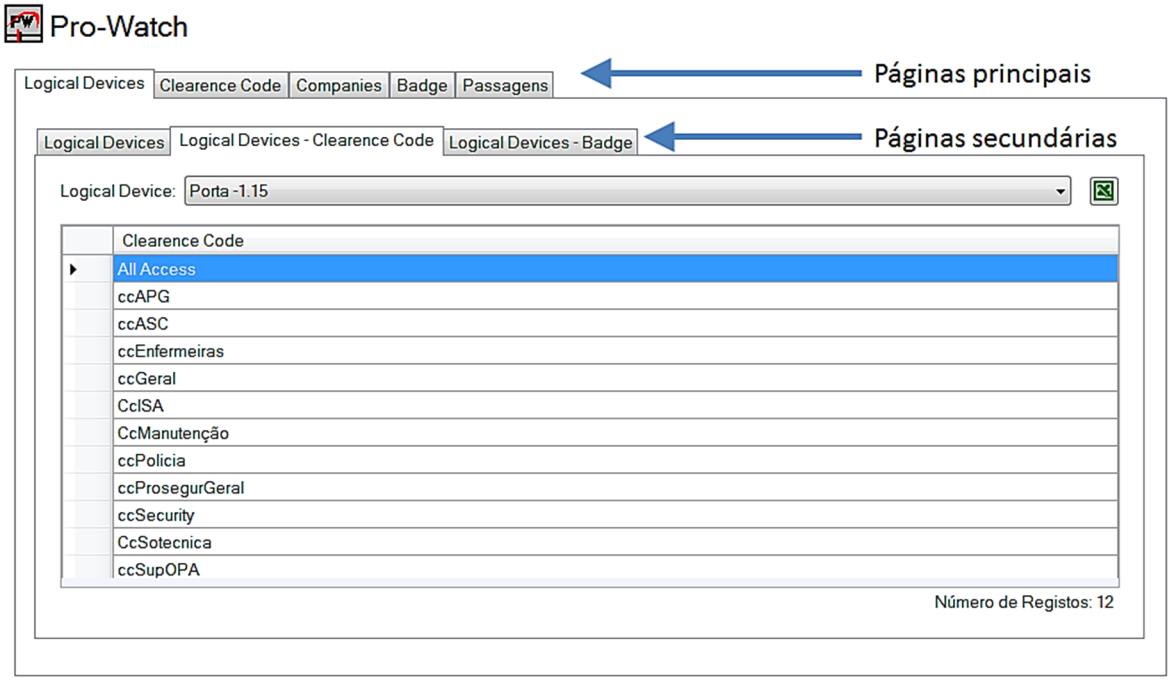


Figura 168 – Ecrã acessos.

Em cada página secundária é possível selecionar um elemento relativo à página principal e obter a lista dos elementos que estão relacionados. No exemplo da Figura 168, na página de *Logical Devices*, selecionando a porta- *Logical Device* -1.15, podemos ver a que grupos - *Clearence Code* a que essa porta pertence. Com esta metodologia podemos saber que portas pertencem a um grupo, que acessos tem cada companhia, que portas estão atribuídas a um cartão, etc. Resumindo podemos analisar todas as perspetivas da relação muitos-para-muitos das entidades *Logical Device*, *Clearence Code*, *Company* e *Badge*.

Em cada página, existe também, um botão com imagem identica à mostrada na Figura 169. Este botão quando pressionado abre uma instancia da aplicação *Microsoft Excel*, cria um novo documento nessa aplicação e exporta a informação que está apresentada na tabela do ecra para *Excel*. Esta funcionalidade permite ir guardando várias pesquisas para poder posteriormente consultar/manipular a informação. Alem da funcionalidade ser implementada no ecrã "Acessos", também foi incluida em vários outros ecras, onde se considerou util a exportação de dados.

Figura 169 – Botão de exportação de dados para Excel.

A ultima página do ecrã "Acessos" denominada "Passagens", permite efetuar pesquisas baseadas na data de apresentação de cartões de acesso nos leitores controlados pelo *Pro-Watch*, mostrando a data/hora em que o evento ocorreu, mostrando o nome o *Logical Device* em causa e apresentando o resultado da permissão de acesso, Figura 170.

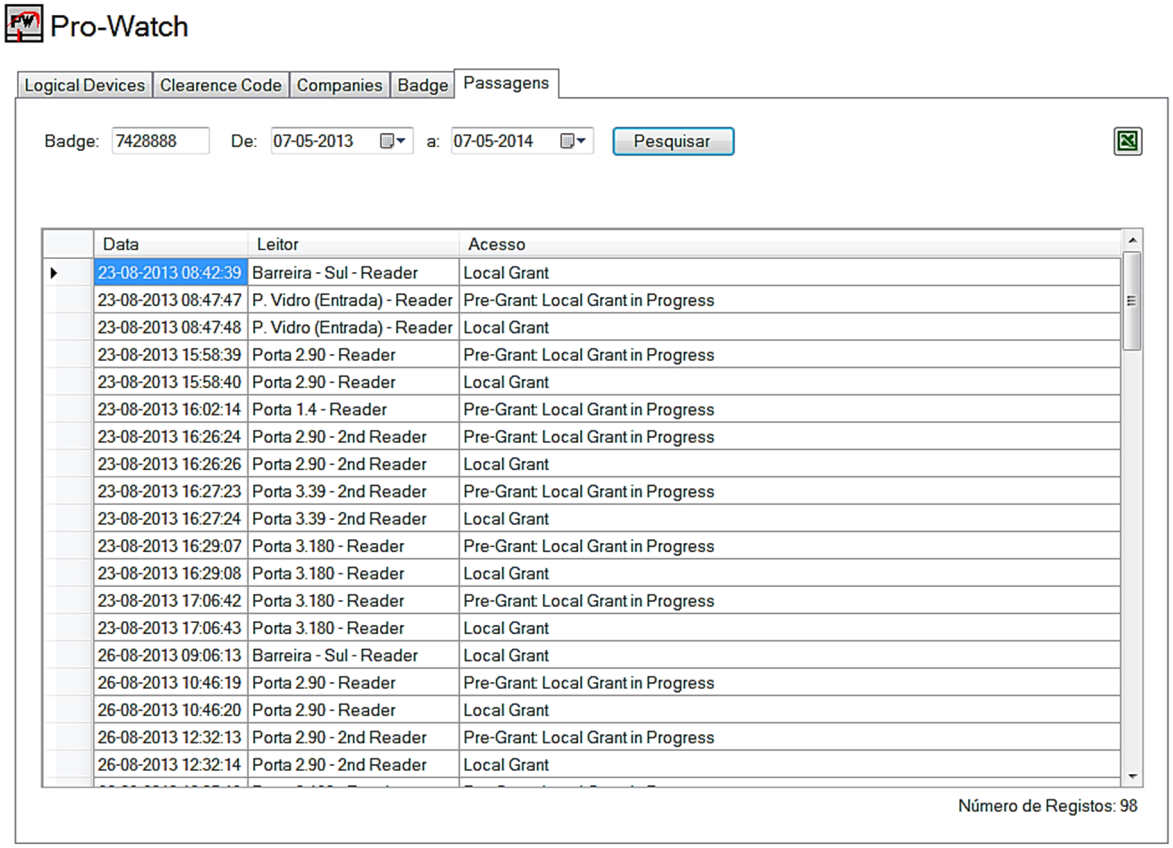


Figura 170 – Registos de abertura de portas de um cartão.

Ecrã Viaturas

O ecrã de viaturas permite criar e gerir o registo de viaturas que circulam em áreas reservadas do Aeroporto, o princípio de uso deste ecrã está em linha com as considerações efetuadas para os ecrãs já apresentados, como mostrado na Figura 171 e Figura 172.

Figura 171 – Ecrã Viaturas.

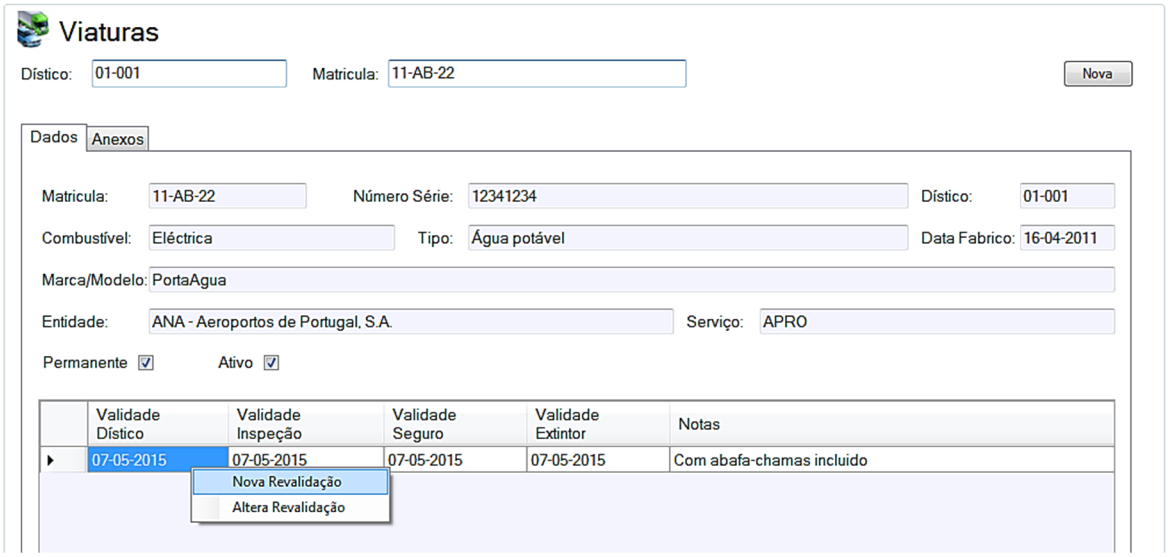


Figura 172 – Dados das viaturas.

Ecrã Dicionários

O ecrã "Dicionários", mostra conjuntos de informação fixa, pré-configurada, que é usada nos vários recursos das aplicações. A lista de tipos de documentos de identificação, que contem os itens "Cartão de cidadão", "Passaporte" e "Bilhete de identidade", é um exemplo de um dos dicionários existentes na plataforma.

No ecrã "Dicionários", na caixa de escolha seleciona-se o tipo de dados que se pretende ver e a informação é apresentada em formato de tabela como mostrado na Figura 173. Este ecrã é um dos que também permite exportar a informação mostrada para *Excel*.

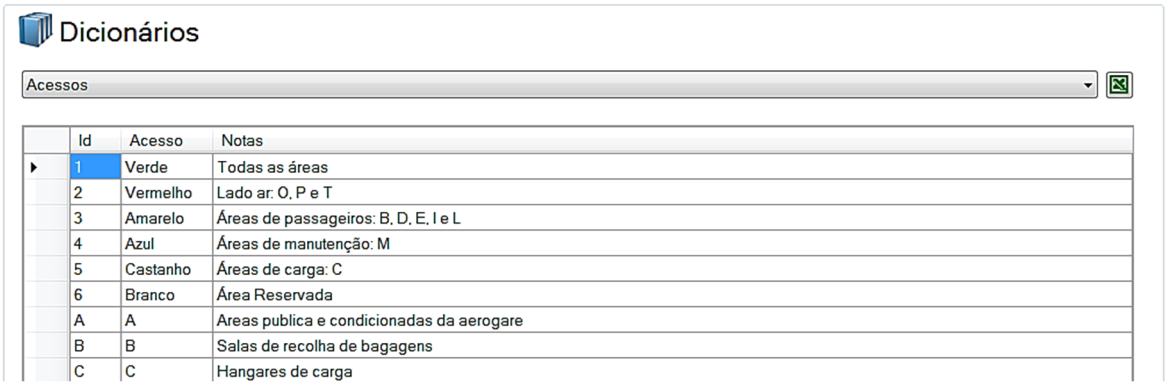


Figura 173 – Ecrã Dicionários.

Ecrã Pesquisa

O ecrã de "Pesquisa", permite efetuar pesquisas de informação na base de dados relativas às pessoas e às portarias, nomeadamente sobre as perspetivas de acessos, perfil, passagens em portarias, etc. As informações podem ser filtradas considerando critérios específicos e podem se exportadas para *Excel*, Figura 174, Figura 175.

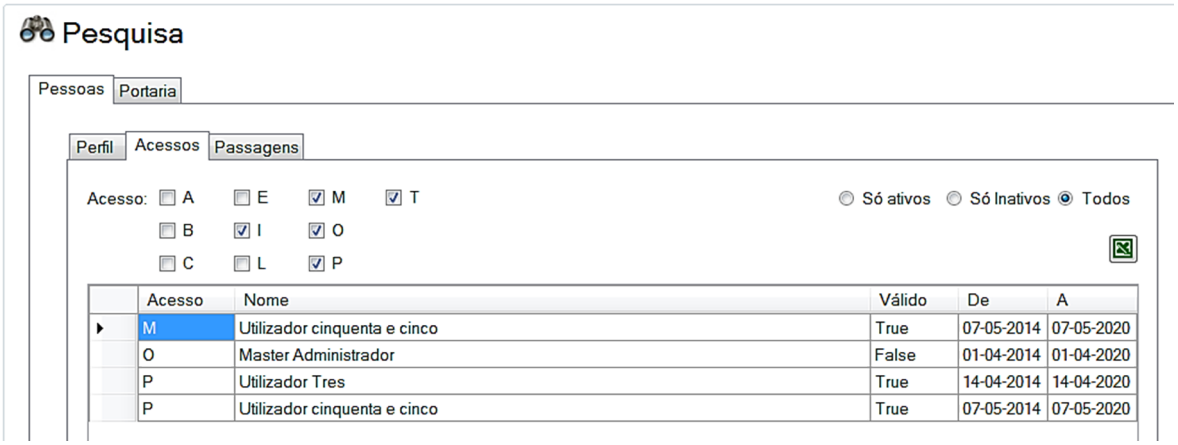


Figura 174 – Ecrã de pesquisa, acessos.

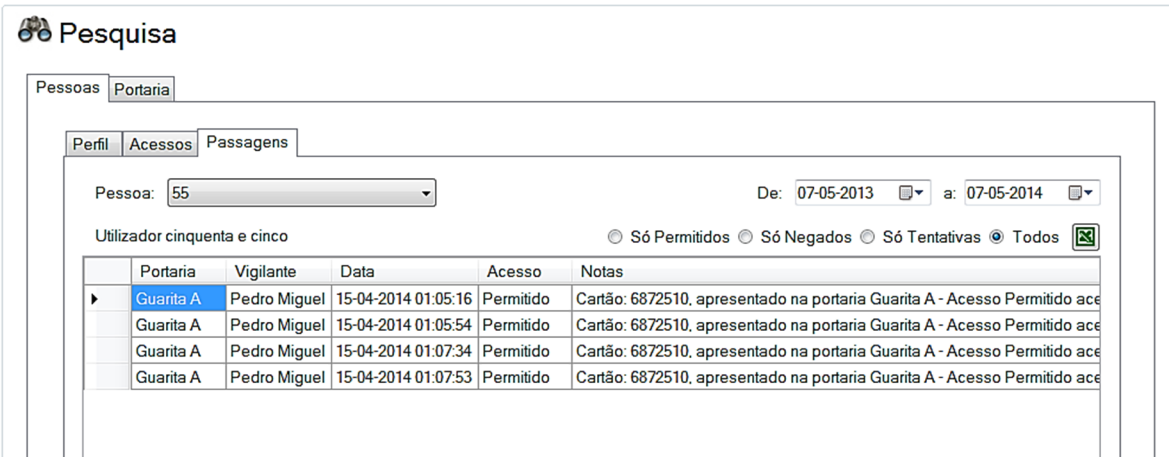


Figura 175 – Ecrã de pesquisa, passagem em portarias.

4.2.2.2. DESCRIÇÃO DA IMPLEMENTAÇÃO DA APLICAÇÃO CRED

Neste subcapítulo apresentam-se os detalhes técnicos de implementação da aplicação CRED.

Do ponto de vista de implementação cada uma das aplicações da plataforma foi desenvolvida sobre um objeto do tipo *MDIform*. Os *MDIform* graficamente são janelas que servem de conter outras janelas. Dentro dessa janela principal foram definidas as barras de estados e as barras de menus que dão acesso às funcionalidades implementadas. As janelas secundárias foram implementadas usando objetos do tipo *Form*, janelas que tem existência dentro do objeto *MDIform*. No trecho de Código 3, é mostrada a função que é executada quando se pressiona o rato sobre o menu "Pessoas", este código, fecha qualquer outra janela secundária que esteja aberta, linha 3, e abre, no estado maximizado, a janela que contem as funcionalidades da entidade Pessoa, linhas 4 a 9.

Código 3 – Exemplo de função de assistência ao uso do menu pessoas.

```
1  Private Sub PessoaToolStripMenuItem_Click(sender As System.Object, e As System.EventArgs)
                                            Handles ToolStripMenuItem_Pessoas.Click

2    On Error Resume Next
3    FechaTodosForms()
4    With frmPessoas
5      .MdiParent = Me
6      .Dock = DockStyle.Fill
7      .WindowState = FormWindowState.Maximized
8      .Show()
9    End With
10 End Sub
```

Na definição dos componentes das barras de menus de todas as aplicações, integrou-se um atalho para que o menu possa ser acedido apenas com o uso do teclado. Esta funcionalidade é implementada acrescentando no nome do menu o caracter "&" antes da letra que se pretende que seja o atalho, Figura 176. Assim quando a aplicação está em execução ao pressionar a tecla ALT, as letras que constituem atalho para o menu apresentam-se sublinhadas, o pressionar dessas letras no teclado é equivalente a pressionar o menu com o botão esquerdo do rato.

Figura 176 – Criação de menus.

No desenvolvimento das aplicações, foram usados os vários objetos que a ferramenta *Visual Basic* disponibiliza, como: botões, caixas de texto, caixas de seleção, tabuladores, imagens, etc. Na distribuição dos objetos pelo espaço disponível tiveram-se alguns cuidados para que o aspeto final fosse o mais "leve" possível, nomeadamente:

- Teve-se o cuidado de manter o alinhamento e o espaçamento entre objetos de forma regular.

- Teve-se o cuidado de manter o espaço, tanto quanto possível, livre de elementos menos usados e por isso se optou pelo uso de menus sensíveis ao contexto que em vez de usar vários botões.

- Implementou-se o acesso aos objetos através do teclado, isto é, estando o cursor num objeto, o pressionar a tecla TAB, transporta o cursor para o objeto imediatamente à direita, ou, se o objeto for o último na linha transporta o cursor para o primeiro objeto da linha seguinte, assim é possível fazer a introdução de dados usando apenas o teclado e não haver a necessidade de movimentar a mão para o rato. Esta funcionalidade é implementada definindo para cada objeto a propriedade `TabStop = True` e a propriedade `TabIndex` com um valor numérico, inteiro, que representa a sequência que se pretende atribuir.

- Nos objetos que têm menus sensíveis ao contexto, os menus podem, também, ser acedidos pelo teclado usando a tecla de menu.

Os menus sensíveis ao contexto são implementados usando objetos do tipo *ContextMenuStrip* que do ponto de vista de desenvolvimento são idênticos aos itens dos menus visíveis. Depois de criados, os *ContextMenuStrip* são associados aos objetos que os usam, através da propriedade `ContextMenuStrip` do objeto em causa. O trecho Código 4

mostra a associação do menu `ContextMenuStrip_DadosPessoaFoto_Altera` ao objeto imagem que contem a foto da pessoa. Na aplicação, o pressionar o botão direito do rato sobre a foto faz aparecer o respetivo menu com mostrado na Figura 177.

Código 4 – Exemplo de associação de menu sensível ao contexto.

```
1  PictureBox_PessoaFoto.ContextMenuStrip = ContextMenuStrip_DadosPessoaFoto_Altera
```

Figura 177 – Menu sensível ao contexto.

Comunicação das aplicações com a base de dados

A implementação da comunicação entre as aplicações e as bases de dados foi efetuada usando a ferramenta do *Visual Basic* denominada *Language-Integrated Query* – LINQ.

O LINQ disponibiliza [87] um ambiente gráfico chamado *Object Relational Designer* – O/R Designer, que possibilita efetuar o mapeamento entre elementos no domínio relacional com objetos no domínio do paradigma de programação orientada por objetos. Esta potencialidade permite [107], [108], que quando se opera com a base de dados em vez de gerir a ligação à base de dados e lançar a execução de *queries* ou *stored procedures* em *Transact SQL* como acontece com outros métodos de acesso, se possam definir objetos que encapsulam as entidades relacionais, que interagem com a base de dados e que são tratados como instâncias de classes.

Na implementação da aplicação em *Visual Basic* foram criadas classes de dados através do LINQ, para conter as *views* e os *stored procedures* que constituem a camada de interface

das bases dados. A criação dessa classe pode ser efetuada graficamente usando o menu do ambiente do *Visual Basic*: "Project – Add New Item…" que apresenta a janela de seleção de itens que se podem adicionar ao projeto. Nesta janela de escolha de objetos, seleciona-se "LINQ to SQL Class", como mostrado na Figura 178.

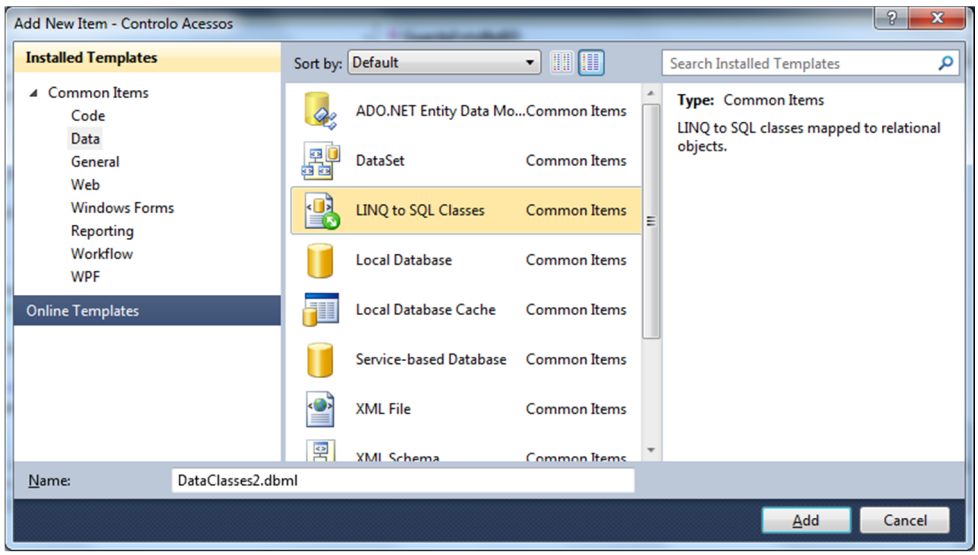


Figura 178 – LINQ, Criação de uma classe para interface com a base de dados.

Como resultado desta operação na área de desenvolvimento do *Visual Basic* é mostrada uma interface gráfica que representa a classe criada, Figura 179. A interface gráfica apresenta-se dividida em duas zonas, o lado esquerdo da interface é relativo às propriedades da classe, o lado direito é relativo aos métodos da classe.

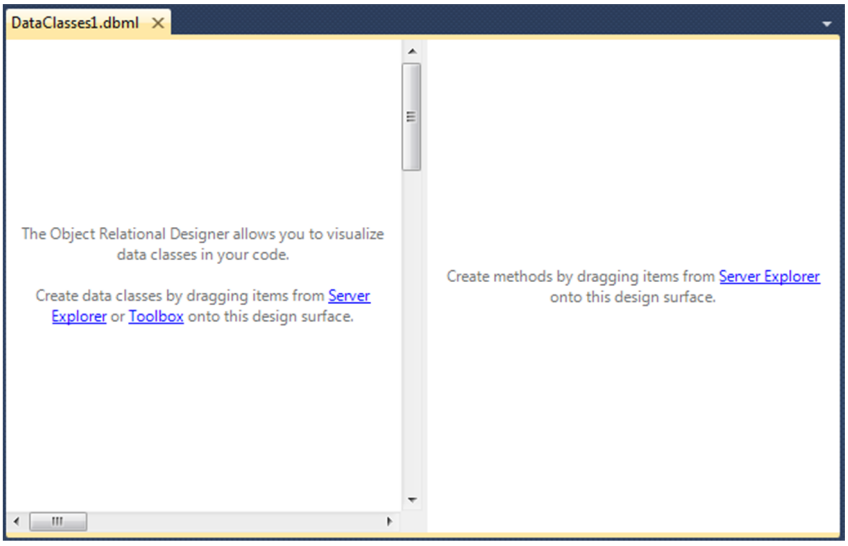


Figura 179 – LINQ, Criação de uma classe: propriedades e métodos.

O objetivo desta classe é conter os *views* que fazem parte da interface da base de dados convertendo-os em propriedades da classe e conter os *stored procedures* da base de dados convertendo-os em métodos da classe. Para atingir este objetivo executam-se os seguintes passos:

1. Cria-se uma ligação de servidor entre a aplicação e a base de dados, ver Anexo H - e).

2. No ambiente do *Visual Basic*, coloca-se visivel simultaneamente a janela *Server Explorer* e a janela da classe de dados que criamos usando o LINQ.

3. Na janela *Server Explorer* expande-se a arvore da ligação ao servidor no ramo *views* e no ramo *stored procedures*.

4. Arrastam-se da janela *Server Explorer* os *views* que se pretende usar na aplicação para área esquerda da janela da classe.

5. Arrastam-se da janela *Server Explorer* os *stored procedures* que se pretende usar na aplicação para área direita da janela da classe.

Com esta operação mapeamos os *views* e *stored procedures* da interface da base de dados na classe de dados que vai ser usada nas aplicações e obtemos um método de acesso à informação sem termos de cuidar dos detalhes da ligação entre a aplicação e o servidor de dados.

Na Figura 180, é mostrado um exemplo de criação de classes usando o LINQ. Neste caso criou-se uma classe para aceder à interface da base de dados PWNT. Do lado esquerdo a imagem estão os *views* e do lado direito os *stored procedures*, uns e os outros foram criados no âmbito deste projeto para interface entre a aplicação CRED e a base de dados do *Pro-Watch.*

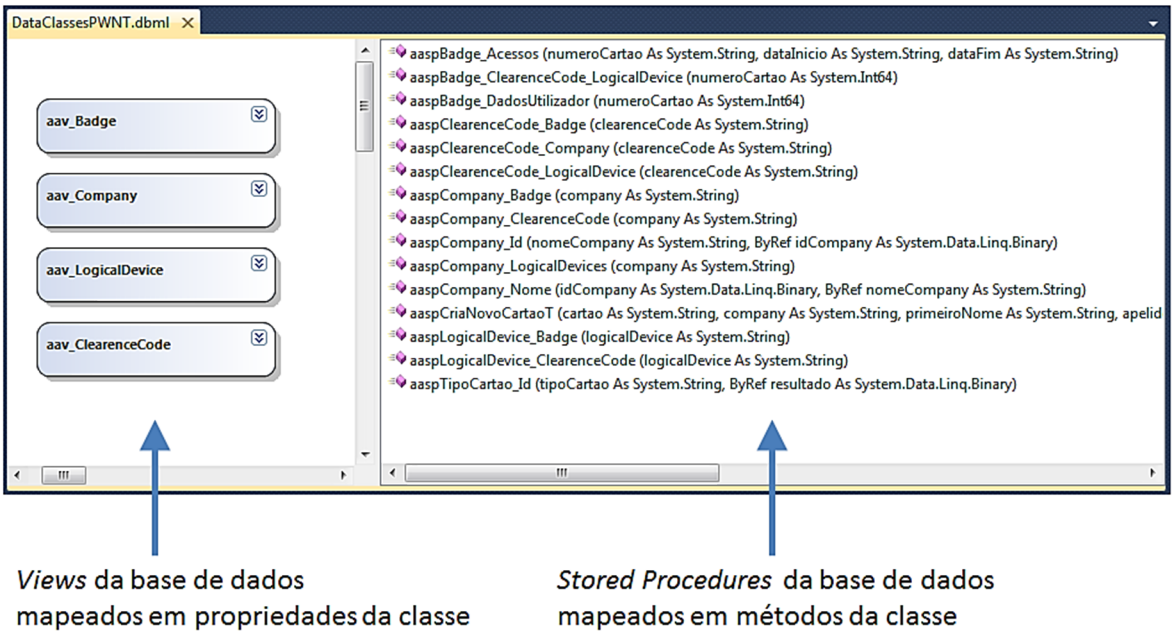


Figura 180 – LINQ, exemplo de classe de dados.

Na linha 1 do trecho Código 5, vemos a criação de uma instância da classe de dados mostrada na Figura 180. Nas linhas 6 a 8 é mostrado o exemplo de utilização de uma propriedade dessa classe. A propriedade esstá associada a um *view* da base de dados, neste caso para carregar numa caixa de seleção o nome das portas configuradas no sistema SACA. E na linha 12 é mostrado um exemplo de utilização dos métodos da classe, associados a um *stored procedure* da base de dados.

Código 5 – LINQ, Exemplo de utilização de uma classe de dados.

```
1  Dim dbPWNT As New DataClassesPWNTDataContext
2
3  . . .
4
5
6  For Each row In dbPWNT.aav_LogicalDevices
7      ComboBox_LogicalDevice.Items.Add(row.ALT_DESCRP)
8  Next
9
10 . . .
11
12 Resultado = dbPWNT.aaspCompany_Badge(ComboBox_Company.Text)
13
14
15 . . .
```

De notar que no exemplo apresentado no Código 5, apesar de estarmos a tratar de ligações à base de dados usando entidades relacionais: *views* e *stored procedures*, no ambiente

Visual Basic, todo o código é escrito com a sintaxe do paradigma de programação orientada por objetos, o que torna o desenvolvimento muito mais intuitivo e menos sujeito a erros.

Estado da comunicação das aplicações com a base de dados

O acesso da aplicação às bases de dados está em constante monitorização. Nas bases de dados foram criados *stored procedures* denominados *BaseDadosPing* que devolvem a data/hora do servidor SQL. Estes *stored procedures* são usados pelas aplicações para periodicamente testar a ligação à base de dados.

No ecrã principal da aplicação CRED está definido um objeto do tipo *Timer* que em intervalos de 1s, executa o método associado ao *BaseDadosPing* para testar a ligação. Se a ligação estiver operativa, na barra de estados da aplicação, é mostrada uma representação gráfica que comuta entre as duas imagens mostradas na Figura 181.

Figura 181 – Ligação da aplicação à base de dados operativa.

Se não existir comunicação entre a aplicação e base de dados, a aplicação entra em modo inoperativo e apresenta ao operador a imagem mostrada na Figura 182.

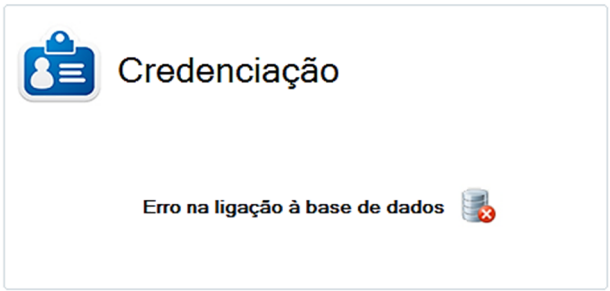


Figura 182 – Ligação da aplicação à base de dados inoperativa.

Apresentação de informação nos ecrãs

O acesso e a gestão da informação definida dos diagramas de caso de uso e apresentado no capítulo de projeto são efetuados em ecrãs que ficam disponíveis pelo uso dos itens da barra de menus. Normalmente quando um desses ecrãs é carregado e mostrado, apresenta duas opções: permite selecionar uma entidade já registada no sistema ou permite criar uma nova entidade.

A seleção de entidade já existente é feita através de caixas de seleção onde se introduz ou número de identificação ou o nome da entidade. A Figura 152 é um exemplo desta questão relativamente à entidade "Pessoas", a Figura 166 é o exemplo aplicado às empresas e a Figura 171 é o exemplo aplicado às viaturas. Quando o utilizador começa a digitar caracteres nestes objetos, é mostrado uma lista com itens que contem os caracteres que estão a ser introduzidos. Esta funcionalidade é implementada com objetos do tipo *ComboBox*, com a propriedade `AutoCompleteSource = ListItens`, e a propriedade `AutoCompleteMode = SuggestAppend`.

Para que os objetos C*omboBox* apresentem informação guardada na base de dados, no momento de carregamento do ecrã usa-se uma instância da classe de acesso a dados criada pelo LINQ, e o carregamento da lista usa um mecanismo similar ao descrito nas linhas 6 a 8 do trecho Código 5.

Quando o utilizador termina a inserção da identificação da entidade pretendida, é executada uma função que usa a identificação introduzida, para pedir à base de dados a informação necessária para preencher os campos do ecrã. O trecho Código 6, mostra a forma como se faz a apresentação dos resultados, nesse código realçam-se as seguintes questões:

- A instrução da linha 7 executa o método da classe que pede à base de dados a informação da viatura identificada pelo valor introduzido na `ComboBox_Id.` A informação obtida é armazenada na variável `Resultado` no formato de um *array* de valores.

- Nas linhas de código 10 a 21, faz-se o preenchimento dos campos do ecrã "Viaturas", com os valores recebidos da base de dados.

- Nas linhas de código 28 e 31 executam-se funções similares, para preenchimento das tabelas com a informação das outras páginas o ecrã.

- De notar que a variável `iExecução` existe para conter o código de saída do *stored procedure*. Após a execução do método da classe, esta variável fica com o valor dos códigos definidos no Anexo K . Esse valor será negativo caso se esteja a reportar um erro.

Código 6 – Exemplo de atualização de informação.

```
1   Private Sub AtualizaDadosViatura()
2       Dim iExecução As Integer
3       Dim Valores() As String
4
5       '-------------------------------------------------------------------------------------------
6       'Informação geral das viaturas
7       Dim Resultado = db.spViatura_Dados(ComboBox_Id.Text, iExecução)
8       If iExecução = 0 Then
9           For Each row In Resultado
10              TextBox_Matricula.Text = row.Matricula
11              TextBox_NumeroSerie.Text = row.NumeroSerie
12              TextBox_Distico.Text = row.Distico
13              TextBox_Combustivel.Text = row.Combustivel
14              TextBox_Tipo.Text = row.TipoVeiculo
15              TextBox_DataFabrico.Text = row.DataFabrico
16              TextBox_MarcaModelo.Text = row.MarcaModelo
17              TextBox_Entidade.Text = row.Nome
18              TextBox_Servico.Text = row.ServicoViatura
19              CheckBox_Permanente.Checked = row.Permanente
20              CheckBox_Ativo.Checked = row.Ativo
21              TextBox_Notas.Text = row.Notas
22          Next
23      Else
24           MsgBox("Na leitura de informação geral da viatura, a base de dados devolveu o erro: "
            & iExecução, vbOK + vbExclamation)
25      End If
26      '-------------------------------------------------------------------------------------------
27      'Informação sobre revalidação
28      AtualizaDataGridView_Revalidacao()
29      '-------------------------------------------------------------------------------------------
30      'Informação sobre anexos
31      AtualizaDataGridView_Anexos()
32      '-------------------------------------------------------------------------------------------
33      'Seleciona a Página inicial
34      TabControl_Geral.SelectTab(0)
35
36  End Sub
37
```

No trecho Código 7, é mostrado um exemplo de preenchimento de dados numa tabela. O exemplo apresenta a implementação da função `AtualizaDataGridView_Revalidacao` invocada no Código 6 que tem como objetivo preencher a tabela com a informação das revalidações da licença de uma viatura. O princípio de implementação é similar ao trecho de código anterior, mas neste caso os campos recebidos pelo método da classe são distribuídos pelas colunas da tabela, linhas 14 e 15.

Código 7 – Exemplo de atualização de informação em tabelas.

```
1   Private Sub AtualizaDataGridView_Revalidacao()
2       Dim iExecução As Integer
3       Dim Valores() As String
4
5       On Error Resume Next
6
7       DataGridView_Revalidacao().Rows.Clear()
8
9       'Pede informação à bd
10      Dim Resultado = db.spViatura_Revalidacao(ComboBox_Id.Text, iExecução)
11
12      If iExecução = 0 Then
13          For Each row In Resultado
14              Valores = {row.Id, row.RefViatura, row.DataRegisto, row.DataValidadeDistico, _
                           row.DataValidadeInspecao, _
                           row.DataValidadeSeguro, row.DataValidadeExtintor, row.Notas}
15
16              DataGridView_Revalidacao.Rows.Add(Valores)
17          Next
18      Else
19
20      End If
21             MsgBox("Na leitura de informação sobre revalidações, a base de dados devolveu o
                       erro: " & iExecução, vbOK + vbExclamation)
22      GroupBox_AdicionaRevalidacao.Visible = False
23  End Sub
```

Em todos os ecrãs das aplicações, os mecanismos de preenchimento de informação são idênticos aos explicados nestes exemplos.

Guardar informação na base de dados

A forma de implementação do envio de informação para a base de dados é exemplificada no trecho Código 8, que é parte da função que dá assistência ao evento de pressionar o botão esquerdo do rato no botão “Cria Pessoa”. Esta implementação tem os seguintes detalhes:

- O envio de dados para guardar começa com a validação da informação introduzida pelo operador, no exemplo apresentado a função `InformacaoNovaPessoaValida` faz essa tarefa. No caso da informação introduzida estar dentro dos parâmetros esperados procede-se ao armazenamento da informação, caso contrário o operador é informado do problema e a aplicação coloca o cursor de operação no objeto cuja informação não passou na validação.

- Nos campos que apenas admitem informação pré-definida nos dicionários, o operador escolhe a opção pretendida através de uma descrição textual. No entanto, a base de dados guarda a informação com referências às chaves primárias das tabelas. Por isso, existe a necessidade de execução das linhas de código 2 a 6 que tem por objetivo obter os valores das chaves primárias das opções efetuadas pelo operador.

- Na linha de código 11, executa-se o método da classe que corresponde ao *stored procedure* da interface da base de dados que cria um novo registo da entidade "Pessoas".

- Nas linhas de código 12 e 13 efetua-se o tratamento de qualquer erro que possa ter ocorrido na execução do *stored procedure* no servidor da base de dados.

Código 8 – Exemplo envio de informação para a base de dados.

```
    . . .
1           If InformacaoNovaPessoaValida() Then
2               db.spNacionalidade_Id(ComboBox_PessoaAlteraNacionalidade.Text, sNacionaliade)
3               db.spDocumentoId_Id(ComboBox_PessoaAlteraDocumentoId.Text, sDocId)
4               db.spEntidade_Id(ComboBox_PessoaAlteraEntidade.Text, sEntidade)
5               db.spFuncao_Id(ComboBox_PessoaAlteraFuncao.Text, sFuncao)
6               db.spServico_Id(ComboBox_PessoaAlteraServico.Text, sServico)
7
8               If CheckBox_PessoaAlteraObjetosProibidos.Checked Then iObjetosProibidos = 1
9
10              Dim Resultado = _
11              db.spPessoa_Cria(TextBox_PessoaAlteraId.Text, TextBox_PessoaAlteraNome.Text,
                                 DateTimePicker_PessoaAlteraDataNascimento.Text, sNacionaliade,
                                 TextBox_PessoaAlteraFiliacaoMaterna.Text,
                                 TextBox_PessoaAlteraFiliacaoPaterna.Text, sDocId,
                                 DateTimePicker_PessoaAlteraValidadeId.Text, sEntidade, sServico,
                                 sFuncao, iObjetosProibidos, TextBox_PessoaAlteraNotas.Text,
                                 My.Settings.UltimoUtilizador, iExecução)
12              If iExecução <> 0 Then
13                  MsgBox("Na guarda dados na criação da pessoa, a base de dados devolveu o
                            erro: " & iExecução, vbOK + vbExclamation)

14      . . .
```

Todas as operações de guarda de dados seguem mecanismos idênticos ao apresentado neste exemplo, cada um usando os respetivos métodos das respetivas classes criadas no LINQ.

Tratamento de ficheiros anexos

Os ficheiros anexos adicionados aos processos das entidades descritas, normalmente são digitalizações de documentos em formato de papel, como: documentos de identificação, títulos de propriedade, documento de seguradoras, listas de ferramentas, etc. Estas digitalizações são frequentemente feitas a cores e tipicamente apresentadas em formato *pdf*. Documentos com estas características tem dimensões da ordem dos megabyte e atingem por vezes algumas dezenas de megabyte.

Face à dimensão destes documentos, as boas práticas, [109], não recomendam que os ficheiros sejam guardados nas bases de dados mas sim em servidores de ficheiros. Para fazer face a esta questão, foi implementada na base de dados uma tabela com caminhos de diretórios para serem usados como local de armazenamento de ficheiros anexos, Figura 183. Com este mecanismo, as aplicações em cada situação usam a chave primária associada ao tipo de ficheiro que se pretende guardar e perguntam à base de dados o diretório onde o ficheiro vai ser guardado. Esta forma de implementação permite que se possa alterar o *file server* de uma forma simples, apenas alterando um campo da base de dados.

SQLQuery1.sql - ASC-SFMP\...\ad...))*

```
1
2  SELECT *  FROM [Credenciacao].[dbo].[tblCaminhoFicheiros]
```

Results | Messages

	IdLocal	Caminho	Descricao
1	AnexosCredenciacaoLostFound	D:\ISEP\2A - Tese\AppCred_Anexos\AnexosCredenciacaoLostFound\	Caminho para o diretorio de armazenamento de ficheiros relacionados com emissão d
2	AnexosCredenciacaoPontual	D:\ISEP\2A - Tese\AppCred_Anexos\AnexosCredenciacaoPontual\	Caminho para o diretorio de armazenamento de ficheiros relacionados com emissão d
3	AnexosEntidades	D:\ISEP\2A - Tese\AppCred_Anexos\AnexosEntidades\	Caminho para o diretorio de armazenamento de ficheiros relacionados com entidades
4	AnexosPessoas	D:\ISEP\2A - Tese\AppCred_Anexos\AnexosPessoas\	Caminho para o diretorio de armazenamento de ficheiros relacionados com pessoas
5	AnexosViaturas	D:\ISEP\2A - Tese\AppCred_Anexos\AnexosViaturas\	Caminho para o diretorio de armazenamento de ficheiros relacionados com viaturas
6	Lista Artigos Proibidos	D:\ISEP\2A - Tese\AppPort_Anexos\ObjetosProibidos\	Caminho para o diretorio de ficheiros de artigos proibidos no servidor das portarias
7	Temporarios	D:\ISEP\2A - Tese\AppCred_Anexos\Temporarios\	Caminho para o diretorio de ficheiros temporarios

Figura 183 – Tabela com registo de servidores de ficheiros para armazenamento de anexos.

No trecho Código 9 é mostrado parte de uma função que regista um novo anexo no sistema. Além da verificação da existência do ficheiro a guardar, efetuada pela função `InformacaoAnexoValido` , este código requere as seguintes explicações:

- Na linha de código 3, procura-se a chave primária do tipo de anexo que o operador selecionou.

- Na linha de código 4, faz-se o registo do anexo na base de dados, isto é, guarda-se a informação de que o processo da pessoa tem um novo anexo. Deste registo resulta uma chave primária que é guardada na variável `lPerfixo`.

- Na linha 12 constrói-se o nome do ficheiro que vai ser armazenado no *file system* da plataforma. O nome do ficheiro a ser armazenado é constituído pela chave primária de registo na base de dados contida na variável `lPerfixo`, concatenada com o caracter "-" e com o nome do ficheiro original. Por exemplo se o ficheiro original tiver o nome BISergioMartins.pdf, e o registo na base de dados tiver o id = 1000, o ficheiro guardado no *file system* tem o nome 1000-BISergioMartins.pdf. Desta forma garante-se cada anexo que se guarda no *file system*, tem um nome único, mesmo que o operador anexe o mesmo ficheiro mais de uma vez.

- O código apresentado na linha 13 faz a cópia do ficheiro original para o *file system*.

- Nas linhas de código 15 a 18 verifica-se se o armazenamento do ficheiro foi efetuado corretamente, caso contrário, apaga-se o registo na base de dados, linha18 e reporta-se o erro ao operador.

Código 9 – Exemplo de código de registo de um novo anexo.

```
1   . . .
2               If InformacaoAnexoValido() Then
3                   db.spTipoAnexo_Id(ComboBox_PessoaTipoAnexo.Text, iTipoAnexo)
4                   db.spPessoa_AnexoCria(ComboBox_PessoaId.Text, Label_PessoaAnexo.Text,
                                          iTipoAnexo, TextBox_PessoaAnexoNotas.Text,
                                          My.Settings.UltimoUtilizador, lPerfixo)
5                   If iExecucao < 0 Then
6                       MsgBox("Na guarda dados de anexo, a base de dados Credenciação devolveu o
                               erro: " & iExecucao, vbOK + vbExclamation)
7                   Else
8                       'copia o ficheiro para o servidor geral
9                       'Nome do ficheiro original
10                      sFicheiroOrigem = Label_PessoaAnexoCaminhoCompleto.Text &
```

```
                                    Label_PessoaAnexo.Text
11                      'nome do ficheiro de destino: ID_BaseDados-Nome_Ficheiro
12                      sFicheiroDestino = My.Settings.AnexosPessoas & lPerfixo & "-" &
                                    Label_PessoaAnexo.Text
13                      My.Computer.FileSystem.CopyFile(sFicheiroOrigem, sFicheiroDestino)
14                      'verifica a existencia do ficheiro no destino
15                      If Not (My.Computer.FileSystem.FileExists(sFicheiroDestino)) Then
16                          MsgBox("Erro na anexação do ficheiro, no servidor principal",
                                vbInformation + vbOKOnly)
17                          db.spPessoa_ErroAnexacaoApagaAnexo(iExecucao,
                                                    My.Settings.UltimoUtilizador,
                                                    iExecucao)
18                      End If
19  . . .
```

O preenchimento das listas de ficheiros anexos que são mostradas aos operadores nos ecrãs de determinada entidade é efetuado com base na informação contida na tabela correspondente que está na base de dados.

Exportação de informação para *Excel*

Para efetuar a exportação do conteúdo de objetos do tipo *DataGridView* - tabelas mostradas no ecrã, para *Excel*, desenvolveu-se uma função mostrada no trecho Código 10. Esta função requere explicação dos seguintes pontos:

- A função tem como parâmetro de entrada a referencia à tabela cujo conteúdo se pretende exportar, defina da variável `Tabela`.
- Nas linhas de código 7 a 10, criam-se variáveis para conter objetos do tipo: aplicação *Excel*, ficheiro de *Excel* e folha dentro do ficheiro de *Excel*. Estes objetos são iniciados nas linhas de código 17 a 23.
- No código das linhas 26 a 29, implementa-se uma estrutura repetitiva para copiar o cabeçalho da tabela de entrada para a primeira linha do ficheiro *Excel*. E formata-se a letra com o estilo negrito.
- Nas linhas 31 a 36, usam-se duas estruturas cíclicas encadeadas para fazer a cópia sequencial das células – linhas/colunas da tabela para o ficheiro.

- O código da linha 38, faz o ajuste da largura das colunas do ficheiro ao conteúdo das células.

Código 10 – Código para exportação de informação para Excel.

```
1   Public Sub Tabela2Excel(Tabela As DataGridView)
2           'Exporta o conteudo da Tabela para uma folha de Excel
3           On Error Resume Next
4
5           If Tabela.Rows.Count > 0 Then
6               'variaveis para conter a estrutura do ficheiro de excel
7               Dim wapp As Microsoft.Office.Interop.Excel.Application
8               Dim wbook As Microsoft.Office.Interop.Excel.Workbook
9               Dim wsheet As Microsoft.Office.Interop.Excel.Worksheet
10              Dim chartRange As Microsoft.Office.Interop.Excel.Range
11              'variaveis para conter referencias a localizações
12              Dim iX As Integer
13              Dim iY As Integer
14              Dim iC As Integer
15
16              'abertura do excel
17              wapp = New Microsoft.Office.Interop.Excel.Application
18
19              wapp.Visible = True
20              'criação de workbook
21              wbook = wapp.Workbooks.Add()
22              'criação de folha
23              wsheet = wbook.ActiveSheet
24
25              'Preenchimento do cabeçalho
26              For iC = 0 To Tabela.Columns.Count - 1
27                  wsheet.Cells(1, iC + 1).Value = Tabela.Columns(iC).HeaderText
28                  wsheet.Cells(1, iC + 1).font.bold = True
29              Next
30              'Preenchimento do conteudo da tabela
31              For iX = 0 To Tabela.Rows.Count - 1
32                  For iY = 0 To Tabela.Columns.Count - 1
33                      wsheet.Cells(iX + 2, iY + 1).value = Tabela(iY, iX).Value.ToString
34                  Next
35
36              Next
37              'ajuste de colunas
38              wsheet.Columns.AutoFit()
39          End If
40  GestorErros:
41          If (Err.Number <> 0) Then
42              MsgBox("Erro na Função: Tabela2Excel" & vbCrLf & Err.Number & " - " &
                        Err.Description, vbCritical + vbOKOnly)
43          End If
44      End Sub
```

Tratamento da fotografia no ecrã "Pessoas"

A página "Dados" do ecrã "Pessoas", contem um objeto do tipo *PictureBox*, para conter a foto da pessoa. Esta é a foto que posteriormente vai ser usada na impressão no cartão de acessos.

Os cartões de acesso usados no Aeroporto Francisco Sá Carneiro são do tipo ID-1, ver Anexo C . Nesses cartões, usa-se uma foto com dimensões de 105x140 *pixeis*. Um ficheiro com estas dimensões guardado no formato *bmp* de 24-bit, que é um formato de imagem a cores, sem compressões nem otimizações ocupa 44KByte. As boas regras de armazenamento de informação, [109], dizem que para blocos de informação com dimensão inferior a 250KBytes, é mais vantajoso guarda-los em bases de dados do que em *file systems*. Por isso definiu-se que a foto das pessoas vai ser armazenada na tabela `tblPessoa`, num campo do tipo *varbinary*.

Para obter da base de dados a foto de uma pessoa, desenvolveu-se o *stored procedure* `spPessoa-DevolveFoto` cuja instrução principal é a mostrada no Código 11 e para guardar a foto na base de dados, desenvolveu-se o *stored procedure* `spPessoa-GuardaFoto` cuja instrução principal é a mostrada no Código 12.

Código 11 – Instrução para obter a foto de uma pessoa.

```
1   SELECT [Foto] FROM [Credenciacao].[dbo].[tblPessoa] WHERE ([id] = @idPessoa)
```

Código 12 – Instrução para atualizar a foto de uma pessoa.

```
1   UPDATE [Credenciacao].[dbo].[tblPessoa] SET ([Foto] = @Foto) WHERE ([id] = @idPessoa)
```

Conforme apresentado na descrição funcional da aplicação, existem duas formas de atribuir uma imagem para constituição da foto do processo da pessoa. A primeira forma é usar o botão "Foto de ficheiro" do ecrã apresentado na Figura 158. Com esta opção abre-se uma janela para selecionar o ficheiro que contem a foto e depois usa-se o botão "Guardar"

para guardar a imagem na base de dados. A conversão da informação da imagem contida no ficheiro para um *array* de *bytes*, para ser guardado na base de dados é efetuada com a instrução mostrada no Código 13.

Código 13 – Conversão de ficheiro para *array* de *bytes*.

```
1   Dim ImageData As Byte() = IO.File.ReadAllBytes(sCaminhoEFicheiroDaFoto)
```

O outro mecanismo de obter a foto é fazer a aquisição de uma imagem usando uma camara fotográfica ligada ao computador. Esta opção está acessível através do botão "Aquisição de foto" que apresenta a janela mostrada na Figura 184, para seleção da máquina fotográfica pretendida.

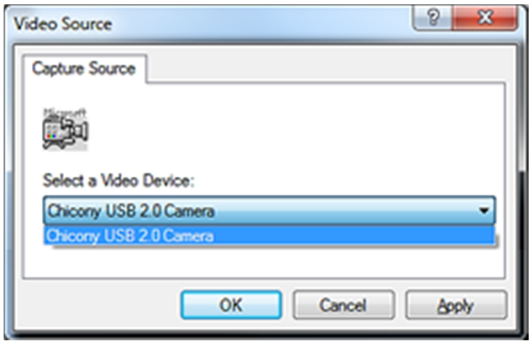


Figura 184 – Janela de seleção da máquina fotográfica.

Após essa seleção, o pressionar o botão "Inicializar Visualização", apresenta o ecrã mostrado na Figura 159. Neste ecrã a imagem capturada pela máquina fotográfica é direcionada para um objeto do tipo *PictureBox*.

O trecho Código 14 mostra as instruções de acesso à imagem da camara fotográfica. Na linha 2, cria-se um manipulador de dispositivo associado à camara escolhida na janela da Figura 184. Nas linhas 7 a 15, faz-se a adaptação da imagem capturada ao objeto *PictureBox* que está no ecrã para visualização pelo operador.

Código 14 – Código de apresentação no ecrã de imagem capturada pela máquina fotográfica.

```
. . .
' Abre a janela de visualização no picturebox
hHwnd = capCreateCaptureWindowA(iDevice, WS_VISIBLE Or WS_CHILD, 0, 0, 640, _
    480, picCaptura.Handle.ToInt32, 0)
' Conecta com o drive selecionado
If SendMessage(hHwnd, WM_CAP_DRIVER_CONNECT, iDevice, 0) Then
    'Define a escala de previsão
    SendMessage(hHwnd, WM_CAP_SET_SCALE, True, 0)
    'Define a taxa de visualização em milisegundos
    SendMessage(hHwnd, WM_CAP_SET_PREVIEWRATE, 66, 0)
    'Iniciar a visualização da imagem a partir da camara
    SendMessage(hHwnd, WM_CAP_SET_PREVIEW, True, 0)
    ' Redimensiona a janela para se ajustar no picturebox
    SetWindowPos(hHwnd, HWND_BOTTOM, 0, 0, picCaptura.Width, picCaptura.Height, SWP_NOMOVE Or
                 SWP_NOZORDER)
        Button_AquisicaoFotoTerminar.Enabled = True
        Button_AquisicaoFotoIniciar.Enabled = False
    Else
        ' Erro de conexão fecha a janela de dispostivos
         DestroyWindow(hHwnd)
         Button_AquisicaoFotoTerminar.Enabled = False
End If
. . .
```

Quando se pressiona o botão "Termina Aquisição" desliga-se a camara fotográfica, Código 15, e a *PictureBox* permanece com a última imagem capturada.

Código 15 – Instrução para desligar a camara fotográfica.

```
SendMessage(hHwnd, WM_CAP_DRIVER_DISCONNECT, iDevice, 0)
```

O passo que seguinte na aquisição da foto é: dentro da imagem capturada pela camara, escolher a parte que vai constituir a foto a guardar. Este processo é feito usando o evento *MouseMove* da *PictureBox*. Quando o utilizador pressiona o botão esquerdo do rato na imagem capturada, é mostrado um retângulo, com as dimensões finais da foto. Esse retângulo, de seleção de imagem, desloca-se com o ponteiro do rato, Código 16. Para selecionar a área de interesse para a foto o utilizador solta o botão rato e a parte da imagem selecionada pelo retângulo é copiada para a *PictureBox* classificada com a etiqueta "Nova foto".

Código 16 – Função de posicionamento de um retângulo para seleção da foto.

```
Private Sub CentraRetanguloDeFoto(iX As Integer, iY As Integer)

        Dim iMetadaLargura As Integer = FotoLagura \ 2
        Dim iMetadeAltura As Integer = FotoAltura \ 2

        On Error Resume Next
```

```
 7
 8          With LineShapeV1
 9              .X1 = iX - iMetadaLargura
10              .X2 = iX - iMetadaLargura
11              .Y1 = iY - iMetadeAltura
12              .Y2 = iY + iMetadeAltura
13          End With
14          With LineShapeV2
15              .X1 = iX + iMetadaLargura
16              .X2 = iX + iMetadaLargura
17              .Y1 = iY - iMetadeAltura
18              .Y2 = iY + iMetadeAltura
19          End With
20          With LineShapeH1
21              .X1 = iX - iMetadaLargura
22              .X2 = iX + iMetadaLargura
23              .Y1 = iY - iMetadeAltura
24              .Y2 = iY - iMetadeAltura
25          End With
26          With LineShapeH2
27              .X1 = iX - iMetadaLargura
28              .X2 = iX + iMetadaLargura
29              .Y1 = iY + iMetadeAltura
30              .Y2 = iY + iMetadeAltura
31          End With
32
33          MostraRetanguloDeFoto(True)
34
35      End Sub
```

O trecho apresentado no Código 17, mostra o uso do método *DrawImage* do objeto do tipo *FormImage*, para copiar a área da imagem definida pelo retângulo para a *PictureBox* que vai conter a foto a ser guarda.

Código 17 – Código de seleção a área da foto.

```
. . .
1   ' Associa o objeto gráfico ao bitmap
2   Using gr As Graphics = Graphics.FromImage(ImagemTemporaria)
3       ' Define as areas das imagens.
4       iTopo = LineShapeV1.Y1
5       iEsquerda = LineShapeV1.X1
6       Dim RetanguloInicial As New Rectangle(iEsquerda, iTopo, FotoLagura, FotoAltura)
7       Dim RetanguloFinal As New Rectangle(0, 0, FotoLagura, FotoAltura)
8       ' Copia a imagem selecionada.
9       gr.DrawImage(PictureBox_PessoaFotoNova.Image, RetanguloFinal, RetanguloInicial,
                     GraphicsUnit.Pixel)
10  End Using
11
12  ' Apresenta o resultado.
13  PictureBox_PessoaFotoFinal.Image = ImagemTemporaria
14 . . .
```

No trecho Código 18, é mostrado como se guarda na base de dados a imagem capturada pela camara fotográfica:

- Nas linhas 1 a 5, guarda-se a imagem da *PictureBox* que contêm a nova foto, num ficheiro temporário.
- Nas linhas 13 a 15 cria-se o *array* de *bytes*, desse ficheiro temporário e guarda-se a informação na base de dados.

Código 18 – Código para guardar na base de dados a imagem capturada pela camara.

```
. . .
1   imgImage = PictureBox_PessoaFotoFinal.Image
2   sFicheiroTemporario = My.Settings.FicheirosTemporarios & "ImagemTemporaria.png"
3
4   On Error Resume Next
5   imgImage.Save(sFicheiroTemporario, Imaging.ImageFormat.Png)
6
7   If (Err.Number = 0) Or (Err.Number = 5) Then
8       'tratamento do erro de leitura e escrita no mesmo ficheiro
9       If Err.Description = "A generic error occurred in GDI+." Then Err.Clear()
10
11      On Error GoTo GestorErros
12
13      Dim ImageData As Byte() = IO.File.ReadAllBytes(sFicheiroTemporario)
14
15      db.spPessoa_GuardaFoto(ComboBox_PessoaId.Text, ImageData)
16  . . .
```

Impressão do cartão de acessos

Na página "Cartão de acesso" do ecrã "Pessoas", depois de registar um cartão é possível proceder à impressão da face do cartão de acesso. Esta funcionalidade é implementada num objeto do tipo *Form*, que tem por base uma imagem modelo, com as dimensões físicas do cartão e apresenta os elementos pictóricos estáticos como: o logotipo da empresa, o contorno amarelo e os retângulos onde ser apresentam as permissões de acesso codificadas por letras. A Figura 185 mostra a imagem usada como fundo do cartão e o ecrã com um exemplo de um cartão.

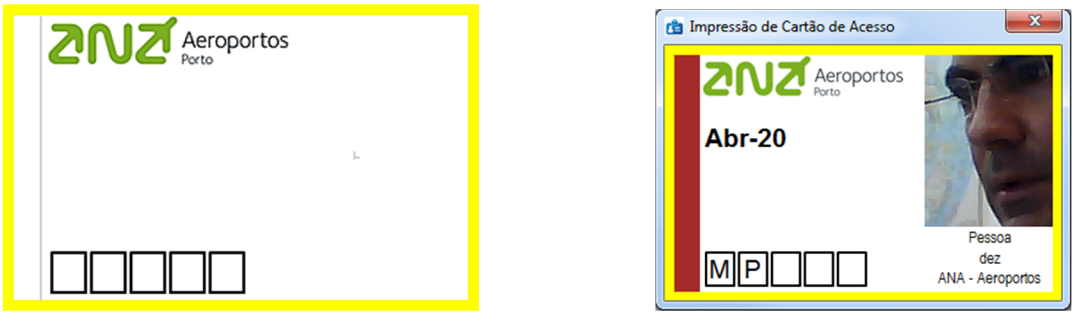


Figura 185 – Imagem do fundo da face principal do cartão de acesso permanente e pontual.

Além da imagem de fundo, o ecrã que mostra o cartão é constituído por objetos do tipo *Label* para conter as informações textuais do cartão como: data de validade, nome da pessoa, códigos de acesso, etc. E por dois objetos do tipo *PictureBox*, um para conter a foto da pessoa e outro para conter a barra de cor para atribuição de acessos codificados por cor.

Na classe do ecrã de impressão de cartões, foi desenvolvido um método para efetuar a entrada de valores e fazer o preenchimento da informação do cartão. Quando se está no ecrã "Pessoas" e se dá instruções para impressão do cartão, esse método do *form* `frmImprimirCartao` é invocado para transferir os dados dos objetos do ecrã "Pessoas" para o ecrã de impressão de cartão.

Pressionando o botão esquerdo do rato sobre o ecrã do cartão de acessos, ou pressionando a tecla "Menu", é apresentado o menu mostrado na Figura 162. O trecho Código 19 mostra a função que é executada quando se usa o item "Imprimir" do menu, nesta função é importante realçar as seguintes questões:

- Como o cartão está na iminência de ser impresso, é necessário cria-lo e a ativa-lo no sistema SACA para que possa abrir as portas que lhe estão atribuídas pelos acessos definidos. Esta operação é efetuada pela função `EnviaCartaoParaProWatch`, linha 6. A função `EnviaCartaoParaProWatch` usa um método da classe de dados que faz a interface com a base de dados PWNT e executa a criação do cartão no *Pro-Watch* e atribui-lhe a *company* definida. No fim da execução, a função devolve um valor booleano que representa o sucesso da criação do cartão no PWNT. Se não for possível criar o cartão no SACA o cartão não é enviado para a impressora e o operador é informado do problema.

- O código das linhas 9 e 10 é o código que efetua a impressão do cartão. Este código usa o objeto de sistema *Printing* para enviar para a impressora definida a imagem do ecrã do cartão.

- Após o cartão ser enviado para a impressora, passa a ser considerado "Emitido". O código da linha 12, atualiza a base de dados *Credenciação* com esta informação e a instrução da linha 14 atualiza a informação que está a ser mostrada ao operador.

Código 19 – Função de impressão de cartão de acesso.

```
1       Private Sub ImprimirToolStripMenuItem_Click(sender As System.Object, e As
                                                    System.EventArgs) Handles
                                                    ImprimirToolStripMenuItem.Click
2           Dim lResultado As Long
3
4           On Error Resume Next
5
6           If EnviaCartaoParaProWatch() Then
7
8               Me.ContextMenuStrip_ImpressaoCartoes.Visible = False
9               Me.PrintForm_Cartoes.PrintAction = Printing.PrintAction.PrintToPrinter
10              Me.PrintForm_Cartoes.Print(Me,
                        PowerPacks.Printing.PrintForm.PrintOption.CompatibleModeClientAreaOnly)
11
12              db.spPessoa_CartaoEmitido(CLng(Label_NumeroDoCartao.Text),
                                          CLng(Label_RefPessoa.Text),
                                          My.Settings.UltimoUtilizador, lResultado)
13
14              frmPessoas.AtualizaDataGridView_Cartoes()
15
16              Me.Close()
17          Else
18              MsgBox("O cartão não foi guardado no Pro-Watch!" & vbCrLf & vbCrLf & vbCrLf &
                       "Por isso não foi impresso", vbOKOnly + vbCritical)
19          End If
20
21      End Sub
22
```

O trecho Código 20 mostra a função que é executada quando se usa o item "Visualizar" do menu apresentado quando se pressiona com o botão direito sobre o cartão de acessos. Nesta função também se usa objeto de sistema *Printing* mas neste caso com a opção `PrintToPreview` que força a apresentação de uma nova janela com a previsualização do cartão impresso.

Código 20 – Função de previsualização do cartão de acesso.

```
1       Private Sub VisualizarToolStripMenuItem_Click(sender As System.Object, e As
                                                      System.EventArgs) Handles
                                                      VisualizarToolStripMenuItem.Click
2           On Error Resume Next
```

```
3           Me.ContextMenuStrip_ImpressaoCartoes.Visible = False
4           Me.PrintForm_Cartoes.PrintAction = Printing.PrintAction.PrintToPreview
5           Me.PrintForm_Cartoes.Print(Me,
               PowerPacks.Printing.PrintForm.PrintOption.CompatibleModeClientAreaOnly)
6
7       End Sub
```

A seleção da impressora para a qual o cartão vai ser impresso segue os padrões usados nas aplicações de ambiente *Windows*, ao ativar a opção de menu "Escolha de impressora" é executada a função mostrada no trecho Código 21 em que a linha 3 apresenta a janela mostrada na Figura 186. Esta janela lista as impressoras instaladas no sistema e permite ao utilizador selecionar a impressora pretendida. A seleção de impressora efetuada pelo utilizador fica ativa na aplicação CRED devido à execução do código da linha 4.

Código 21 – Função de seleção de impressora.

```
1       Private Sub EscolhaDeImpressoraToolStripMenuItem_Click(sender As System.Object, e As
                                                    System.EventArgs) Handles
                                                    EscolhaDeImpressoraToolStripMenuItem.Click
2           On Error Resume Next
3           PrintDialog_ImprimirCartoes.ShowDialog()
4           Me.PrintForm_Cartoes.PrinterSettings =
                                            Me.PrintDialog_ImprimirCartoes.PrinterSettings
5       End Sub
```

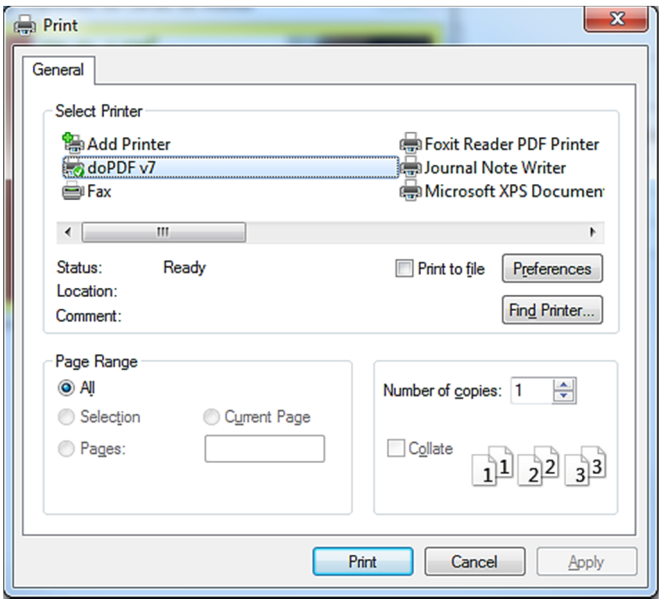


Figura 186 – Janela de seleção de impressora.

A opção do menu “Re-Impressão”, usada para reutilização do cartão de acessos, esconde todos os objetos que constituem a face visual do cartão, com exceção da linha que risca a data de validade atual e da nova data de validade, imprimido apenas uma das imagens mostradas na Figura 163, dependendo se é a primeira ou a segunda reimpressão.

Registo permanente de dados locais

Nas aplicações desenvolvidas para a plataforma de credenciação, há necessidade de manter informação entre execuções da mesma aplicação. Por exemplo para que quando se apresenta o ecrã de *login* se possa preencher o campo com a identificação da pessoa que usou aplicação da última vez, é necessário guardar essa informação de uma execução para ser apresentada na execução seguinte. Outro exemplo, quando se deteta um problema de comunicações com a base de dados, pretende-se guardar na base de dados o registo desse evento, mas como não há ligação à base de dados é necessário guardar essa informação noutro local para que logo que a comunicação seja restabelecida se fazer o registo do evento na tabela apropriada.

O *Visual Basic* possui um recurso que permite guardar dados permanentemente entre instâncias de execução das aplicações, este recurso chama-se *Settings* do projeto, Figura 187.

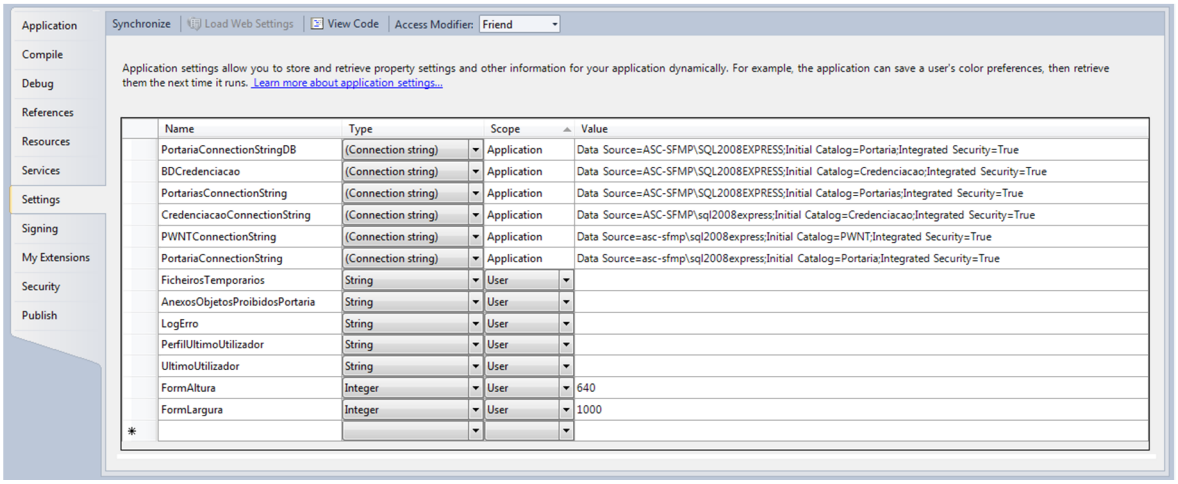


Figura 187 – Registo de dados permanentes na aplicação.

No fundo os *settings* são declarações de variáveis que são vistas em todo o projeto da aplicação e cujo valor é permanente entre execuções da aplicação. O trecho Código 22, linha 3 mostra um exemplo da forma de leitura do valor de variável que contém a identificação do último utilizador e na linha 7 mostra-se a atualização do valor dessa variável. Na realidade estas variáveis são usadas como qualquer outra variável, no entanto tem a característica do seu valor não ser volátil entre execuções da aplicação.

Código 22 – Uso de variáveis *settings* do projeto da aplicação.

```
1  . . .
2
3  TextBox_User.Text = My.Settings.UltimoUtilizador
4  . . .
5  My.Settings.UltimoUtilizador = TextBox_User.Text
6  . . .
7
```

4.2.3. IMPLEMENTAÇÃO DA APLICAÇÃO PORT

O módulo PORT é a aplicação de *software*, desenvolvida para ser instalada nas portarias que constituem uma fronteira entre zonas de diferentes níveis de segurança. Nessas portarias podem passar pessoas a pé ou a conduzir viaturas e o rastreio é efetuado por um elemento humano. O módulo PORT existe para auxiliar o vigilante na determinação das permissões de passagem e de condução de viaturas.

Esta secção faz a apresentação das funcionalidades da aplicação e a descrição dos detalhes de implementação.

4.2.3.1. DESCRIÇÃO FUNCIONAL DA APLICAÇÃO PORT

Á semelhança da aplicação CRED, o uso da aplicação PORT, é restringida por palavra-chave, Figura 188.

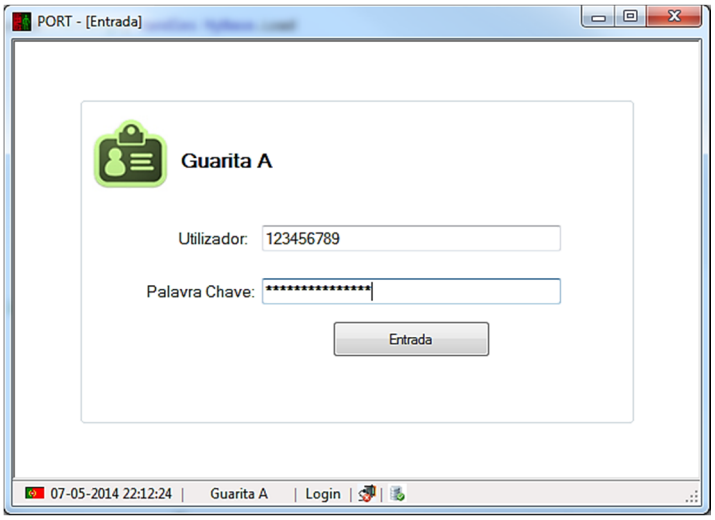


Figura 188 – Aplicação PORT, *login*.

Tem acesso a esta aplicação pessoas com o perfil de "Administrador" e com o perfil de "Vigilante". Fazendo o registo com perfil de "Administrador" é apresentado o ecrã mostrado na Figura 189. Na barra de estado, é mostrada a informação data/hora do sistema, o nome da portaria em que a aplicação está configurada, o nome do operador que está a

usar o sistema e duas imagens que representam o estado das ligações da aplicação ao leitor de cartões e à base de dados da portaria.

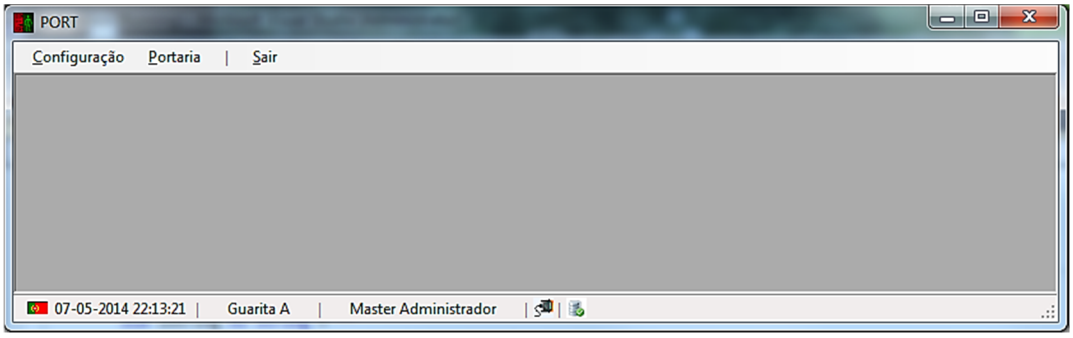


Figura 189 – Aplicação PORT, ecrã principal.

Na barra de menu para o perfil "Administrador", estão disponíveis duas funcionalidades, uma de configuração do sistema e outra de funcionamento operacional da portaria tal como se fosse um vigilante.

Ecrã de Configuração

No ecrã configuração o Administrador define os detalhes de funcionamento da portaria sob duas perspetivas: configurando a porta serie que faz a comunicação com o leitor de cartões de acesso, Figura 190. E configurando portaria em si, isto é, indicanso à aplicação em que portaria está a ser executada, Figura 191.

Figura 190 – Aplicação PORT, configuração da porta serie do leitor de cartões.

De notar que as informações mostradas na página de configuração da portaria, ou seja, o nome da portaria e os acessos que lhe estão associados, são definidos na aplicação CRED e neste ecrã estão diponiveis apenas para leitura. Na aplicação PORT, o Administrador apenas indica que essa instância específica da aplicação PORT, é relativa à portaria selecionada.

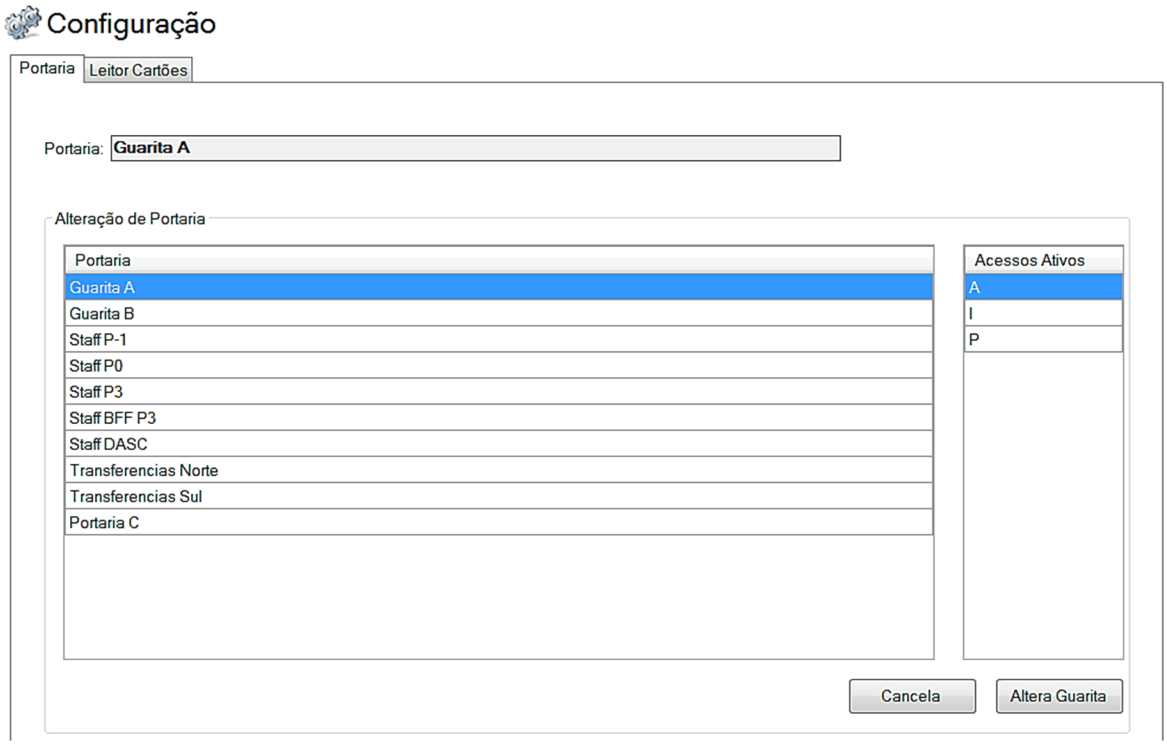


Figura 191 – Aplicação PORT, configuração da portaria.

Ecrã de Vigia

Quando a pessoa que se identifica no ecrã de entrada da aplicação, tem o perfil de "Vigilante", apenas é mostrado o ecrã de vigia de portaria, Figura 192, este, é também o ecrã que é apresentado ao administrador quando pressiona o menu "Portaria".

O ecrã "Portaria", em estado de operação, apresenta uma foto de um leitor de cartões, com uma imagem animada, para indicar ao operador que a aplicação está em execução. Nesta situação, a aplicação PORT está a aguardar que seja apresentado um cartão de acessos no respetivo leitor.

Figura 192 – Aplicação PORT, ecrã de vigia.

Quando é apresentado um cartão de acesso no leitor, a aplicação recebe o número do cartão e pergunta à base de dados *Portarias* se a pessoa identificada por esse cartão tem acesso de passagem nessa portaria. A resposta à pergunta é mostrada ao vigilante, num ecrã composto por três partes, Figura 193. Na primeira parte é apresentada informação relativa à pessoa e ao cartão que foi lido, nomeadamente: número do cartão, validade, nome, empresa, foto, etc. Se a pessoa tiver permissões de passagem na portaria, com artigos perigosos, como por exemplo ferramentas, no campo "Artigos", é mostrado um botão com a etiqueta "Sim", Figura 193. Ao usar esse o botão é apresentado ao vigilante o conteúdo do ficheiro anexo ao processo da pessoa (introduzido na plataforma pela aplicação CRED) e classificado com o tipo "Lista de artigos proibidos". Se a pessoa não tiver este tipo de permissões no local do botão é mostrada uma etiqueta com a indicação "Não".

Figura 193 – Aplicação PORT, ecrã análise de cartão de acesso.

O segundo bloco de dados do ecrã de análise do cartão de acesso apresenta um pictograma e um texto relativo à permissão de passagem da pessoa pela portaria. O texto pode ser "Acesso Permitido" ou "Acesso Negado". No caso de o acesso ser negado é também mostrada uma explicação para o facto, como mostrado na Figura 194.

O terceiro bloco de dados é relativo às permissões de condução de viaturas. Neste bloco é mostrado um pictograma que representa a permissão de passagem pela portaria a conduzir um veículo. Nesta área também é mostrada a indicação se pode conduzir veículos especiais e se pode conduzir em condições de LVO.

De notar, no exemplo da Figura 194, que apesar de a pessoa ter permissões de condução, como o cartão está fora da validade o pictograma de permissão de condução é vermelho.

Figura 194 – Aplicação PORT, Acesso Negado.

No caso de ser apresentado um cartão que não esteja registado na base de dados, a aplicação PORT, não permite o acesso e dá indicações ao vigilante para apreensão do cartão, Figura 195.

Em qualquer dos casos, sempre que um cartão é apresentado no leitor e o ecrã de informação é mostrado, o Vigilante tem que fechar explicitamente esse ecrã, pressionando o botão "Visto". Mesmo que outros cartões sejam passados no leitor, a informação do primeiro cartão não saí do monitor sem a operação do Vigilante. Nesses casos o sistema regista na base de dados que os cartões foram apresentados ao leitor, mas que a análise não foi mostrada ao vigilante, e por isso, vão ter de ser passados outra vez.

Figura 195 – Aplicação PORT, Cartão desconhecido.

Operacionalidade do sistema

As ligações à base de dados e ao conversor *Weigand/RS232*, estão sob constante monitorização pela aplicação PORT, qualquer falha de comunicação coloca a aplicação fora de serviço, com as indicações mostradas na Figura 196.

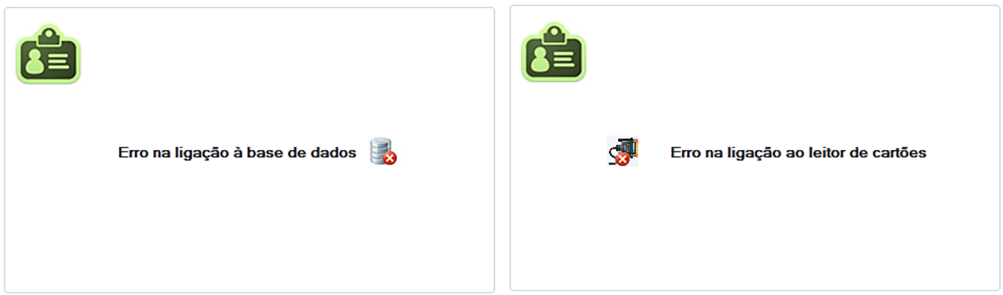


Figura 196 – Aplicação PORT, erros de ligação.

O leitor de cartões tem a linha de *tamper* ligada diretamente ao conversor *Weigand/RS232*, qualquer alteração do estado da instalação do leitor é reportada pelo conversor à aplicação PORT, que por sua vez, coloca-se fora se serviço mostrando o ecrã apresentado na Figura 197.

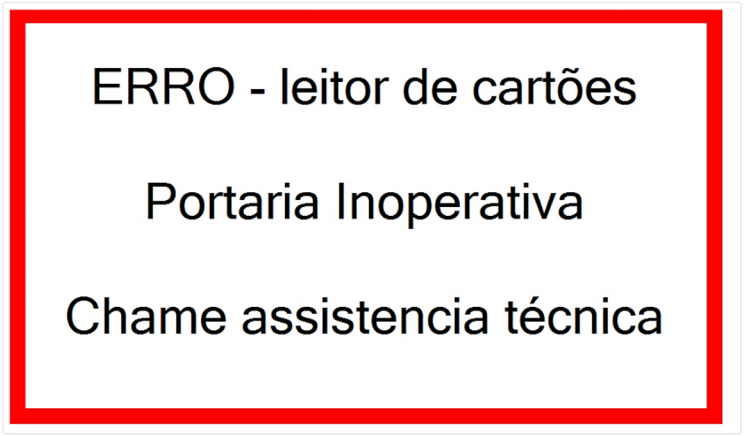
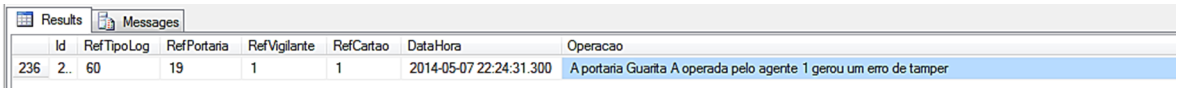


Figura 197 – Aplicação PORT, alteração da instalação do leitor de cartões.

Qualquer erro de funcionamento da aplicação é guardado, logo que possível, na base de dados, mesmo que o erro seja de ligação à base de dados, nestes casos logo que a comunicação é restabelecida o registo de erro é efetuado. A Figura 198, mostra um exemplo do registo de um problema na instalação do leitor de cartões da Guarita A.


Results | Messages

	Id	RefTipoLog	RefPortaria	RefVigilante	RefCartao	DataHora	Operacao
236	2..	60	19	1	1	2014-05-07 22:24:31.300	A portaria Guarita A operada pelo agente 1 gerou um erro de tamper

Figura 198 – Aplicação PORT, registo de erro de funcionamento.

4.2.3.2. DESCRIÇÃO DA IMPLEMENTAÇÃO DA APLICAÇÃO PORT

Na descrição funcional da aplicação PORT, explicou-se que a aplicação tem o único propósito de aguardar que um cartão de acesso seja apresentado ao leitor de cartões e mostrar ao vigilante se a pessoa tem permissões de passagem na portaria a pé ou a conduzir um veículo.

Quando se entra na aplicação com um utilizador de perfil "Administrador", além do ecrã operacional da portaria também é possível efetuar configurações sobre a aplicação que está a ser executada. Esta secção apresenta os detalhes técnicos de implementação que são particulares desta aplicação.

Configuração da porta serie

No diagrama apresentado na Figura 142 verifica-se que o leitor de cartões está ligado ao computador da portaria através de uma porta série. O ecrã mostrado na Figura 190 apresenta a forma de como essa porta é selecionada. Existe uma caixa de escolha onde estão listadas todas as portas série desse computador, usando este recurso o Administrador seleciona porta COM onde o leitor de cartões está ligado e pressiona o botão "Configurar" para que seja essa porta a considerada. No trecho Código 23, apresenta-se a função que carrega o objeto do tipo *ComboBox* com o nome das portas série do sistema para que o utilizador possa selecionar a porta onde o leitor está ligado.

Código 23 – Função de carregamento das portas série.

```
1       Private Sub GetSerialPortNames()
2           ' Carrega portas COM
3           For Each sp As String In My.Computer.Ports.SerialPortNames
4               cmbPortas.Items.Add(sp)
5           Next
6           cmbPortas.Text = cmbPortas.Items(0)
7       End Sub
```

Quando o Administrador seleciona a porta série a ser usada, esse valor é guardado na variável `My.Settings.PortaSerie`. Esta, é umas das variáveis de projeto que mantem o valor entre execuções até que uma instrução de código a altere explicitamente.

Para usar as funcionalidades relativas à porta série selecionada, no objeto *MDIForm* da aplicação PORT, foi incluído um objeto do tipo *System.IO.Ports.SerialPort*. E foi implementada a função `ConfiguraPortaSerie` apresentada no Código 24, para fazer a configuração dessa porta. Sempre que a identificação da porta usada é alterada, ou sempre que a aplicação é iniciada esta função é executada. No código dessa função há que realçar os seguintes pontos:

- Na linha 11, faz a configuração da porta série. A variável sString contem a identificação da porta definida pelo Administrador e guardada na variável `My.Settings.PortaSerie`, os outros parâmetros de configuração: o *baud rate* = 9600bps, a paridade = nenhuma, o número de *bits* de dados = 8 e o número de *Stop bits* = 1, que são imposições do conversor *Wiegand*/RS-232 são configurados no código como valores estáticos.

- Logo que a porta é configurada coloca-se no estado operativo, linha13, para poder começar a receber informação.

Código 24 – Função de configuração da porta série.

```
 1      Public Function ConfiguraPortaSerie() As Integer
 2
 3          Dim sString As String
 4          On Error GoTo GestorErros
 5          ConfiguraPortaSerie = 0
 6          sString = My.Settings.PortaSerie
 7
 8          If sString <> "" Then
 9              With frmPrincipal.SerialPort_Cartoes
10                  If .IsOpen Then .Close()
11                  frmPrincipal.SerialPort_Cartoes = My.Computer.Ports.OpenSerialPort(sString,
                                        9600, IO.Ports.Parity.None, 8, IO.Ports.StopBits.One)
12              End With
13              frmAgentePortaria.Timer_PortaSerie.Enabled = True
14          Else
15              MsgBox("ERRO!, porta serie desconhecida" & vbCrLf & vbCrLf & "Leitor de cartões
                       inoperativo", vbCritical + vbOKOnly)
16          End If
17
18  . . .
19
```

Configuração da portaria

Na Figura 191 apresenta-se o ecrã de configuração da portaria, nesse ecrã, na lista de portarias, o operador seleciona a portaria de interesse e pressionando o botão "Altera", atribuí a seleção efetuada à aplicação que está a ser executada. A tabela com a lista de portarias é carregada usando a função apresentada no Código 25, que tem as seguintes particularidades:

- A informação do preenchimento da tabela está na propriedade da classe de dados que mapeia o *view* `vPortarias`, linha 10.
- A tabela tem duas colunas, uma para conter o nome da portaria e outra para conter o valor da chave-primária da portaria. Na tabela apenas a coluna do nome é visível para o utilizador. Quando é feita a seleção da portaria é o valor da chave primária, contido na coluna não visível, que é guardado na variável permanente `My.Settings.IdPortaria`, Código 26.

Código 25 – Função de carregamento de lista de portarias.

```
1      Private Sub AtualizaListBoxPortarias()
2          Dim Valores() As String
3          On Error Resume Next
4
5          With DataGridView_ListaGuarita
6              'limpa a lista
7              .Rows.Clear()
8              DataGridView_AcessoPortaria.Rows.Clear()
9              'Preenche a lista
10             For Each row In dbPortarias.vPortarias
11                 Valores = {row.Id, row.Nome}
12                 .Rows.Add(Valores)
13             Next
14             .Refresh()
15         End With
16     End Sub
```

Código 26 – Código para guardar o identificador da portaria.

```
1  . . .
2
3  My.Settings.IdPortaria = DataGridView_ListaGuarita.SelectedCells(0).Value
4
5  . . .
```

Operação da portaria

Em funcionamento normal a portaria apresenta o ecrã mostrado na Figura 192, neste ecrã de vigia é mostrada uma fotografia de um leitor de cartões, sobre a qual está um objeto *PictureBox* carregado com uma imagem do tipo *Gif* animado. A *PictureBox* quando está no estado visível mostra uma sequência de imagens que dá a ilusão de movimento. Esta funcionalidade foi implementada para o vigilante verificar facilmente que a aplicação está em execução.

No ecrã de vigia foi configurado um objeto do tipo *Timer*, que periodicamente verifica se existem dados no *buffer* da porta serie onde está ligado o leitor de cartões. Conforme foi apresentado na secção 3.6.1.2, a aplicação PORT pode receber do conversor tramas do tipo 01_H ou do tipo 02_H, a primeira com dimensão de vinte *bytes* e a segunda com seis *bytes*. Como as comunicações são efetuadas à velocidade de 9600 bit/s, a maior trama demora 8x20/9600 ≈ 17mS a ser transmitida do conversor para o computador. Considerando este tempo definiu-se que a periodicidade para a aplicação verificar se tem informação do leitor é de 350mS, tempo cerca de vinte vezes superior ao tempo de transmissão. Este tempo é suficientemente grande para que a probabilidade ler só parte da trama seja baixa, mas ainda assim, é um tempo suficientemente pequeno para dar a ilusão ao vigilante de que tem uma resposta instantânea à apresentação do cartão.

No trecho Código 27, apresenta-se a forma de leitura dos *bytes* do *buffer* da porta serie. Nesse código realçam-se os seguintes pontos:

- A leitura da informação é feita *byte* a *byte* usando o método `.ReadByte` em vez de ler todo o *buffer* de uma só vez usando o método `.ReadLine`. Esta opção foi efetuada porque o método `.ReadLine` converte o valor hexadecimal do *byte* lido num caracter e esse caracter depende do código de caracteres definido no computador, o que provoca a alteração da informação envida pelo conversor. Usando o método `.ReadByte` obtém-se um *array* dos valores hexadecimais exatos enviados pelo conversor, mas o *buffer* tem de ser lido um *byte* de cada vez.

- A leitura dos bytes é feita para um *array* dinâmico. Na primeira execução do ciclo repetitivo o *array* tem dimensão zero e que vai crescendo à medida que se leem os *bytes* do *buffer*.

Código 27 – Leitura do *buffer* de dados da porta serie.

```
1  . . .
2
3  Do While frmPrincipal.SerialPort_Cartoes.BytesToRead > 0 And
           UBound(BytesEntrada) < frmPrincipal.SerialPort_Cartoes.ReadBufferSize - 1
4       ReDim Preserve BytesEntrada(UBound(BytesEntrada) + 1)
5       BytesEntrada(UBound(BytesEntrada)) = frmPrincipal.SerialPort_Cartoes.ReadByte
6  Loop
7
8  . . .
```

Depois de obter o número do cartão de RFID, a aplicação pergunta à base de dados sobre as permissões de acesso do cartão e apresenta o resultado ao vigilante. O trecho Código 28, apresenta a implementação desta operação de onde são de realçar os seguintes aspetos:

- Na linha 13 efetua-se a consulta à base de dados sobre a permissão de acessos. De notar que este método da classe tem como parâmetros de entrada além do número do cartão e da identificação da portaria, tem também, a identificação do vigilante que está a operar, isto para que nos registos de histórico, essa informação seja associada à apresentação do cartão e seja possível em cada passagem, saber quem passou, quando, onde e quem é que estava de serviço de vigia.

- Nas linhas 20 a 33, faz-se a análise de algum possível erro que a base de dados tenha devolvido. Estes erros, estão dentro da gama de erros apresentados no Anexo K e são relativos à operação em questão.

- Nas linhas 38 a 81 faz o tratamento da informação relativa a um cartão registado no sistema. Neste bloco de código usa-se frequentemente a função `PreencheDadosPessoa` que aceita informações e atribui-as aos diversos objetos que constituem o ecrã de informação de acessos, como os apresentados nas Figura 193, Figura 194 e Figura 195.

Código 28 – Verificação das permissões de um cartão.

```
1       Private Sub ValidaCartao(idCartao As String, idPortaria As String, idAgente As String)
2           Dim bAcessoPermitido As Boolean
3           Dim sNomePessoa As String = ""
4           Dim sIdPessoa As String = ""
5           Dim lResultado As Long
6           Dim bMostraFoto As Boolean = False
7
8           On Error GoTo GestorErros
9
10          bAVerUmCartão = True
11
12          'verifica se a pessoa tem acesso
13          Dim Resultado = dbPortarias.spCartaoAcedePortaria(idCartao, idPortaria, idAgente,
                            bAcessoPermitido)
14          For Each linha In Resultado
15              lResultado = linha.Column1
16          Next
17
18          '-------------------------------------------------------------------------------
19          'Tratamento dos casos de erro
20          Select Case lResultado
21              Case -2
22                  ' -2  - agente inválido
23                  CorFundo = Color.Red
24                  PreencheDadosPessoa(, , , , , , , , "Acesso Negado", "Vigilante
                                        desconhecido", const_MOSTRA_SO_CARTAO)
25              Case -4
26                  ' -4  - Portaria desconhecida
27                  CorFundo = Color.Red
28                  PreencheDadosPessoa(, , , , , , , , "Acesso Negado", "Portaria desconhecida",
                                       const_MOSTRA_SO_CARTAO)
29              Case -6
30                  ' -6  - Cartão desconhecido
31                  CorFundo = Color.Red
32                  PreencheDadosPessoa(idCartao, , , , , , , , "Acesso Negado", "Cartão
                                        desconhecido. Apreender cartão", const_MOSTRA_SO_CARTAO)
33          End Select
34
35
36          '-------------------------------------------------------------------------------
37          'Tratamento do cartão reconhecido
38          Dim DadosPessoa = dbPortarias.spPessoa_Dados(idCartao)
39          For Each row In DadosPessoa
40
41              If IsNothing(row.LicencaConducao) Then row.LicencaConducao = ""
42              If IsNothing(row.LVO) Then row.LVO = False
43              If IsNothing(row.VeiculosEspeciais) Then row.VeiculosEspeciais = False
44
45              NomeAnexo = ""
46              If Not (IsNothing(row.ListaObjetosProibidos)) Then
                     NomeAnexo = row.ListaObjetosProibidos
47              'retira o perfixo do nome
48              'NomeAnexo = NomeAnexo.Substring(0, InStr(NomeAnexo, "-") - 1)
49
50              Select Case lResultado
51                  Case Is > 0
52                      '  > 0  - O cartão tem acesso à portaria indicada
53                      CorFundo = Color.Green
54                      bMostraFoto = True
55                      PreencheDadosPessoa(idCartao, row.DataValidadeCartao, row.Nome,
                                            row.Entidade, row.ObjetosProibidos,
                                            row.LicencaConducao, row.LVO, row.VeiculosEspeciais,
                                            "Acesso Permitido", "", const_MOSTRA_TODOS_CAMPOS)
56                  Case 0
57                      '  0  - O cartão não tem acesso à portaria indicada
58                      CorFundo = Color.Red
59                      bMostraFoto = True
60                      PreencheDadosPessoa(idCartao, row.DataValidadeCartao, row.Nome,
```

```
                                        row.Entidade, row.ObjetosProibidos,
                                        row.LicencaConducao, row.LVO, row.VeiculosEspeciais,
                                        "Acesso Negado", "Sem acesso a esta portaria",
                                        const_MOSTRA_TODOS_CAMPOS)
61                  Case -34, -13
62                      '-34  - Cartão fora de data de validade
63                      bMostraFoto = True
64                      CorFundo = Color.Red
65                      PreencheDadosPessoa(idCartao, row.DataValidadeCartao, row.Nome,
                                        row.Entidade, row.ObjetosProibidos,
                                        row.LicencaConducao, row.LVO, row.VeiculosEspeciais,
                                        "Acesso Negado", "Cartão fora da validade.",
                                        const_MOSTRA_TODOS_CAMPOS)
66                  Case -35
67                      '-35  - Cartão não ativo
68                      bMostraFoto = True
69                      CorFundo = Color.Red
70                      PreencheDadosPessoa(idCartao, row.DataValidadeCartao, row.Nome,
                                        row.Entidade, row.ObjetosProibidos,
                                        row.LicencaConducao, row.LVO, row.VeiculosEspeciais,
                                        "Acesso Negado", "Cartão desativado.",
                                        const_MOSTRA_TODOS_CAMPOS)
71              End Select
72              '-------------------------------------------------------------------------------
73              'Tratamento da foto
74              If bMostraFoto Then
75                  dbPortarias.spCartao_PessoaId(idCartao, sIdPessoa)
76                  CarregaFotoDaBDEmPictureBox(sIdPessoa, PictureBox_Foto)
77                  PictureBox_Foto.Visible = True
78              Else
79                  PictureBox_Foto.Visible = False
80              End If
81          Next
82
83          '-------------------------------------------------------------------------------
84          'Tratamento da moldura
85          AlertaContorno(CorFundo)
86          GroupBox_LeituraCartao.Visible = True
87
88  GestorErros:
89          If Err.Number <> 0 Then
90              MsgBox("Erro na Função: ValidaCartao" & vbCrLf & Err.Number & " - " &
                        Err.Description, vbCritical + vbOKOnly)
91              Err.Clear()
92          End If
93      End Sub
```

4.2.4. IMPLEMENTAÇÃO DA APLICAÇÃO PONTU

A aplicação PONTU permite fazer a gestão e emissão de cartões acesso de duração muito curta, sejam cartões para visitas, denominados pontuais, quer sejam cartões para passageiros de chegadas que acederem às zona *Lost&Found* para recolha de bagagem.

4.2.4.1. DESCRIÇÃO FUNCIONAL DA APLICAÇÃO PONTU

Á semelhança das aplicações já apresentadas, a aplicação PONTU é iniciada no ecrã de identificação do utilizador, Figura 200, que dá acesso ao ecrã principal onde são apresentados os menus com as funcionalidades disponíveis, Figura 201. Um menu dá acesso as funcionalidades relativas à credenciação pontual e o outro menu permite aceder às operações relacionadas com credenciação de passageiros. A aplicação tem um terceiro menu que apresenta os dicionários das listas usadas pela aplicação.

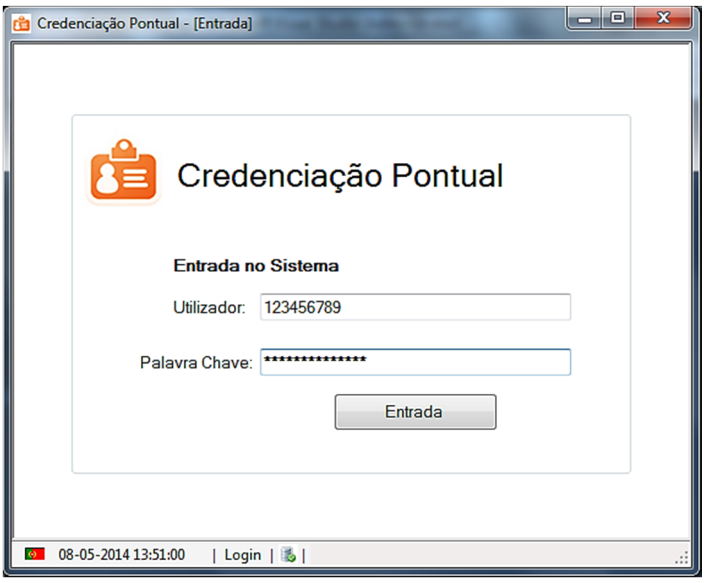


Figura 200 – Aplicação PONTU, ecrã de entrada.

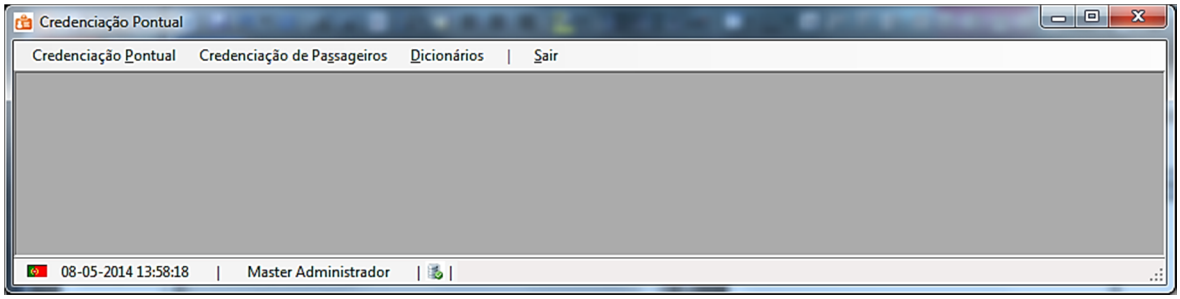


Figura 201 – Aplicação PONTU, ecrã principal.

Credenciação pontual

No ecrã de credenciação pontual, faz-se o registo da informação relevante do visitante, nomeadamente: nome, número de documento de identificação, entidade que representa, data de nascimento, etc. Estas funcionalidades, usam a metodologia de implementação que foi apresentada nas outras aplicações, Figura 202.

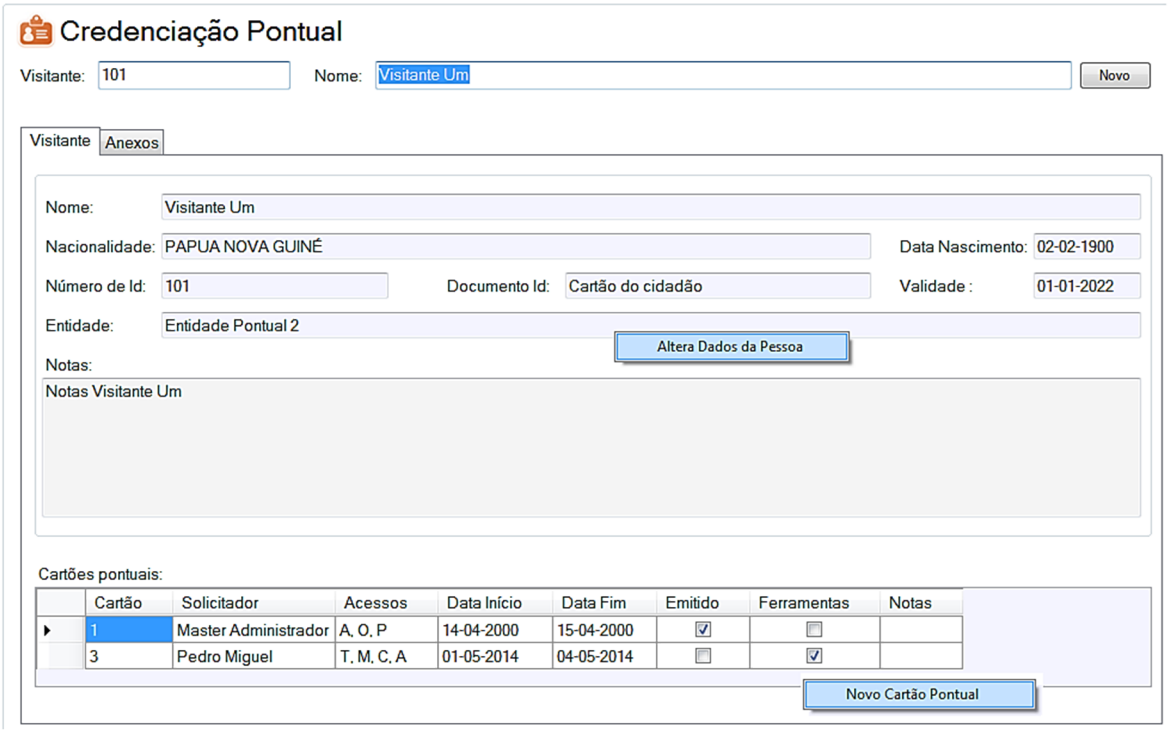


Figura 202 – Aplicação PONTU, ecrã de credenciação pontual.

Neste ecrã, também se procede à atribuição dos cartões de acesso e das respetivas permissões. Para o fazer, pressiona-se com o botão direito do rato na tabela de cartões pontuais e seleciona-se a opção "Novo Cartão Pontual", que apresenta o ecrã mostrado na Figura 203.

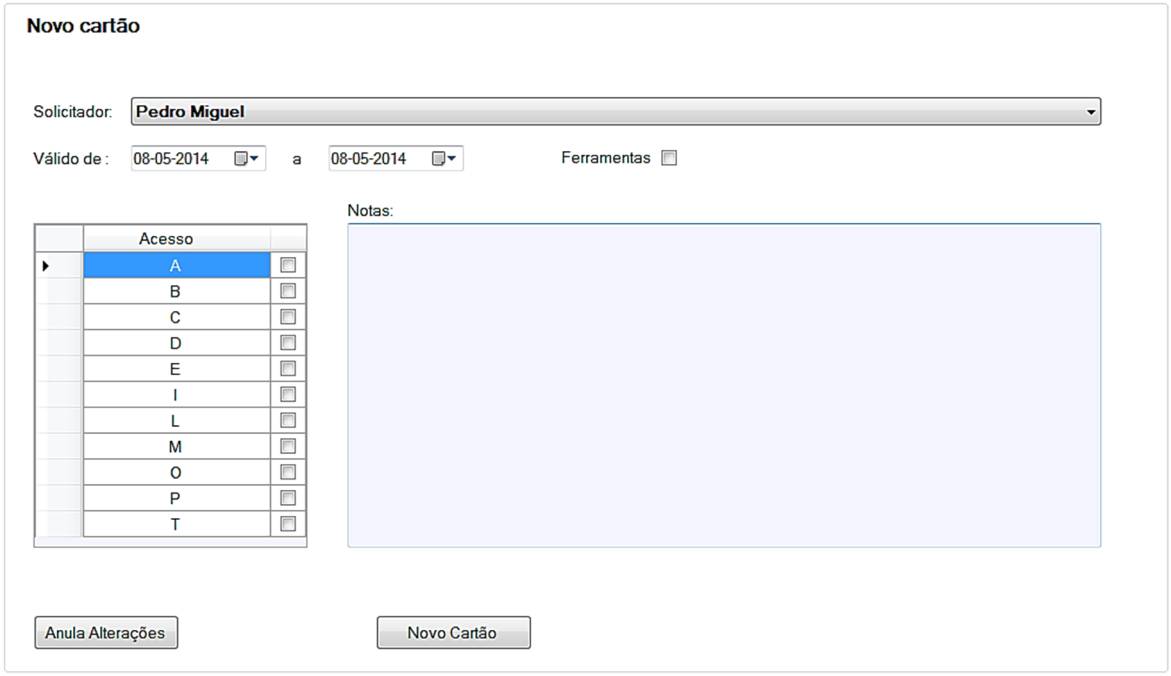


Figura 203 – Aplicação PONTU, ecrã de credenciação pontual, novo cartão.

No ecrã de criação de novo cartão, além da indicação da data da visita, da informação se o visitante é portador de ferramentas e dos acessos pretendidos é necessário indicar a pessoa interna que solicitou o cartão e é responsável por acompanhar a visita.

Impressão de cartão pontual

Na tabela de cartões de acesso, no ecrã de credenciação pontual, se pressionarmos o botão direito do rato sobre um cartão ainda não emitido, o menu que é mostrado, Figura 204, apresenta as opções de alteração da informação do cartão e de impressão do mesmo cartão.

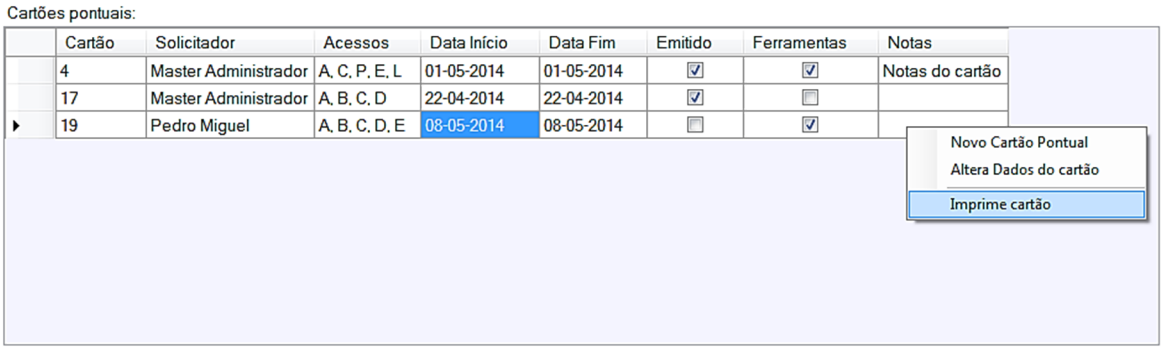


Figura 204 – Aplicação PONTU, ecrã de credenciação pontual, impressão de cartão.

Usando a opção impressão de cartões, é apresentada uma réplica da imagem do cartão pontual, preenchido com a informação da respetiva pessoa. Á semelhança do tratamento efetuado nos cartões permanentes e temporários, ao pressionar com o botão direito do rato no ecrã do cartão pontual, são disponibilizadas as funcionalidades de visualização impressão do cartão, assim como de seleção da impressora, Figura 205. Após a ordem de impressão, o cartão passa ao estado de "Emitido" e na lista de cartões deixa de ser possível alterar os dados ou fazer uma nova impressão.

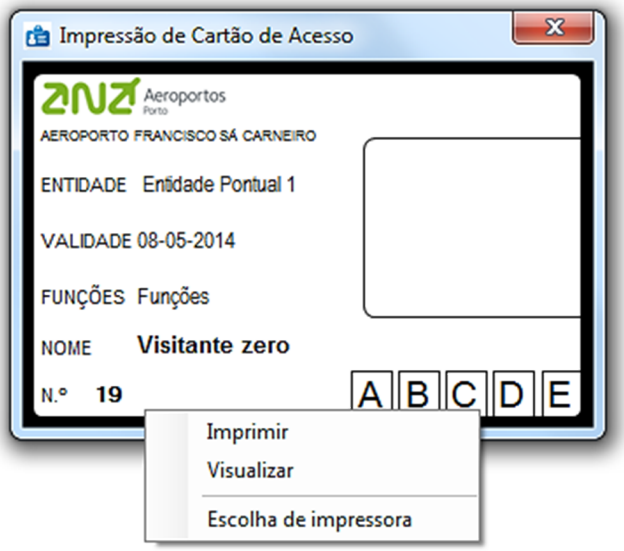


Figura 205 – Aplicação PONTU, cartão pontual.

Credenciação de passageiros

O ecrã de credenciação de passageiros permite fazer o registo da informação do passageiro, Figura 206, e o registo do respetivo cartão de acesso, Figura 207, usando metodologias já explicadas.

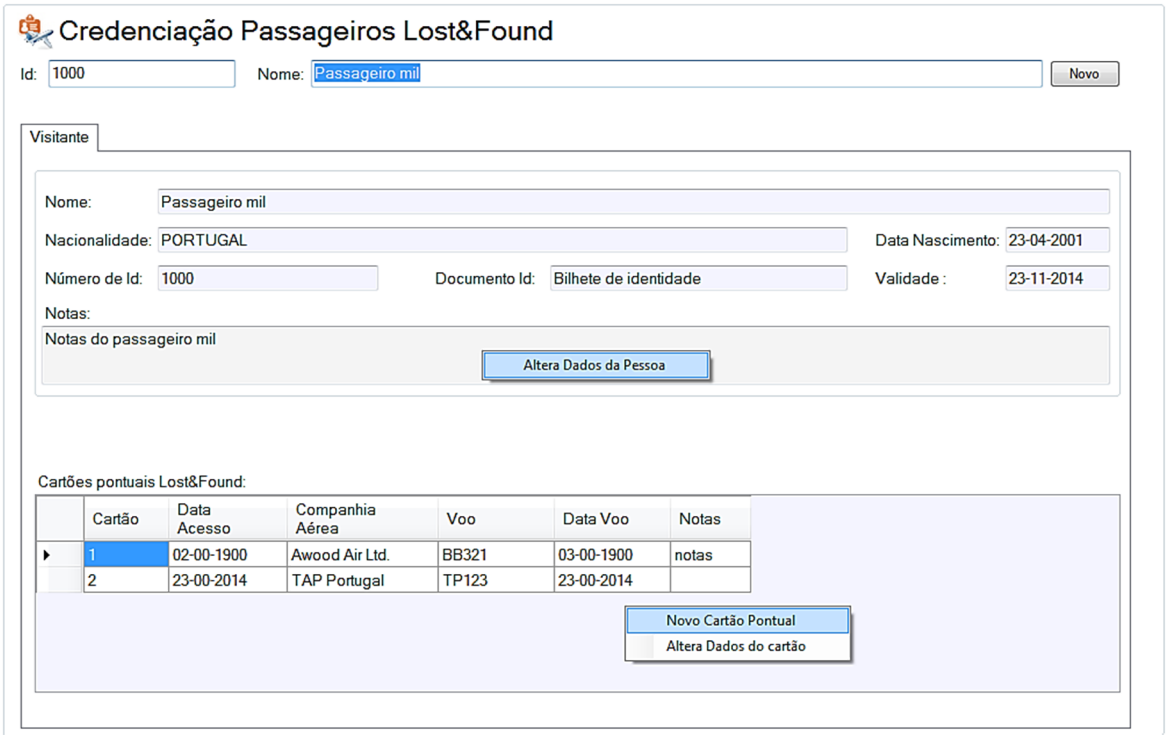


Figura 206 – Aplicação PONTU, Credenciação de passageiros.

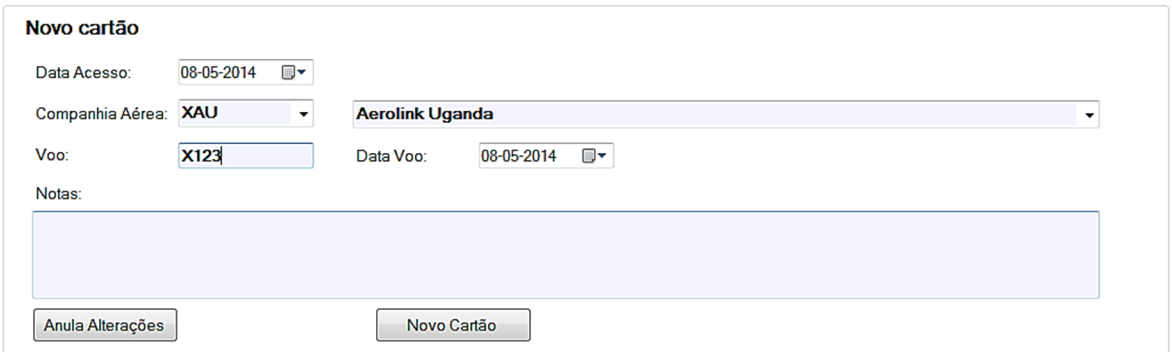


Figura 207 – Aplicação PONTU, Cartão de acesso *Lost&Found*.

Ecrã de dicionários

No ecrã de dicionários, é possível ao agente que está a operar a aplicação, introduzir novas entidades, que são representadas pelos visitantes que vão usar a credenciação pontual, Figura 208.

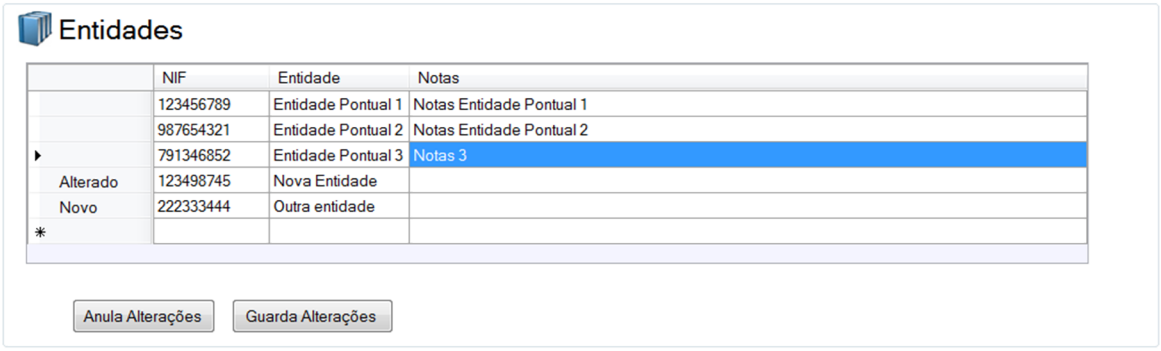


Figura 208 – Aplicação PONTU, dicionário de entidades.

4.2.4.2. DESCRIÇÃO DA IMPLEMENTAÇÃO DA APLICAÇÃO PONTU

A implementação da aplicação PONTU, baseia-se em mecanismos de carregamento de informação, seja em caixas seja em tabelas. Baseia-se, em menus sensíveis ao contexto, quer em objetos com informação quer sejam para emissão e impressão de cartões. A aplicação PONTU tem as funcionalidades apresentadas na secção anterior e é implementada com as soluções idênticas às apresentadas na explicação da implementação das aplicações CRED e PORT.

5. CONCLUSÕES

No fim do trabalho desenvolvido, obteve-se um protótipo que agrega numa única plataforma as funcionalidades relativas à credenciação e ao controlo de acessos atualmente existentes nas diversas aplicações e procedimentos em operação no Aeroporto Francisco Sá Carneiro.

A plataforma cobre as necessidades de registo, gestão e controlo das credenciações de acessos de pessoas e de viaturas, cobre as habilitações de condução em áreas reservadas, engloba a verificação eletrónica de acesso de passagem em portarias e permite gerir os acessos de visitas e passageiros.

Na implementação das bases de dados, quer nas que foram desenvolvidas especificamente para este projeto, quer na base de dados existente no sistema de controlo de portas, criaram-se camadas de encapsulamento que escondem ao exterior a complexidade da implementação do modelo relacional e que constituem um mecanismo fiável, flexível e robusto de registo de informação.

Ainda no âmbito das bases de dados, implementaram-se soluções de registos de histórico, para todas as operações e eventos que ocorrem no sistema. Soluções que são executadas de forma independente e transparente das aplicações externas. Desta capacidade resulta uma

compilação de informação poderosa do ponto de vista de análise e rastreio de eventos, recurso fundamental em sistemas de segurança.

No desenvolvimento das aplicações da plataforma, apenas foram usados recursos das ferramentas selecionadas, descartando as potencialidades de soluções de outros fornecedores que a longo prazo necessitam de suporte específico, tornando-se um problema para a plataforma como um todo.

No caso da ligação a dispositivos externos, como a máquina fotográfica, implementaram-se mecanismos de uso de equipamento universal em detrimento de dispositivos de um fabricante em particular, que no logo prazo, também eles podem representar uma fragilidade do sistema.

Relativamente à interação entre a plataforma desenvolvida e o sistema de controlo de portas, implementou-se uma solução que faz o sincronismo de informação de forma transparente, isto é, a emissão de cartões de acesso na plataforma, despoleta de forma automática e sem mais intervenções do operador, a criação dos respetivos elementos no sistema de abertura de portas, ficando o cartão emitido apto a abrir as portas associadas às suas permissões.

Do exposto conclui-se que os objetivos propostos para este trabalho foram alcançados e foram implementadas soluções adequadas ao âmbito do projeto.

Desenvolvimentos futuros

Com investimento de mais tempo de trabalho na:

- Análise das fontes de informação em uso nas soluções atuais. E migrando essa informação para as bases de dados desenvolvidas nesta plataforma.
- E, submetendo o protótipo ao uso de operadores e implementando as adaptações sugeridas pela experiencia do uso.

O protótipo funcional desenvolvido no âmbito nesta Tese transforma-se numa solução produtiva pronta para ser usada.

Referências Documentais

[1] Gabinete de Segurança ASC – *Manual de Formação de Segurança Aeroportuária Nível 6*, ANA – Aeroportos de Portugal S.A., Outubro 2008.

[2] Gabinete de Segurança ASC – *Manual de acessos às áreas críticas das zonas restritas de segurança e áreas reservadas*, 2ª revisão, ANA – Aeroportos de Portugal S.A., Outubro 2010.

[3] Airport Council International – *ASQ Awards*. http://www.aci.aero/Airport-Service-Quality/ASQ-Awards/Past-Winners/2012. Consultado em Janeiro de 2014.

[4] Google Maps, Consultado em Janeiro de 2014.

[5] MARRANA, João; PIRES, Paulo; et al- *Aeroporto Francisco Sá Carneiro, um novo aeroporto para o noroeste peninsular*, 2007 ISBN 978-989-95428-1-5

[6] BAKKER, Paul; FRIED, Stephen; et al- *Official Guide To The CISSP CBK Secutity Transcends Technology*, 2nd Edition 2010, CRC Press. ISBN 978-1-4398-0959-4.

[7] MACGREGOR, William; MEHTA, Ketan; et al – *A Recommendation for the Use of PIV Credentials in Physical Access Control Systems (PACS)*. National Institute of Standards and Technology, U.S. Departement of Commerce, 2008.

[8] Smart Card Alliance – *Smart Card Technology and Application Glossary*, 2009.

[9] *What is the difference between "Biometric identification" and "biometric verification"?*. http://arindamcctvaccesscontrol.blogspot.pt/2011/01/what-is-difference -between-biometric.html, consultado em Setembro de 2013.

[10] Smart Card Alliance – *Physical Access Control System Migration Options for Using FIPS 201-1 Compliant Credentials*. PAC-07002, 2007

[11] Smart Card Alliance – *The top 10 hot identity topics*. IC-06001, 2006.

[12] Open Security Exchange – *Open Secutiy Exchage Best Practices*. IEEE – Open Security Exchange, 2004

[13] MERKERT, Robert J. – *Smart Cards and Biometrics in Physical Access Control Systems*. Biometric Consortium 2005 Conference.

[14] Smart Card Alliance – *Using Smart Cards for Secure Physical Access*. ID-03003, 2003.

[15] Datacenter Dynamics – *Portugal Telecom inaugura CPD referência em tecnologia na Covilhã*, consultado em Setembro de 2013.

[16] SIEMENS – *SiPass Integrated System architecture*, http://www.sipass-access-control.com/ssp-sipass/downloads/sipass_integrated_diagram.pdf , consultado em Setembro de 2013.

[17] HONEYWELL – *Nexsentry Star II controller with Miro options*, 92995100049-A Honeywell International Inc.

[18] HONEYWELL- *Pro-Watch Software Suite Release 3.8.0 Guide*, Honeywell International Inc, 2010.

[19] HELIX – *Wiegand Format connections*. Technical Tip No 13.

[20] *Wiegand data formats*. http://www.proxmark.org/forum/viewtopic.php?pid=8036, consultado em Novembro de 2013.

[21] SIMONS VOSS – *System 3060 – Digital Access Components* – Smart Handle.

[22] SIMONS VOSS – *Transponder 3064*.

[23] SIMONS VOSS – SYSTEM 3060 – *Digital Access Components – Smart Handle*

[24] HONEYWELL – *The Nexsentry Star II Access Control Unit*, 6600058, Honeywell International Inc.

[25] Honeywell – *OmniProx™ Reader Model Series OP10/20/30/40/45/90*. K5336 Rev. 7, 07/2001.

[26] BANERJEE, Bob – *The PSIM conversation is changing from "integration" to "impacts"*. www.securityinfowatch.com/article/10892410/video-surveillance-4-bold-predictions-for-psim, consultado em Janeiro de 2014.

[27] GRIFFIN, Joel – *Moving beyond integration with PSIM*. www.securityinfowatch .com /article/10777266/psim-vendors-discuss-industry-trends-at-asis-2012, consultado em Janeiro de 2014

[28] O'MARA, Deborah – *PSIM: Navigating the Great Unknown.* http://www.securityinfowatch .com/article/10516404/psim-navigating-the-great-unknown, consultado em Janeiro de 2014

[29] FINKENZELLER, Klaus – *RFID handbook fundamentals and applications in contactless smart cards, radio frequency identification and near-field communication*, 3rd Edition, 2010. John Wiley & Sons, Ltd. ISBN: 978-0-470-69506-7

[30] KEYENCE – *Bar code Reader technical guide*. 2000. Keyence Corporation BL-KA-HB2-1-1200.

[31] MOTA, Nuno Miguel Silva – *Integração de Sistemas de Identificação Automáticos*. Dissertação de Mestrado, 2010. Faculdade de Engenharia da Universidade do Porto.

[32] MASABI – *Connecting the Dots: An Introduction to 2D Barcodes.* www.masabi.com/2011/03/04/connecting-the-dots-an-introduction-to-2d-barcodes-3/, Consultado em Fevereiro de 2014.

[33] IATA – *In the fast lane*. Passenger terminal world magazine, Jan 2014.

[34] *Automated access control for intra-Schengen passengers.* www.brusselsairport.be /en/blog_bru/oct2011/18919/, consultado em Setembro de 2013.

[35] Wikipedia- *Magnetic stripe card*, http://en.wikipedia.org/wiki/Magnetic_card, consultado em Setembro de 2013.

[36] Segure-Teck – *Blank Cards*, http://www.securetech-corp.com/store/magnetic-card/mag-blankcards, consultado em Setembro de 2013.

[37] SILICON LABS – *Magnetic Stripe Reader*, Application Note AN148, rev.1.3.

[38] RANKL, Wolfgang; EFFING, Wolfgang – *Smart Card Handbook*, 4th Edition, 2010. John Wiley and Sons, Ltd. ISBN 978-0-470-74367-6.

[39] WARD, Matt; KRANENBURG, Rob van – *RFID: Frequency, standards, adoption and innovation*, 2006. JISC Technology and Standards Watch.

[40] WEINSTEIN, Ron – *RFID: A Technical Overview and Its Application to the Enterprise*. 520-9202/05 IEEE Computer Society, 2005.

[41] EBVELKTRONIK - *RFID Selection Guide*, version 1 2010

[42] NAKAMORI, Emi; TSUKUDA Daiki; et al.- *A New Indoor Position Estimation Method of RFID Tags for Continuous Moving Navigation Systems*. Japão, 978-1-4673-1954-6/12, IEEE, 2012.

[43] YANG, Po; WU, Wenyan; et al.- *Efficient Object Localization Using Sparsely Distributed Passive RFID Tags*. 0278-0046, IEEE, 2012

[44] Codegate – *Which RFID technology?*. www.codegate.co.uk/rfid/whichrfidtechnology, consultado em Novembro de 2013

[45] Via Verde – *Como Funciona a via Verde*. http://www.viaverde.pt/Website /Section.jsf?TopFolderPath=%5CRoot%5CContents%5CWebsite%5CProdutosServicos&SelectedSub FolderId=109/, consultado em Setembro de 2013.

[46] WARD, Matt; KRANENBURG. Rob- *RFID: Frequency, standards, adoption and innovation*. JISC Technology and Standards Watch, 2006.

[47] Wikipedia- *Oersted (unidade))*.http://pt.wikipedia.org/wiki/Oersted_(unidade), consultado em Dezembro de 2013.

[48] Wikipedia- *Electromagnetic radiation*, *EM spectrum.svg*. http://en.wikipedia.org/wiki, consultado em Setembro de 2013.

[49] Wikipedia- *Microchip implant (animal)*.http://en.wikipedia.org/wiki/, consultado em Setembro de 2013.

[50] MATIS, Ross – *Port Logistics Group deploys TAGSYS RFID solutions for retail 272érea272tí*, http://secureidnews.com/news-item/port-logistics-group-deploys-tagsys-rfid-solutions-for-retail-clients/?tag=rfid#, consultado em Setembro de 2013.

[51] RFID WORLD CANADA- *World RFID Market to Reach 20 Billion USD in 2014?* http://www.rfidworld.ca/ world-rfid-market-to-reach-20-billion-usd-in-2014/769, consultado em Setembro de 2013.

[52] KIP, Arthur – *Fundamentals of electricity and magnetism*. 1969, McGraw-Hill ISBN 07-034780-8.

[53] BOAVENTURA, Alírio – *Leitor/Gravador RFID – Banda HF (13.56MHz)*, Dissertação de Mestrado, 2009, Departamenteo de Engenharia de Eletrónica, Telecomunicações e Informática da Universidade de Aveiro

[54] HID – *HID Global Product Catalog*, MKT-PRODCAT-RG-EN, 2008.

[55] NFC Forum – *NFC and Contactless Technologies*. http://www.nfc-forum.org/aboutnfc/nfc_and_contactless/, consultado em Setembro de 2013.

[56] Near Field Communication (NFC) Channel – *Provisioning physical access credentials to mobile phones*. http://secureidnews.com/technologies/, consultado em Setembro de 2013.

[57] SIMONS VOSS – *System 3060 – Identification Media – Mobile Key*. www.simons-voss.com , consultado em Janeiro de 2014.

[58] BURKE, Oliver – *Pave the way for NFC-based ID Services*. Security Magazine, Janeiro de 2014.

[59] SIMONS VOSS – *WaveNet*. www.simons-voss.com , consultado em Janeiro de 2014.

[60] LE, Chien – *A survey of biometrics security systems*. http://www.cse.wustl.edu/ ~jain/cse571-11/ftp/biomet/273érea.html, consultado em Janeiro de 2014.

[61] ADEOYE, Elufemi - *A survey of emerging biometric Technologies*, International Journal of Computer Applications (0975 – 8887) Volume 9– No.10, 2010.

[62] PRIMO, Patricia – *Projeto de desenvolvimento de sistemas ópticos para módulos biométricos*. Dissertação de Mestrado, 2009. Universidade de Coimbra.

[63] MATOS, Hélder – *Reconhecimento biométrico baseado na geometria da mão*, Dissertação de Mestrado, 2011. Faculdade de Engenharia do Porto.

[64] CANEDO, José – *Visão geral de um sistema biométrico*. http://www. Forumbiometria.com/ fundamentos-de-biometria/129-visao-geral-de-um-sistema-biometrico.html, consultado em Setembro de 2013.

[65] PARK, Gitae; KIM, Soown – *Hand Biometric Recognition Based on Fused Hand Geometry and Vascular Patterns*, Journal Sensors, 2013. ISSN 1424-8220.

[66] GALBALLY, Javier; ROSS, Arun; et al. – *Íris image reconstruction from binary templates: An efficient probabilistic approach based on genetic algorithms*. 2013. Computer Vision and Image Understanding, 2013.

[67] DAUGMAN, John – *Envolving Methods in 273ére recognition*, University of Cambridge.

[68] TISSE, Christel; MARTIN, Lionel – *Person identification technique using human 273ére recognition*. Université de Montpellier.

[69] National Science and Technology Council – *Íris Recognition*. http://www.fbi.gov/about-us /cjis /fingerprints_biometrics/biometric-center-of-excellence/files/iris-recognition.pdf, consultado em Fevereiro de 2014.

[70] APPLE – *iPhone 5s: About Touch ID security*. http://support.apple.com/kb/HT5949, consultado em Fevereiro de 2014.

[71] BIMA – *Biometrics glossary*. Version 6.0. Biometrics Identity Management Agency, 2012.

[72] PRABHAKAR, Salil; IVANISOV, Alexander; JAIN, Anil – *Biometric recognition: Sensor characteristics and image quality*, 2011. IEEE Instrumentation & Measurement Magazine, ISSN 1094-6969.

[73] BRENNER, Gabriel; BIZARRIA, Walter – *Sistema de controle de acesso com biometria da digital*. SEGeT – VIII Simpósio de Excelência em Gestão e Tecnologia.

[74] JAIN, Anil; PRABHAKAR, Jain; ROSS, Arun – *Fingerprint matching: data acquisition and performance evaluation*. Michigan State University MI48824.

[75] EKEY – *eKey finger scanners business*, www.ekey.net, consultado em Fevereiro de 2014.

[76] HID – *bioClass RKLB57, RWKLB575 and BIO500*. www.hidglobal, consultado em Fevereiro de 2014.

[77] *Imagem eye-diagram-front*, http://healthfavo.com consultado em Fevereiro de 2014.

[78] MASEK, Libor – *Recognition of human patterns for biometric identification*, 2003. University of Western Australia.

[79] http://www.wired.co.uk/

[80] KIMALDI – *Terminal de reconhecimento facial 3D Suprema FaceStation*. http://www.kimaldi.com/kimaldi_por/produtos/sistemas_biometricos/controlo_de_acesso_biometrico/terminal_de_reconhecimento_facial_3d_suprema_facestation, consultado em Setembro de 2013.

[81] KIMALDI – *Terminal de biometria vascular Kimaldi FingerVein*. http://www.kimaldi.com /kimaldi_por/produtos/sistemas_biometricos/274érea274tív_digital_vascular_e_facial/biometria_vascular/terminal_de_biometria_vascular_kimaldi_fingervein. Consultado em Fevereiro de 2014.

[82] http://www.hertasecurity.com/wp-content/uploads/2012/10/scanner_retina.png

[83] ANDRADE, Christopher – *Investigating and comparing multimodal biometric techniques*. University of Johannesburg.

[84] STMicroelectronics – *RM0008, Reference Manual*, 2088. www.st.com , consultado em Janeiro de 2014.

[85] Microsoft - *Overview of the .NET Framework*, http://msdn.microsoft.com/en-us/library/zw4w595w.aspx, consultado em Setembro de 2013.

[86] MAGALHÃES, Alberto – *SQL Server 2008*, 2ª Edição- FCA Editora de Informática, LDA. ISBN: 978-792-722-684-9.

[87] EVJEN, Bill; HOLLIS, Billy; et al.- *Visual Basic 2008*, 1st Edition. Wrox. ISBN: 978-0-470-19136-1

[88] HONEYWELL INTERNATIONA INC. – *Pro-Watch Software Suite Guide*, 2011. Release 3.81 revision J.

[89] NEX WATCH - *The Nexsentry StarII Access Control Unit User Guide*, Revision B, Part #6600058.

[90] DATE, C.J – *SQL and Relational Theory*. 2 edition, O´Reilly, ISBN: 978-1-449-31640-2, 2012.

[91] MICROSOFT – *Information Schema Views (Transact-SQL)*. http://technet.microsoft.com/en-us/library/ms186778.aspx, consultado em Março de 2014.

[92] NATARAJAN, Jay; BRUCHEZ, Rudi; SHAW Scott – *Pro T-SQL 2012 Programmer's Guide*, 3rd Edition. Apress, ISBN: 978-1-4302-4597-1.

[93] BEAULIEU, Alan - *Learning SQL*, 2nd Edition, 2009. O'Reilly, ISBN: 978-0-596-52083-0

[94] BEN-GAN, Itzik - Microsoft® SQL Server® 2012 T-SQL Fundamentals, 2012. O'Reilly, ISBN: 978-0-735-65814-1.

[95] RATIONAL– *UML Notarion Guide*. Version 1.1, 1997.

[96] HAMILTON, Kim; MILES, Russell – *Learning UML 2.0*, 2006. O'Reilly, ISBN: 978-0-59-600982-3

[97] PILONE, By Dan; PITMAN, Neil – *UML 2.0 in a Nutshell*, 2005. O'Reilly, ISBN: 0-596-00795-7

[98] MILICEV, Dragan – *Model-Driven Development with Executable UML*, 2009. Wiley Publishing, Inc. ISBN: 978-0-470-48163-9.

[99] Wikipedia- *UML*. http://pt.wikipedia.org/wiki/UML, consultado em Março de 2014.

[100] HONEYWELL – *OmniProx, HAS-OMNIPROX*, Honeywell, 2007.

[101] ADEMCO – OmniProx™ Reader Installation Instructions, K5336 Rev. 7, 2001.

[102] ETConcept – Conversor Wiegand para RS232 Manual Técnico, V1.3, 2009.

[103] HONEYWELL – *OmniProx, HAS-OMNIPROX*, Honeywell, 2007.

[104] ADEMCO – *OmniProx™ Reader Installation Instructions*, K5336 Rev. 7, 2001.

[105] ETConcept – *Conversor Wiegand para RS232 Manual Técnico*, V1.3, 2009.

[106] Mueller – LiYCY Shielded PVC Data Cable.

[107] MSDN – *LINQ to SQL*, http://msdn.microsoft.com/en-us/library/bb386976.aspx, consultado em Novembro de 2013.

[108] MSDN – *Introduction to LINQ in Visual Basic*, http://msdn.microsoft.com/en-us/library/bb763068.aspx, consultado em Novembro de 2013.

[109] SEARS, Russell; INGEN, Catharine van; GRAY, Jim -*Large Object Storage in a Database or a Filesystem?*, Technical Report MSR-TR-2006-45, Microsoft Corporation, 2006.

Anexo A Códigos de áreas restritas e reservadas no ASC

O Aeroporto Francisco Sá Carneiro tem vigente uma classificação de áreas restritas e reservadas codificadas em letras e cores. As letras representam áreas com características de restrições específicas, os códigos de cores representam grupos de letras. Na Figura 209 é apresentado um extrato do documento [1] com a classificação das áreas em códigos de letras e na Figura 210 é apresentado a descrição do código de cores.

Áreas restritas:

Área O - constituída pelas pistas, caminhos de circulação *(taxiways)* que lhes dão acesso, áreas de segurança operacional envolventes e pelos órgãos/equipamentos afectos aos Serviços de Navegação Aérea, quando incluídos no perímetro aeroportuário;

Área P - constituída pelas plataformas de estacionamento de aeronaves, respectivos acessos e áreas de protecção envolventes;

Área I - constituída pelas salas de embarque, desembarque e trânsito e outras áreas das aerogares situadas entre as posições de controlo de imigração (passaportes) e as portas de embarque e desembarque;

Área M - constituída pelas instalações de manutenção de aeronaves e outras instalações técnicas ligadas à actividade que se processa nas plataformas de estacionamento das mesmas;

Área C - constituída pelos terminais e armazéns de carga ou sua fracção incluídos no "lado ar";

Área T - constituída pelos terminais de bagagem de partidas e de chegadas;

Área B - constituída pela sala de recepção de bagagem da área de desembarque dos passageiros;

Área E - constituída pelas salas de embarque, para além das posições de controlo de segurança, guarnecidas pela PSP;

Área L - constituída pela área das lojas *free-shop*, restaurantes e *lounges*;

Área D - constituída pelas salas de embarque dos voos domésticos.

Áreas reservadas:

Área A - área de acesso condicionado da aerogare.

Figura 209 – Classificação de áreas no ASC, extraído de [1].

Códigos de controlo de cores:

Cor verde - todas as áreas;

Cor vermelha - lado ar (áreas O, P e T);

Cor amarela - área de passageiros (áreas B, L, E, D, e I);

Cor azul - área de manutenção (área M);

Cor castanha - área de carga (área C);

Cor branca - área reservada (área A).

Figura 210 – Código de cores para áreas no ASC, extraído de [1].

A face impressa dos cartões de identificação de pessoas e de viaturas contém a especificação das zonas a que os elementos têm acesso. Estes previlégios podem ser apresentados pelo código de cores, por grupos de letras ou por uma combinação das duas. Normalmente, as cores apresentam as áreas onde o elemento exerce a maioria das suas funções e as letras poderão ser conjugadas com as cores para alargar as áreas de acesso [1]. Na Figura 211 é mostrado um exemplo de um cartão permanente em que o acesso é definido pela lista de cor castanha no lado esquerdo do cartão e pelo conjunto de letras "P", "T" e "I".

Figura 211 – Exemplo de cartão de acesso.

Anexo B Radiação eletromagnética

A radiação eletromagnética é a uma forma de energia constituída pela auto-propagação no espaço, de ondas do campo elétrico – E em fase, com ondas do campo magnético – B. Os campos E e B dispõem-se perpendicularmente um ao outro e os dois perpendicularmente à direção do movimento, Figura 212.

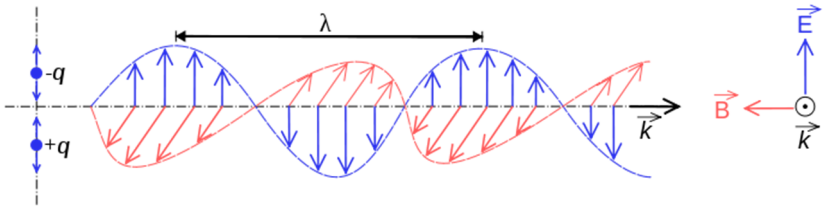


Figura 212 – Onda eletromagnética, [48].

As cargas elétricas em movimento emitem radiação eletromagnética que se manifesta de energeticamente de duas formas distintas, Figura 213.

- No espaço próximo das cargas que originam a radiação, conhecido por *near field*, se as cargas deixarem de se movimentarem, deixa de haver energia.

- No espaço "longe" das cargas, conhecido por *far field*, a radiação manifesta-se em de forma ondulatória que não depende das cargas elétricas que lhe deram origem podendo-se propagar indefinidamente pelo vazio à velocidade da luz.

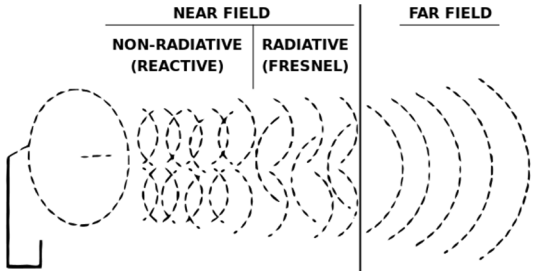


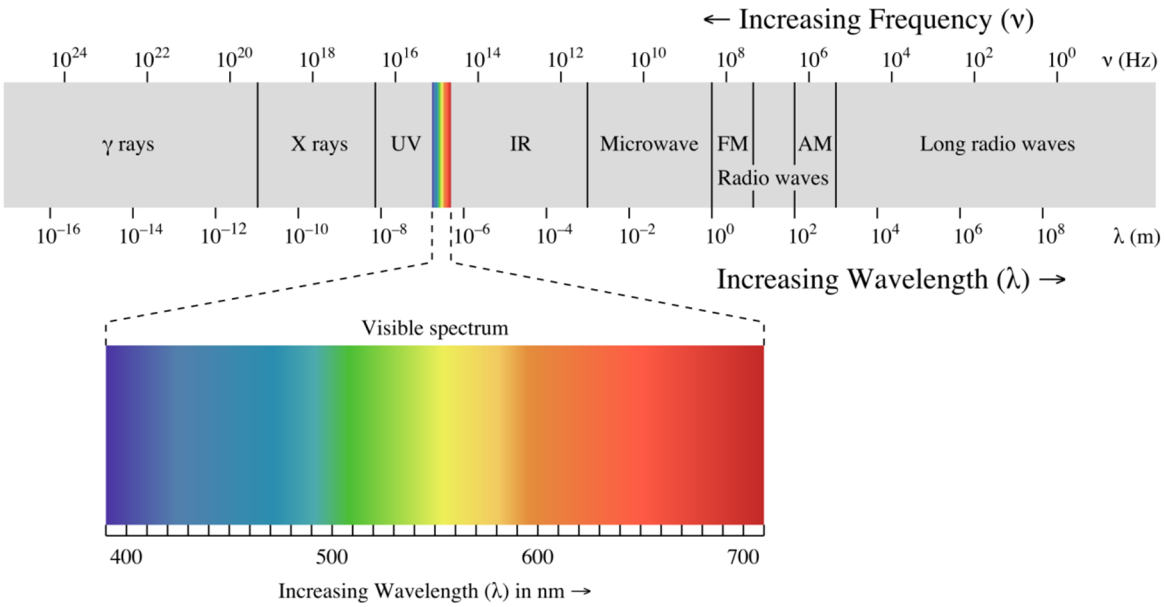
Figura 213 – *Near Field* / *Far Field*, [48].

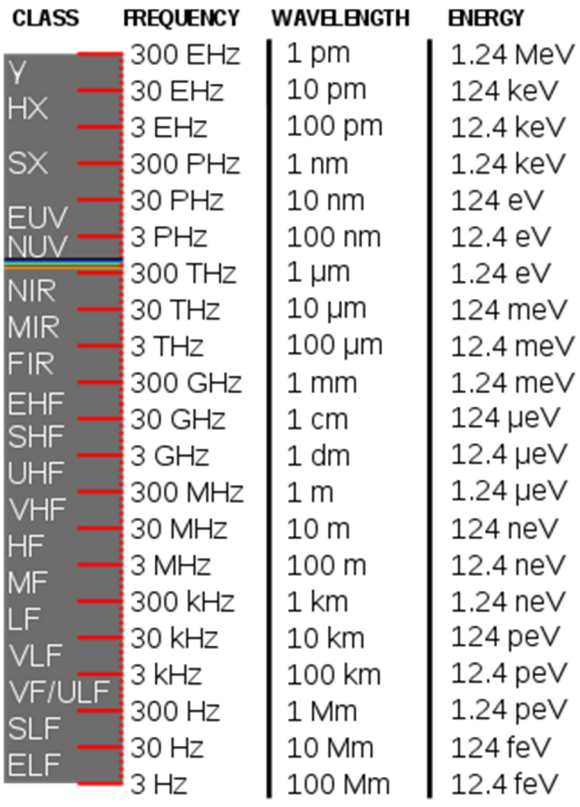
A radiação eletromagnética, segundo a teoria ondulatória, [52], é caracterizada pela frequência e pelo comprimento de onda. A frequência caracteriza a periodicidade no tempo. O período – T (segundo) é a medida de tempo em que a onda inicia a repetição de uma forma já passada. O comprimento de onda caracteriza a periodicidade no espaço, a distância mínima entre os pontos no espaço em que a onda passa a ter as mesmas características chama-se comprimento de onda – λ (metro). O período e o comprimento de onda estão relacionados pela expressão $V = \lambda / T$, em que V(metro/segundo) representa a velocidade de propagação da onda no meio.

Ainda considerando o caracter ondulatório da radiação eletromagnética, os campos elétricos e magnéticos obedecem às regras ondulatórias como o princípio de sobreposição, refração, difração e interação com a matéria que podem provocar absorção ou emissão de ondas com outras frequências.

Considerando o modelo de partículas da radiação eletromagnética, a radiação é composta partículas sem massa denominadas fotões, que são emitidos e absorvidos pelas partículas dos materiais atuando como transportadores de energia. A energia dos fotões é proporcional à frequência da onda associada e relacionada pela equação de Planck-Einstein $E = h \bullet \nu$, Em que E (eV) representa a energia o *quantum* do fotão, $h = 6{,}6260693(11)\text{x}10^{-34}$ Js, representa constante de Planck e ν (Hz) representa a frequência da onda.

As ondas eletromagnéticas são classificadas segundo a sua frequência variando de ondas de rádio, as de menor frequência e maior comprimento de onda, até à radiação gama, as de maior frequência e consequentemente as de maior energia, Figura 214.





Legend:
γ = Gamma rays

HX = Hard X-rays
SX = Soft X-Rays

EUV = Extreme-ultraviolet
NUV = Near-ultraviolet

Visible light (colored bands)

NIR = Near-infrared
MIR = Moderate-infrared
FIR = Far-infrared

EHF = Extremely high frequency (microwaves)
SHF = Super-high frequency (microwaves)

UHF = Ultrahigh frequency (radio waves)
VHF = Very high frequency (radio)
HF = High frequency (radio)
MF = Medium frequency (radio)
LF = Low frequency (radio)
VLF = Very low frequency (radio)
VF = Voice frequency
ULF = Ultra-low frequency (radio)
SLF = Super-low frequency (radio)
ELF = Extremely low frequency(radio)

Figura 214 – Espectro eletromagnético, [48].

Anexo C Características físicas dos cartões de identificação

Os cartões usados nos processos de identificação e autenticação podem assumir vários formatos normalizados, nomeadamente formato ID-1 cujos exemplos de aplicação mais conhecido são os cartões bancários, o formato ID-000 usado por exemplo nos cartões dos telefones móveis "mais antigos" e que estão a ser substituídos por cartões mais pequenos no formato MINI-UICC [38]. A Figura 215, Figura 216, Figura 217, e Figura 218 apresentam as características dimensionais destes formatos normalizados.

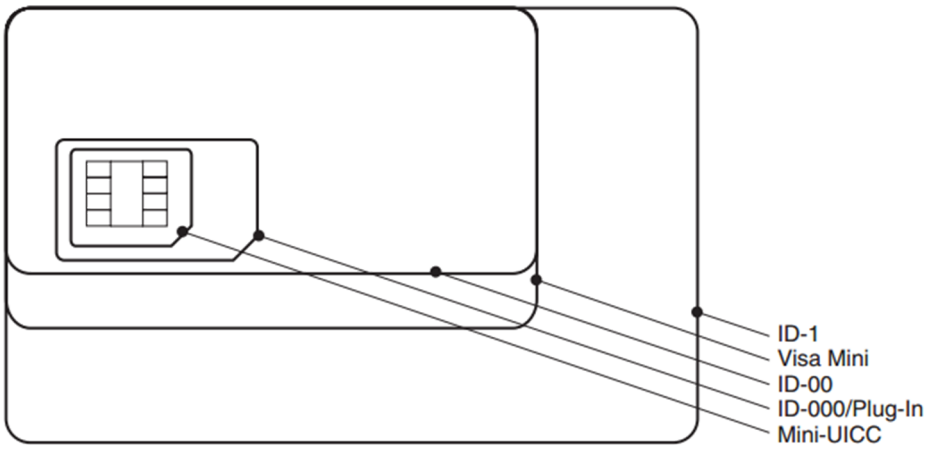


Figura 215 – Comparação das dimensões dos formatos normalizados de cartões, [38].

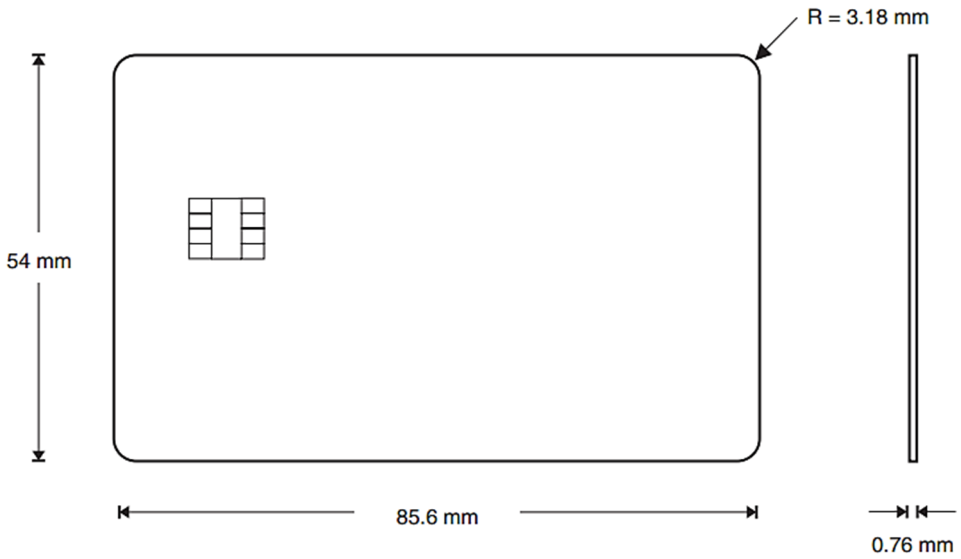


Figura 216 – Dimensões do formato normalizado de cartões ID-1, [38].

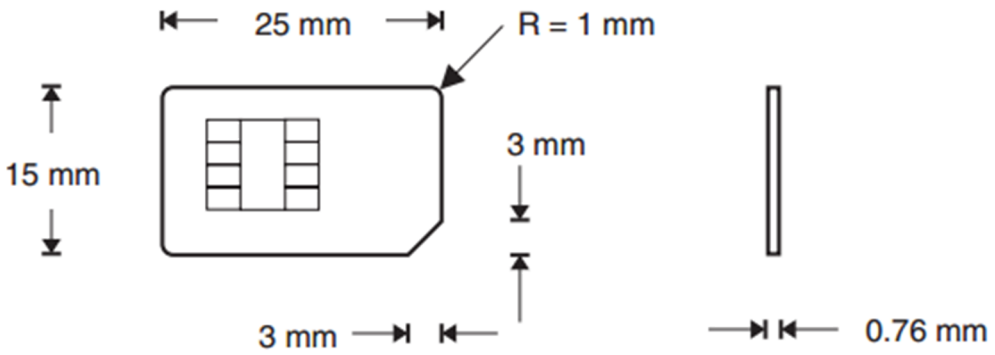


Figura 217 – Dimensões do formato normalizado de cartões ID-000, [38].

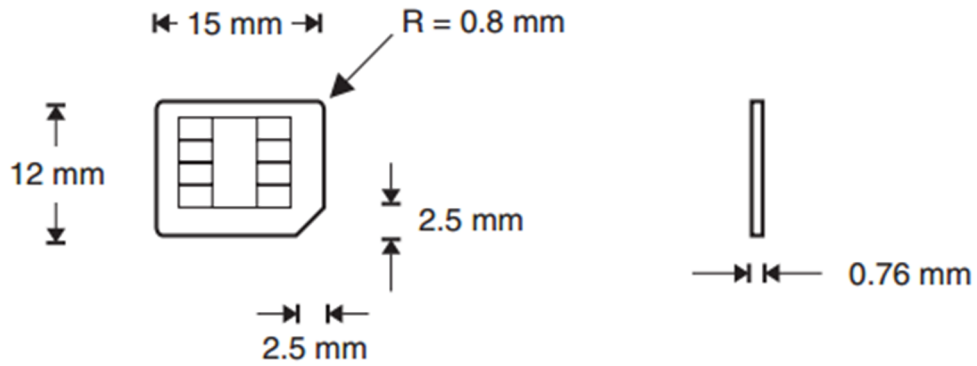


Figura 218 – Dimensões do formato normalizado de cartões MINI-UICC, [38].

Anexo D Contactos elétricos dos cartões de identificação

Em determinados cartões de memória e cartões com processador, existe a possibilidade de comunicar com os recursos do cartão de forma elétrica através de contactos existentes na superfície dos cartões. Estas implementações podem usar seis ou oito contactos. A norma ISO7816-2 especifica as dimensões dos contactos com superfície mínima de 1.7x2mm, e define a sua posição dos contactos no cartão, na Figura 219 são apresentados exemplos de tomadas de contactos.

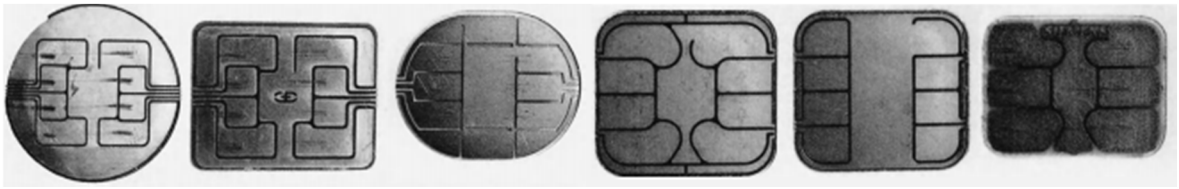


Figura 219 – Exemplos de contactos elétricos usados nos cartões de identificação, [38].

A função de cada pino de contacto também está normalizada e apresenta a distribuição mostrada na Figura 220 e descritos na Tabela 24.

Tabela 24 – Funções dos contactos elétricos dos cartões, [38].

Nome	Função
Vcc	Tensão de alimentação que depende do tipo de integrados: Class A – 5V, Class B – 3V, Class C – 1,8V.
GND	Tensão de referência.
RST	Sinal de *Reset*.
SPU	Para usos específicos, como tensão de programação de EEPROM, ou comunicação SWP.
CLK	Sinal de Relógio.
I/O	Sinal de comunicação de Entrada/Saída.
AUX1	Para uso específico como D+ na comunicação USB.
AUX2	Para uso específico como D- na comunicação USB.

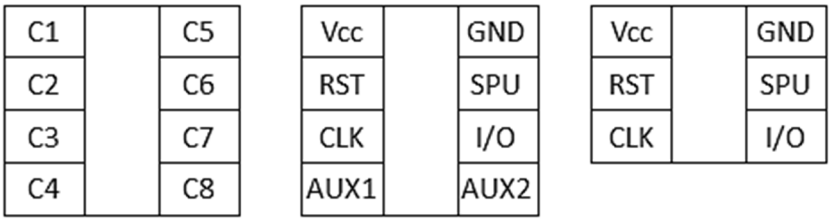


Figura 220 – Diagrama de contactos elétricos dos cartões, [38].

Na Figura 221, é mostrado um exemplo de diagrama de implementação de um *chip* com interface elétrica por contactos, onde se pode também ver a localização da instalação da antena para comunicação por radiofrequência. Na zona de baixo da figura são apresentadas fotografias de um módulo completo para ser integrado no corpo dos cartões e fotografias de detalhe das ligações elétricas.

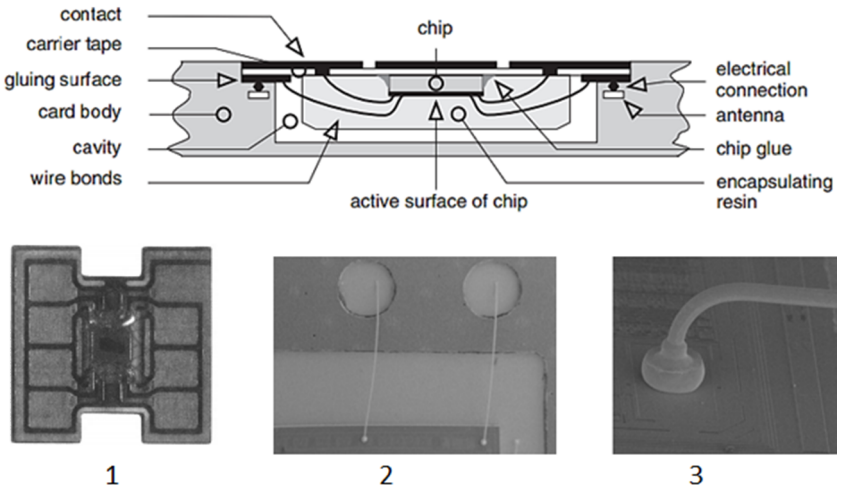


Figura 221 – Exemplo de instalação de um processador, [38].

Anexo E Protocolo *Inter-Integrated Circuit* - I^2C

O I^2C – *Inter-Integrated Circuit* ou *Inter IC* é protocolo de comunicação, série, síncrono, desenvolvido inicialmente para efetuar comunicações entre *motherboard* e periféricos nomeadamente com memórias EEPROM, SDRAM, DIMM, conversores analógico-digitais, digital-analógicos, dipositivos LCD, etc. Ao longo dos tempos a frequência de comunicação evoluiu dos 100KHz iniciais, até aos 5MHz da versão 5. O I^2C é um protocolo do tipo *master-slave*, de barramento multiponto e *multimaster* em que a função de geração do sinal de relógio é executada apenas pelo *master* e só admite comunicação num sentido – *half duplex*.

O barramento I^2C é implementado sobre duas linhas: a SDA que é a linha de dados e a SCL que contém o sinal de relógio. O andar de saída dos dispositivos *master* funciona na topologia dreno aberto e por isso o barramento de comunicação tem de ser equipado com resistências de fixação de tensão – *pull-up*, Figura 222.

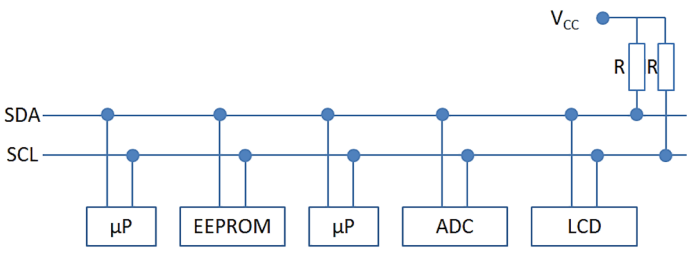


Figura 222 – Barramento I^2C.

Como o I^2C é um protocolo síncrono, o sinal de relógio define os instantes de escrita/leitura de informação na linha de dados que é fixado quando está no nível lógico baixo. Existem duas exceções a esta regra que definem o início e o fim da transmissão de dados, Figura 223 [84]:

- Condição de início de transmissão: estando as duas linhas em repouso, no estado lógico alto imposto pelas resistências de *pull-up*. O estado das linhas do barramento que marca o início de transmissão de dados é definido por colocar a linha de dados no nível lógico baixo, enquanto se mantem a linha de *clock* no nível alto. Este estado indica aos participantes na rede I^2C que a próxima vez que o *clock* estiver no estado lógico baixo, está-se a transmitir o primeiro *bit* da trama I^2C.

- Condição de fim de transmissão: a indicação de fim de transmissão é efetuada por: estado o sinal de *clock* no nível alto, faz-se uma transição do nível baixo para o nível alto na linha de dados.

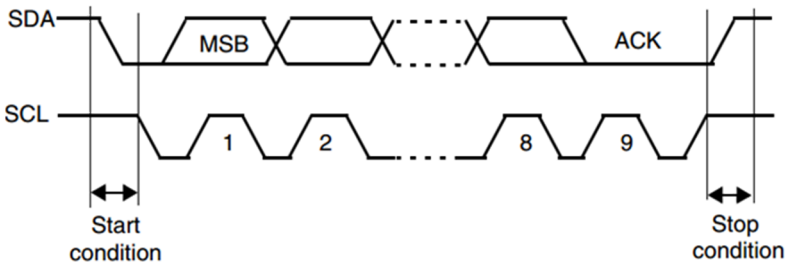


Figura 223 – Diagrama temporal do protocolo I^2C, [84].

Sendo o I^2C um protocolo multiponto, cada dispositivo presente no barramento tem associado um endereço que o identifica na rede. As tramas I^2C, são compostas por cinco blocos como mostrado na Figura 224:

- A transmissão começa por um *master* impor a condição de início.

- Depois segue-se a transmissão do endereço de destino, que dependendo da implementação pode ser composto por sete ou dez *bit,* mas o *bit* mais significativo é sempre transmitido primeiro.

- O bloco seguinte é constituído por um *bit* que transmite ao destinatário que a trama é do tipo de leitura ou de escrita, sob o ponto de vista do *master*.

- Seguem-se os *bits* de informação.
- A trama termina com a condição de fim.

A cada bloco de informação e a cada conjunto de oito *bits*, quem estiver a recebe informação, o *master* ou o *slave*, tem o uso do barramento para transmitir um *bit* de reconhecimento de dados recebidos. De notar, Figura 224, que nas operações de leitura, quando do *master* não pretender receber mais informação, no fim do pacote de oito *bits* envia ao *slave* no bloco de reconhecimento o estado negado.

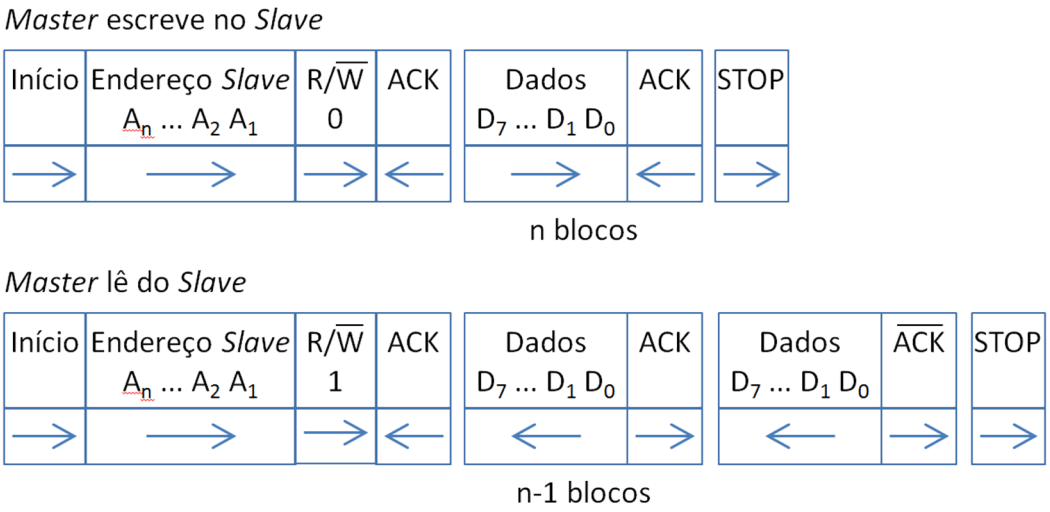


Figura 224 – Trama de leitura e escrita do protocolo I^2C, adaptado de [84].

Com este tipo de tramas, os dispositivos do barramento podem estar a operar num de quatro modos:

- *Master* a transmitir.
- *Master* a receber.
- *Slave* a transmitir.
- *Slave* a receber.

Anexo F Protocolo *Single Wire Protocol* – SWP

As soluções para atingir os requisitos de segurança na tecnologia NCF que usam como mecanismo de autenticação os cartões (U)SIM[21] - (*Universal) Subscriber Identity Module*, levantaram a necessidade de se efetuar comunicação de dados entre o modulo NCF e o respetivo cartão.

Os cartões (U)SIM fazem interface com o exterior por tomada de contactos, ver Anexo D para mais detalhes. O reduzido número contactos disponíveis levou à criação de um protocolo denominado *Single Wire Protocol* – SWP que usa apenas um fio para efetuar a comunicação. O contacto elétrico C6 do cartão (U)SIM, Figura 225, que originalmente era usado para programação da memoria EEPROM, passou também a ser usado para se efetuar a comunicação SWP, [38] e [39].

---

[21] **SIM** –*Subscriber Identity Module* é um cartão usado nos dispositivos de comunicação móvel que permite identificar e autenticar de forma segura o subscritor do serviço de comunicações GSM. Os cartões com as mesmas características mas que acedem a serviços de comunicação UTMS denominam-se USIM – *Universal Subscriber Identity Module*.

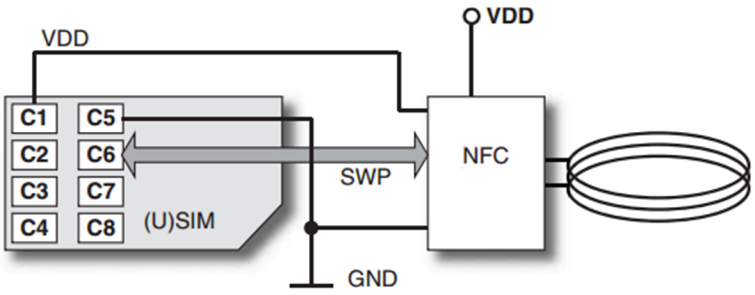


Figura 225 – Diagrama de ligações do protocolo SWP, [38].

O SWP é um protocolo série, bidirecional com arquitetura *master-slave*. Nas comunicações entre o NCF e o cartão, modulo NCF assume sempre o papel de *master* e o cartão o papel de *slave*.

A bidirecionalidade é implementada sobre o mesmo fio da seguinte forma: o cartão envia a sua informação com modelação de corrente e recebe dados através de deteção de sinais modulados em tensão.

Quando o *master* envia um estado lógico "0" ou "1", define um nível de tensão na linha de comunicações que é interpretada pelo *slave*, O *slave* por sua vez no nível de tensão definido pelo master pode deixar passar corrente para transmitir o estado lógico alto ou não deixar passar corrente transmitindo um estado lógico baixo, esta codificação é interpretada pelo *master* como a comunicação enviada pelo *slave.* Para suportar estas funcionalidades o microcontrolador do cartão está equipado com um periférico específico para trabalhar com estes níveis de tensão e corrente, Figura 227.

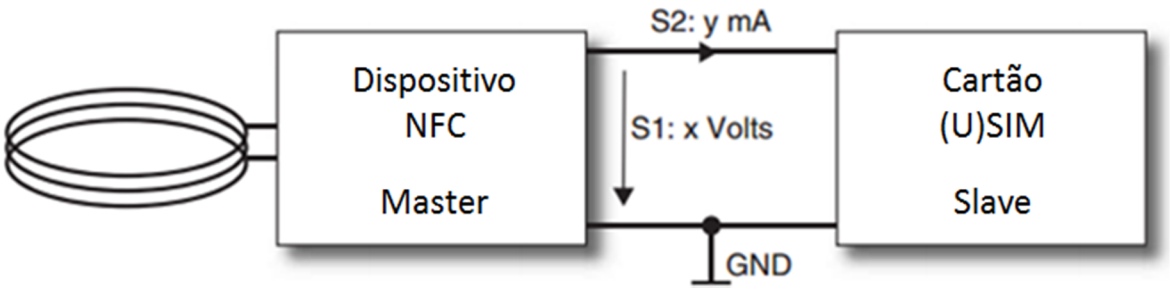


Figura 226 – Comunicação via protocolo SWP, [38].

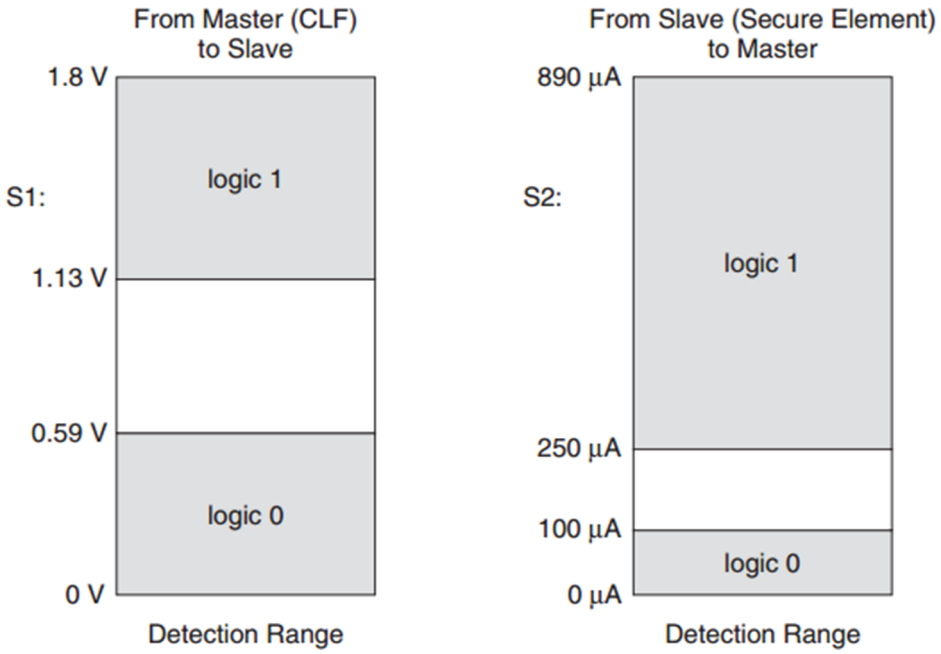


Figura 227 – Níveis de tensão e corrente dos estado lógicos do protocolo SWP, [38].

Anexo G Técnicas de representação e modelação de aplicações de *software*

O desenvolvimento de projetos, nomeadamente de *software*, nas suas diferentes etapas de implementação requer que os sistemas sejam representados em modelos simples de entender e que realcem apenas os pontos que se pretendem abordar ocultando os detalhes não relevantes.

Os modelos usam conceitos abstratos e podem ser apresentados em vários formatos como: representações matemáticas, diagramas, esquemas, plantas, etc, não podendo normalmente ser considerados representações completas de todo um sistema. Em engenharia de *software*, usam-se modelos como representações simplificadas do mundo real, sobre as seguintes abordagens, [98]:

- Visualização: modelos que mostram como o sistema é ou como o sistema vai ser;
- Especificação: modelos que apresentam a estrutura e/ou o comportamento do sistema;
- Construção: modelos que apresentam as formas de implementação;
- Documentação: modelos que evidenciam as decisões de implementação durante o desenvolvimento.

Os modelos mais usados em projetos de *software* são apresentados em formado de diagramas como por exemplo: diagramas descritivos, diagramas de pacotes, diagramas de caso de uso, diagramas de classes, diagramas relacionais e diagramas de interação ou sequência.

a) *Unified Modeling Language* - UML

A *Unified Modelling Language* – UML, [95], [96], é uma linguagem que utiliza uma simbologia específica para modelar projetos de *software* e documenta-los ao longo do seu ciclo de vida usando vários tipos de diagramas para focar características especificas do projeto. Na Figura 228 é mostrado a hierarquia de diagramas UML.

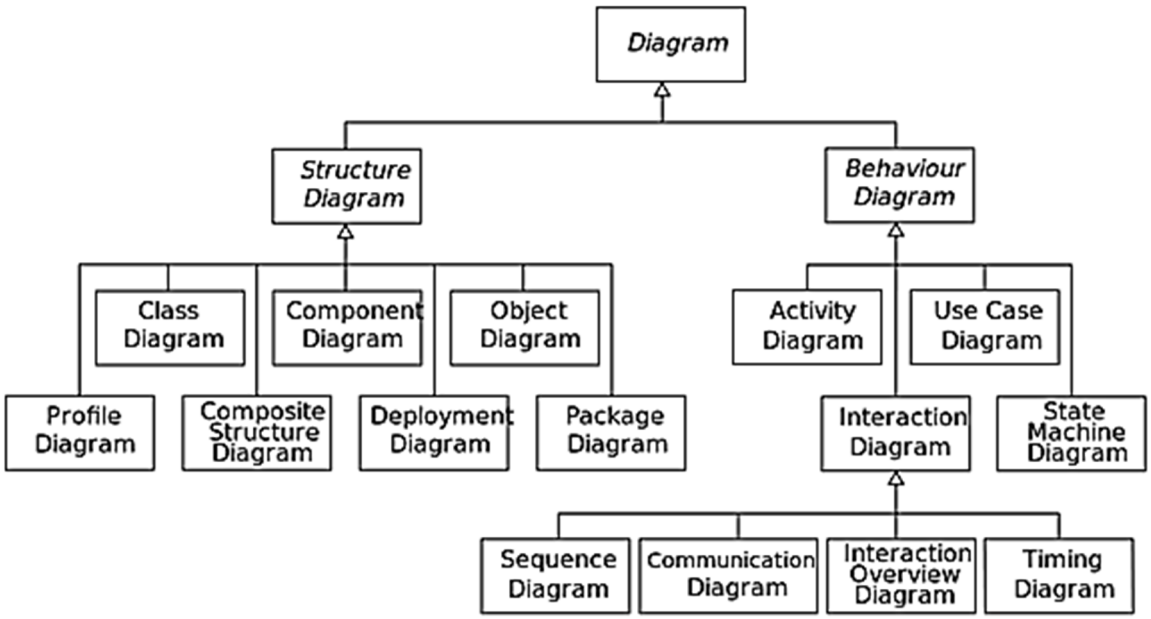


Figura 228 – Tipos de diagramas especificados pela UML, [99].

No projeto de *software* apresentado neste trabalho, usam-se diagramas com notação UML, para descrever os conceitos do projeto, nomeadamente os diagramas apresentados nos pontos seguintes.

b) Diagramas de pacotes

Os diagramas de pacote são representações que agrupam funcionalidades. A entidade fundamental deste tipo de diagramas é o elemento pacote, Figura 229, que tem um nome e que representa o conjunto de funcionalidades.

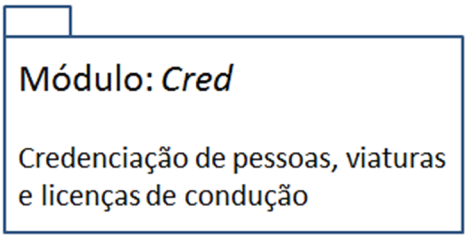


Figura 229 – Diagramas de pacotes – Pacote.

Os pacotes podem estar relacionados entre si: uns pacotes contendo outros pacotes, por acesso a funcionalidades de outros pacotes e por agrupamento das funcionalidades de outros pacotes, [97].

O diagrama da Figura 102, é um exemplo de uso de diagramas de pacotes em que o pacote "Plataforma de credenciação", contem os pacotes dos módulos aplicacionais a criar.

c) Diagramas de caso de uso

Os diagramas de caso de uso são representações que descrevem funcionalidades e requisitos de um sistema. Os diagramas são compostos por blocos, desenhados em elipses que apresentam a funcionalidade – caso de uso, e por atores que interagem com as funcionalidades, as interações são representadas por linhas [97]. O diagrama mostrado na Figura 230 é um exemplo de diagrama de caso de uso.

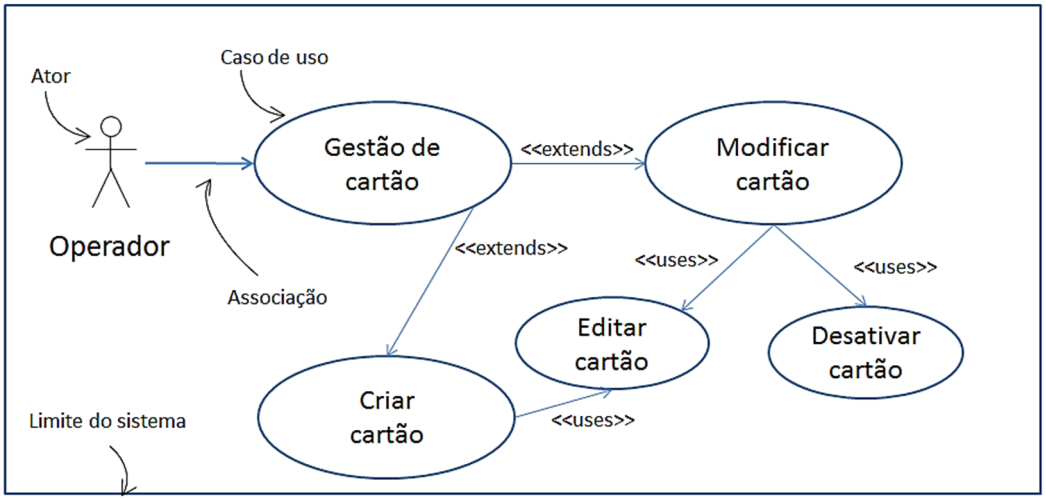


Figura 230 – Exemplo de diagrama de caso de uso.

Entre os casos de uso podem haver relações do tipo *include* ou do tipo *extended*, [95]. A relação do tipo *include* aplica-se quando se pretende representar um comportamento comum a vários casos de uso, nesta situação, cria-se um caso de uso que represente o comportamento comum que é utilizado pelos outros caso de uso, evitando a repetição. A relação do tipo *extended* aplica-se quando se pretende implementar variações ou acréscimos a um comportamento, nesta situação, cria-se um caso de uso para o "comportamento normal" e cria-se casos de uso para as variações que são usadas pelo primeiro caso de uso.

Os diagramas apresentados na Figura 106 e na Figura 107 são exemplos de diagramas de caso de uso. O diagrama da Figura 103 é um exemplo de diagrama de caso de uso onde se incluem pacotes e relações entre atores.

d) Diagramas de classes

As classes são modelos de conceitos que representam "coisas" físicas ou lógicas e são compostas por informação – atributos ou propriedades e são compostas por métodos que executam operações sobre os seus atributos, Figura 231.

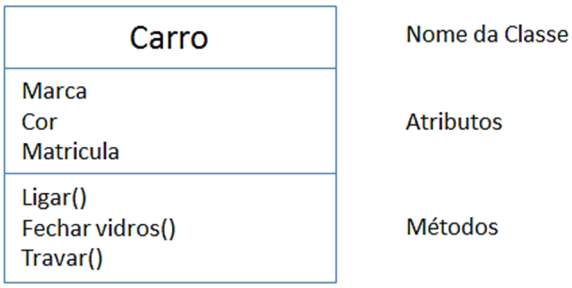


Figura 231 – Exemplo de representação de uma classe.

Um diagrama de classes, além das classes, representa as relações entre classes que podem ser de vários tipos, [97], [96]:

- Associação: representa a ligação física ou concetual entre duas classes, por exemplo a classe pessoa tem uma associação de posse com objetos da classe cartão, Figura 232.

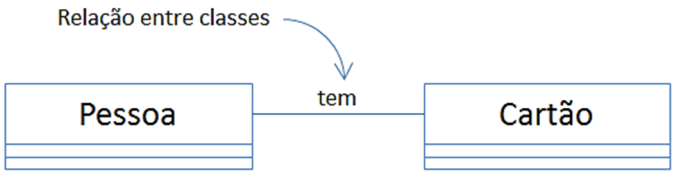


Figura 232 – Exemplo de representação de relação entre classes.

- Composição: relação em que uma classe é composta por outras no sentido em que a primeira contem a segunda e a segunda não tem existência independente da primeira, Figura 233.

Figura 233 – Exemplo de representação de relação de composição.

- Agregação: quando os objetos de uma classe são parte de um todo, composto por outra classe, um objeto "Equipa", agrega objetos "Jogador". Neste tipo de relação o objeto "Jogador", tem uma existência própria, independente da equipa, Figura 234.

Figura 234 – Exemplo de representação de relação de agregação.

- Dependência: quando a alteração de uma classe altera as características da classe de que lhe depende.

- Generalização – especialização: é uma relação hierárquica de classes, em que num sentido da hierarquia o contexto é mais generalista e no outro sentido é mais especializado sobre determinada faceta.

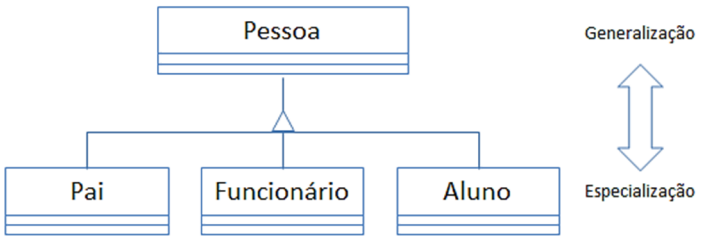


Figura 235 – Exemplo de representação de relação generalização-especialização.

- Classes a efetuar a relação entre outras classes: quando a relação em si, tem características mais complexas com dados e métodos. Por exemplo a relação entre a entidade "Pessoa" e a entidade "Perfil", tem por exemplo a data de início de atribuição do perfil, tem a data de validade, tem associados mecanismos de identificação e de autenticação. E por isso a relação entre as duas entidades também pode ser implementada à custa de uma classe de relação, constituído um relação ternária.

- Multiplicidade: a multiplicidade caracteriza a "quantidade" de relações que uma classe tem com outra, por exemplo um carro tem uma relação de quatro com o objeto pneu, uma equipa tem uma relação com um ou mais jogadores e cada jogador pode estar relacionado com uma ou mais equipas. As multiplicidades mais usuais são:

  0, 1- zero ou uma relação.

  1 – Exatamente uma relação.

0, *- zero ou mais relações.

1, *- Uma ou mais ou mais

As relações de multiplicidade são marcadas no diagrama junto à classe que goza dessa multiplicidade. No exemplo apresentado na Figura 236, o diagrama indica que, os objetos da classe pessoa podem ter a posse de um ou mais objetos da classe cartão, e que os objetos da classe cartão apenas podem pertencer a um objeto da classe pessoa.

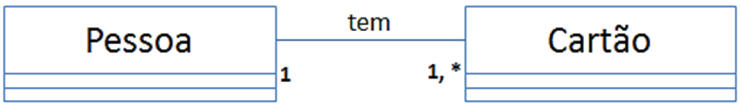


Figura 236 – Exemplo de representação de relação de multiplicidade.

e) Diagramas de interações

Os diagramas de interações são representações que apresentam as comunicações e interações entre objetos, [97]. Estes diagramas apresentam os objetos e a sequência de mensagens entre eles. O diagrama mostrado na Figura 117 é um exemplo de diagrama de interação.

f) Diagramas descritivos

Os diagramas descritivos são representações livres que não seguem a terminologia UML e cuja funcionalidade é a de representação gráfica de uma situação ou realidade. Estes digramas são compostos por:

- Caixas de texto que descrevem uma operação ou um estado;
- Linhas com setas que definem o fluxo de ações, informação ou sequências de estados;

- Texto, junto a linhas ou caixas que acrescentam informação sobre o que se pretende ilustrar.

Os diagramas apresentados na Figura 83, Figura 84 e Figura 85, são exemplos de diagramas descritivos.

Anexo H *SQL Server*

Este anexo apresenta detalhes relacionadas com a ferramenta *SQL Server*, que de alguma forma foram usados neste projeto. As questões abordadas são classificadas em duas vertentes:

- Explicações sobre de recursos.
- Trechos de código de *Transact SQL* que respondem a necessidades específicas.

a) Metadata

O *SQL* Server apresenta vários recursos onde se encontram informação sobre os objetos e estruturas que constituem as bases de dados, nomeadamente: tabelas, relações, restrições, índices, procedimentos, funções, etc. Esta informação descreve a forma como a informação é organizada, a ideia de suporte destes mecanismos é que toda a informação pode ser representada em tabelas mesmo a informação sobre as tabelas. Nos textos em idioma inglês este tipo de dados são referidos como *metadata*, isto é, dados que descrevem outros dados.

Exemplos de mecanismos que trabalham com *metadata* são os *information_scheme* e nas versões posteriores ao *SQL Server* 2005, os *Catalog views*, [92]. Estes recursos são

particularmente úteis quando se pretende analisar a estrutura de bases de dados desenvolvidas por terceiros.

Information_scheme

O termo *scheme* no *SQL Server* refere-se a uma base de dados. *Information_scheme*, [90], [92], [93] é o conjunto de tabelas em formato *view* com informação *metadata* da estrutura de uma base de dados. Exemplos de tabelas *information_scheme* são:

- *information_schema.tables*: apresenta informação sobre todas a tabelas de uma base de dados, nomeadamente o nome das tabelas e o tipo de tabela.

- *information_schema.columns*: que apresenta informação sobre todas as colunas de todas as tabelas de uma base de dados.

- *information_schema.check_constraints*: apresenta informação sobre todas a relações e restrições implementadas numa base de dados.

- *information_schema.routines*: apresenta informação sobre todas as funções e *stored procedures* definidos.

- *information_schema.parameters*: apresenta informação sobre cada parâmetro de entrada e saída das funções e *stored procedures* definidos na base de dados.

Rotinas de sistema - *metadata*

O *SQL Server*, para cada base de dados disponibiliza um conjunto alargado de rotinas – *stored procedures* e funções – *functions*, que devolvem resultados, mais trabalhados em determinado sentido, sobre as tabelas de *metadata*. Por exemplo o *stored procedure sp_columns* devolve a informação existente no *information_schema.columns*, esta rotina admite parâmetros de entrada como por exemplo a identificação da tabela onde se pretende fazer a procura, o que facilita a localização da informação pretendida.

No exemplo seguinte, Código 29, são apesentadas duas soluções equivalentes, ambas devolvem informação sobre as colunas da tabela *tblPessoa*. Na linha 1 obtém-se o resultado por execução de *stored procedure* e na linha 2 obtém-se o resultado por execução

de um comando *Transact-SQL*. Neste exemplo verifica-se que o uso do *stored procedure* torna o código mais simples e mais independente da estrutura de dados.

Código 29 – Código *Transct SQL*: para pesquisa de informação de colunas numa tabela.

```
1   EXEC sys.sp_columns @table_name = 'tblPessoa'
2   SELECT * FROM INFORMATION_SCHEMA.COLUMNS WHERE TABLE_NAME = 'tblPessoa'
```

Na Figura 237, é apresentado o resultado da execução destes comandos onde ser pode ler entre outras informações, as identificações dos campos da tabela, o tipo de dados, a dimensão dos dados, se admite valores nulos, etc.

	TABLE_NAME	COLUMN_NAME	TYPE_NAME	LENGTH	NULLABLE
1	tblPessoa	Id	varchar	15	0
2	tblPessoa	RefTipoId	bigint	8	0
3	tblPessoa	ValidadeDocId	datetime	16	0
4	tblPessoa	Nome	varchar	100	0
5	tblPessoa	DataNascimento	datetime	16	0
6	tblPessoa	RefNacionalidade	bigint	8	0
7	tblPessoa	FiliacaoMaterna	varchar	100	0
8	tblPessoa	FiliacaoPaterna	varchar	100	0
9	tblPessoa	Foto	image	2147483647	1
10	tblPessoa	PalavraChave	varchar	20	1
11	tblPessoa	RefEntidade	varchar	9	0
12	tblPessoa	RefServico	bigint	8	1
13	tblPessoa	RefFuncao	bigint	8	1

Figura 237 – Exemplo da informação de uma tabela.

b) Lista de tabelas e funções de uma base de dados

Usando os recursos *information_schema* podemos chegar ao conhecimento do número de tabelas e de funções de uma base dados, por exemplo usando os trechos os *Transct-SQL* apresentados no bloco Código 30, que executados sobre na base de dados do *Pro-Watch* apresenta os resultados mostrados na Figura 238, onde podemos verificar que a base de dados tem quinhentas e dez tabelas.

Código 30 – Código *Transct SQL*: Lista de tabela e funções.

```
1   SELECT * FROM INFORMATION_SCHEMA.TABLES WHERE TABLE_TYPE = 'base table'
    ORDER BY TABLE_NAME

2   SELECT * FROM INFORMATION_SCHEMA.TABLES WHERE TABLE_TYPE = 'view'
    ORDER BY TABLE_NAME

3   SELECT * FROM INFORMATION_SCHEMA.ROUTINES  ORDER BY ROUTINE_NAME
```

Neste trecho de código notar os seguintes detalhes:

- No *information_schema* a referência `TABLES` permite aceder a informação sobre tabelas e a referência `ROUTINES` permite aceder a informação sobre as funções: *stored procedures* e *functions*.

- Relativamente à informação de tabelas, existem dois tipos de tabelas: o tipo `base table` que representa as tabelas propriamente ditas e o tipo `view` que representa as *views* – tabelas apenas de leitura.

Results | Messages

	TABLE_CATALOG	TABLE_SCHEMA	TABLE_NAME	TABLE_TYPE
503	PWNT	dbo	WORKSP_EV_LOG	BASE TABLE
504	PWNT	dbo	WRKST	BASE TABLE
505	PWNT	dbo	WRKST_A	BASE TABLE
506	PWNT	dbo	WRKST_E	BASE TABLE
507	PWNT	dbo	WRKST_HG_CONF...	BASE TABLE
508	PWNT	dbo	WRKST_I	BASE TABLE
509	PWNT	dbo	WRKST_M	BASE TABLE
510	PWNT	dbo	WRKST_R	BASE TABLE

	TABLE_CATALOG	TABLE_SCHEMA	TABLE_NAME	TABLE_TYPE
487	PWNT	dbo	VW_V5_PORTAL_OBJECTS	VIEW
488	PWNT	dbo	WATCHTOWER_TIMER_DET	VIEW
489	PWNT	dbo	WRKST_INTERCOM	VIEW
490	PWNT	dbo	WRKST_LOGICAL	VIEW
491	PWNT	dbo	WRKST_LOGICAL_EVENTS	VIEW
492	PWNT	dbo	WRKST_MONITOR	VIEW
493	PWNT	dbo	WRKST_MONITOR_1	VIEW
494	PWNT	dbo	WRKST_MONITOR_2	VIEW

	SPECIFIC_CATALOG	SPECIFIC_SCHEMA	SPECIFIC_NAME	ROUTINE_CATALOG	ROUTINE_SCHEMA	ROUTINE_NAME
285	PWNT	dbo	V5_LAST_MESSA...	PWNT	dbo	V5_LAST_MESSAGE_TIME
286	PWNT	dbo	V5_PANEL_MOVE	PWNT	dbo	V5_PANEL_MOVE
287	PWNT	dbo	V5_PANEL_VALID	PWNT	dbo	V5_PANEL_VALID
288	PWNT	dbo	VAL_PANEL_MOVE	PWNT	dbo	VAL_PANEL_MOVE
289	PWNT	dbo	VISTA_PANEL_M...	PWNT	dbo	VISTA_PANEL_MOVE
290	PWNT	dbo	VISTA_PANEL_VA...	PWNT	dbo	VISTA_PANEL_VALID
291	PWNT	dbo	WATCHTOWER_...	PWNT	dbo	WATCHTOWER_PANEL_MOVE
292	PWNT	dbo	WATCHTOWER_...	PWNT	dbo	WATCHTOWER_PANEL_VALID
293	PWNT	dbo	WRKST_UNION	PWNT	dbo	WRKST_UNION

Figura 238 – *Transct SQL*: Lista de tabelas e funções.

c) Número de registos das tabelas

Quando se está a analisar a interação entre as funcionalidades de uma interface gráfica e as suas repercussões na respetiva base de dados, é muito útil conseguir contar os registos das tabelas para podermos comparar o número de registos antes e depois da execução da funcionalidade e desta forma perceber que tabelas estão envolvidas numa operação.

O código *Transct-SQL* apresentado no bloco Código 31, faz uma listagem de todas as tabelas de uma base de dados e apresenta na coluna *Nregistos* o número de registos dessa tabela. Executando este código sobre a base de dados inicial do *Pro-Watch*, obtém-se o resultado mostrada na Figura 239.

Código 31 – Código *Transct SQL*: Número de registos das tabelas de uma base de dados.

```
1   SELECT SC.NAME +'.'+ TA.NAME NomeTabela,SUM(pa.rows) Nregistos
    FROM sys.tables TA INNER JOIN sys.partitions PA ON PA.OBJECT_ID =
    TA.OBJECT_ID INNER JOIN sys.schemas SC ON TA.schema_id = SC.schema_id
    WHERE TA.is_ms_shipped = 0 AND PA.index_id IN (1,0)
    GROUP BY SC.NAME,TA.NAME
    ORDER BY NomeTabela
```

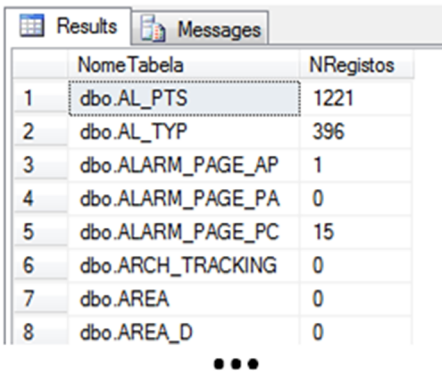


Figura 239 – *Transct SQL*: Número de registos das tabelas de uma base de dados.

d) Tabelas com *triggers* implementados

Os *triggers* são blocos de código [94], que se executam quando ocorre um evento de alteração de dados numa tabela.

O *SQL Server*, permite três tipos de eventos que despoletam a execuação de um *trigger*: inserir de um registo, alteração de um campo ou apagar de um registo.

Cada bloco de código do tipo *trigger* pode ser associado a um ou mais tipo de eventos. A indicação do tipo de evento que faz despoletar a execução do código pode ser: `AFTER UPDATE` que executa o *trigger* quando a informação é alterada, `AFTER INSERT` que executa o *trigger* após a introdução de um novo registo ou `AFTER DELETE` que executa o código após a remoção de um registo. O *SQL Server* permite usar qualquer associação destes três tipos de eventos para despoletar a execução do *trigger*. Por exemplo, se pretendermos executar o mesmo código quando se insere, apaga e altera um registo deve-se usar a instrução: `AFTER INSERT, UPDATE, DELETE`.

Ao contrário dos *stored procedures*, os *triggers* não podem ser executados diretamente pelo utilizador e não aceitam nem devolvem parametros.

Para ser conhecer os *triggers* implementados numa base de dados e as tabelas a que estão associados pode-se usar o comando apresentados no Código 32. Que executado sobre a base de dados Credenciação apresenta o resultado mostrado na Figura 240.

Código 32 – Código *Transct SQL*: *Triggers* associados a tabelas.

```
1   SELECT DISTINCTo.[name] AS [Table], tr.[name] AS [Trigger]
    FROM [sysobjects] o JOIN [sysobjects] tr ON o.[id] = tr.[parent_obj]
    WHERE tr.[type] = 'tr'
    ORDER BY [Table], [Trigger]
```

	Table	Trigger
1	tblLicencaConducao	trgLicencaConducaoApaga
2	tblLicencaConducao	trgLicencaConducaoCria
3	tblPessoa	trgPessoaAtualiza
4	tblPessoa	trgPessoaCria
5	tblPortaria	trgPortariaApaga
6	tblPortaria	trgPortariaAtualiza
7	tblPortaria	trgPortariaCria
8	tblValidadeAcessoPessoaCor	trgPessoaAcessoCorAltera
9	tblValidadeAcessoPessoaCor	trgPessoaAcessoCorApaga
10	tblValidadeAcessoPessoaLetra	trgPessoaAcessoLetraAltera

Figura 240 – *Transct SQL*: *Triggers* associados a tabelas.

Quando a propriedade da base de dados *Allow Trigger to Fire Others* está ativa a alteração de dados efetuada por *triggers* despoleta a execuação de outros *triggers* criando uma estrutura de *triggers* encadeados que pode ir até trinta e dois níveis.

e) Criação de ligação de servidor entre uma aplicação *VB* e o *SQL Server*

No ambiente *Visual Studio – Visual Basic*, a ligação da aplicação a uma base de dados residente num servidor *SQL Server* faz-se executando os passos seguintes.

1. Menu "View – Server Explorer", Figura 241.

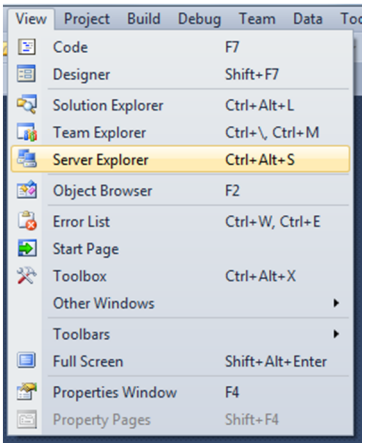


Figura 241 – VB - Ligação a um servidor de dados.

2. Surge a janela mostrada na Figura 242, onde é necessário configurar:
    a. O tipo de servidor, neste caso: "Microsoft SQL Server (Sql Client)".
    b. O endereço do servidor de dados neste caso: "asc-sfmp\sql2008express".
    c. Selecionar o nome da base de dados, neste caso "Credenciação".
    d. Testar se as configurações são válidas: botão "Test Connection".

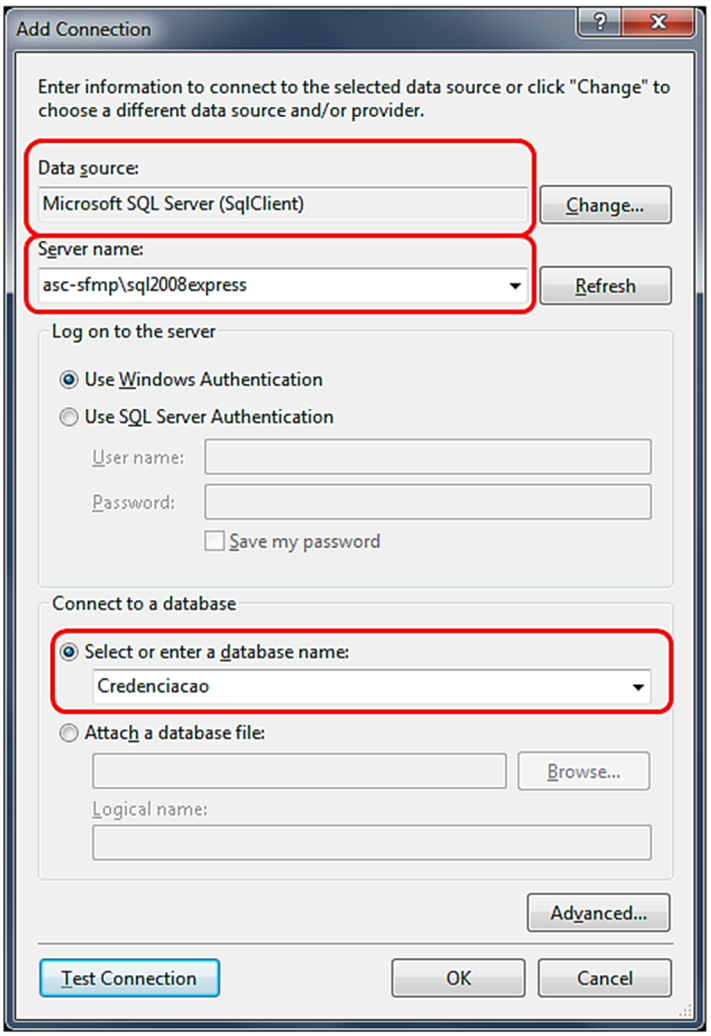


Figura 242 – VB - Ligação a um servidor de dados, configuração do acesso.

A execução destes passos cria uma ligação à base de dados definida, num servidor específico que pode ser verificada na janela "Server Explorer" do ambiente de trabalho do *Visual Studio*, Figura 243. Nessa janela é mostrada a ligação ao servidor "asc-sfmp\sql2008express.Credenciacao.dbo". A base de dados é representada numa estrutura em árvore onde podemos aceder aos componentes da base de dados, nomeadamente aos *views* e aos *stored procedures.*

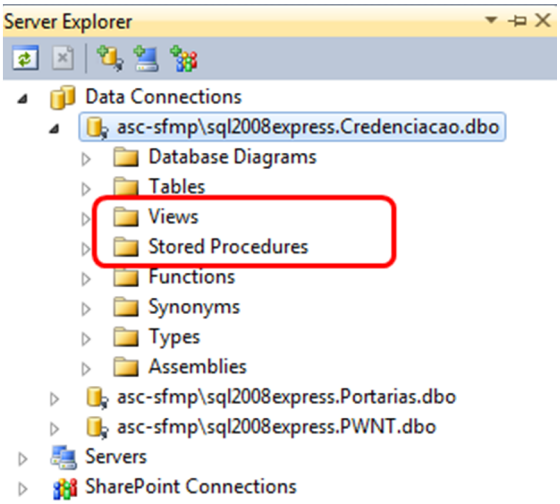


Figura 243 – VB – Janela *Server Explorer*.

Anexo I Tabelas da base de dados

Este anexo faz a apresentação de todas as tabelas que compõem a base de dados da plataforma e da portaria segundo o modelo de relacional de implementação descrito na secção 3.4.3.

Inicialmente é apresenta-se uma lista das tabelas e depois são mostrados os detalhes de cada tabela: são indicados os nomes das tabelas e dos campos para implementação no *SQL Server*, assim como uma descrição da funcionalidade de cada campo e indica-se qual(ais) a(s) chave(s) primária(s) das tabelas.

Tabela 25 – Lista das tabelas da base de dados Credenciação

Tabela	Descrição
tblCaminhoFicheiros	Para guardar os caminhos dos locais de armazenamento dos ficheiros de anexo.
tblCartao	Para conter informação dos cartões permanentes e temporários.
tblCartaoPassageiro	Para conter informação sobre os cartões de acesso concedidos a passageiros.
tblCartaoPontual	Para conter informação sobre cartões de acesso pontual.
tblCartaoPontualAcesso	Para conter informação sobre as permissões acesso dos

	cartões pontuais.
tblCartaoPontualAnexo	Para conter informação sobre os anexos associados a cartões pontuais.
tblCartaoPontualPortaria	Para conter informação sobre as portarias que permitem o acesso a um cartão pontual.
tblDicAcessoCor	Dicionário de cores que codificam níveis de acesso.
tblDicAcessoLetra	Dicionário de letras que codificam níveis de acesso.
tblDicAplicacao	Dicionário de informações sobre as aplicações que constituem a plataforma.
tblDicClassificacaoTipoAnexo	Dicionário da classificação dos tipos de anexos.
tblDicClassificacaoTipoLog	Dicionário de classificação dos tipos de registos de eventos.
tblDicCodigosErro	Dicionário de códigos de erro
tblDicCombustivel	Dicionário de tipos de combustível usado nas viaturas.
tblDicCompanhiaAerea	Dicionário de companhias aéreas.
tblDicDocumentoId	Dicionário de documentos de identificação aceite.
tblDicEstadoCartao	Dicionário do estado processual de cartões de acesso
tblDicFuncao	Dicionário de funções laborais.
tblDicInfracao	Dicionário de infrações associadas à vertente de segurança.
tblDicInfracaoConducao	Dicionário de infrações associadas à vertente de condução de viaturas em áreas reservadas.
tblDicPais	Dicionário de países.
tblDicPenalidade	Dicionário de penalidades associadas à vertente de segurança.
tblDicPenalidadeConducao	Dicionário de penalidades associadas à vertente de condução de viaturas em áreas reservadas.
tblDicPerfil	Dicionário de perfil de utilizador
tblDicServico	Dicionário de serviços associados a pessoas.
tblDicServicoViatura	Dicionário de serviços associados a viaturas
tblDicTipoAnexo	Dicionário de tipos de anexos
tblDicTipoCartaConducao	Dicionário de tipos de carta de condução emitidas pelos países.

tblDicTipoLicencaConducao	Dicionário de tipos de licença de condução para áreas reservadas.
tblDicTipoLog	Dicionário de tipos de registos de eventos.
tblDicTipoPagamento	Dicionário de tipo de pagamentos de custos associados a credenciação.
tblDicTipoVeiculo	Dicionário de tipos de veículos.
tblDicZonasViatura	Dicionário de zonas de operação de viaturas.
tblEmail	Para conter informação sobre e-mail de pessoas e empresas.
tblEntidade	Para conter informação sobre empresas.
tblEntidadeAnexo	Para conter informação sobre anexos associados a empresas.
tblEntidadePontual	Para conter informação sobre empresas de relativas a visitas pontuais.
tblInfracao	Para conter informação de infrações associadas à vertente de segurança.
tblInfracaoConducao	Para conter informação de infrações associadas à vertente de condução de viaturas.
tblLicencaConducao	Para conter informação das licenças de condução.
tblLicencaConducaoRenovacao	Para conter informação de renovação de licenças de condução.
tblLog	Tabela de registos de eventos gerais.
tblLogAcessoPortaria	Tabela de registos de eventos de acesso a portarias.
tblLogInfEntidade	Tabela de registos de eventos relacionados com empresas.
tblLogInfPassageiro	Tabela de registos de eventos associados a passageiros.
tblLogInfPessoa	Tabela de registos de eventos associados às pessoas.
tblLogInfPortaria	Tabela de registos de eventos associados às portarias.
tblLogInfViatura	Tabela de registos de eventos associados às viaturas.
tblLogInfVisitante	Tabela de registos de eventos associados a visitantes.
tblLogLogin	Tabela de registos de eventos associados a entradas e saídas de uso de aplicações.
tblLogPassagemPortaria	Tabela de registos de eventos associados à apresentação de cartões de acesso nas portarias.

tblLoteCartoes	Para conter informação sobre os lotes de cartões.
tblMorada	Para conter informação sobre as moradas de pessoas e empresas.
tblPassageiro	Para conter informação sobre passageiros que entram em áreas reservadas.
tblPessoa	Para conter informação sobre pessoas registadas no sistema e portadores de cartões de acesso.
tblPessoaAnexo	Para conter informação sobre ficheiros anexos associados a pessoas
tblPortaria	Para conter informação sobre portarias.
tblTelefone	Para conter informação sobre telefones de pessoas e empresas.
tblValidadeAcessoPessoaCor	Para conter informação sobre os acessos de pessoas, codificados por cores.
tblValidadeAcessoPessoaLetra	Para conter informação sobre os acessos de pessoas, codificados por letras.
tblValidadeAcessoPortaria	Para conter informação sobre os acessos de portarias, codificados por letras.
tblValidadePerfil	Para conter informação sobre o perfil de pessoas registadas no sistema.
tblViatura	Para conter informação sobre viaturas.
tblViaturaAnexo	Para conter informação sobre ficheiros anexos associados a viaturas.
tblViaturaRevalidacao	Para conter informação sobre as revalidações periódicas de licenças de viaturas.
tblVisitante	Para conter informação sobre as pessoas a que são atribuídos cartões pontuais.
tblVisitanteAnexo	Para conter informação sobre ficheiros anexos associados a visitantes.
tblZonasViatura	Para conter informação sobre zonas de operação de viaturas.

Tabela 26 – Lista de tabelas da base de dados Portarias

Tabela	Descrição
tblCaminhoFicheiros	Para guardar os caminhos dos locais de armazenamento dos ficheiros de anexo.
tblLog	Tabela de registos de eventos gerais.
tblLogLoginPortaria	Tabela de registos de eventos de entrada e saída de uso da aplicação PORT.
tblLogPassagemPortaria	Tabela de registos de eventos de apresentação de cartões de acesso nas portarias.
tblPessoa	Para conter informação sobre os portadores de cartões de acesso.
tblPortaria	Para conter informação sobre as portarias.

a) Tabelas de registo de dados relacionados com pessoas e entidades

Tabela 27 – Tabela da base de dados – tblPessoa.

tblPessoa		Para conter a informação pessoal unitária, relativas as pessoas registadas no sistema.	
Id	String [12]	Número do documento de identificação: Passaporte, Cartão do cidadão, etc.	Chave primária
Nome	String [100]	Nome da pessoa	
Data Nascimento	Date	Data de nascimento	
RefNacionalidade	Long	Nacionalidade	
Filiação Materna	String [100]	Nome da mãe	
Filiação Paterna	String [100]	Nome do pai	
RefTipoID	Long	Referencia ao dicionário do tipo de documento usado para identificação	
ValidadeID	Date	Validade do documento de identificação	
Foto	Object	Foto	
Palavra-chave	String [20]	Palavra-chave	
RefEntidade	Long	Referencia à entidade que representa	
RefServico	Long	Referencia ao dicionário do serviço a que faz assistência	
RefFuncao	Long	Referencia ao dicionário da função que desempenha	
DataRegisto	Date	Data em que a pessoa é registada	
ObjetosProibidos	Boolean	Marcador que indica se a pessoa pode transportar objetos proibidos.	
Notas	Text	Campo de texto livre para informações complementares	

Tabela 28 – Tabela da base de dados – tblEntidade.

tblEntidade		Para conter a informação sobre as empresas que as pessoas que acedem a áreas reservadas representam.	
NIF	Long	Número de identificação fiscal.	Chave primária
Nome	String [50]	Nome da empresa	
RefSignatario	String [12]	Pessoa registada que representa a empresa	
RefNacionalidade	String [20]	Nacionalidade	
RefTipoPagamento	Long	Referencia ao dicionário de tipo de pagamento, sobre os custos de credenciação	
DataRegisto	Date	Data em que a pessoa é registada	
Notas	Text	Campo de texto livre para informações complementares	

Tabela 29 – Tabela da base de dados – tblEMail.

tblEMail		Para conter contacto de correio eletrónico	
Id	Long	Numero sequencial que identifica o registo	Chave primária
RefDono	String [50]	Referência a quem pertence o registo: pessoa ou empresa	
Pessoa	Boolean	Marcador que assume o valor “Verdade” se o dono o registo for uma pessoa e assume o valor “falso” se o dono for um empresa	
Endreço	String [70]	Endereço de correio eletrónico	
Ativo	Boolean	Marcador que assume o valor “Verdade” se o registo é usável e assume o valor “falso” se o registo for histórico	
DataRegisto	Date	Data em que o registo foi efetuado	
Notas	Text	Campo de texto livre para informações complementares	

Tabela 30 – Tabela da base de dados – tblTelefone

tblTelefone		Para conter contactos telefónicos	
Id	Long	Numero sequencial que identifica o registo	Chave primária
RefDono	String [50]	Referência a quem pertence o registo: pessoa ou empresa	

Pessoa	Boolean	Marcador que assume o valor “Verdade” se o dono do registo for uma pessoa e assume o valor “falso” se o dono for um empresa
Telefone	Num [15]	Numero de telefone
Ativo	Boolean	Marcador que assume o valor “Verdade” se o registo é usável e assume o valor “falso” se o registo for histórico
DataRegisto	Date	Data em que o registo foi efetuado
Notas	Text	Campo de texto livre para informações complementares

Tabela 31 – Tabela da base de dados – tblMorada

tblMorada		Para conter moradas	
Id	Long	Numero sequencial que identifica o registo	Chave primária
RefDono	String [50]	Referência a quem pertence o registo: pessoa ou empresa	
Pessoa	Boolean	Marcador que assume o valor “Verdade” se o dono do registo for uma pessoa e assume o valor “falso” se o dono for um empresa	
Morada	String [150]	Descritivo da morada	
Localidade	String [50]	Localidade	
Código-Postal	num [7]	Código-Postal	
País	String [25]	País	
Ativo	Boolean	Marcador que assume o valor “Verdade” se o registo é usável e assume o valor “falso” se o registo for histórico	
DataRegisto	Date	Data em que o registo foi efetuado	
Notas	Text	Campo de texto livre para informações complementares	

Tabela 32 – Tabela da base de dados – tblDicServico

tblDicServico		Dicionário para conter listas de serviços	
Id	Long	Numero sequencial que identifica o registo	Chave primária
Servico	String [50]	Nome/Descrição do serviço: Manutenção, vigilância, limpeza, etc.	
Notas	Text	Campo de texto livre para informações complementares	

Tabela 33 – Tabela da base de dados – tblDicFuncoes

tblDicFuncao		Dicionário para conter listas de funções	
Id	Long	Numero sequencial que identifica o registo	Chave primária
Funcao	String [50]	Nome/Descrição da função: Supervisor, operador, etc.	
Notas	Text	Campo de texto livre para informações complementares	

Tabela 34 – Tabela da base de dados – tblDicTipoPagamento

tblDicTipoPagamento		Dicionário para conter listas de formas de pagamento dos custos associadas às credenciações.	
Id	Long	Numero sequencial que identifica o registo	Chave primária
TipoPagamento	String [50]	Nome do tipo de pagamento: Isento, Numerário, Faturação, etc.	
Notas	Text	Campo de texto livre para informações complementares	

Tabela 35 – Tabela da base de dados – tblDicDocumentoID

tblDicDocumentoID		Dicionário para conter listas de tipos de documentos de identificação.	
Id	Long	Numero sequencial que identifica o registo	Chave primária
DocumentoID	String [50]	Nome do tipo de documento de identificação: Passaporte, Cartão d cidadão, etc.	
Notas	Text	Campo de texto livre para informações complementares	

Tabela 36 – Tabela da base de dados – tblPessoaAnexo

tblPessoaAnexo		Para conter documentos anexados à entidade pessoa	
Id	Long	Numero sequencial que identifica o registo	Chave primária
RefPessoa	String [12]	Referencia à pessoa a que este registo pertence	
Anexo	String [200]	Campo para conter o nome do documento	

RefTipoAnexo	Long	Classificação do tipo de anexo
DataRegisto	Date	Data de criação do registo
Notas	Text	Campo de texto livre para informações complementares

Tabela 37 – Tabela da base de dados – tblEntidadeAnexo

tblEntidadeAnexo		Para conter documentos anexados às entidades	
Id	Long	Numero sequencial que identifica o registo	Chave primária
RefEntidade	String [12]	Referencia à pessoa a que este registo pertence	
Anexo	String [200]	Campo para conter o nome do documento	
RefTipoAnexo	Long	Classificação do tipo de anexo	
DataRegisto	Date	Data de criação do registo	
Notas	Text	Campo de texto livre para informações complementares	

b) Tabelas de registo de dados relacionados com cartões de acesso

Tabela 38 – Tabela da base de dados – tblDicEstadoCartão

tblDicEstadoCartao		Dicionário para conter a lista de possíveis estados de operacionalidade de cartões.	
Id	Long	Numero sequencial que identifica o registo	Chave primária
Estado	String [50]	Estado do cartão: Em processamento, Em renovação etc.	
Notas	Text	Campo de texto livre para informações complementares	

Tabela 39 – Tabela da base de dados – tblCartão

tblCartao		Dicionário para conter informação dos cartões atribuídos a pessoas.	
Id	Long	Número do cartão	Chave primária
RefPessoa	String [12]	Referencia ao dono do cartão	
DataInicio	Date	Data de atribuição do cartão	
DataFim	Date	Data de validade do cartão	
Ativo	Boolean	Marcador para indicar se o cartão pode ser usado	
Permanente	Boolean	Marcador que indica se o cartão é do tipo permanente ou temporário	
RefEstado	Long	Referencia ao estado de processamento do cartão	
DataRegisto	Date	Data de criação do registo	
Company	nvarchar(128)	Referencia à entidade *company* da base de dados PWNT, que está associada ao cartão.	
Notas	Text	Campo de texto livre para informações complementares	

Tabela 40 – Tabela da base de dados – tblLoteCartões

tblLoteCartões		Dicionário para conter listas de lotes de cartões.	
Id	Long	Numero sequencial que identifica o registo	Chave primária
PrimeiroNumero	Long	Número do primeiro cartão do lote	

UltimoNumero	Long	Número do último cartão do lote
DataInicio	Date	Data de início de uso do lote
DataFim	Date	Data de fim do uso do lote
Ativo	Boolean	Marcador para indicar se o lote está ativo
DataRegisto	Date	Data de criação do registo
Notas	Text	Campo de texto livre para informações complementares

c) Tabelas de registo de dados relacionados com o perfil dos utilizadores

Tabela 41 – Tabela da base de dados – tblDicPerfil

tblDicPerfil		Dicionário para conter a lista de perfis dos utilizadores.	
Id	Long	Numero sequencial que identifica o registo	Chave primária
Perfil	String [50]	Perfil de utilizador: Administrador, Gestor, etc.	
Notas	Text	Campo de texto livre para informações complementares	

Tabela 42 – Tabela da base de dados – tblValidadePerfil

tblValidadePerfil		Dicionário para conter informação sobre os perfis atribuídos a pessoas.	
Id	Long	Numero sequencial que identifica o registo	Chave primária
RefPessoa	String [12]	Referencia ao dono do cartão	
RefPerfil	Long	Referencia ao dicionário de perfis	
DataInicio	Date	Data de atribuição do perfil	
DataFim	Date	Data de validade do perfil	
Ativo	Boolean	Marcador para indicar se o perfil está a ser usado	
DataRegisto	Date	Data de criação do registo	
Notas	Text	Campo de texto livre para informações complementares	

d) Tabelas de registo de dados relacionados com infrações de segurança

Tabela 43 – Tabela da base de dados – tblInfracao

tblInfracao		Para conter infrações associadas à vertente de segurança.	
Id	Long	Numero sequencial que identifica o registo	Chave primária
RefPessoa	String [12]	Referencia à pessoa a que este registo pertence.	
RefInfracao	Long	Referencia ao tipo de infração cometida	
DataInfracao	Date	Data da infração	
RefPenalidade	Long	Referencia ao tipo de penalidade	
DataPenalidade	Date	Data de execução da penalidade	
DataRegisto	Date	Data de criação do registo	
Notas	Text	Campo de texto livre para informações complementares	

Tabela 44 – Tabela da base de dados – tblDicInfracao

tblDicInfracao		Dicionário para conter listas de tipos de infrações associadas à vertente de segurança.	
Id	Long	Numero sequencial que identifica o registo	Chave primária
Infração	String [200]	Nome/descrição da infração.	
Notas	Text	Campo de texto livre para informações complementares	

Tabela 45 – Tabela da base de dados – tblDicPenalidade

tblDicPenalidade		Dicionário para conter listas de tipos de penalidades associadas à vertente de segurança.	
Id	Long	Numero sequencial que identifica o registo	Chave primária
Penalidade	String [200]	Nome/descrição da Penalidade: advertência, desativação de cartão 8 dias, etc.	
Notas	Text	Campo de texto livre para informações complementares	

e) Tabelas de registo de dados relacionados com permissões de acessos

Tabela 46 – Tabela da base de dados – tblDicAcessoLetra

tblDicAcessoLetra		Tabela dicionário com lista de acessos codificados em letras.	
AcessoLetra	String [1]	Nome do acesso	Chave primária
Notas	Text	Campo de texto livre para informações complementares	

Tabela 47 – Tabela da base de dados – tblDicAcessoCor

tblDicAcessoCor		Tabela dicionário com lista de acessos codificados em cores.	
Id	Long	Numero sequencial que identifica o registo	Chave primária
Acesso	String [15]	Nome do acesso	
Notas	Text	Campo de texto livre para informações complementares	

Tabela 48 – Tabela da base de dados – tblPortaria

tblPortaria		Tabela para conter informação relativa a portarias.	
Id	Long	Numero sequencial que identifica o registo	Chave primária
Portaria	String [50]	Nome da portaria	
Notas	Text	Campo de texto livre para informações complementares	

Tabela 49 – Tabela da base de dados – tblValidadeAcessoPessoaLetra

tblValidadeAcessoPessoaLetra		Tabela com registo que fazem a atribuição às pessoas de acessos codificados em letras.	
Id	Long	Numero sequencial que identifica o registo	Chave primária
RefPessoa	String [12]	Referencia à pessoa	

RefAcesso	String [1]	Referencia ao Acesso
DataInicio	Date	Data de atribuição do acesso
DataFim	Date	Data de validade do acesso
Ativo	Boolean	Marcador para indicar se o acesso está a ser usado
DataRegisto	Date	Data de criação do registo
Notas	Text	Campo de texto livre para informações complementares

Tabela 50 – Tabela da base de dados – tblValidadeAcessoPessoaCor

tblValidadeAcessoPessoaCor		Tabela com registo que fazem a atribuição às pessoas de acessos codificados em cores.	
Id	Long	Numero sequencial que identifica o registo	Chave primária
RefPessoa	String [12]	Referencia à pessoa	
RefAcesso	String [15]	Referencia ao Acesso	
DataInicio	Date	Data de atribuição do acesso	
DataFim	Date	Data de validade do acesso	
Ativo	Boolean	Marcador para indicar se o acesso está a ser usado	
DataRegisto	Date	Data de criação do registo	
Notas	Text	Campo de texto livre para informações complementares	

Tabela 51 – Tabela da base de dados – tblValidadeAcessoPortaria

tblValidadeAcessoPortaria		Tabela com registo que fazem a atribuição de acessos às portarias.	
Id	Long	Numero sequencial que identifica o registo	Chave primária
RefPortaria	Long	Referencia à portaria	
RefAcesso	String [1]	Referencia ao Acesso	
DataInicio	Date	Data de atribuição do acesso	
DataFim	Date	Data de validade do acesso	
Ativo	Boolean	Marcador para indicar se o acesso está a ser usado	
DataRegisto	Date	Data de criação do registo	
Notas	Text	Campo de texto livre para informações complementares	

f) Tabelas de registo de dados relacionados com licenças de condução

Tabela 52 – Tabela da base de dados – tblLicencaConducao

tblLicencaConducao		Tabela de registo de informação relacionada com as licenças de condução	
Id	Long	Número sequencial que identifica o registo	Chave primária
NumeroLicenca	Long	Número da licença de condução	
ValidadeLConducao	Date	Date de validade da licença de condução.	
RefPessoa	String [12]	Referencia à pessoa	
NcartaConducao	String [15]	Número da carta de condução emitida pelo país de origem do portador	
ValidadeCConducao	Date	Data de validade da carta de condução	
RefTipoCConducao	Long	Referencia ao tipo de carta de condução emitida pelo país de origem do portador	
RefTipoLConducao	Long	Referencia ao tipo de licença de condução em áreas reservadas do aeroporto	
NotaFormação	0-100	Resultado da avaliação do exame teórico e prático para atribuição de licença de condução.	
DataFormação	Date	Resultado da avaliação do exame teórico e prático para atribuição de licença de condução.	
VeiculosEspeciais	Boolean	Se a licença permite manobrar veículos especiais	
LVO	Boolean	Se a licença permite condução em condições LVO – *Low Visibility Operation*.	
LicencaPermanente	Boolean	Se a licença é do tipo permanente ou temporário.	
Ativo	Boolean	Marcador para indicar se o acesso está a ser usado	
DataRegisto	Date	Data de criação do registo	
Notas	Text	Campo de texto livre para informações complementares	

Tabela 53 – Tabela da base de dados – tblDicTipoCartaConducao

tblDicTipoCartaConducao		Tabela dicionário com lista de tipos de carta de condução emitida pelo país de origem do portador.	
Id	Long	Numero sequencial que identifica o registo	Chave primária
TipoCarta	String [15]	Nome do tipo de carta	
Notas	Text	Campo de texto livre para informações complementares	

Tabela 54 – Tabela da base de dados – tblDicTipoLicencaConducao

tblDicTipoLicencaConducao		Tabela dicionário com lista de tipos de licença de condução em áreas restritas do aeroporto.	
Id	Long	Numero sequencial que identifica o registo	Chave primária
TipoLicenca	String [15]	Nome da licença	
Notas	Text	Campo de texto livre para informações complementares	

Tabela 55 – Tabela da base de dados – tblLicencaConducaoRenovacao

tblLicencaConducaoRenovacao		Tabela com informação sobre as renovações das licenças de condução.	
Id	Long	Numero sequencial que identifica o registo	Chave primária
DataRenovacao	Date	Data da renovação	
CursoExtintores	Boolean	Se efetuou o curso de manuseamento de extintores	
CartaConducao	Boolean	Se foram verificados os dados da carta de condução e se estão atualizados	
Exames	Boolean	Se os exames de renovação foram efetuados com sucesso	
DataRegisto	Date	Data de criação do registo	
Notas	Text	Campo de texto livre para informações complementares	

Tabela 56 – Tabela da base de dados – tblInfracaoConducao

tblInfracaoConducao		Para conter infrações associadas à licença de condução.	
Id	Long	Numero sequencial que identifica o registo	Chave primária
RefPessoa	String [12]	Referencia à pessoa a que este registo pertence.	

RefInfracao	Long	Referencia ao tipo de infração cometida
DataInfracao	Date	Data da infração
RefPenalidade	Long	Referencia ao tipo de penalidade
DataPenalidade	Date	Data de execução da penalidade
DataRegisto	Date	Data de criação do registo
Notas	Text	Campo de texto livre para informações complementares

Tabela 57 – Tabela da base de dados – tblDicInfracaoConducao

tblDicInfracaoConducao		Dicionário para conter listas de tipos de infrações associadas à licença de condução.	
Id	Long	Numero sequencial que identifica o registo	Chave primária
Infração	String [200]	Nome/descrição da infração.	
Notas	Text	Campo de texto livre para informações complementares	

Tabela 58 – Tabela da base de dados – tblDicPenalidadeConducao

tblDicPenalidadeConducao		Dicionário para conter listas de tipos de penalidades associadas à licença de condução.	
Id	Long	Numero sequencial que identifica o registo	Chave primária
Penalidade	String [200]	Nome/descrição da Penalidade:.	
Notas	Text	Campo de texto livre para informações complementares	

g) Tabelas de registo de dados relacionados com viaturas

Tabela 59 – Tabela da base de dados – tblViatura

tblViatura		Tabela de registo de informação relacionada viaturas que circulam em áreas restritas	
Id	Long	Número sequencial que identifica o registo	Chave primária
Matricula	String [6]	Matrícula da viatura	
NumeroSerie	String[40]	Número de série da viatura	
MarcaModelo	String[50]	Marca e modelo da viatura	
RefCombustivel	Long	Referencia ao tipo de combustível usado pelo veículo	
DataFabrico	Date	Data de fabrico da viatura	
RefTipoVeiculo	Long	Referencia à classificação de tipo de veículo	
RefEntidade	Long	Referencia à entidade possuidora do veículo	
RefServico	Long	Referencia ao serviço normalmente prestado pelo veículo	
Distico	String[6]	Dístico identificativo da viatura	
Permanente	Boolean	Se o acesso da viatura às áreas restritas é do tipo permanente ou temporário	
Ativo	Boolean	Marcador para indicar se o acesso está a ser usado	
DataRegisto	Date	Data de criação do registo	
Notas	Text	Campo de texto livre para informações complementares	

Tabela 60 – Tabela da base de dados – tblViaturaAnexo

tblViaturaAnexo		Para conter documentos anexados às credenciações de viaturas	
Id	Long	Numero sequencial que identifica o registo	Chave primária
RefViatura	String [12]	Referencia à viatura a que este registo pertence	
Anexo	String [200]	Campo para conter o nome do documento	
RefTipoAnexo	Long	Classificação do tipo de anexo	
Notas	Text	Campo de texto livre para informações complementares	

Tabela 61 – Tabela da base de dados – tblDicCombustivel

tblDicCombustivel		Tabela dicionário com lista de tipos de combustíveis usados em viaturas.	
Id	Long	Numero sequencial que identifica o registo	Chave primária
Combustível	String [20]	Tipo de combustível	
Notas	Text	Campo de texto livre para informações complementares	

Tabela 62 – Tabela da base de dados – tblDicTipoVeiculo

tblDicTipoVeiculo		Tabela dicionário com lista de tipos de viaturas.	
Id	Long	Numero sequencial que identifica o registo	Chave primária
TipoVeiculo	String [30]	Tipo de veículo	
Notas	Text	Campo de texto livre para informações complementares	

Tabela 63 – Tabela da base de dados – tblDicServicoViatura

tblDicServicoViatura		Tabela dicionário com lista de serviços prestados por viaturas.	
Id	Long	Numero sequencial que identifica o registo	Chave primária
TipoServicoViatura	String [30]	Nome do serviço prestado pelas viaturas	
Notas	Text	Campo de texto livre para informações complementares	

Tabela 64 – Tabela da base de dados – tblZonasViatura

tblZonasViatura		Tabela de registo das zonas a que as viaturas tem acesso.	
Id	Long	Numero sequencial que identifica o registo	Chave primária
RefViatura	Long	Referencia à viatura	
RefZona	Long	Referencia à zona a que a viatura tem acesso	
DataValidade	Date	Data do acesso à zona	
Ativo	Boolean	Se o acesso à zona indicada está ativo	
DataRegisto	Date	Data de criação do registo	
Notas	Text	Campo de texto livre para informações complementares	

Tabela 65 – Tabela da base de dados – tblDicZonasAcessoViatura

tblDicZonasAcessoViatura		Tabela dicionário com lista de zonas acessíveis por viaturas.	
Id	Long	Numero sequencial que identifica o registo	Chave primária
Zona	String [30]	Nome da zona	
Notas	Text	Campo de texto livre para informações complementares	

Tabela 66 – Tabela da base de dados – tblViaturaRevalidacao

tblViaturaRevalidacao		Tabela com informação sobre a revalidação da credenciação de viaturas para acesso a zonas restritas.	
Id	Long	Numero sequencial que identifica o registo	Chave primária
RefViatura	Long	Referencia à viatura	
DataValidadeDistico	Date	Data de validade do dístico	
DataValidadeInspecao	Date	Data de validade da inspeção do veículo	
DataValidadeSeguro	Date	Data de validade do seguro do veículo	
DataValidadeExtintor	Date	Data de validade do extintor	
DataRegisto	Date	Data de criação do registo	
Notas	Text	Campo de texto livre para informações complementares	

h) Tabelas de registo de dados relacionados com cartões pontuais de visitas

Tabela 67 – Tabela da base de dados – tblVisitante

tblVisitante	Tabela de registo de informação relacionada com visitantes		
Id	String[12]	Número do documento de identificação	Chave primária
Nome	String[100]	Nome do visitante	
DataNascimento	Date	Data de nascimento	
RefNacionalidade	Long	Nacionalidade	
RefTipoId	Long	Referencia ao tipo de documento de identificação	
ValidadeDocID	Date	Data de validade do documento de identificação	
RefEntidade	Long	Referencia à entidade que o visitante representa	
DataRegisto	Date	Data de criação do registo	
Notas	Text	Campo de texto livre para informações complementares	

Tabela 68 – Tabela da base de dados – tblCartaoPontual

tblCartaoPontual		Tabela de registo de informação relacionada com cartões de acesso pontual de visitantes	
Id	Long	Número sequencial que identifica o registo	Chave primária
RefSolicitador	String[12]	Referencia ao colaborador autorizado que solicita credenciação pontual para o visitante	
RefVisitante	String[12]	Referencia à informação do visitante	
DataInicio	Date	Data de início de validade do cartão pontual	
DataValidade	Date	Data de validade do cartão	
Emitido	Boolean	Marcador para indicar se o acesso está a ser usado	
Ferramentas	Boolean	Marcador que indica se o visitante é portador de ferramentas	
DataRegisto	Date	Data de criação do registo	
Notas	Text	Campo de texto livre para informações complementares	

Tabela 69 – Tabela da base de dados – tblCartaoPontualAnexo

tblCartaoPontualAnexo		Para conter documentos anexados às credenciações pontuais de visitantes	
Id	Long	Numero sequencial que identifica o registo	Chave primária
RefCartaoPontual	String [12]	Referencia ao cartão pontual a que este registo pertence	
Anexo	objecto	Campo para conter o documento	
RefTipoAnexo	Long	Classificação do tipo de anexo	
DataRegisto	Date	Data de criação do registo	
Notas	Text	Campo de texto livre para informações complementares	

Tabela 70 – Tabela da base de dados – tblCartaoPontualPortaria

tblCartaoPontualPortaria		Para fazer a atribuição de permissão de passagem em portarias a visitantes	
Id	Long	Numero sequencial que identifica o registo	Chave primária
RefCartaoPontual	String [12]	Referencia ao cartão pontual a que este registo pertence	
RefPortaria	Long	Referencia a portarias	
DataRegisto	Date	Data de criação do registo	

Tabela 71 – Tabela da base de dados – tblCartaoPontualAcesso

tblCartaoPontualAcesso		Para fazer a atribuição de permissão acessos a visitantes	
Id	Long	Numero sequencial que identifica o registo	Chave primária
RefCartaoPontual	String [12]	Referencia ao cartão pontual a que este registo pertence	
RefAcesso	String [1]	Referencia ao acesso	
DataRegisto	Date	Data de criação do registo	

Tabela 72 – Tabela da base de dados – tblVisitanteAnexo

tblVisitanteAnexo		Para conter documentos anexados à entidade Visitante	
Id	Long	Numero sequencial que identifica o registo	Chave primária
RefVisitante	String [12]	Referencia à pessoa a que este registo pertence	
Anexo	String [200]	Campo para conter o nome do documento	
RefTipoAnexo	Long	Classificação do tipo de anexo	
DataRegisto	Date	Data de criação do registo	
Notas	Text	Campo de texto livre para informações complementares	

i) Tabelas de registo de dados relacionados com cartões pontuais de passageiros

Tabela 73 – Tabela da base de dados – tblPassageiro

tblPassageiro		Tabela de registo de informação relacionada com passageiros que necessitam de cartões pontuais	
Id	String[12]	Número do documento de identificação	Chave primária
Nome	String[60]	Nome do passageiro	
DataNascimento	Date	Data de nascimento	
RefNacionalidade	Long	Nacionalidade	
RefTipoId	Long	Referencia ao tipo de documento de identificação	
ValidadeDocID	Date	Data de validade do documento de identificação	
DataRegisto	Date	Data de criação do registo	
Notas	Text	Campo de texto livre para informações complementares	

Tabela 74 – Tabela da base de dados – tblCartaoPontualPassageiro

tblCartaoPontualPassageiro		Tabela de registo de informação relacionada com cartões de acesso pontual de passageiros	
Id	Long	Número sequencial que identifica o registo	Chave primária
RefPassageiro	String[12]	Referencia à informação do passageiro	
DataValidade	Date	Data de validade do cartão	
RefCompanhiaAerea	Long	Companhia aérea que serviu o passageiro	
Voo	String[20]	Designação do voo de chegadas	
DataHoraVoo	DateTime	Data e hora do voo	
DataRegisto	Date	Data de criação do registo	
Notas	Text	Campo de texto livre para informações complementares	

Tabela 75 – Tabela da base de dados – tblDicCompanhiaAerea

tblDicCompanhiaAerea		Dicionário de companhias aéreas	
Id	Long	Numero sequencial que identifica o registo	Chave primária
IATA	String[15]	Código IATA	
ICAO	String[15]	Código ICAO	
Companhia	String [20]	Identificação da companhia aérea	
Pais	String(350)	Pais de origem da companhia aérea	
Notas	Text	Campo de texto livre para informações complementares	

j) Tabelas de registo de dados relacionados com acontecimentos no sistema

Tabela 76 – Tabela da base de dados – tblDicTipoLog

tblDicTipoLog		Tabela dicionário com lista de tipos de registo de ações do sistema	
Id	Long	Numero sequencial que identifica o registo	Chave primária
TipoLog	String [20]	Nome do tipo de registo de log	
Notas	Text	Campo de texto livre para informações complementares	

Tabela 77 – Tabela da base de dados – tblDicClassificacaoTipoLog

tblDicClassificacaoTipoLog		Tabela dicionário com lista de tipos de classificação de registo de ações do sistema	
Id	Long	Numero sequencial que identifica o registo	Chave primária
ClassificacaoTipoLog	String [20]	Nome do tipo classificação: pessoa, portaria, sistema, etc.	
Notas	Text	Campo de texto livre para informações complementares	

Tabela 78 – Tabela da base de dados – tblLogInfPessoa

tblLogInfPessoa		Tabela de registo de ações de alteração de dados relacionados com pessoas	
Id	Long	Número sequencial que identifica o registo	Chave primária
RefTipoLog	Long	Referencia ao tipo de registo	
RefPessoa	String[12]	Referencia à pessoa	
RefOperador	String[12]	Referencia ao operador que efetuou a operação	
DataHora	DateTime	Data e hora de alteração de dados	
Operação	Text	Descrição da operação efetuada	

Tabela 79 – Tabela da base de dados – tblLogInfEntidade

tblLogInfEntidade		Tabela de registo de ações de alteração de dados relacionados com entidades	
Id	Long	Número sequencial que identifica o registo	Chave primária
RefTipoLog	Long	Referencia ao tipo de registo	
RefEntidade	String[12]	Referencia à entidade	
RefOperador	String[12]	Referencia ao operador que efetuou a operação	
DataHora	DateTime	Data e hora de alteração de dados	
Operação	Text	Descrição da operação efetuada	

Tabela 80 – Tabela da base de dados – tblLogInfPessoa

tblLogInfPessoa		Tabela de registo de ações de alteração de dados relacionados com pessoas	
Id	Long	Número sequencial que identifica o registo	Chave primária
RefTipoLog	Long	Referencia ao tipo de registo	
RefPessoa	String[12]	Referencia à pessoa	
RefOperador	String[12]	Referencia ao operador que efetuou a operação	
DataHora	DateTime	Data e hora de alteração de dados	
Operação	Text	Descrição da operação efetuada	

Tabela 81 – Tabela da base de dados – tblLogInfPortaria

tblLogInfPortaria		Tabela de registo de ações de alteração de dados relacionados com portarias	
Id	Long	Número sequencial que identifica o registo	Chave primária
RefTipoLog	Long	Referencia ao tipo de registo	
RefPortaria	Long	Referencia à portaria	
RefOperador	String[12]	Referencia ao operador que efetuou a operação	
DataHora	DateTime	Data e hora de alteração de dados	
Operação	Text	Descrição da operação efetuada	

Tabela 82 – Tabela da base de dados – tblLogInfViatura

tblLogInfViatura		Tabela de registo de ações de alteração de dados relacionados com viaturas	
Id	Long	Número sequencial que identifica o registo	Chave primária
RefTipoLog	Long	Referencia ao tipo de registo	
RefViatura	Long	Referencia à viatura	
RefOperador	String[12]	Referencia ao operador que efetuou a operação	
DataHora	DateTime	Data e hora de alteração de dados	
Operação	Text	Descrição da operação efetuada	

Tabela 83 – Tabela da base de dados – tblLogLogin

tblLogLogin		Tabela de registo de ações de acessos às aplicações da plataforma: *Login* e *Logout*.	
Id	Long	Número sequencial que identifica o registo	Chave primária
RefTipoLog	Long	Referencia ao tipo de registo	
RefOperador	String[12]	Referencia ao operador que efetuou a operação	
DataHora	DateTime	Data e hora de alteração de dados	
Operação	Text	Descrição da operação efetuada	

Tabela 84 – Tabela da base de dados – tblLogAcessoPortaria

tblLogAcessoPortaria		Tabela de registo de ações de acessos à aplicação Port: *Login* e *Logout*.	
Id	Long	Número sequencial que identifica o registo	Chave primária
RefTipoLog	Long	Referencia ao tipo de registo	
RefVigilante	String[12]	Referencia ao vigilante que efetuou a operação	
RefPortaria	Long	Referencia à portaria	
DataHora	DateTime	Data e hora de alteração de dados	
Operação	Text	Descrição da operação efetuada	

Tabela 85 – Tabela da base de dados – tblLogPassagemPortaria

tblLogPassagemPortaria		Tabela de registo de passagens em portaria	
Id	Long	Número sequencial que identifica o registo	Chave primária
RefTipoLog	Long	Referencia ao tipo de registo	
RefPortaria	Long	Referencia à portaria	
RefVigilante	String[12]	Referencia ao vigilante que está registado no sistema	
RefCartão	Long	Referencia ao cartão que foi apresentado no leitor da portaria	
DataHora	DateTime	Data e hora de alteração de dados	
Operação	Text	Descrição da operação efetuada	

k) Tabelas de registo de dados da base de dados Portarias

Tabela 86 – Tabela de registo de informação de pessoas, na base de dados das portarias.

tblPessoa		Para conter a informação pessoal e respetiva informação de acessos.	
Id	String [12]	Número do documento de identificação	Chave primária
Nome	String [100]	Nome da pessoa	
NumeroCartao	Long	Número do cartão ativo	
DataValidadeCartao	Date	Data de validade do cartão	
Foto	Object	Foto do portador	
Entidade	String[50]	Nome da entidade do portador	
ObjetosProibidos	Boolean	Marcador que indica se a pessoa pode transportar objetos proibidos	
ListaObProibidos	Object	Lista de objetos proibidos	
Perfil	bigint	Referencia ao perfil da pessoa	
PalavraChave	String[20]	Palavra -chave	
LicencaConducao	String[20]	Tipo de licença de condução	
LVO	Boolean	Se a pessoa pode conduzir em condições LVO	
VeiculosEspeciais	Boolean	Se a pessoa pode conduzir veículos especiais	
A	Boolean	Marcador que indica se o portador tem acesso a esta área restrita	
B	Boolean	Marcador que indica se o portador tem acesso a esta área restrita	
C	Boolean	Marcador que indica se o portador tem acesso a esta área restrita	
D	Boolean	Marcador que indica se o portador tem acesso a esta área restrita	
E	Boolean	Marcador que indica se o portador tem acesso a esta área restrita	
I	Boolean	Marcador que indica se o portador tem acesso a esta área restrita	
L	Boolean	Marcador que indica se o portador tem acesso a esta área restrita	

M	Boolean	Marcador que indica se o portador tem acesso a esta área restrita
O	Boolean	Marcador que indica se o portador tem acesso a esta área restrita
P	Boolean	Marcador que indica se o portador tem acesso a esta área restrita
T	Boolean	Marcador que indica se o portador tem acesso a esta área restrita

Tabela 87 – Tabela de registo de informação das portarias e respetivos acessos.

tblPortaria		Para conter a informação da portaria	
Id	Long	Número do documento de identificação	Chave primária
Nome	String [50]	Nome da portaria	
A	Boolean	Marcador que indica se o portador tem acesso a esta área restrita	
B	Boolean	Marcador que indica se o portador tem acesso a esta área restrita	
C	Boolean	Marcador que indica se o portador tem acesso a esta área restrita	
D	Boolean	Marcador que indica se o portador tem acesso a esta área restrita	
E	Boolean	Marcador que indica se o portador tem acesso a esta área restrita	
I	Boolean	Marcador que indica se o portador tem acesso a esta área restrita	
L	Boolean	Marcador que indica se o portador tem acesso a esta área restrita	
M	Boolean	Marcador que indica se o portador tem acesso a esta área restrita	
O	Boolean	Marcador que indica se o portador tem acesso a esta área restrita	
P	Boolean	Marcador que indica se o portador tem acesso a esta área restrita	
T	Boolean	Marcador que indica se o portador tem acesso a esta área restrita	

Tabela 88 – Tabela da base de dados – tblLogAcessoPortaria

tblLogLoginPortaria		Tabela de registo de ações de acessos à aplicação Port: *Login* e *Logout*.	
Id	Long	Número sequencial que identifica o registo	Chave primária
RefTipoLog	Long	Referencia ao tipo de registo	
RefOperador	String[12]	Referencia ao operador que efetuou a operação	
RefPortaria	Long	Referencia à portaria	
DataHora	DateTime	Data e hora de alteração de dados	
Operação	Text	Descrição da operação efetuada	

Tabela 89 – Tabela da base de dados – tblLogPassagemPortaria

tblLogPassagemPortaria		Tabela de registo de passagens em portaria	
Id	Long	Número sequencial que identifica o registo	Chave primária
RefTipoLog	Long	Referencia ao tipo de registo	
RefPortaria	Long	Referencia à portaria	
RefVigilante	String[12]	Referencia ao vigilante que está registado no sistema	
RefCartão	Long	Referencia ao cartão que foi apresentado no leitor da portaria	
DataHora	DateTime	Data e hora de alteração de dados	
Operação	Text	Descrição da operação efetuada	

Tabela 90 Tabela da base de dados – tblLog

tblLog		Tabela de registo de outros eventos	
Id	Long	Número sequencial que identifica o registo	Chave primária
Operação	Text	Descrição da operação efetuada	

l) Tabelas para implementação de dicionários Gerais

Tabela 91 – Tabela da base de dados – tblDicAplicacoes

tblDicAplicacao		Dicionário de aplicações que constituem a plataforma	
Id	Long	Número sequencial que identifica o registo	Chave primária
Aplicação	String[20]	Nome da aplicação	
Notas	Text	Campo de texto livre para informações complementares	

Tabela 92 – Tabela da base de dados – tblDicCodigosDeErro

tblDicCodigosErro		Dicionário de aplicações que constituem a plataforma	
Id	Long	Número de erro	Chave primária
Erro	String[100]	Descrição do erro	

Tabela 93 – Tabela da base de dados – tblDicTipoAnexo

tblDicTipoAnexo		Dicionário de classificação dos ficheiros em anexo	
Id	Long	Número sequencial que identifica o registo	Chave primária
Tipo	String[50]	Descrição do tipo	
RefClassificacao	bigint	Classificação do tipo de anexo	
Notas	Text	Campo de texto livre para informações complementares	

Tabela 94 – Tabela da base de dados – tblDicClassificacaoTipoAnexo

tblDicClassificacaoTipoAnexo		Dicionário de classificação dos tipos de ficheiros em anexo	
Id	Long	Número sequencial que identifica o registo	Chave primária
Classificacao	String[20]	Classificação do tipo de anexo	
Notas	Text	Campo de texto livre para informações complementares	

Tabela 95 – Tabela da base de dados – tblDicPais

tblDicPais		Dicionário para conter uma lista de paises	
Id	Long	Número sequencial que identifica o registo	Chave primária
Pais	String[30]	Nome do país	

m) Outras tabelas

Tabela 96 – Tabela da base de dados – tblCaminhoFicheiro

tblCaminhoFicheiro		Tabela para conter o caminho de localização dos diretório do *file server* para armazenamento dos ficheiros anexos.	
Id	String[100]	Designação do local	Chave primária
Caminho	String[500]	Caminho para o diretório	
Descricao	Text	Informação complementar sobre o registo	

Anexo J Bateria de testes de *Hardware*

Formulário de apoio à realização dos testes do módulo de *hardware* de leitura de cartões.

Teste de *Hardware*					
Número de cartão	Número lido pelo sistema	Leitura Válida			
Observações:					
Resultado do teste:		Aprovado		Reprovado	

Critério de aceitação: boa execução da leitura do número de identificação de pelo menos cinco cartões.

Anexo K Lista de Erros: *Stored Procedures*

Este anexo apresenta uma lista de códigos que são devolvidos pelos *stored procedures* no parâmetro de saída que reporta o sucesso da execução da rotina. Estes códigos estão associados à validação dos parâmetros de entrada e aos possíveis erros que ocorrem durante a execução do código.

Tabela 97 – Lista de erros de retorno de *stored procedures*.

Erro	Descrição
Gerais – Erros gerais	
4	Portaria - Login não autorizado
3	Portaria - Login autorizado
2	Portaria - Passagem não permitida
1	Portaria - Passagem permitida
0	Registo efetuado com sucesso
-1	Rotina não executada
-2	Erro – Pessoa desconhecida
-3	Erro – Entidade Desconhecida
-4	Erro – Portaria Desconhecida
-5	Erro – Viatura Desconhecida
-6	Erro - Cartão desconhecido
-7	Erro – Palavra-chave errada
-8	Erro – Perfil desconhecido

-9	Erro – Solicitador de cartão pontual desconhecido
-10	Erro – Visitante desconhecido
-11	Erro – Passageiro desconhecido
-12	Erro – Companhia aérea desconhecida
Datas – Erros relacionados com datas	
-10	Erro – Data de início posterior à data de fim
-11	Erro – Data de fim anterior à data atual
-12	Erro – Documento de identificação caducado
-13	Erro – Data invalida
Informação - Erros relacionados à informação existente	
-20	Erro – Número de início maior que o número de fim
-21	Erro – Endereço de e-mail desconhecido
-22	Erro – Numero de telefone desconhecido
-23	Erro – Morada desconhecida
-24	Erro – Anexo desconhecido
-25	Erro – Palavra-chave Errada
-26	Erro – Já existe um cartão ativo para este utilizador
-27	Erro – Números de lote já existente
-28	Erro – Já existe um perfil ativo para este utilizador
-29	Erro – Já existe este acesso ativo para este utilizador
-35	Erro – Não existe este acesso ativo para este utilizador
-30	Erro – Atribuição de acesso desconhecida
-31	Erro – Registo de infração desconhecido
-32	Erro – Registo de infração de condução desconhecido
-33	Erro – Cartão pontual desconhecido
-34	Erro - Cartão de identificação fora da data de validade
-35	Erro - Cartão de identificação não ativo
Dicionários – Erros relacionados com operações associadas a tabelas-dicionário	
-50	Erro – Tipo de documento de identificação desconhecido
-51	Erro – País desconhecido
-52	Erro – Tipo estado de cartão
-53	Erro – Entidade desconhecida
-54	Erro – Serviço desconhecido
-55	Erro – Função desconhecida
-56	Erro – Aplicação desconhecida
-57	Erro – Tipo de pagamento desconhecido
-58	Erro – Estado de cartão desconhecido
-59	Erro – Perfil de utilizador desconhecido
-60	Erro – Acesso desconhecido

-61	Erro – Infração desconhecida
-62	Erro – Penalidade desconhecida
-63	Erro – Tipo carta condução desconhecido
-64	Erro – Tipo licença de condução desconhecida
-65	Erro – Tipo de combustível desconhecido
-66	Erro – Tipo de veículo desconhecido
-67	Erro – Zona desconhecida
-68	Erro – Infração de condução desconhecida
-69	Erro – Penalidade desconhecida
-70	Erro – Companhia aérea desconhecida
-71	Erro na anexação de ficheiro, apaga o registo na base de dados
Base de dados – Erros relacionados com a base de dados	
-500	Erro no registo de dados
Pro-Watch – Erros relacionados com a interface da base de dados PWNT	
-100	Erro - *Company* desconhecida
-101	Erro - *Clearance Code* desconhecido
-102	Erro - Cartão desconhecido

Anexo L Interface com a base de dados

Este anexo apresenta a lista de todas as interfaces implementadas nas bases de dados da plataforma e do *Pro-Watch*. Para cada interface faz-se uma descrição da funcionalidade, apresentando os parâmetros de entrada e os resultados que são devolvidos.

Na apresentação da definição das interfaces usa-se a seguinte nomenclatura:

- In – Parâmetro de entrada.
- [In] – Parâmetro de entrada opcional.
- Out – Parâmetro de saída.

a) Interfaces relacionadas com pessoas

Pessoa-Cria *Stored procedure*		Cria um registo de pessoas
In	Id	Número do documento de identificação
In	Nome	Nome

In	DataNascimento		Data de nascimento
In	RefNacionalidade		Indicação da nacionalidade
In	Filiação Materna		Nome da mãe
In	Filiação Paterna		Nome do pai
In	RefTipoID		Referencia ao tipo de documento usado para identificação
In	ValidadeID		Validade do documento de identificação
[In]	Palavra-chave		Palavra-chave
In	RefEntidade		Referencia à entidade que representa
In	RefServico		Referencia ao dicionário do serviço a que faz assistência
In	RefFuncao		Referencia ao dicionário da função que desempenha
In	ObjetosProibidos		Marcador que indica se a pessoa pode transportar objetos proibidos
[In]	Notas		Campo de texto livre para informações complementares
In	RefOperador		Pessoa que está a executar a operação
Out	Execucao	-12	Erro – Documento de identificação caducado
		-51	Erro – País desconhecido
		-52	Erro – Tipo de documento de identificação desconhecido
		-53	Erro – Entidade desconhecida
		-54	Erro – Serviço desconhecido
		-55	Erro – Função desconhecida
		-2	Erro – Operador inválido
		0	Registo efetuado com sucesso

Pessoa-Edita *Stored procedure*		Permite alterar a informação sobre uma pessoa
In	Id	Número do documento de identificação
[In]	Nome	Nome
[In]	DataNascimento	Data de nascimento
[In]	RefNacionalidade	Indicação da nacionalidade
[In]	Filiação Materna	Nome da mãe
[In]	Filiação Paterna	Nome do pai
[In]	RefTipoID	Referencia ao dicionário do tipo de documento usado para identificação
[In]	ValidadeID	Validade do documento de identificação
[In]	Foto	Foto

[In]	Palavra-chave		Palavra-chave
[In]	RefEntidade		Referencia à entidade que representa
[In]	RefServico		Referencia ao dicionário do serviço a que faz assistência
[In]	RefFuncao		Referencia ao dicionário da função que desempenha
[In]	Notas		Campo de texto livre para informações complementares
In	RefOperador		Pessoa que está a executar a operação
Out	Execucao	-2	Erro – Pessoa desconhecida
		-12	Erro – Documento de identificação caducado
		-51	Erro – País desconhecido
		-52	Erro – Tipo de documento de identificação desconhecido
		-53	Erro – Entidade desconhecida
		-54	Erro – Serviço desconhecido
		-55	Erro – Função desconhecida
		-13	Erro – Data invalida
		0	Registo efetuado com sucesso

Pessoa-GuadaFoto *Stored procedure*			Guarda a foto de uma pessoa.
In	IdPessoa		Número de identificação da pessoa
In	Foto		*Array* de *bytes* com a foto da pessoa
In	RefOperador		Pessoa que está a executar a operação
Out	Execucao	-2	Erro – Pessoa desconhecida
		0	Registo efetuado com sucesso

Pessoa-DevolveFoto *Stored procedure*			Devolve a foto de uma pessoa.
In	IdPessoa		Número de identificação da pessoa
Out	Execucao	-2	Erro – Pessoa desconhecida
		0	Registo efetuado com sucesso
	Resultado		Foto

Email-Cria *Stored procedure*			Adiciona um endereço de correio eletrónico a uma pessoa ou a uma entidade.
In	IdDono		Número de identificação da pessoa ou da entidade
In	EnderecoCorreio		Endereço do correio eletrónico da pessoa
In	PessoaEmpresa		Pessoa = 1, Empresa = 0
In	Notas		Notas
In	RefOperador		Pessoa que está a executar a operação
Out	Execucao	-2	Erro – Pessoa desconhecida
		-3	Erro – Entidade desconhecida
		0	Registo efetuado com sucesso

Email-Edita *Stored procedure*			Edita um registo existente de correio eletrónico, permitindo alterar o campo "Ativo" e o campo "Notas"
In	IdDono		Número de identificação da pessoa ou da entidade
In	ID-EnderecoCorreio		Identificador do registo de correio eletrónico
[In]	Notas		Notas
In	RefOperador		Pessoa que está a executar a operação
Out	Execucao	-2	Erro – Pessoa desconhecida
		-3	Erro – Entidade desconhecida
		-21	Erro – Endereço de e-mail desconhecido
		0	Registo efetuado com sucesso

Email-Apaga *Stored procedure*			Apaga um registo de correio eletrónico
In	ID-EnderecoCorreio		Identificador do registo de correio eletrónico
In	RefOperador		Pessoa que está a executar a operação
Out	Execucao	-21	Erro – Endereço de e-mail desconhecido
		0	Registo efetuado com sucesso

Pessoa-Email *Stored procedure*			Devolve os e-mail ativos da pessoa
In	IdPessoa		Identificador do utilizador
Out	Execucao	-2	Erro – Pessoa desconhecida

		0	Registo efetuado com sucesso
Out	Resultado	Identificador do e-mail	
		E-mail	
		Notas	

Telefone-Cria *Stored procedure*			Adiciona um contacto telefónico a uma pessoa ou a uma empresa
In	IdDono		Número de identificação da pessoa ou entidade
In	PessoaEmpresa		Pessoa = 1, Empresa = 0
In	Telefone		Número de telefone, nos números estrageiros este campo deve conter o indicativo do país.
In	Notas		Notas
In	RefOperador		Pessoa que está a executar a operação
Out	Execucao	-2	Erro – Pessoa desconhecida
		-3	Erro – Entidade desconhecida
		0	Registo efetuado com sucesso

Telefone-Edita *Stored procedure*			Edita um registo existente de telefone, permitindo alterar o campo "Ativo" e o campo "Notas"
In	IdDono		Número de identificação da pessoa ou entidade
In	ID-Telefone		Identificador do registo de telefone
[In]	Ativo		Indica se o endereço introduzido está a ser usado
[In]	Notas		Notas
In	RefOperador		Pessoa que está a executar a operação
Out	Execucao	-2	Erro – Pessoa desconhecida
		-22	Erro – Numero de telefone desconhecido
		0	Registo efetuado com sucesso

Telefone -Apaga *Stored procedure*			Apaga um registo de telefone
In	ID-Telefone		Identificador do registo de telefone
In	RefOperador		Pessoa que está a executar a operação
Out	Execucao	-23	Erro – Numero de telefone desconhecido
		0	Registo efetuado com sucesso

Pessoa-Telefones *Stored procedure*			Devolve os telefones ativos da pessoa
In	IdPessoa		Identificador do utilizador
Out	Execucao	-2	Erro – Pessoa desconhecida
		0	Registo efetuado com sucesso
Out	Resultado	Id do número de telefone	
		Número de telefone	
		Notas	

Morada-Cria *Stored procedure*			Adiciona uma morada postal a uma pessoa ou entidade
In	Id		Número de identificação da pessoa
In	PessoaEmpresa		Pessoa = 1, Empresa = 0
In	Morada		Descrição da morada postal
In	Localidade		Localidade da morada
In	CodigoPostal		Código Postal
In	RefPais		Referencia ao País
In	Notas		Notas
In	RefOperador		Pessoa que está a executar a operação
Out	Execucao	-2	Erro – Pessoa desconhecida
		-51	Erro – País desconhecido
		0	Registo efetuado com sucesso

Morada-Edita *Stored procedure*		Edita um registo de Morada existente, permitindo alterar todos os campos.
In	Id	Número de identificação da pessoa
In	PessoaEmpresa	Pessoa = 1, Empresa = 0
In	MoradaID	Identificador da morada a alterar
[In]	Morada	Descrição da morada postal
[In]	Localidade	Localidade da morada
[In]	CodigoPostal	Código Postal
[In]	RefPais	Referencia ao País
[In]	Notas	Notas

In	RefOperador		Pessoa que está a executar a operação
Out	Execucao	-2	Erro – Pessoa desconhecida
		-23	Erro – Morada desconhecida
		-24	Erro – Pais desconhecido
		0	Registo efetuado com sucesso

Morada -Apaga *Stored procedure*			Apaga um registo de morada
In	ID-Morada		Identificador do registo de Morada
In	RefOperador		Pessoa que está a executar a operação
Out	Execucao	-24	Erro – Morada desconhecida
		0	Registo efetuado com sucesso

Pessoa-Morada *Stored procedure*		Devolve a informação das moradas de uma pessoa identificada pelo Id
In	RefPessoa	Referencia à pessoa
Out	Resultado	Id da morada
		Morada
		Localidade
		CodigoPostal
		Pais
		Ativo
		Notas

Pessoa-AnexoCria *Stored procedure*			Adiciona um anexo ao registo de uma pessoa.
In	Id		Número de identificação da pessoa
In	NomeAnexo		Nome do ficheiro para anexar
In	RefTipoAnexo		Tipo de anexo
In	Notas		Notas
In	RefOperador		Pessoa que está a executar a operação
Out	Execucao	-2	Erro – Pessoa desconhecida
		0	Registo efetuado com sucesso

Pessoa-ApagaAnexo *Stored procedure*			Apaga um anexo ao registo de uma pessoa.
In	Id		Número de identificação da pessoa
In	AnexoID		Indentificador de anexo existentes
In	Notas		Notas
In	RefOperador		Pessoa que está a executar a operação
Out	Execucao	-2	Erro – Pessoa desconhecida
		-24	Erro – Anexo desconhecido
		0	Registo efetuado com sucesso

Pessoa-Anexos *Stored procedure*			Devolve os anexos de uma pessoa.
In	Id		Número de identificação da pessoa
In	AnexoID		Indentificador de anexo existentes
In	Notas		Notas
In	RefOperador		Pessoa que está a executar a operação
Out	Execucao	-2	Erro – Pessoa desconhecida
		-24	Erro – Anexo desconhecido
		0	Registo efetuado com sucesso
	Resultado		Id
			NomeAnexo
			RefTipoAnexo
			TipoAnexo
			Notas

Pessoa-ErroAnexacaoApagaAnexo *Stored procedure*			Apaga um anexo ao registo de uma pessoa.
In	Id		Número de identificação da pessoa
In	AnexoID		Indentificador de anexo existentes
In	Notas		Notas
In	RefOperador		Pessoa que está a executar a operação
Out	Execucao	-2	Erro – Pessoa desconhecida

		-24	Erro – Anexo desconhecido
		0	Registo efetuado com sucesso

Pessoa-VerificaPalavraChave *Stored procedure*			Verifica se a palavra-chave de uma pessoa é válida em caso positivo devolve a referência ao seu perfil de acesso
In	Id		Número de identificação da pessoa
In	PalavraChave		Palavra-chave da pessoa identificada
In	RefAplicacao		Referencia à aplicação em que a pessoa está a tentar aceder
Out	Execucao	-2	Erro – Pessoa desconhecida
		-56	Erro – Aplicação desconhecida
		-25	Erro – Palavra-chave Errada
		Perfil	Em caso da palavra-chave corresponder à pessoa identificadas, é devolvido o identificador de perfil da pessoa (número positivo)

Pessoa-Logout *Stored procedure*			Faz o registo de *logout* de uma aplicação
In	Id		Número de identificação da pessoa
Out	Execucao	-2	Erro – Pessoa desconhecida
		0	Registo efetuado com sucesso

Pessoa-Nome *Stored procedure*			Devolve o nome de uma pessoa identificada pelo identificador
In	Id		Número de identificação da pessoa
Out	Execucao	-2	Erro – Pessoa desconhecida
		0	Registo efetuado com sucesso
Out	Resultado	Nome	

Pessoa-Dados *Stored procedure*		Informação geral de uma pessoa identificada pelo Id
In	RefPessoa	Referencia ao portador do cartão
Out	Resultado	Id
		Nome

		DataNascimento
		TipoDocumentoID
		ValidadeDocumentoID
		Entidade
		Funcao
		Servico
		Filiação Materna
		Filiação Paterns
		Objetos proibidos

Infracao-Cria *Stored procedure*			Cria uma infração de segurança associada a uma pessoa
In	RefPessoa		Referencia ao portador do cartão
In	RefInfracao		Definição da infração
In	DataInfração		Data de registo da infração
In	RefPenalidade		Definição da infração
In	DataPenalidade		Data de execução da penalidade
[In]	Notas		Notas
In	RefOperador		Pessoa que está a executar a operação
Out	Execucao	-2	Erro – Pessoa Desconhecida
		-61	Erro – Infração desconhecida
		-62	Erro – Penalidade desconhecida
		0	Registo efetuado com sucesso

Infracao-Edita *Stored procedure*			Altera a informação de uma infração de segurança associada a uma pessoa.
In	Id		Identificador da infração
[In]	RefInfracao		Definição da infração
[In]	DataInfração		Data de registo da infração
[In]	RefPenalidade		Definição da infração
[In]	DataPenalidade		Data de execução da penalidade
[In]	Notas		Notas
Out	Execucao	-2	Erro – Pessoa Desconhecida
		-61	Erro – Infração desconhecida

		-62	Erro – Penalidade desconhecida
		0	Registo efetuado com sucesso

Pessoa-Infracao *Stored procedure*		Devolve a informação sobre todas as infrações atribuídas a uma pessoa	
In	IdPessoa		Identificador do utilizador
Out	Execucao	-2	Erro – Pessoa desconhecida
		0	Registo efetuado com sucesso
Out	Resultado	IdPessoa	
		Nome	
		Infração	
		IdInfracao	
		DataInfracao	
		IdPenalizacao	
		Penalizaçao	
		DataPenalizacao	
		Notas	

b) Interfaces relacionadas com perfil

Perfil-Atribui *Stored procedure*	Cria um perfil associado a um utilizador		
In	IdPessoa		Identificador do utilizador
In	IdPerfil		Identificador do perfil
In	DataInicio		Data de início de validade do perfil
In	DataFim		Data de validade
In	Ativo		Marcador que indica se o perfil está ativo
[In]	Notas		Campo de texto livre para informações complementares
In	RefOperador		Pessoa que está a executar a operação
Out	Execucao	-2	Erro – Pessoa desconhecida
		-59	Erro – Perfil de utilizador desconhecido
		-28	Erro – Já existe um perfil ativo para este utilizador
		-10	Erro – Data de início posterior à data de fim
		-11	Erro – Data de fim anterior à data atual
		0	Registo efetuado com sucesso

Perfil-Retira *Stored procedure*	Retira um perfil associado a um utilizador		
In	IdPessoa		Identificador do utilizador
[In]	IdPerfil		Identificador do perfil
[In]	Notas		Campo de texto livre para informações complementares
In	RefOperador		Pessoa que está a executar a operação
Out	Execucao	-2	Erro – Pessoa desconhecida
		-59	Erro – Perfil de utilizador desconhecido
		-28	Erro – Já existe um perfil ativo para este utilizador
		-10	Erro – Data de início posterior à data de fim
		-11	Erro – Data de fim anterior à data atual
		0	Registo efetuado com sucesso

Perfil-Pessoa *Stored procedure*			Devolve a informação sobre todos os perfis atribuídos a uma pessoa
In	IdPessoa		Identificador do utilizador
Out	Execucao	-2	Erro – Pessoa desconhecida
		0	Registo efetuado com sucesso
Out	Resultado	IdPessoa	
		Nome	
		IdPerfil	
		Perfil	
		DataInicio	
		DataFim	
		Ativo	
		Notas	

Pessoa-ComPerfil *Stored procedure*			Devolve a lista de pessoas com determinado perfil
In	RefPerfil		Identificador do perfil
Out	Execucao	-59	Erro – Perfil de utilizador desconhecido
		0	Registo efetuado com sucesso
		Perfil	
		DataInicio	
		DataFim	
		Ativo	
		Notas	

c) Interfaces relacionadas com entidades

Entidade-Cria *Stored procedure*	Cria um registo de empresa		
In	Nome		Nome da empresa
In	NIF		Número de identificação fiscal da empresa
[In]	RefSignatário		Referencia à pessoa que representa a empresa
In	RefNacionalidade		Referencia ao país de origem da empresa
In	RefTipoPagamento		Referencia à forma de pagamento dos custos associados à credenciação de pessoa e viaturas
[In]	Notas		Campo de texto livre para informações complementares
In	RefOperador		Pessoa que está a executar a operação
Out	Execucao	-2	Erro – Pessoa desconhecida
		-51	Erro – Nacionalidade desconhecida
		-57	Erro – Tipo de pagamento desconhecido
		0	Registo efetuado com sucesso

Entidade-Edita *Stored procedure*	Altera a informação do registo de uma empresa		
In	NIF		Número de identificação fiscal da empresa
[In]	RefSignatário		Referencia à pessoa que representa a empresa
[In]	RefNacionalidade		Referencia ao país de origem da empresa
[In]	RefTipoPagamento		Referencia à forma de pagamento dos custos associados à credenciação de pessoa e viaturas
[In]	Notas		Campo de texto livre para informações complementares
In	RefOperador		Pessoa que está a executar a operação
Out	Execucao	-3	Erro – Entidade Desconhecida
		-2	Erro – Pessoa desconhecida
		-51	Erro – Nacionalidade desconhecida
		-57	Erro – Tipo de pagamento desconhecido
		0	Registo efetuado com sucesso

d) Interfaces relacionadas com portarias

Portaria-Cria *Stored procedure*	Cria um registo de portaria		
In	Nome		Nome da portaria
[In]	Notas		Campo de texto livre para informações complementares
Out	Execucao	0	Registo efetuado com sucesso

Portaria-Edita *Stored procedure*	Altera a informação do registo de uma portaria		
In	ID		Identificador da portaria
[In]	Notas		Campo de texto livre para informações complementares
In	RefOperador		Pessoa que está a executar a operação
Out	Execucao	-4	Erro – Portaria Desconhecida

e) Interfaces relacionadas com cartões

Cartao –Cria *Stored procedure*			Adiciona um cartão.
In	Id		Número de identificação do cartão
In	RefPessoa		Referencia ao portador do cartão
In	DataInicio		Data de início de validade do cartão
In	DataFim		Data de validade do cartão
In	Ativo		Marcador que indica se o cartão está em uso
In	Permanente		Marcador que indica se o cartão é do tipo permanente ou temporário
In	RefEstado		Referencia ao estado do cartão
In	Company		Referencia a entidade do tipo Company da base de dados PWNT
[In]	Notas		Notas
In	RefOperador		Pessoa que está a executar a operação
Out	Execucao	-2	Erro – Pessoa Desconhecida
		-10	Erro – Data de início posterior à data de fim
		-11	Erro – Data de fim anterior à data atual
		-26	Erro – Já existe um cartão ativo para este utilizador
		-58	Erro – Estado de cartão desconhecido
		0	Registo efetuado com sucesso

Cartao –Edita *Stored procedure*		Altera informação de um cartão.
In	Id	Número de identificação do cartão
[In]	RefPessoa	Referencia ao portador do cartão
[In]	DataInicio	Data de início de validade do cartão
[In]	DataFim	Data de validade do cartão
[In]	Ativo	Marcador que indica se o cartão está em uso
[In]	Permanente	Marcador que indica se o cartão é do tipo permanente ou temporário
[In]	RefEstado	Referencia ao estado do cartão
[In]	Company	Referencia a entidade do tipo Company da base de dados

			PWNT
[In]	Notas		Notas
In	RefOperador		Pessoa que está a executar a operação
Out	Execucao	-2	Erro – Pessoa Desconhecida
		-10	Erro – Data de início posterior à data de fim
		-11	Erro – Data de fim anterior à data atual
		-26	Erro – Já existe um cartão ativo para este utilizador
		-58	Erro – Estado de cartão desconhecido
		0	Registo efetuado com sucesso

LoteCartões-Cria *Stored procedure*			Para criar um registo de lote de cartões de acesso
in	PrimeiroNumero		Primeiro número do lote
in	UltimoNumero		Ultimo número do lote
in	DataInicio		Data de início de uso do lote
in	DataFim		Data de fim de uso do lote
in	Ativo		Marcador que informa se o lote está em uso
[In]	Notas		Notas
In	RefOperador		Pessoa que está a executar a operação
out	Execucao	-10	Erro – Data de início posterior à data de fim
		-11	Erro – Data de fim anterior à data atual
		-20	Erro – Número de início maior que o número de fim
		-27	Erro – Números de lote já existente
		0	Registo efetuado com sucesso

LoteCartões-Edita *Stored procedure*			Altera informação num registo de lote existente
In	PrimeiroNumero		Primeiro número do lote
[In]	UltimoNumero		Ultimo número do lote
[In]	DataInicio		Data de início de uso do lote
[In]	DataFim		Data de fim de uso do lote
[In]	Ativo		Marcador que informa se o lote está em uso
[In]	Notas		Notas
In	RefOperador		Pessoa que está a executar a operação
out	Execucao	-10	Erro – Data de início posterior à data de fim

		-11	Erro – Data de fim anterior à data atual
		-20	Erro – Número de início maior que o número de fim
		-27	Erro – Números de lote já existente
		0	Registo efetuado com sucesso

LoteCartões-CartãoValido *Stored procedure*			Devolve *Verdade* se o número de cartão pertence ao um lote ativo
In	NumeroCartao		Número de cartão a verificar
out	Resultado		Verdade/Falso

Cartoes-Pessoa *Stored procedure*			Devolve a informação sobre todos os cartões atribuídos a uma pessoa
In	IdPessoa		Identificador do utilizador
Out	Execucao	-2	Erro – Pessoa desconhecida
		0	Registo efetuado com sucesso
Out	Resultado	IdPessoa	
		Nome	
		NumeroCartao	
		DataInicio	
		DataFim	
		Ativo	
		Permanente	
		IdEstado	
		Estado	
		Company	
		Notas	

CartaoEmitido *Stored procedure*			Coloca um cartão no estado de emitido
In	Id		Identificador do cartão
In	IdPessoa		Identificação do portador do cartão
In	RefOperador		Pessoa que está a executar a operação
Out	Execucao	-2	Erro – Pessoa desconhecida

f) Interfaces relacionadas com acessos

Acessos-AtribuiLetraPessoa *Stored procedure*			Associa um acesso codificado em letras a um utilizador
In	IdPessoa		Identificador do utilizador
In	IdAcesso		Identificador do Acesso
In	DataInicio		Data de início de validade do acesso
In	DataFim		Data de validade
In	Ativo		Marcador que indica se o acesso está ativo
[In]	Notas		Campo de texto livre para informações complementares
In	RefOperador		Pessoa que está a executar a operação
Out	Execucao	-2	Erro – Pessoa desconhecida
		-60	Erro – Acesso desconhecido
		-29	Erro – Já existe este acesso ativo para este utilizador
		-10	Erro – Data de início posterior à data de fim
		-11	Erro – Data de fim anterior à data atual
		0	Registo efetuado com sucesso

Acessos-AtribuiLetraPortaria *Stored procedure*			Associa um acesso codificado em letras a uma portaria
In	IdPortaria		Identificador da portaria
In	IdAcesso		Identificador do Acesso
In	DataInicio		Data de início de validade do acesso
In	DataFim		Data de validade
In	Ativo		Marcador que indica se o acesso está ativo
[In]	Notas		Campo de texto livre para informações complementares
In	RefOperador		Pessoa que está a executar a operação
Out	Execucao	-2	Erro – Pessoa desconhecida
		-4	Erro – Portaria desconhecida
		-60	Erro – Acesso desconhecido
		-10	Erro – Data de início posterior à data de fim
		-11	Erro – Data de fim anterior à data atual
		0	Registo efetuado com sucesso

Acessos-AtribuiCorPessoa *Stored procedure*			Associa um acesso codificado em cores a um utilizador
In	IdPessoa		Identificador do utilizador
In	IdAcesso		Identificador do Acesso
In	DataInicio		Data de início de validade do acesso
In	DataFim		Data de validade
In	Ativo		Marcador que indica se o acesso está ativo
[In]	Notas		Campo de texto livre para informações complementares
In	RefOperador		Pessoa que está a executar a operação
Out	Execucao	-2	Erro – Pessoa desconhecida
		-60	Erro – Acesso desconhecido
		-29	Erro – Já existe este acesso ativo para este utilizador
		-10	Erro – Data de início posterior à data de fim
		-11	Erro – Data de fim anterior à data atual
		0	Registo efetuado com sucesso

AcessosLetra-Pessoa *Stored procedure*			Devolve a informação sobre todos os acessos, codificados em letras, atribuídos a uma pessoa
In	IdPessoa		Identificador do utilizador
Out	Execucao	-2	Erro – Pessoa desconhecida
		0	Registo efetuado com sucesso
Out	Resultado	IdPessoa	
		Nome	
		IdAcesso	
		Acesso	
		DataInicio	
		DataFim	
		Ativo	
		Estado	
		Notas	

AcessosLetra-Portaria *Stored procedure*	Devolve a informação sobre todos os acessos, codificados em letras, atribuídos a uma portaria

In	IdPortaria		Identificador da portaria
Out	Execucao	-4	Erro – Portaria desconhecida
		0	Registo efetuado com sucesso
Out	Resultado	IdPessoa	
		Nome	
		IdAcesso	
		Acesso	
		DataInicio	
		DataFim	
		Ativo	
		Estado	
		Notas	

AcessosCor-Pessoa *Stored procedure*		Devolve a informação sobre todos os acessos, codificados em cores, atribuídos a uma pessoa	
In	IdPessoa		Identificador do utilizador
Out	Execucao	-2	Erro – Pessoa desconhecida
		0	Registo efetuado com sucesso
Out	Resultado	IdPessoa	
		Nome	
		IdAcesso	
		Acesso	
		DataInicio	
		DataFim	
		Ativo	
		Estado	
		Notas	

Acessos-Pessoa *Stored procedure*		Devolve a identificação de todas as pessoas associadas a um acesso	
In	IdAcesso		Identificador do acesso
Out	Execucao	-60	Erro – Acesso desconhecido
		0	Registo efetuado com sucesso
Out	Resultado	IdAcesso	

		Acesso	
		IdPessoa	
		Nome	
		DataInicio	
		DataFim	
		Ativo	
		Estado	
		Notas	

Acessos-Portaria *Stored procedure*		Devolve a identificação de todas as portarias associadas a um acesso	
In	IdAcesso		Identificador do acesso
Out	Execucao	-60	Erro – Acesso desconhecido
		0	Registo efetuado com sucesso
Out	Resultado	IdAcesso	
		Acesso	
		IdPortaria	
		NomePortaria	
		DataInicio	
		DataFim	
		Ativo	
		Estado	
		Notas	

Acessos-RetiraLetraPessoa *Stored procedure*		Desativa um acesso codificado em letras de um utilizador	
In	IdPessoa		Identificador do utilizador
In	IdAcesso		Identificador do Acesso
In	RefOperador		Pessoa que está a executar a operação
Out	Execucao	-2	Erro – Pessoa desconhecida
		-35	Erro – Não existe este acesso ativo para este utilizador
		0	Registo efetuado com sucesso

Acessos-RetiraCorPessoa *Stored procedure*			Desativa um acesso codificado em cores de um utilizador
In	IdPessoa		Identificador do utilizador
In	IdAcesso		Identificador do Acesso
In	RefOperador		Pessoa que está a executar a operação
Out	Execucao	-2	Erro – Pessoa desconhecida
		-35	Erro – Não existe este acesso ativo para este utilizador
		0	Registo efetuado com sucesso

Acessos-TodasLetrasAtribuidas *Stored procedure*			Verifica se uma pessoa tem cinco letras de acesso ativas
In	IdPessoa		Identificador do utilizador
Out	Execucao	-2	Erro – Pessoa desconhecida
		1	A pessoa tem cinco letras de acesso ativas
		0	A pessoa não tem cinco letras de acesso ativas

Acessos-RetiraLetraPortaria *Stored procedure*			Desativa um acesso codificado em letras de uma portaria
In	IdPortaria		Identificador da portaria
In	IdAcesso		Identificador do Acesso
In	RefOperador		Pessoa que está a executar a operação
Out	Execucao	-4	Erro – Portaria desconhecida
		-35	Erro – Não existe este acesso ativo para este utilizador
		0	Registo efetuado com sucesso

Pessoas-ComAcessos *Stored procedure*			Devolve uma lista de pessoas com determinado acesso
In	IdAcesso		Identificador do acesso
Out	Execucao	-60	Erro – Acesso desconhecido
		0	Registo efetuado com sucesso
Out	Resultado	IdPessoa	
		Nome	
		DataInicio	

		DataFim
		Ativo
		Estado
		Notas

g) Interfaces relacionadas com licenças de condução

LicencaConducao-Edita *Stored procedure*			Se a licença referida pelo indicador não existir, cria uma licença de condução, senão altera a informação.
In	RefPessoa		Identificação de pessoa
In	NcartaConducao		Número da carta de condução emitida pelo país de origem da pessoa
In	ValidadeCConducao		Data de validade da carta de condução
In	RefTipoCConducao		Tipo de carta de condução
In	ValidadeLConducao		Data de validade da licença de condução
In	RefTipoLConducao		Referência do tipo de licença de condução
In	DataExame		Data do exame teórico-prático
In	NotaExame		Nota da avaliação do exame
In	CursoExtintores		Marcador que indica que se efetuou o curso de manuseio de extintores
In	VeiculosEspeciais		Marcador que indica se a pessoa tem permissões de condução de veículos especiais
In	LVO		Marcador que indica se a pessoa pode conduzir em LVO
In	Permanente		Marcado que indica se a licença é do tipo permanente ou temporária
In	Ativo		Marcador que indica se a licença está ativa
[In]	Notas		Notas
In	RefOperador		Pessoa que está a executar a operação
Out	Execucao	-2	Erro – Pessoa desconhecida
		-63	Erro – Tipo carta condução desconhecido
		-64	Erro – Tipo licença de condução desconhecida
		-13	Erro – Data invalida
		0	Registo efetuado com sucesso

LicencaConducaoRenovacao-Nova *Stored procedure*			Cria um registo de renovação da licença de condução
In	RefPessoa		Identificação de pessoa
In	DataRenovacao		Data de renovação da licença de condução
In	CursoExtintores		Marcador que indica que se efetuou o curso de manuseio de extintores
In	CartaCondução		Marcador que indica que os dados da carta de condução estão atualizados e conforme os requisitos da licença
In	Exames		Marcador que indica que os exames associados foram realizados com sucesso
[In]	Notas		Notas
In	RefOperador		Pessoa que está a executar a operação
Out	Execucao	-2	Erro – Pessoa desconhecida
		-13	Erro – Data invalida
		0	Registo efetuado com sucesso

LicencaConducaoRenovacao-Edita *Stored procedure*			Edita um registo de renovação da licença de condução
In	NumeroLicenca		Identificação da licença a editar
[In]	DataRenovacao		Data de renovação da licença de condução
[In]	CursoExtintores		Marcador que indica que se efetuou o curso de manuseio de extintores
[In]	CartaCondução		Marcador que indica que os dados da carta de condução estão atualizados e conforme os requisitos da licença
[In]	Exames		Marcador que indica que os exames associados foram realizados com sucesso
[In]	Notas		Notas
In	RefOperador		Pessoa que está a executar a operação
Out	Execucao	-2	Erro – Pessoa desconhecida
		-13	Erro – Data invalida
		0	Registo efetuado com sucesso

InfracaoConducao-Cria *Stored procedure*		Cria uma infração de condução
In	RefPessoa	Referencia ao portador do cartão

In	RefInfracao		Definição da infração
In	DataInfração		Data de registo da infração
In	RefPenalidade		Definição da infração
In	DataPenalidade		Data de execução da penalidade
[In]	Notas		Notas
In	RefOperador		Pessoa que está a executar a operação
Out	Execucao	-2	Erro – Pessoa Desconhecida
		-68	Erro – Infração de condução desconhecida
		-69	Erro – Penalidade de condução desconhecida
		-13	Erro – Data invalida
		0	Registo efetuado com sucesso

InfracaoConducao-Edita *Stored procedure*			Altera a informação de uma infração de condução associada a uma pessoa
In	Id		Identificador da infração
[In]	RefInfracao		Definição da infração
[In]	DataInfração		Data de registo da infração
[In]	RefPenalidade		Definição da infração
[In]	DataPenalidade		Data de execução da penalidade
[In]	Notas		Notas
In	RefOperador		Pessoa que está a executar a operação
Out	Execucao	-2	Erro – Pessoa Desconhecida
		-68	Erro – Infração de condução desconhecida
		-69	Erro – Penalidade de condução desconhecida
		-13	Erro – Data invalida
		0	Registo efetuado com sucesso

Pessoa-InfracaoConducao *Stored procedure*			Devolve a informação sobre todas as infrações de condução atribuídas a uma pessoa
In	IdPessoa		Identificador do utilizador
Out	Execucao	-2	Erro – Pessoa desconhecida
		0	Registo efetuado com sucesso
Out	Resultado	IdPessoa	

		Nome
		Infração
		IdInfracao
		DataInfracao
		IdPenalizacao
		Penalizaçao
		DataPenalizacao
		Notas

Pessoa-LicencaConducao *Stored procedure*			Devolve a informação sobre todas a licença de condução atribuída a uma pessoa
In	IdPessoa		Identificador do utilizador
Out	Execucao	-2	Erro – Pessoa desconhecida
		0	Registo efetuado com sucesso
Out	Resultado	NcartaConducao	
		ValidadeCConducao	
		RefTipoCConducao	
		ValidadeLConducao	
		RefTipoLConducao	
		DataExame	
		NotaExame	
		CursoExtintores	
		VeiculosEspeciais	
		LVO	
		Permanente	
		Ativo	
		Notas	

h) Interfaces relacionadas com viaturas

Viatura-Cria *Stored procedure*			Cria registo de viatura
In	Matricula		Matricula
In	NumeroSerie		Número de serie da viatura
In	DataFabrico		Data de fabrico da viatura
In	RefTipoCombustivel		Tipo de combustível
In	MarcaModelo		Texto a descrever a marca e o modelo da viatura
In	RefTipoVeiculo		Indicação do tipo de veículo
In	RefEntidade		Entidade possuidora da viatura
In	RefServico		Serviço normalmente efetuado pela viatura
In	Distico		Dístico identificador
In	Permanente		Marcador que indica se o dístico é do tipo permanente ou temporário
In	Ativo		Marcador que indica se a licença está ativa
[In]	Notas		Notas
In	RefOperador		Pessoa que está a executar a operação
Out	Execucao	-65	Erro – Tipo de combustível desconhecido
		-66	Erro – Tipo de veículo desconhecido
		-53	Erro – Entidade desconhecida
		-54	Erro – Serviço desconhecido
		-13	Erro – Data invalida
		0	Registo efetuado com sucesso

Viatura-Edita *Stored procedure*		Altera a informação de um registo de viatura
In	ID	Identificador da viatura
[In]	Matricula	Matricula
[In]	NumeroSerie	Número de serie da viatura
[In]	DataFabrico	Data de fabrico da viatura
[In]	RefTipoCombustivel	Tipo de combustível
[In]	MarcaModelo	Texto a descrever a marca e o modelo da viatura

[In]	RefTipoVeiculo		Indicação do tipo de veículo
[In]	RefEntidade		Entidade possuidora da viatura
[In]	RefServico		Serviço normalmente efetuado pela viatura
[In]	Distico		Dístico identificador
[In]	Permanente		Marcador que indica se o dístico é do tipo permanente ou temporário
[In]	Ativo		Marcador que indica se a licença está ativa
[In]	Notas		Notas
In	RefOperador		Pessoa que está a executar a operação
Out	Execucao	-65	Erro – Tipo de combustível desconhecido
		-66	Erro – Tipo de veículo desconhecido
		-53	Erro – Entidade desconhecida
		-54	Erro – Serviço desconhecido
		-13	Erro – Data invalida
		0	Registo efetuado com sucesso

Viatura –Dados *Stored procedure*			Devolve a informação de uma viatura
In	Id		Número de identificação da viatura
Out	Execucao	-5	Erro – Viatura Desconhecida
		0	Registo efetuado com sucesso
	Resultado		ID
			Matricula
			NumeroSerie
			DataFabrico
			RefTipoCombustivel
			Combustivel
			MarcaModelo
			RefTipoVeiculo
			TipoVeiculo
			RefEntidade
			Entidade
			RefServico
			Servico
			Distico

		Permanente
		Ativo
		Notas

Viatura –AnexoCria *Stored procedure*			Adiciona um anexo ao registo de uma viatura
In	Id		Número de identificação da viatura
In	Anexo		Ficheiro com documento para anexar
[In]	Notas		Notas
In	RefOperador		Pessoa que está a executar a operação
Out	Execucao	-5	Erro – Viatura Desconhecida
		0	Registo efetuado com sucesso

Viatura –ApagaAnexo *Stored procedure*			Apaga um anexo ao registo de uma viatura.
In	Id		Número de identificação da vitura
In	AnexoID		Indentificador do anexo a apagar
[In]	Notas		Notas
In	RefOperador		Pessoa que está a executar a operação
Out	Execucao	-5	Erro – Viatura Desconhecida
		-24	Erro – Anexo desconhecido
		0	Registo efetuado com sucesso

Viatura –Anexos *Stored procedure*			Devolve a lista de anexos de uma viatura
In	Id		Número de identificação da viatura
Out	Execucao	-5	Erro – Viatura Desconhecida
		0	Registo efetuado com sucesso
	Resultado		ID
			Anexo
			RefTipoAnexo
			TipoAnexo
			DataRegisto

ViaturaZona-Cria *Stored procedure*			Adiciona um acesso de uma zona, a uma viatura
In	Id		Número de identificação da viatura
In	Zona		Referencia à zona a adicionar
In	DataValidade		Data de validade da atribuição da zona à viatura
In	Ativo		Marcador que indica que a viatura tem acesso à zona
[In]	Notas		Notas
In	RefOperador		Pessoa que está a executar a operação
Out	Execucao	-5	Erro – Viatura Desconhecida
		-67	Erro – Zona desconhecida

ViaturaZona-Edita *Stored procedure*			Altera um acesso de uma zona existente, a uma viatura
In	Id		Número de identificação da viatura
In	Zona		Referencia à zona a adicionar
[In]	DataValidade		Data de validade da atribuição da zona à viatura
[In]	Ativo		Marcador que indica que a viatura tem acesso à zona
[In]	Notas		Notas
In	RefOperador		Pessoa que está a executar a operação
Out	Execucao	-5	Erro – Viatura Desconhecida
		-67	Erro – Zona desconhecida

ViaturaRevalidacao-Cria *Stored procedure*			Cria um registo de revalidação de viatura
In	ID		Identificador da viatura
In	DataValidadeDistico		Data de validade do dístico
In	DataValidadeInspecao		Data de validade da inspeção periódica
In	DataValidadeSeguro		Data de validade do seguro
In	DataValidadeExtintor		Data de validade do extintor
[In]	Notas		Notas
In	RefOperador		Pessoa que está a executar a operação
Out	Execucao	-13	Erro – Data inválida
		0	Registo efetuado com sucesso

ViaturaRevalidacao-Edita *Stored procedure*			Altera a informação de um registo de revalidação de viatura
In	ID		Identificador da viatura
[In]	DataValidadeDistico		Data de validade do dístico
[In]	DataValidadeInspecao		Data de validade da inspeção periódica
[In]	DataValidadeSeguro		Data de validade do seguro
[In]	DataValidadeExtintor		Data de validade do extintor
[In]	Notas		Notas
In	RefOperador		Pessoa que está a executar a operação
Out	Execucao	-13	Erro – Data inválida
		0	Registo efetuado com sucesso

Viatura –Revalidacao *Stored procedure*			Devolve a informação das revalidações de uma viatura
In	Id		Número de identificação da viatura
Out	Execucao	-5	Erro – Viatura Desconhecida
		0	Registo efetuado com sucesso
	Resultado		ID
			DataValidadeDistico
			DataValidadeInspecao
			DataValidadeSeguro
			DataValidadeExtintor
			Notas
			Ativo

i) Interfaces relacionadas com cartões pontuais de visitas

Visitante-CriaEdita *Stored procedure*			Cria um registo de visitante ou altera informação de um já existente
In	Id		Número do documento de identificação
In	Nome		Nome
In	DataNascimento		Data de nascimento
In	RefNacionalidade		Indicação da nacionalidade
In	RefTipoID		Referencia ao tipo de documento usado para identificação
In	ValidadeID		Validade do documento de identificação
In	RefEntidade		Referencia à entidade que representa
[In]	Notas		Campo de texto livre para informações complementares
In	RefOperador		Pessoa que está a executar a operação
Out	Execucao	-12	Erro – Documento de identificação caducado
		-51	Erro – País desconhecido
		-52	Erro – Tipo de documento de identificação desconhecido
		-53	Erro – Entidade desconhecida
		-13	Erro – Data invalida
		0	Registo efetuado com sucesso

Visitante-Dados *Stored procedure*			Devolve a informação de um visitante
In	Id		Número do documento de identificação
Out	Execucao	-10	Erro – Visitante desconhecido
	Resultado		Id
			Nome
			DataNascimento
			RefNacionalidade
			Nacionalidade
			RefTipoID
			TipoId

			ValidadeID
			RefEntidade
			Notas

Visita –AnexoCria *Stored procedure*			Adiciona um anexo ao registo de um cartão pontual de visitante
In	Id		Número de identificação do cartão pontual
In	Anexo		Ficheiro com documento para anexar
[In]	Notas		Notas
In	RefOperador		Pessoa que está a executar a operação
Out	Execucao	-33	Erro – Cartão pontual desconhecido
		0	Registo efetuado com sucesso

Visita –ApagaAnexo *Stored procedure*			Apaga um anexo ao registo de um cartão pontual de visitante.
In	Id		Número de identificação do cartão pontual
In	AnexoID		Indentificador do anexo a apagar
In	RefOperador		Pessoa que está a executar a operação
Out	Execucao	-33	Erro – Cartão pontual desconhecido
		-24	Erro – Anexo desconhecido
		0	Registo efetuado com sucesso

Visita –Anexos *Stored procedure*			Devolve os anexos de um visitante
In	String[12]		Identificação do visitante
Out	Execucao	-10	Erro – Visitante desconhecido
		0	Registo efetuado com sucesso
	Resultado		Id
			Anexo
			Notas

CartaoPontual-Cria *Stored procedure*			Cria cartão pontual para visita.
In	RefVisita		Referencia ao visitante
In	RefSolicitador		Referencia a pessoa registada que solicita credenciação para o visitante
In	DataInicio		Data de inicio da visita
In	DataFim		Data de fim da visita
In	Ferramentas		Marcador que indica que o visitante é portador de ferramentas
In	Ativo		Estado do cartão ático
[In]	Notas		Campo de texto livre para informações complementares
In	RefOperador		Pessoa que está a executar a operação
Out	Execucao	-2	Erro – Pessoa desconhecida
		-13	Erro – Data invalida
		0	Registo efetuado com sucesso

CartaoPontual-Edita *Stored procedure*			Altera informação de cartão pontual para visita.
In	Id		Identificador do cartão
[In]	RefSolicitador		Referencia a pessoa registada que solicita credenciação para o visitante
[In]	DataInicio		Data de inicio da visita
[In]	DataFim		Data de fim da visita
[In]	Ferramentas		Marcador que indica que o visitante é portador de ferramentas
[In]	Ativo		Estado do cartão ático
In	RefOperador		Pessoa que está a executar a operação
Out	Execucao	-2	Erro – Pessoa desconhecida
		0	Registo efetuado com sucesso

CartaoPontual-Emitido *Stored procedure*			Coloca um cartão pontual no estado de emitido
In	Id		Identificador do cartão
In	IdPessoa		Identificação do portador do cartão
In	RefOperador		Pessoa que está a executar a operação
Out	Execucao	-2	Erro – Pessoa desconhecida
		0	Registo efetuado com sucesso

CartaoPontual-AtribuiAcesso *Stored procedure*			Associa um acesso codificado em letras a um cartão pontual de visitante
In	IdCartao		Identificador do do cartão pontual
In	IdAcesso		Identificador do Acesso
In	DataInicio		Data de início de validade do acesso
In	DataFim		Data de validade
In	Ativo		Marcador que indica se o acesso está ativo
[In]	Notas		Campo de texto livre para informações complementares
In	RefOperador		Pessoa que está a executar a operação
Out	Execucao	-2	Erro – Pessoa desconhecida
		-6	Erro – Cartão desconhecido
		-60	Erro – Acesso desconhecido
		-29	Erro – Já existe este acesso ativo para este utilizador
		-10	Erro – Data de início posterior à data de fim
		-11	Erro – Data de fim anterior à data atual
		0	Registo efetuado com sucesso

CartaoPontual-Acesso Edita *Stored procedure*			Altera a informação da associação um acesso codificado em letras a um cartão pontual de visitante.
In	IdCartao		Identificador do cartão pontual
[In]	DataInicio		Data de início de validade do acesso
[In]	DataFim		Data de validade
[In]	Ativo		Marcador que indica se o acesso está ativo
[In]	Notas		Campo de texto livre para informações complementares
In	RefOperador		Pessoa que está a executar a operação
Out	Execucao	-30	Erro – Atribuição de acesso desconhecida
		-6	Erro – Cartão desconhecido
		-10	Erro – Data de início posterior à data de fim
		-11	Erro – Data de fim anterior à data atual
		0	Registo efetuado com sucesso

CartaoPontual-Acessos *Stored procedure*			Acessos devolve os acessos de um cartão pontual
In	IdCartao		Identificador do do cartão pontual
Out	Execucao	-10	Erro – Cartão desconhecido
	Resultado		IdCartao
			DataInicio
			DataFim
			Ativo
			Notas

Visitante-CartoesPontuais *Stored procedure*			Devolve a informação dos cartões pontuais de um visitante
In	Strin[12]		Identificação da pessoa
Out	Execucao	-2	Erro – Visitante desconhecido
	Resultado		Id
			Nome
			DataNascimento
			RefNacionalidade
			Nacionalidade
			RefTipoID
			TipoId
			ValidadeID
			RefEntidade
			Notas

j) Interfaces relacionadas com cartões pontuais de passageiros

Passageiro-CriaEdita *Stored procedure*			Cria um registo de um passageiro ou atualiza a informação de registos já existentes
In	Id		Número do documento de identificação
In	Nome		Nome
In	DataNascimento		Data de nascimento
In	RefNacionalidade		Indicação da nacionalidade
In	RefTipoID		Referencia ao tipo de documento usado para identificação
In	ValidadeID		Validade do documento de identificação
[In]	Notas		Campo de texto livre para informações complementares
In	RefOperador		Pessoa que está a executar a operação
Out	Execucao	-12	Erro – Documento de identificação caducado
		-51	Erro – País desconhecido
		-52	Erro – Tipo de documento de identificação desconhecido
		-13	Erro – Data invalida
		0	Registo efetuado com sucesso

Passageiro-Dados *Stored procedure*			Devolve a informação de um passageiro
In	Id		Número do documento de identificação
Out	Execucao	-2	Erro – Pessoa desconhecida
		0	Registo efetuado com sucesso
	Resultado		Id
			Nome
			DataNascimento
			RefNacionalidade
			Nacionalidade
			RefTipoID
			TipoID
			ValidadeID

CartaoPax-Cria *Stored procedure*			Cria um registo de cartão de acesso pontual para passageiro.
In	PassageiroId		Identificador do passageiro
In	DataValidadeCartao		Data de validade do cartão pontual
In	RefCompanhia		Companhia aérea
In	Voo		Identificação do voo
In	DataVoo		Data do voo
In	HoraVoo		Hora do voo
In	Ativo		Marcador para indicar se o registo está ativo
[In]	Notas		Campo de texto livre para informações complementares
Out	Execucao	-12	Erro – Documento de identificação caducado
		-11	Erro – Visitante desconhecido
		-70	Erro – Companhia aérea desconhecida
		-13	Erro – Data invalida
		0	Registo efetuado com sucesso

CartaoPax-Edita *Stored procedure*			Altera a informação de um registo de cartão de acesso pontual para passageiro.
In	CartaoId		Identificador do cartão temporário
[In]	DataValidadeCartao		Data de validade do cartão pontual
[In]	RefCompanhia		Companhia aérea
[In]	Voo		Identificação do voo
[In]	DataVoo		Data do voo
[In]	HoraVoo		Hora do voo
[In]	Ativo		Marcador para indicar se o registo está ativo
[In]	Notas		Campo de texto livre para informações complementares
Out	Execucao	-12	Erro – Documento de identificação caducado
		-70	Erro – Companhia aérea desconhecida
		-13	Erro – Data invalida
		0	Registo efetuado com sucesso

Passageiro-CartoesPontuais *Stored procedure*			Devolve a informação dos cartões pontuais de um passageiro
In	Strin[12]		Identificação da pessoa
Out	Execucao	-2	Erro – Visitante desconhecido
	Resultado		Id
			DataValidadeCartao
			RefCompanhia
			CompanhiaAerea
			Voo
			DataVoo
			HoraVoo
			Ativo
			Notas

k) Interfaces criadas na base de dados PWNT

Nesta secção são apresentadas as interfaces a serem implementadas na base de dados do *Pro-Watch*. O nome das interfaces – *stored procedures*, a criar é iniciado pelo prefixo spana_, para os distinguir das implementações nativas.

aaspBadge_Acessos	Devolve os acessos de um cartão.
aaspBadge_ClearenceCode_LogicalDevice	Devolve os *clearence codes* e os respetivos *logical devices* atribuídos a um cartão.
aaspBadge_DadosUtilizador	Devolve a informação do utilizador de um cartão.
aaspClearenceCode_Badge	Devolve os *clearence codes* atribuídos a um cartão.
aaspClearenceCode_Company	Devolve as *company* que estão associadas a um *clearence codes.*
aaspClearenceCode_LogicalDevice	Devolve os *logical devices* associados a um *clearence codes.*
aaspCompany_Badge	Devolve a lista de cartões associados a uma *company.*
aaspCompany_ClearenceCode	Devolve os *clearence codes* associados a uma *company.*
aaspCompany_Id	Devolve a informação de uma *company* identificada por um id.
aaspCompany_LogicalDevices	Devolve os *logical devices* associados a uma *company.*
aaspCriaNovoCartao	Cria um novo cartão permanente
aaspCriaNovoCartaoT	Cria um novo cartão temporário
aaspLogicalDevice_Badge	Devolve a lista de cartões associados a um cartão
aaspLogicalDevice_ClearenceCode	Devolve a *logical devices* associados a um *Clearence code*

CriaNovoCartao *Stored procedure*			Cria um registo de cartão de identificação de carater temporario
In	NumeroCartão		Número do cartão
In	IdCompany		Número identificador da *company* definida no *Pro-Watch*
In	DataEmissão		Data de emissão do cartão
In	DataValidade		Data de validade do cartão
In	PrimeiroNome		Primeiro nome do portador
In	UltimoNome		Ultimo nome do portador
In	NomeCompleto		Nome completo do portador
In	NDocId		Número do documento de identificação do portador
In	Entidade		Entidade que o portador representa
In	Funcao		Função que o portador do cartão desempenha
Out	Execucao	-53	Erro – Entidade desconhecida
		-54	Erro – Serviço desconhecido
		-55	Erro – Função desconhecida
		-2	Erro – Pessoa Desconhecida
		-10	Erro – Data de início posterior à data de fim
		-11	Erro – Data de fim anterior à data atual
		-13	Erro – Data invalida
		0	Registo efetuado com sucesso

CriaNovoCartaoT *Stored procedure*			Cria um registo de cartão de identificação.
In	NumeroCartão		Número do cartão
In	IdCompany		Número identificador da *company* definida no *Pro-Watch*
In	DataEmissão		Data de emissão do cartão
In	DataValidade		Data de validade do cartão
In	PrimeiroNome		Primeiro nome do portador
In	UltimoNome		Ultimo nome do portador
In	NomeCompleto		Nome completo do portador
In	NDocId		Número do documento de identificação do portador
In	Entidade		Entidade que o portador representa
In	Funcao		Função que o portador do cartão desempenha
Out	Execucao	-53	Erro – Entidade desconhecida

		-54	Erro – Serviço desconhecido
		-55	Erro – Função desconhecida
		-2	Erro – Pessoa Desconhecida
		-10	Erro – Data de início posterior à data de fim
		-11	Erro – Data de fim anterior à data atual
		-13	Erro – Data invalida
		0	Registo efetuado com sucesso

l) Interface para ser usada com o Portal Cartão do Aeroporto

Esta secção faz a listagem das interfaces implementadas para troca de informação entre a plataforma de credenciação e a plataforma do Cartão do Aeroporto.

NovaPessoa *Stored procedure*			Envia informação para a base de dados da plataforma de credenciação para a criação de uma nova pessoa.
Executa a interface: Pessoa-Cria			
In	Id		Número do documento de identificação
In	Nome		Nome
In	DataNascimento		Data de nascimento
In	RefNacionalidade		Indicação da nacionalidade
In	Filiação Materna		Nome da mãe
In	Filiação Paterna		Nome do pai
In	RefTipoID		Referencia ao tipo de documento usado para identificação
In	ValidadeID		Validade do documento de identificação
[In]	Foto		Foto
[In]	Palavra-chave		Palavra-chave
In	RefEntidade		Referencia à entidade que representa
In	RefServico		Referencia ao dicionário do serviço a que faz assistência
In	RefFuncao		Referencia ao dicionário da função que desempenha
[In]	Notas		Campo de texto livre para informações complementares
Out	Execucao	-12	Erro – Documento de identificação caducado
		-51	Erro – País desconhecido
		-52	Erro – Tipo de documento de identificação desconhecido
		-53	Erro – Entidade desconhecida
		-54	Erro – Serviço desconhecido
		-55	Erro – Função desconhecida
		-13	Erro – Data invalida
		0	Registo efetuado com sucesso

NovaEntidade *Stored procedure*			Envia informação para a base de dados da plataforma de credenciação para a criação de uma nova entidade.
Executa a interface: Entidade-Cria			
In	Nome		Nome da empresa
In	NIF		Número de identificação fiscal da empresa
[In]	RefSignatário		Referencia à pessoa que representa a empresa
In	RefTipoPagamento		Referencia à forma de pagamento dos custos associados à credenciação de pessoa e viaturas
[In]	Notas		Campo de texto livre para informações complementares
Out	Execucao	-2	Erro – Pessoa desconhecida
		-57	Erro – Tipo de pagamento desconhecido
		0	Registo efetuado com sucesso

AtribuicaoCartao *Stored procedure*			Envia informação para a base de dados da plataforma de credenciação para a criação ou renovação de um cartão de uma pessoa
Executa a interface: Cartao –Cria ou Cartao –Revalidação dependendo do valor no campo "Novo" **.**			
In	Id		Número de identificação do cartão
In	RefPessoa		Referencia ao portador do cartão
In	DataInicio		Data de início de validade do cartão
In	DataFim		Data de validade do cartão
In	Ativo		Marcador que indica se o cartão está em uso
In	Permanente		Marcador que indica se o cartão é do tipo permanente ou temporário
In	RefEstado		Referencia ao estado do cartão
In	Novo		Marcador que indica se o cartão é novo ou revalidação
[In]	Notas		Notas
Out	Execucao	-2	Erro – Pessoa Desconhecida
		-10	Erro – Data de início posterior à data de fim
		-11	Erro – Data de fim anterior à data atual
		-26	Erro – Já existe um cartão ativo para este utilizador
		-58	Erro – Estado de cartão desconhecido
		0	Registo efetuado com sucesso

ListaCartoesPessoa *Stored procedure*			Devolve a informação sobre os cartões de uma pessoa
Executa a interface: Cartao-Pessoa			
In	IdPessoa		Identificador do utilizador
Out	Execucao	-2	Erro – Pessoa desconhecida
		0	Registo efetuado com sucesso
Out	Resultado	IdPessoa	
		Nome	
		NumeroCartao	
		DataInicio	
		DataFim	
		Ativo	
		Permanente	
		IdEstado	
		Estado	
		Notas	

ListaPessoasEntidade *Stored procedure*			Devolve informação sobre todas as pessoas associadas a uma entidade
In	IdEntidade		Identificador da entidade
Out	Execucao	-3	Erro - Entidade Desconhecida
		0	Sucesso
Out	Resultado	IdPessoa	
		Nome	
		NumeroCartao	
		DataInicio	
		DataFim	
		Ativo	
		Permanente	
		IdEstado	
		Estado	
		Notas	

ListaErros *View*		Devolve informação sobre os erros despoletados pela interface da plataforma de credenciação
Out	Resultado	Número de erro
		Descrição do erro

m) Interfaces da base de dados Portarias

Esta secção faz a listagem das interfaces implementadas na base de dados que apoiam o funcionamento da aplicação *Port*.

<table>
<tr><td colspan="2">CartaoAcedePortaria
Stored procedure</td><td colspan="2">Esta rotina recebe o número de um cartão e devolve três tipos de resultados:
Passagem permitida
Passagem proibida
Cartão desconhecido
Nos dois primeiros casos devolve também informações sobre o portador do cartão, como nome, foto, etc.</td></tr>
<tr><td>In</td><td colspan="2">IdCartao</td><td>Número de série do cartão RFID</td></tr>
<tr><td>In</td><td colspan="2">IdPortaria</td><td>Identificador da portaria</td></tr>
<tr><td>In</td><td colspan="2">IdVigilante</td><td>Identificação do vigilante que opera na aplicação</td></tr>
<tr><td rowspan="5">Out</td><td rowspan="5">Execucao</td><td>-6</td><td>Erro - Cartão desconhecido</td></tr>
<tr><td>-4</td><td>Erro - Portaria Desconhecida</td></tr>
<tr><td>-2</td><td>Erro - Pessoa desconhecida - Vigilante</td></tr>
<tr><td>-34</td><td>Erro - Cartão de identificação caducado</td></tr>
<tr><td>0</td><td>Registo efetuado com sucesso</td></tr>
<tr><td rowspan="2"></td><td rowspan="2">Passagem</td><td>1</td><td>Portaria - Passagem permitida</td></tr>
<tr><td>2</td><td>Portaria - Passagem proibida</td></tr>
<tr><td rowspan="8">Out</td><td colspan="2" rowspan="8">[Resultado]

Apenas apresentado se o cartão estiver registado</td><td>Nome</td></tr>
<tr><td>NumeroCartao</td></tr>
<tr><td>DataValidade</td></tr>
<tr><td>Foto</td></tr>
<tr><td>Entidade</td></tr>
<tr><td>TipoLicencaCondução</td></tr>
<tr><td>ObjetosProibidos</td></tr>
<tr><td>AnexoObjetosProibidos</td></tr>
</table>

RegistaPassagemSemResultado *Stored procedure*			Regista a apresentação de um cartão numa portaria, sem mostrar dados ao vigilante
In	IdCartao		Número de série do cartão RFID
In	IdPortaria		Identificador da portaria
In	IdVigilante		Identificação do vigilante que opera na aplicação
Out	Execucao	-2	Erro - Pessoa desconhecida - Vigilante
		-4	Erro - Portaria Desconhecida

RegistaTamper *Stored procedure*			Regista a ativação do sinal de tamper
In	IdPortaria		Identificador da portaria
In	IdVigilante		Identificação do vigilante que opera na aplicação
Out	Execucao	-2	Erro - Pessoa desconhecida - Vigilante
		-4	Erro - Portaria Desconhecida

Pessoa-VerificaPalavraChave *Stored procedure*			Esta rotina recebe o número de identificação do vigilante, e faz a autenticação de entrada na aplicação *Port*
In	IdCartao		Número de série do cartão RFID do vigilante
In	IdPortaria		Identificador da portaria
Out	Execucao	-6	Erro - Cartão desconhecido
		-4	Erro - Portaria Desconhecida
		-34	Erro - Cartão de identificação caducado
		0	Registo efetuado com sucesso
	Passagem	3	Portaria - Login autorizada
		4	Portaria - Login não autorizada
Out	[Resultado]		Nome Vigilante
			NumeroCartao

Pessoa-Logout *Stored procedure*		Esta rotina recebe o número de identificação do vigilante, e faz o registo de saída da aplicação *Port*
In	IdCartao	Número de série do cartão RFID do vigilante
In	IdPortaria	Identificador da portaria

Out	Execucao	-6	Erro - Cartão desconhecido
		-4	Erro - Portaria Desconhecida
		-34	Erro - Cartão de identificação caducado
		0	Registo efetuado com sucesso

Portarias-PassagemEm *Stored procedure*		Devolve a informação de passagens numa portaria, no período indicado
In	IdPortaria	Identificador da portaria
In	DataInicio	
In	DataFim	
Out	Resultado	RefTipoLog
		TipoLog
		RefVigilante
		Vigilante
		Cartao
		Operacao
		DataHora

Pessoa-PassagemEm *Stored procedure*		Devolve a informação de passagens de uma pessoa em portarias, no período indicado
In	IdPPessoas	Identificador de pessoas
In	DataInicio	
In	DataFim	
Out	Resultado	RefTipoLog
		TipoLog
		RefPortaria
		Portaria
		RefVigilante
		Vigilante
		Operacao
		DataHora

n) Interfaces relacionadas com palavras-chave

Existem um conjunto de *stored procedures* que face a um valor de texto devolvem a respetiva chave primária. A tabela seguinte apresenta a lista desses interfaces.

Acessos-Id
CaminhoDeFicheiros-Id
Combustivel-Id
CompanhiaAereaICAO-Id
DocumentoId-Id
Entidade-Id
EstadoCartao-Id
Funcao-Id
Infracao-Id
Nacionalidade-Id
PenalidadeConducao-Id
Penalidade-Id
Perfil-Id
Portaria-Id
Servico-Id
ServicoViatura-Id
TipoAnexo-Id
TipoCartaConducao-Id
TipoLicencaConducao-Id
TipoPagamento-Id
Veiculo-Id

o) Interfaces relacionadas com dicionários

DicServiço *View*	Lista de serviços que as pessoas executam
Id	
Servico	
Notas	

DicFunções *View*	Lista de funções que as pessoas desempenham
Id	
Funcao	
Notas	

DicTipoPagamento *View*	Lista de formas de pagamento dos custos associados à credenciação
Id	
TipoPagamento	
Notas	

DicDocumentoID *View*	Lista de documentos de identificação aceitáveis
Id	
TipoDocumento	

DicEstadoCartao *View*	Lista de estados de processamento dos cartões de identificação
Id	
EstadoCartao	
Notas	

DicPerfil *View*	Lista de perfis dos utilizadores
Id	
Perfil	

DicInfracao *View*	Lista infrações relacionadas com segurança
Id	
Infracao	
Notas	

DicPenalidade *View*	Lista de penalidades relacionadas com segurança
Id	
Penalidade	
Notas	

DicAcessoLetra *View*	Lista códigos de zonas de acesso representadas em letras
LAcesso	
Notas	

DicAcessoCor *View*	Lista códigos de zonas de acesso representadas em cores
Id	
CAcesso	
Notas	

DicTipoCartaConducao *View*	Lista de tipos de cartas de condução emitidas pelos países
Id	
TipoCConducao	

Notas

DicTipoLicencaConducao *View*	Lista de tipos de licença de condução para condução em zonas restritas
Id	
TipoLConducao	
Notas	

DicInfracaoConducao *View*	Lista infrações relacionadas com condução em zonas restritas
Id	
InfracaoC	
Notas	

DicPenalidadeConducao *View*	Lista de penalidades relacionadas com condução em zonas restritas
Id	
PenalidadeC	
Notas	

DicCombustivel *View*	Lista de tipos de combustíveis usados em viaturas
Id	
Combustivel	
Notas	

DicTipoVeiculo *View*	Lista de tipos de veículos
Id	
TipoVeiculo	

Notas

DicServicoViatura *View*	Lista de serviços a que as viaturas estão adstritas
Id	
ServicoV	
Notas	

DicZonasViatura *View*	Lista de zonas em que as viaturas podem circular
Id	
Zonas	
Notas	

DicCompanhiaAerea *View*	Lista de companhias aéreas
Id	
CompanhiaA	
Notas	

DicTipoLog *View*	Lista tipos de registo de eventos
Id	
TipoLog	
Notas	

DicClassificacaoLog *View*	Lista de classificações dos registos de eventos
Id	
ClassificacaoLog	

Notas

DicAplicacao *View*	Lista das aplicações que constituem a plataforma
Id	
Aplicacao	

DicCodigoErro *View*	Lista de códigos de erro devolvidos pelas rotinas programadas no *SQL Server*
NumeroErro	
DescricaoErro	

DicTipoAnexo *View*	Lista de classificações dos documentos guardados em anexo
Id	
TipoAnexo	
Notas	

DicPais *View*	Lista de paises
Id	
Pais	

p) Outras Interfaces

BaseDadosPing *Stored procedure*	Devolve a informação data/hora do servidor da base de dados	
Out	Resultado	DataHora

Anexo M Ecrãs da aplicação CRED

Este anexo apresenta a lista de ecrãs da aplicação CRED, organizados por funcionalidades.

a) Ecrãs do menu “Pessoas”

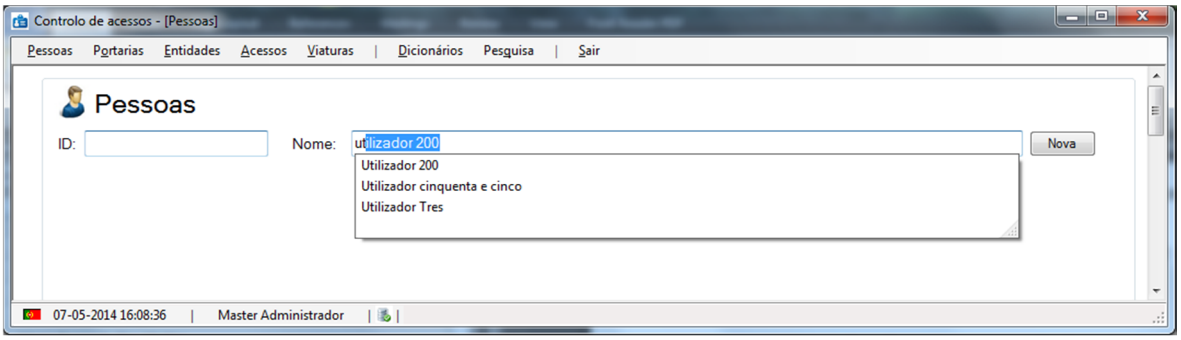


Figura 244 – CRED – Ecrã do menu “Pessoas”: seleção de pessoa.

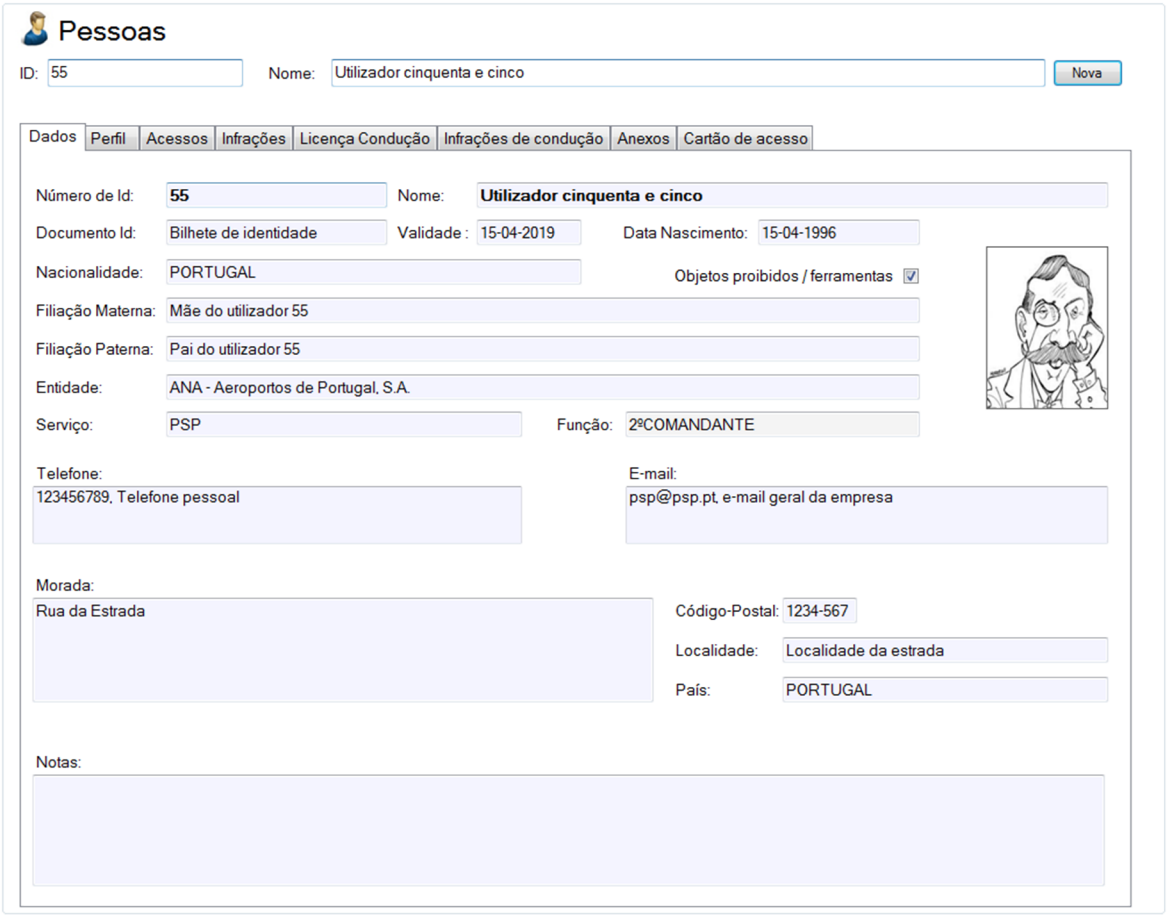


Figura 245 – CRED – Ecrã do menu "Pessoas": Dados gerais.

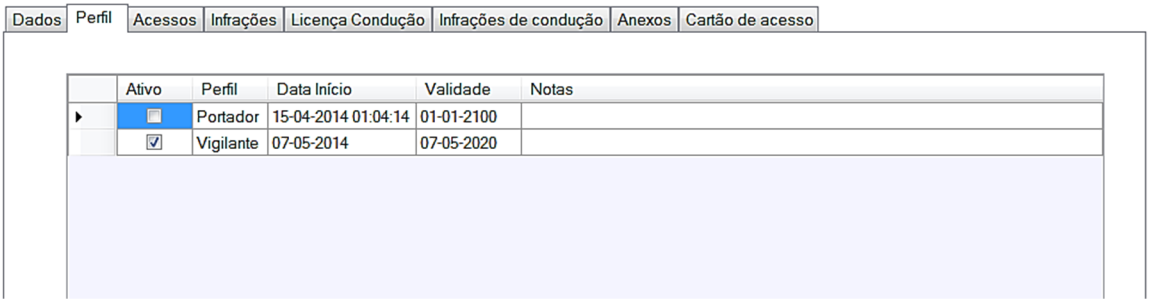


Figura 246 – CRED – Ecrã do menu "Pessoas": Perfil.

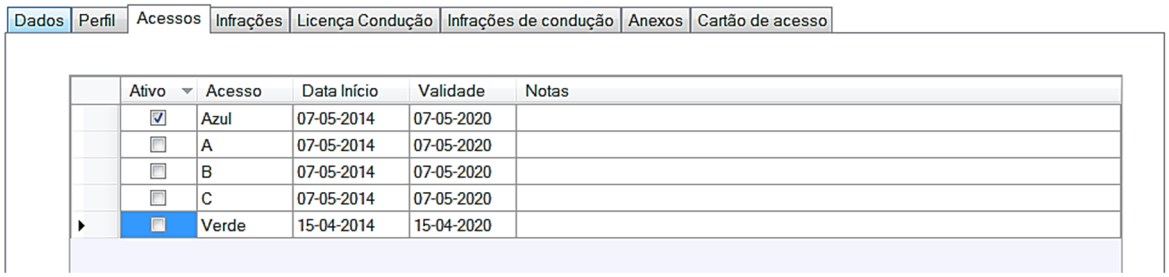


Figura 247 – CRED – Ecrã do menu "Pessoas": Acessos.

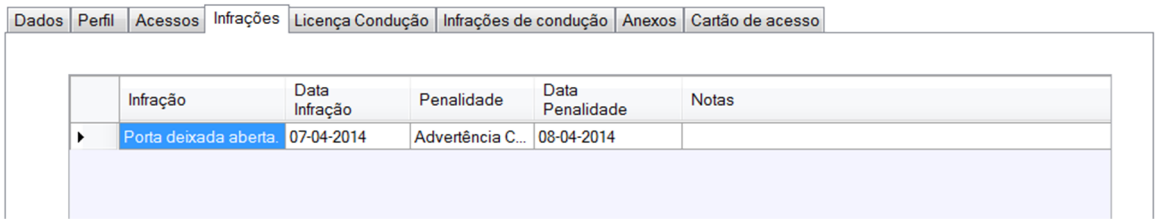


Figura 248 – CRED – Ecrã do menu "Pessoas": Infrações.

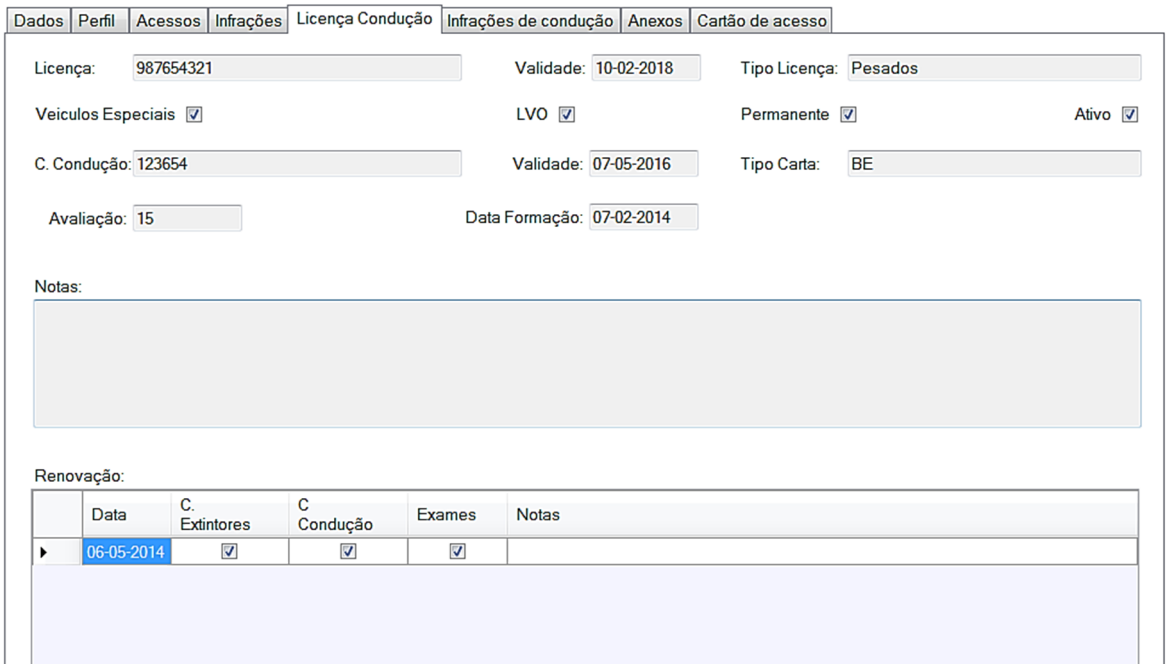


Figura 249 – CRED – Ecrã do menu "Pessoas": Licença de condução.

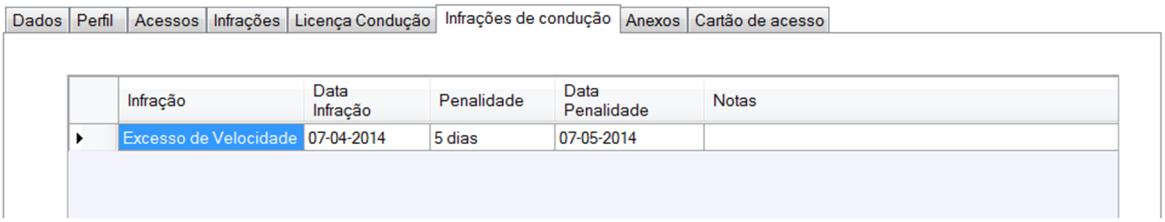


Figura 250 – CRED – Ecrã do menu "Pessoas": Infrações de condução.

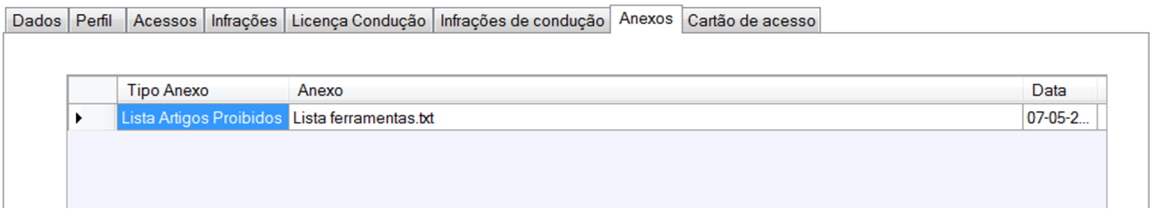


Figura 251 – CRED – Ecrã do menu "Pessoas": Anexos.

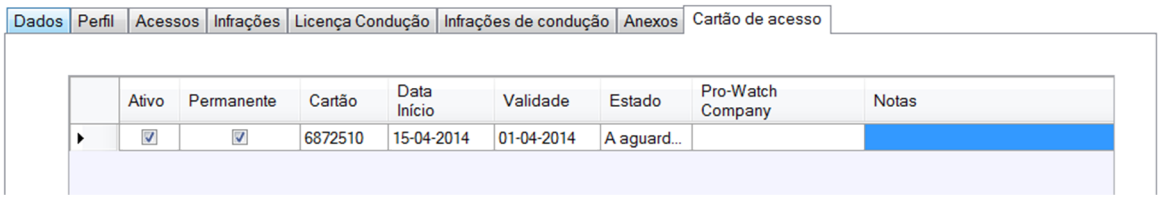


Figura 252 – CRED – Ecrã do menu "Pessoas": Cartão de acesso.

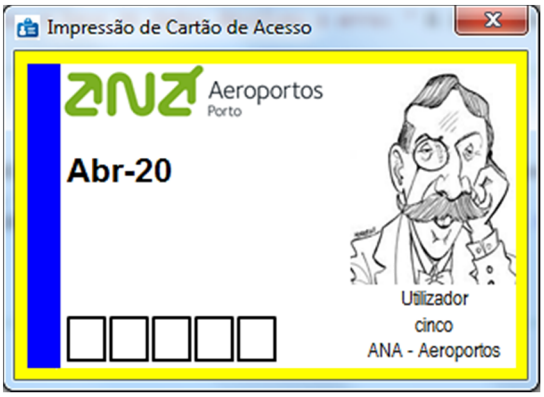


Figura 253 – CRED – Ecrã do menu "Pessoas": Impressão de Cartão de acesso.

b) Ecrãs do menu “Portarias”

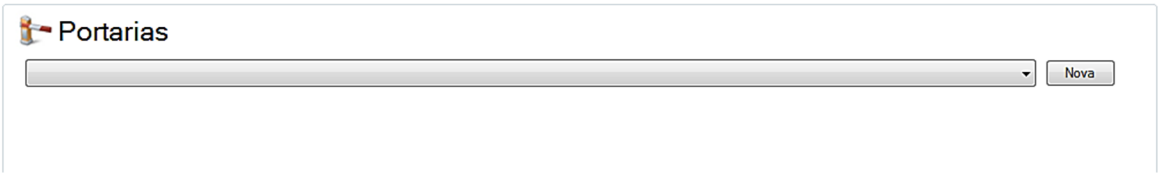


Figura 254 – CRED – Ecrã do menu “Portarias”.

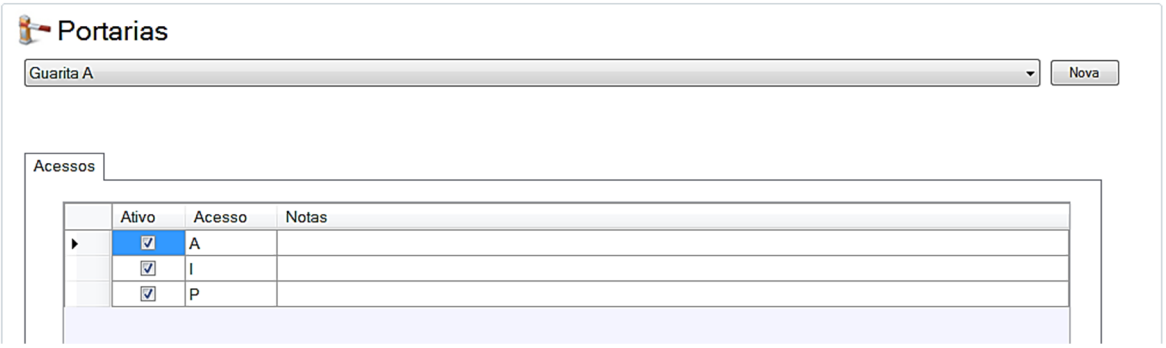


Figura 255 – CRED – Ecrã do menu “Portarias”: Acessos.

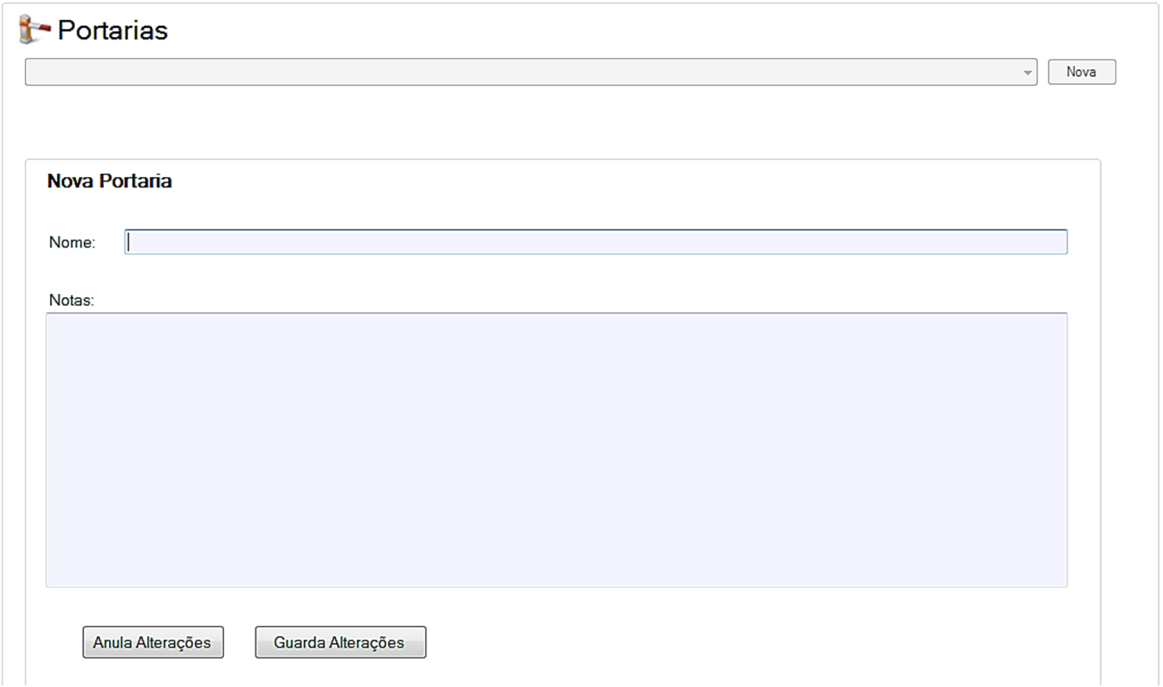


Figura 256 – CRED – Ecrã do menu “Portarias”: Nova portaria.

c) Ecrãs do menu “Entidades”

Figura 257 – CRED – Ecrã do menu “Entidades”.

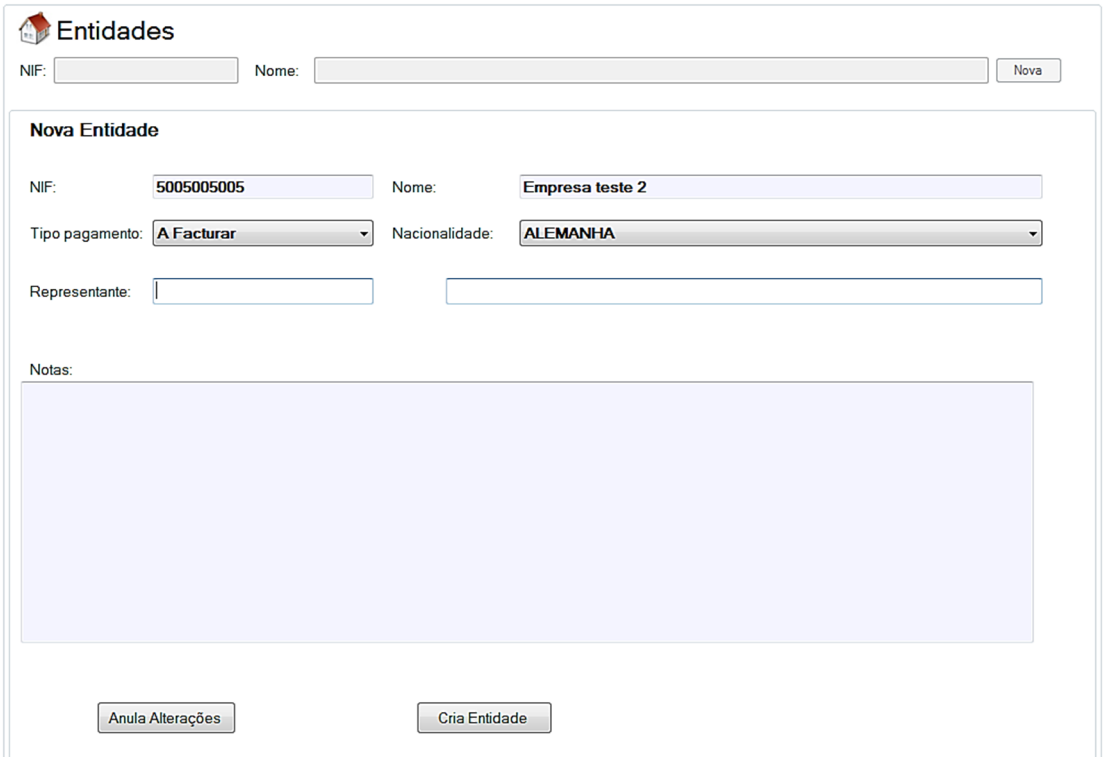


Figura 258 – CRED – Ecrã do menu “Entidades” – Dados gerais.

d) Ecrãs do menu “Acessos”

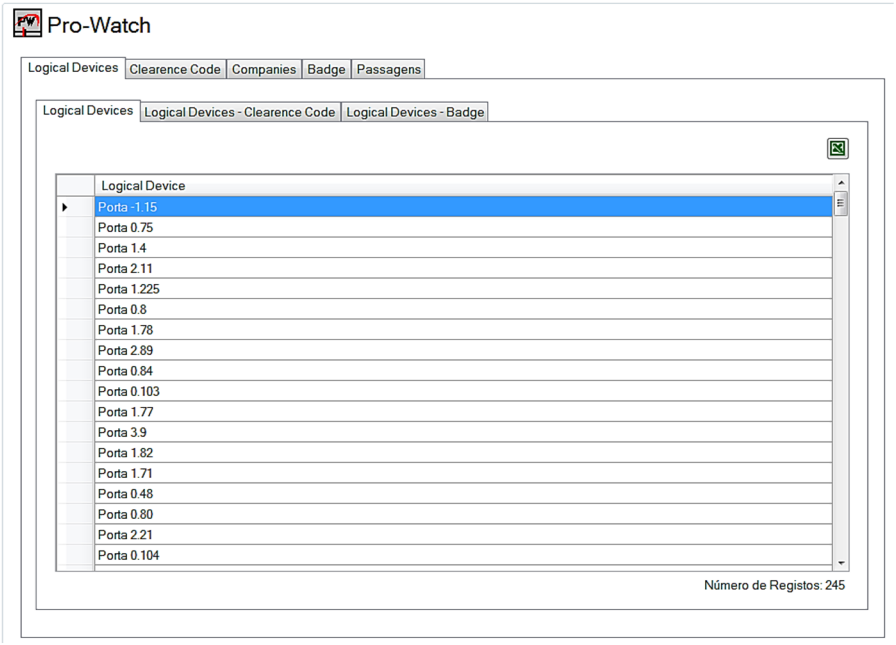


Figura 259 – CRED – Ecrã do menu “Acessos” – *Logical Devices*.

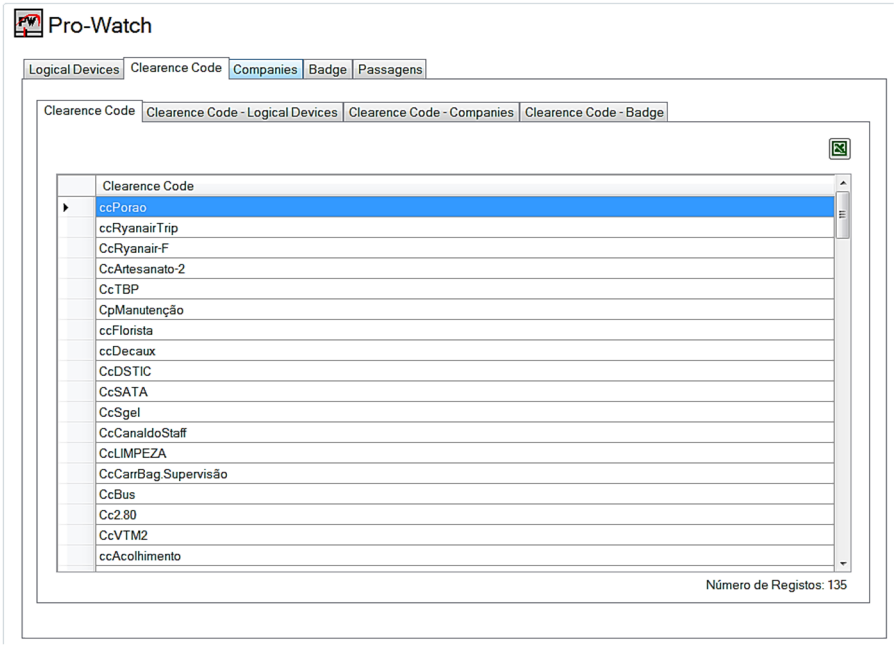


Figura 260 – CRED – Ecrã do menu “Acessos” – *Clearance Codes*.

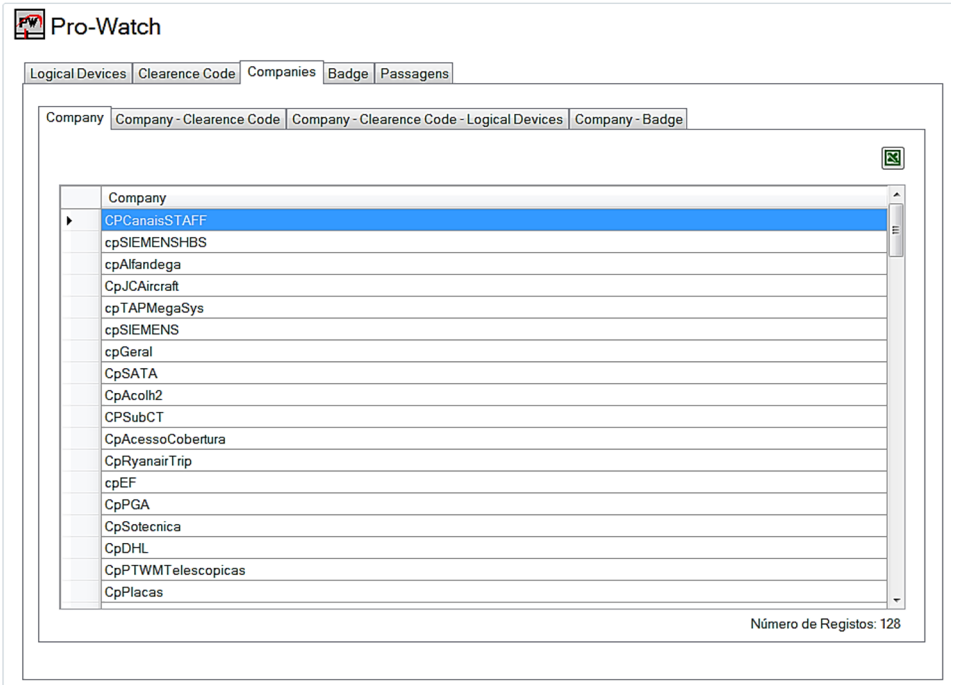


Figura 261 – CRED – Ecrã do menu "Acessos" - *Companies*.

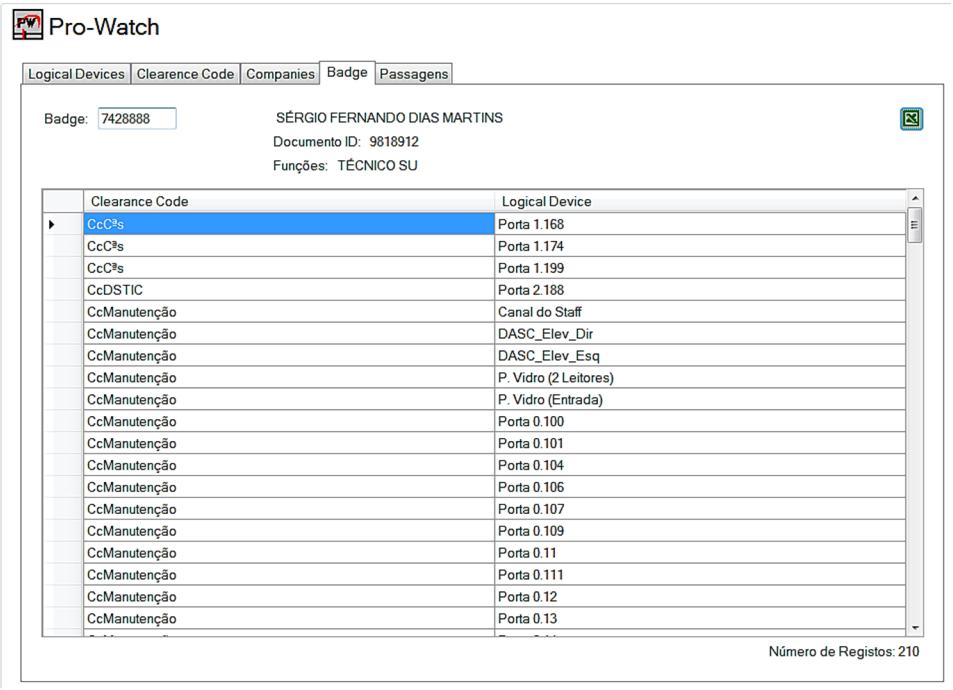


Figura 262 – CRED – Ecrã do menu "Acessos" - *Badge*.

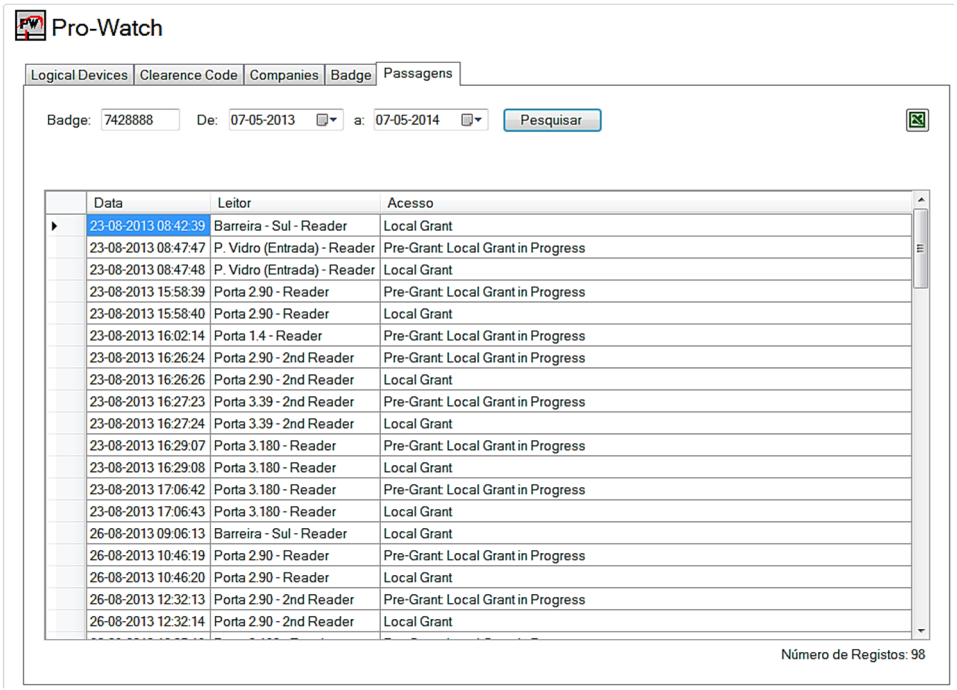


Figura 263 – CRED – Ecrã do menu "Acessos" - Passagens.

e) Ecrãs do menu “Viaturas”

Figura 264 – CRED – Ecrã do menu “Viaturas”.

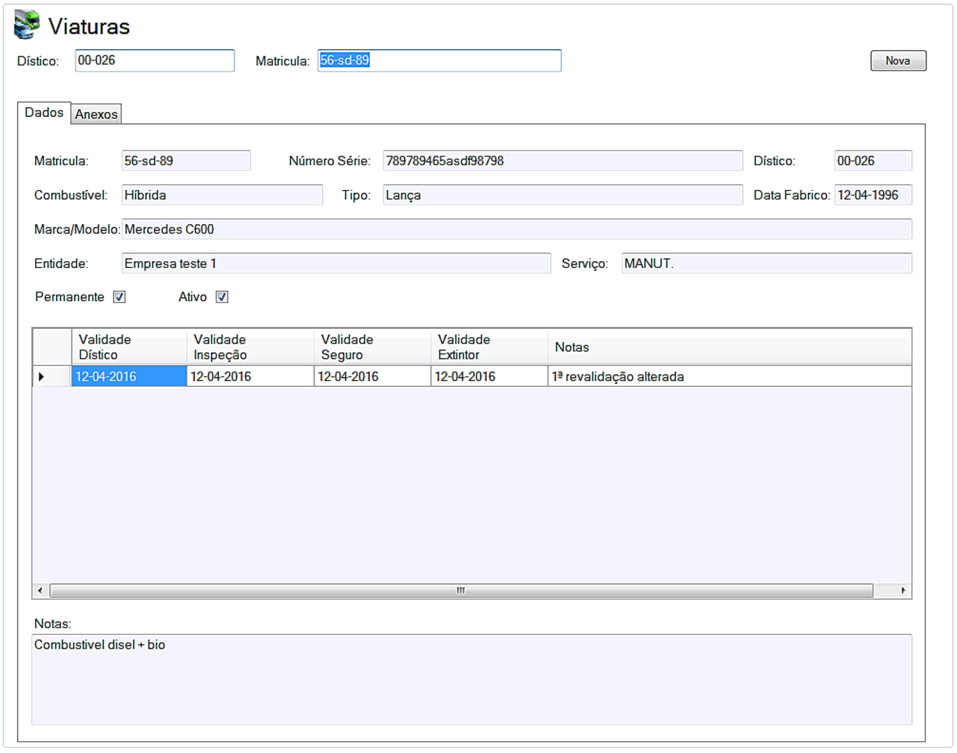


Figura 265 – CRED – Ecrã do menu “Viaturas” – Dados Gerais.

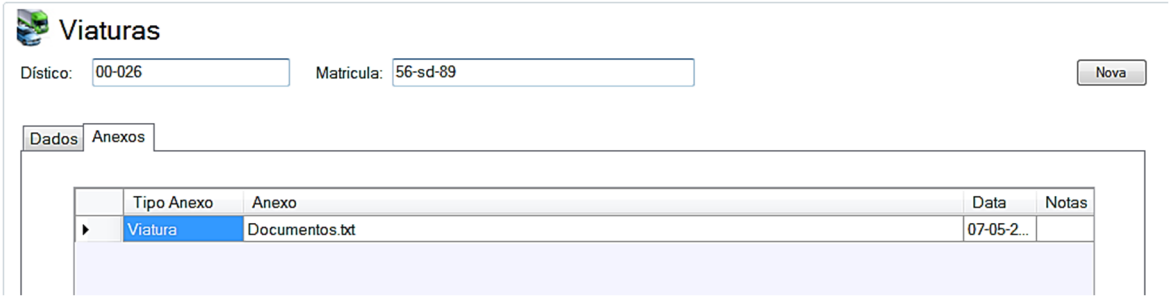


Figura 266 – CRED – Ecrã do menu “Viaturas” - Anexos.

f) Ecrãs do menu “Dicionários”

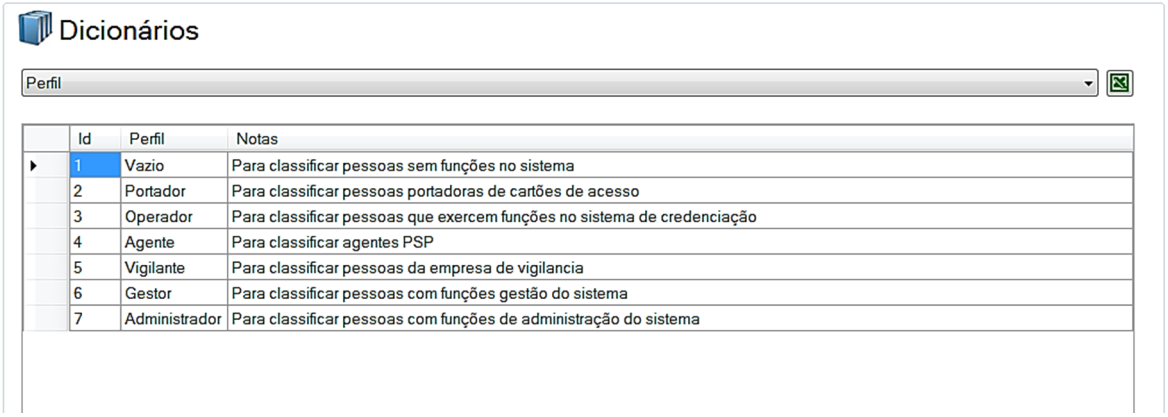


Figura 267 – CRED – Ecrã do menu “Dicionários” – Tipos de perfil.

g) Ecrãs do menu “Pesquisa”

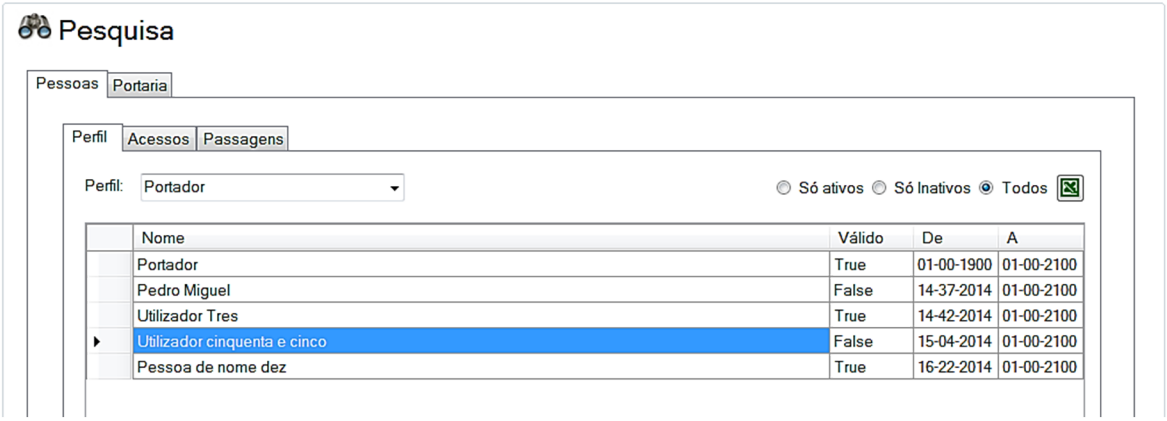


Figura 268 – CRED – Ecrã do menu “Pesquisa” – Pessoas, perfil.

Figura 269 – CRED – Ecrã do menu “Pesquisa” – Pessoas, acessos

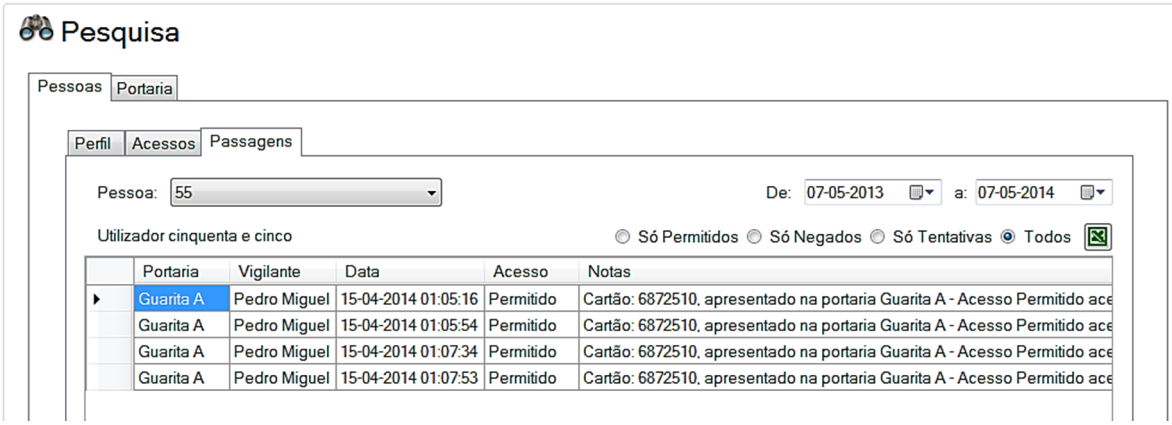


Figura 270 – CRED – Ecrã do menu "Pesquisa" – Pessoas, passagens.

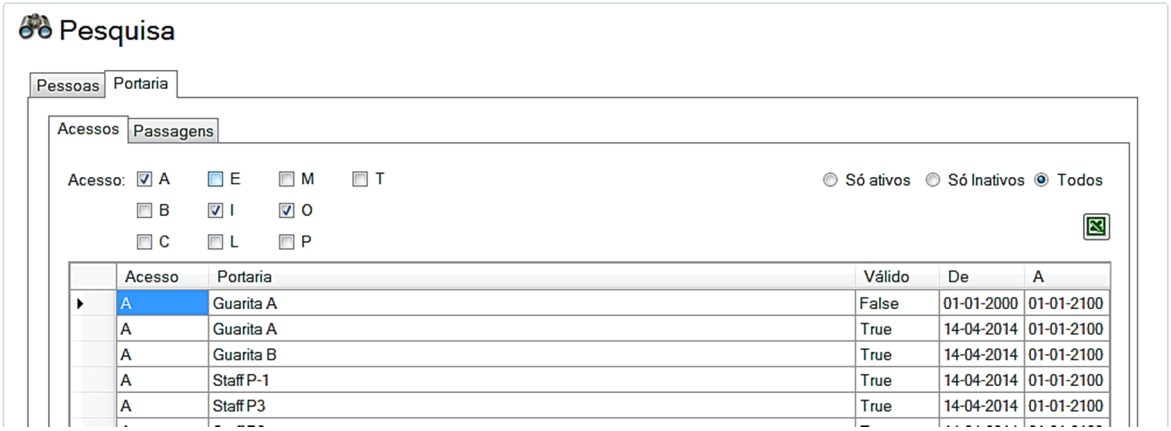


Figura 271 – CRED – Ecrã do menu "Pesquisa" – Portarias, acessos.

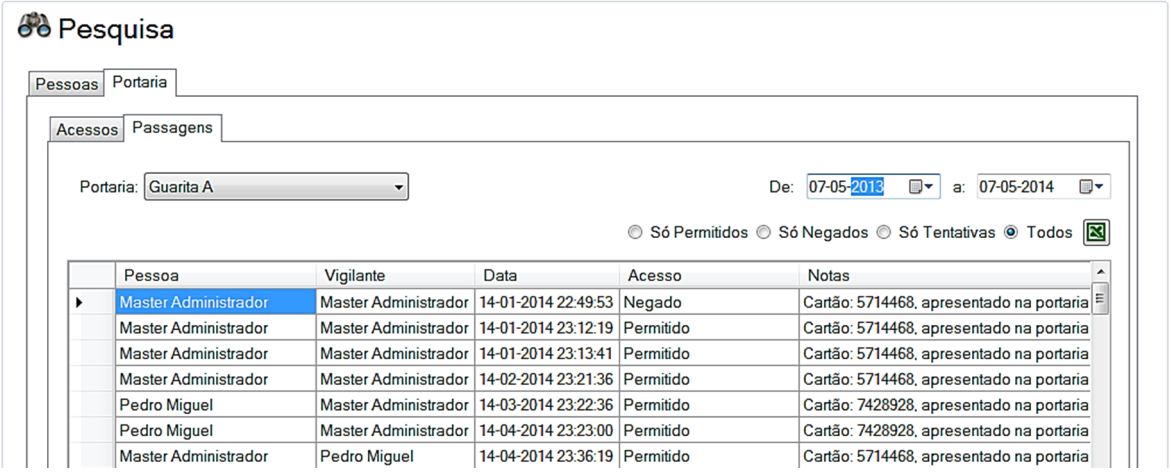


Figura 272 – CRED – Ecrã do menu "Pesquisa" – Portarias, passagens.