

SISTEMA INFORMÁTICO DE GESTÃO E REGISTO DE ASSIDUIDADE NA DIREÇÃO- GERAL DE ALIMENTAÇÃO E VETERINÁRIA

EDUARDO MANUEL JACINTO BRÁS

Julho de 2020

SISTEMA INFORMÁTICO DE GESTÃO E REGISTO DE ASSIDUIDADE NA DIREÇÃO- GERAL DE ALIMENTAÇÃO E VETERINÁRIA

Eduardo Manuel Jacinto Brás

Departamento de Engenharia Electrotécnica

Mestrado em Engenharia Electrotécnica e de Computadores

Área de Especialização em Sistemas e Planeamento Industrial

Relatório elaborado para satisfação parcial dos requisitos da Unidade Curricular de
Tese/Dissertação do Mestrado em Engenharia Electrotécnica e de Computadores

Candidato: Eduardo Manuel Jacinto Brás, Nº 1040232, 1040232@isep.ipp.pt

Orientação científica: Professora Cecília Maria Reis, cmr@isep.ipp.pt



Departamento de Engenharia Electrotécnica

Mestrado em Engenharia Electrotécnica e de Computadores

Área de Especialização em Sistemas e Planeamento Industrial

2020

Aos meus pais, Eduardo e Helena, de quem me orgulho, pelo constante apoio e incentivo na procura do conhecimento e, fundamentalmente, por terem sido sempre um exemplo para mim em todas as etapas da minha vida...

Agradecimentos

À minha Orientadora, Professora Doutora Cecília Maria Reis que imediatamente se identificou com a minha proposta de projeto e que sempre se mostrou totalmente disponível para me ajudar nesta caminhada tirando as minhas dúvidas, guiando-me e motivando-me sempre a concluir mais esta etapa do meu percurso acadêmico.

À Mestre Elsa Machado, Diretora de Serviços de Alimentação e Veterinária da Região Norte pela oportunidade que me deu de ingressar neste Organismo, pela confiança, pelos sábios conselhos, pela sua paciência e acima de tudo pelo enorme carinho com que sempre me tratou. Um enorme ser humano que sempre lutou pelo meu bem-estar.

À minha namorada Joana Cancela, pelo apoio, carinho e sobretudo paciência durante a realização deste trabalho. Agradeço também enquanto Coordenadora Regional dos Recursos Humanos da Direção de Serviços de Alimentação e Veterinária da Região Norte, pela valiosa ajuda no entendimento legal e administrativo de todas as variáveis presentes no âmbito da gestão de assiduidade, assim como pelas ideias e propostas de melhoria a incorporar no Sistema de Gestão de Assiduidade.

Resumo

O projecto apresentado nesta dissertação foi realizado na Direção-Geral de Alimentação e Veterinária (DGAV) e visava resolver a falta de um sistema digital de registo de assiduidade, nas várias Direções Regionais e Unidades de Abate, que se apresentasse simples de utilizar, fiável e rigoroso, mas que não exigisse um elevado esforço financeiro.

O modelo de registo de assiduidade existente nesses locais, até então, era quase, na sua totalidade, feito de forma manuscrita e a forma de registo nada tinha de rigoroso, tendo como única base a boa-fé e a boa vontade de quem o utilizava. Este procedimento dava origem a uma grande demora e a bastantes erros no apuramento da assiduidade de cada trabalhador.

Sendo assim, foi desenvolvido um sistema informático de registo de assiduidade na linguagem de programação “C#” cujo acesso e registo são feitos através da leitura do nome completo do trabalhador, a partir do Cartão de Cidadão. Os registos de assiduidade são armazenados numa nuvem, uma vez que não haveria a opção de ser criada uma base de dados interna e acedida por entidades externas à DGAV.

Depois de implementado novo sistema informático de assiduidade num número limitado de locais, previamente escolhidos, e ultrapassada a fase de testes, este passou a estar definitivamente em produção.

Para comprovar a mais-valia do sistema desenvolvido, foram elaborados alguns indicadores que permitiram verificar o sucesso da implementação. Para iguais períodos de tempo antes e depois da implementação do novo sistema, o tempo gasto na elaboração dos mapas mensais de assiduidade, por parte dos RH, foi reduzido em aproximadamente 89%, os erros de processamento salarial praticamente deixaram de existir e os casos de trabalhadores, com tempos de trabalho inferiores ao exigido por lei, passaram de 7 para 1.

Actualmente, o sistema informático de registo de assiduidade continua a ser utilizado pelos trabalhadores, é bastante elogiado por todos e demonstrou, até ao momento, ser bastante fiável e robusto, não tendo parado de funcionar em nenhum momento.

Palavras-Chave

Assiduidade Automática, Leitura do Cartão de Cidadão, Nuvem, OneDrive, C#, Excel, Webservice, Smartcards, Cartões Magnéticos, Biometria, RFID, DGAV, Administração Pública, Desmaterialização

Abstract

The project presented in this dissertation was carried out at DGAV and aimed to resolve the lack of a digital attendance registration system in the various Regional Departments and Slaughter Units that was simple to use, reliable and rigorous but that did not require a high financial effort.

The model of registration of attendance existing in those places until then was almost entirely done by hand and the form of registration was not rigorous, having as its sole basis the good faith and goodwill of those who used it. These facts gave rise to a long delay and many errors in determining the attendance of each worker.

Therefore, a computer system for attendance registration was developed in the C# programming language which can be accessed and registered by reading the full name of the worker's Citizen Card (CC). The attendance records are stored in a cloud since there would be no option to create an internal database and accessed by entities external to DGAV.

After the implementation of the new computer system for attendance in a limited number of locations, previously chosen, and after the testing phase has passed, it is now in production.

To prove the added value of this new system, some indicators were elaborated that allowed to verify the success of the implementation. For equal periods of time before and after the implementation of the new system, the time spent now in the preparation of the monthly maps of attendance by the HR was reduced by approximately 89%, the errors of salary processing almost ceased to exist and the cases of workers with time lower than required by law went from 7 to 1.

At present, this new attendance registration system continues to be used by workers, is highly praised by everyone and has shown up to now to be quite reliable and robust, having never stopped working at any time.

Keywords

Automatic Attendance, Citizen Card Reading, Cloud, OneDrive, C#, Excel, Webservice, Smartcards, Magnetic Cards, Biometrics, RFID, DGAV, Public Administration, Dematerialization

Résumé

Le projet présenté dans cette thèse a été réalisé à la DGAV visant à remédier l'absence d'un système d'enregistrement numérique des présences dans les différentes directions régionales et unités d'abattage simple à utiliser, fiable et rigoureux mais ne nécessitant pas d'un effort financier important.

Le modèle d'enregistrement des présences existant jusque-là dans ces lieux était presque entièrement fait à la main et le formulaire d'inscription n'était pas rigoureux, ayant pour seul fondement la bonne foi et la bonne volonté de ceux qui les utilisaient. Cette procédure entraînait un long retard et de nombreuses erreurs dans la détermination de la présence de chaque travailleur. Si cela ne suffisait pas, ce manque de rigueur provoquait aussi, souvent, des conflits entre les travailleurs et entre les travailleurs et leurs supérieurs, en raison de la méfiance à l'égard des données enregistrées et des données validées.

À ce titre, a été développé un nouveau système informatisé d'enregistrement des présences dans le langage de programmation C# avec l'accès et l'enregistrement par la lecture du nom complet du travailleur, à partir de la carte de citoyen. Les enregistrements de la présence d'un travailleur sont stockés dans un nuage, car il n'y aurait pas d'option pour créer une base de données interne et accessible par des entités externes à la DGAV.

Lorsque le nouveau système informatisé d'assiduité a été mis en place et dépassée la phase de tests, celui-ci est devenu définitivement en production.

Pour prouver la valeur ajoutée du système développé, certains indicateurs ont été élaborés qui ont permis de vérifier le succès de la mise en œuvre. Pour des périodes de temps égales avant et après la mise en œuvre du nouveau système, le temps consacré à la préparation des cartes de fréquentation mensuelles, par les RH, a été réduit d'environ 89%, les erreurs de traitement des salaires ont pratiquement cessé d'exister et les cas des travailleurs avec des temps de travail inférieurs à la loi ont passé de 7 à 1.

Actuellement, le nouveau système informatisé d'enregistrement des présences continue d'être utilisé par les travailleurs, il est très apprécié par tout le monde et il s'est révélé jusqu'à présent assez fiable et robuste, n'ayant jamais cessé de fonctionner à aucun moment.

Mots-clés

Fréquentation Automatique, Lire la Carte de Citoyen, Nuage, OneDrive, C#, Excel, Webservice, Smartcards, Cartes magnétiques, Biométrie, RFID, DGAV, Administration Publique, Dématérialisation

Índice

AGRADECIMENTOS	I
RESUMO	III
PALAVRAS-CHAVE	IV
ABSTRACT	V
KEYWORDS	VI
RESUME	VII
MOTS-CLES.....	VIII
ÍNDICE	IX
ÍNDICE DE FIGURAS.....	XI
ÍNDICE DE TABELAS.....	XV
ACRÓNIMOS	XVII
1. INTRODUÇÃO	1
1.1. CONTEXTUALIZAÇÃO	2
1.2. OBJETIVOS	5
1.3. CALENDARIZAÇÃO.....	5
1.4. ORGANIZAÇÃO DO RELATÓRIO.....	6
2. REVISÃO DA LITERATURA	9
2.1. CARTÕES DE BANDA MAGNÉTICA	10
2.2. CARTÕES RFID	15
2.3. SMARTCARDS DE CONTACTO	18
2.4. BIOMETRIA.....	22
2.5. LINGUAGEM .NET	26
3. CARACTERIZAÇÃO DO PROBLEMA	27
3.1. SISTEMA DE REGISTO DE ASSIDUIDADE AS-IS.....	28
3.2. SISTEMA DE REGISTO DE ASSIDUIDADE TO-BE	33
4. ANÁLISE DE REQUISITOS E PROPOSTA DE PROTÓTIPO	35
4.1. REQUISITOS FUNCIONAIS.....	36
4.2. MÉTODO DE IDENTIFICAÇÃO E ACESSO AO SISTEMA	39
4.3. LIGUAGEM DE PROGRAMAÇÃO E FORMA DE ARMAZENAMENTO A ADOPTAR	41
4.4. FLUXOGRAMA DO PROTÓTIPO	43

5.	DESENVOLVIMENTO E APRESENTAÇÃO DO SISTEMA.....	47
5.1.	REQUISITOS PRÉVIOS.....	48
5.2.	<i>LOGIN</i> NO SISTEMA.....	50
5.3.	REGISTO DE ASSIDUIDADE AUTOMÁTICA.....	60
5.4.	REGISTO DE OBSERVAÇÕES/JUSTIFICAÇÕES.....	71
5.5.	VISUALIZAÇÃO DE ASSIDUIDADE.....	83
6.	TESTES DO SISTEMA E ANÁLISE DE RESULTADOS.....	87
6.1.	TESTES DO SISTEMA.....	87
6.2.	ANÁLISE DE RESULTADOS.....	88
7.	CONCLUSÃO.....	93
8.	REFERÊNCIAS DOCUMENTAIS.....	97
	ANEXO A	
	ANEXO B	
	ANEXO C	
	ANEXO D	
	ANEXO E	

Índice de Figuras

Figura 1. Logotipo da Direção Geral de Alimentação e Veterinária	3
Figura 2. Cartão de banda magnética	10
Figura 3. Características físicas dos cartões de banda magnética – ISO 7810	11
Figura 4. Características da banda magnética – ISO 7811	11
Figura 5. Representação do conteúdo da pista 1 – ISO 7811	12
Figura 6. Representação do conteúdo da pista 2 – ISO 7811	12
Figura 7. Representação do conteúdo da pista 3 – ISO 7811	13
Figura 8. Leitor e gravador de cartões de banda magnética	14
Figura 9. Ciclo de um sistema anti-roubo através da tecnologia RFID	16
Figura 10. Diferentes tipos de etiquetas	16
Figura 11. Diversos tipos de leitores RFID	17
Figura 12. Princípio de acoplamento indutivo	18
Figura 13. Cartão de Cidadão Português – Cartão com microcontrolador embutido	19
Figura 14. Cartão de memória com capacidade de armazenamento de 525Kbits	20
Figura 15. Leitor/gravador de <i>smartcards</i>	20
Figura 16. Contactos da área dourada de um <i>smartcard</i>	21
Figura 17. Elementos de um <i>smartcard</i>	21
Figura 18. Reconhecimento da impressão digital	23
Figura 19. Livro de ponto usado para registo de assiduidade diário	29
Figura 20. Mod.75/DGAV preenchido com registos de assiduidade	30
Figura 21. Tabela <i>Excel</i> para aferição de tempos de trabalho mensal por trabalhador	31
Figura 22. Extracto de um Mod.62/DGAV preenchido	31
Figura 23. Fluxograma do processo AS-IS	32
Figura 24. Estrutura de pastas da <i>OneDrive</i>	48
Figura 25. Estrutura de pastas criadas pelo sistema de assiduidade	48
Figura 26. Adição da biblioteca <i>pteidlib_dotnet.dll</i>	49
Figura 27. DCU para a funcionalidade <i>Login</i>	51

Figura 28. Mensagem ao utilizador de CC não detetado	52
Figura 29. Mensagem ao utilizador de validade e tolerância expirada	55
Figura 30. Mensagem ao utilizador de validade do CC expirada mas dentro da tolerância	55
Figura 31. Ficheiro *.txt com a informação do nome da DSAVR	57
Figura 32. Ficheiro *.txt com a informação das unidades orgânicas da DSAVRN	58
Figura 33. Ficheiro *.txt com algumas observações/justificações existentes	59
Figura 34. Página principal do sistema de registo de assiduidade	60
Figura 35. Mensagem de aviso de não selecção da Unidade Orgânica	61
Figura 36. Mensagem de aviso de registo de assiduidade já existente	61
Figura 37. Mensagem de aviso de registo anterior não efectuado	62
Figura 38. Mensagem de aviso de não respeito pela pausa de almoço	62
Figura 39. DCU para a funcionalidade “Registo de Assiduidade Automática”	63
Figura 40. Ficheiro *.txt com o nome do local onde está instalado o sistema	64
Figura 41. Extrato da tabela usada na gestão de <i>macaddress</i>	65
Figura 42. Informações apresentadas ao utilizador aquando do registo de assiduidade	66
Figura 43. Informação dos dados gravados e mensagem de sucesso do registo	68
Figura 44. Extracto da assiduidade mensal de um utilizador com as informações registadas	70
Figura 45. Adição de justificação/observação Erro de Entrada – 1º Período	72
Figura 46. Mensagem de aviso de registo com hora inferior ao registo anterior	72
Figura 47. Extrato da assiduidade mensal de um trabalhador com observações do primeiro grupo	73
Figura 48. DCU para a funcionalidade “Registo de Observações/Justificações” - grupo 1	74
Figura 49. Adição de observação “Férias”	75
Figura 50. Mensagem de aviso de observação já existente	76
Figura 51. Extrato da assiduidade mensal de um trabalhador com observação do segundo grupo	76
Figura 52. Mensagem de aviso de tentativa de justificação de mais de sete horas	77

Figura 53. Extrato da assiduidade mensal de um trabalhador com observação do terceiro grupo	78
Figura 54. Janela de introdução do período de ausência	80
Figura 55. Extrato da assiduidade mensal de um trabalhador com observação do quarto grupo	80
Figura 56. DCU para a funcionalidade “Registo de Observações/Justificações” – grupos 2,3 e 4	81
Figura 57. Área reservada à visualização de registos.	83
Figura 58. Mensagem de aviso de ficheiro não encontrado	83
Figura 59. DCU para a funcionalidade “Visualização de Registos”	84

Índice de Tabelas

Tabela 1. Calendarização do projeto	6
Tabela 2. Vantagens e desvantagens dos cartões de banda magnética	15
Tabela 3. Requisitos funcionais a considerar no desenvolvimento do sistema	36

Acrónimos

AMA	– Agência para a Modernização Administrativa, I.P.
AP	– Administração Pública
API	– Application Programming Interface
ABA	– American Bankers Association
CC	– Cartão de Cidadão
CNPD	– Comissão Nacional de Proteção de Dados
CPU	– Central Processing Unit
DCU	– Diagrama de Casos de Uso
DGAV	– Direção-Geral de Alimentação e Veterinária
DSAVR	– Direção de Serviços de Alimentação e Veterinária Regional
DSAVRN	– Direção de Serviços de Alimentação e Veterinária da Região do Norte
PREMAC	– Plano de Redução e Melhoria da Administração Central
DSAVRC	– Direção de Serviços de Alimentação e Veterinária da Região do Centro
DSAVRLVT	– Direção de Serviços de Alimentação e Veterinária da Região de Lisboa e Vale do Tejo
DSAVRA	– Direção de Serviços de Alimentação e Veterinária da Região do Alentejo
DSAVRALG	– Direção de Serviços de Alimentação e Veterinária da Região do Algarve
DAV	– Divisão de Alimentação e Veterinária

DSAVR	–	Direção de Serviços de Alimentação e Veterinária Regional
EEPROM	–	Electrically Erasable Programmable Read-Only Memory
GUI	–	Graphic User Interface
IP	–	Internet Protocol
IS	–	Inspeção Sanitária
IATA	–	International Air Transport Association
LGTFP	–	Lei Geral do Trabalho em Funções Públicas
OOP	–	Object-Oriented Programming
PDF	–	Portable Document Format
NAV	–	Núcleo de Alimentação e Veterinária
NIF	–	Número de Identificação Fiscal
RAM	–	Random Access Memory
RFID	–	Radio Frequency Identification
RGPD	–	Regulamento Geral de Protecção de Dados
RH	–	Recursos Humanos
ROM	–	Read-Only Memory
SDK	–	Software Development Kit
SO	–	Sistema Operativo
SRACC	–	Sistema de Registo de Assiduidade através do Cartão de Cidadão

VPN – Virtual Private Network

1. INTRODUÇÃO

A crescente competitividade entre organizações e a consequente otimização de Recursos Humanos (RH) gera, cada vez mais, um maior rigor na gestão da sua assiduidade e pontualidade. Ainda que na Administração Pública (AP) não exista a competitividade observada no setor privado, os RH são alvo de uma preocupação constante por serem o ativo mais importante e valioso na prossecução da missão dos vários serviços que a compõem. É fundamental, desde já, fazer a distinção entre assiduidade e pontualidade. É uma confusão de conceitos muito frequente que, por vezes, gera uma má interpretação das próprias Leis do trabalho, levando a inusitadas tensões entre trabalhador e chefia. A pesquisa das duas palavras no dicionário [1] permite constatar que “assiduidade” é a qualidade do que é assíduo, é a presença regular num local em que se tem o compromisso de permanecer, cumprindo um horário previamente estabelecido, geralmente para trabalhar ou estudar, significando ainda frequência, continuação, regularidade. Relativamente a “pontualidade” verifica-se que é a qualidade de pontual,

exatidão, rigor. Sendo a assiduidade e a pontualidade os elementos mais importantes na estabilidade contratual do trabalhador, na sua avaliação de desempenho, assim como no processamento do seu salário, é importante que o processo de controlo e gestão da assiduidade seja o mais rigoroso, expedito e transparente possível.

Quanto ao controlo, existem muitas soluções no mercado, desde sistemas de acionamento mecânico ou por proximidade, até sistemas mais modernos e *bulletproof* como, por exemplo, os biométricos. Relativamente aos primeiros sistemas enunciados, são sem dúvida os mais utilizados, por serem aqueles que exigem menor investimento, mas também são aqueles cujo rigor e transparência mais deixam a desejar.

Na AP esta verdade mantém-se. Embora já existam organismos (muito poucos) que adotaram os sistemas biométricos, muitos outros acabam por escolher os sistemas de acionamento mecânico, uma vez que, os primeiros para serem instalados e implementados necessitam de respeitar uma enorme quantidade de requisitos e boas práticas [2] referenciadas pela Comissão Nacional de Proteção de Dados (CNPD). Existem, ainda que com expressão mínima, outras realidades na AP como é o caso da Direção-Geral de Alimentação e Veterinária (DGAV), que, nas suas unidades regionais desconcentradas é ainda utilizado o controlo, registo e gestão da assiduidade tudo de forma manuscrita. Este projeto nasce precisamente da necessidade de evoluir para um sistema digital que se compadeça com as necessidades de gestão de RH dos dias de hoje.

1.1. CONTEXTUALIZAÇÃO

No âmbito do Compromisso Eficiência, o XIX Governo Constitucional determinou as linhas gerais do Plano de Redução e Melhoria da Administração Central (PREMAC) com o objetivo de a tornar eficiente e racional na utilização dos recursos públicos e, por outro lado, para o cumprimento dos objetivos de redução da despesa. Visava, obviamente, a reorganização da estrutura do Estado através de extinção, criação ou fusão de serviços. E é nesse contexto que DGAV é criada de acordo com o Decreto-lei n.º 7/2012 [3]. O logotipo oficial deste organismo é o apresentado na figura 1.



Figura 1. Logotipo da Direção Geral de Alimentação e Veterinária

A DGAV é um serviço central da administração direta do Estado, dotado de autonomia administrativa e integrado no Ministério que tutela o setor agro pecuário, atualmente o Ministério da Agricultura. Tem como Missão “...a definição, execução e avaliação das políticas de segurança alimentar, de proteção animal e de sanidade animal, proteção vegetal e fitossanidade, sendo investida nas funções de autoridade sanitária veterinária e fitossanitária nacional e autoridade responsável pela gestão do sistema de segurança alimentar” e as suas competências respeitam à saúde e proteção animal; regulamentação e coordenação do controlo alimentar; e sanidade vegetal e fitossanidade [4].

Esta Direção-Geral estrutura -se nas seguintes unidades orgânicas nucleares [5]:

- a) Direção de Serviços de Gestão e Administração;
- b) Direção de Serviços de Estratégia, Comunicação e Internacionalização;
- c) Direção de Serviços de Proteção Animal;
- d) Direção de Serviços de Sanidade Vegetal;
- e) Direção de Serviços de Nutrição e Alimentação;
- f) Direção de Serviços de Segurança Alimentar;
- g) Direção de Serviços de Meios de Defesa Sanitária.

Integram ainda a estrutura nuclear as seguintes unidades orgânicas desconcentradas [5]:

- h) Direção de Serviços de Alimentação e Veterinária da Região do Norte (DSAVRN);
- i) Direção de Serviços de Alimentação e Veterinária da Região do Centro (DSAVRC);

- j) Direção de Serviços de Alimentação e Veterinária da Região de Lisboa e Vale do Tejo (DSAVRLVT);
- k) Direção de Serviços de Alimentação e Veterinária da Região do Alentejo (DSAVRA);
- l) Direção de Serviços de Alimentação e Veterinária da Região do Algarve (DSAVRALG).

Para melhor compreensão da estrutura orgânica da DGAV é possível ver no anexo A o respetivo organograma.

Às Direções de Serviços de Alimentação e Veterinária Regionais, compete, no âmbito das respetivas áreas geográficas, assegurar a execução das ações, planos e programas definidos pelos serviços centrais da DGAV.

De salientar, mais uma vez, que a Sede da DGAV em Lisboa, que alberga as direções de serviço enumeradas imediatamente acima de a) a g), tem um sistema de assiduidade automático e por isso este projeto só fazia sentido ser implementado nas áreas desconcentradas onde este sistema não existe e a dinâmica e organização do trabalho é completamente diferente da existente na Sede. Nos casos das Direções de Serviço de Alimentação e Veterinária Regionais desconcentradas, a desconcentração não se verifica apenas ao nível da estrutura, mas também geograficamente, sendo que estas estão ainda divididas em Divisões e Núcleos de Alimentação e Veterinária (DAV e NAV, respetivamente), não esquecendo os matadouros ou unidades de abate onde exercem funções de inspeção sanitária (IS) espalhados pelo País. Todos estas nuances serão relevadas e detalhadas nas seção 3 deste documento.

O autor da presente dissertação entrou ao serviço na DGAV em Janeiro de 2018 onde foi afeto à DSAVRN, na categoria de Técnico Superior, para as funções de Coordenador Regional do Parque Informático. Foi-lhe feita toda esta contextualização e dado a conhecer todas as restrições a nível de recursos humanos, financeiros e materiais que, hoje em dia, são transversais a toda a AP direta. Entre outras funções intrínsecas ao cargo, foi-lhe atribuído o desafio de criar um sistema de assiduidade moderno e automatizado, tendo em conta todas estas restrições.

1.2. OBJETIVOS

O objetivo principal deste projeto é dotar os serviços desconcentrados da DGAV de uma plataforma informática para o registo da assiduidade dos trabalhadores.

É fundamental desmaterializar o processo de assiduidade, eliminando livros de ponto e outros formulários em papel usados para o efeito, beneficiando também as equipas, com funções de RH, uma vez que terão acesso aos registos de forma totalmente digital e imediata, reduzindo o tempo de processamento e os erros associados.

Sendo assim, seguem-se os objetivos operacionais relativos a este projeto:

- Efetuar o levantamento dos requisitos e funcionalidades do sistema informático de assiduidade;
- Desenvolver o Sistema de Registo de Assiduidade com Cartão de Cidadão (SRACC);
- Elaborar o manual de utilização do SRACC ;
- Instalar o SRACC em todas as DAV, NAV e DSAVR de todo o País;
- Instalar o SRACC em todos os Matadouros elegíveis (onde haja serviços prestados pelo Corpo de Inspeção da DGAV) de todo o país;
- Realizar Ações de Formação/Sensibilização sobre a implementação e funcionamento do SRACC.

1.3. CALENDARIZAÇÃO

Tendo em conta a complexidade de desenvolvimento do sistema e da sua implementação, é pertinente elaborar e apresentar uma calendarização, com o objetivo de constituir uma referência nos trabalhos a executar e dos seus *milestones*, de forma a mitigar possíveis atrasos. Sendo assim, a calendarização é apresentada na tabela 1.

Depois de caracterizado o problema, no capítulo 4 são apresentados os requisitos e funcionalidades pedidas pelos vários intervenientes que atualmente fazem parte do processo de assiduidade. É também neste capítulo que é dada a conhecer qual a tecnologia de registo de assiduidade a seleccionar, assim como, a linguagem de programação escolhida e, ainda, a forma de armazenamento dos dados a adotar. Por fim, é apresentado o fluxograma geral da solução informática proposta pelo autor desta dissertação.

O capítulo 5, mais técnico, refere-se ao desenvolvimento do SRACC. São apresentados, em quatro pontos fundamentais, as funções da aplicação assim como o seu desenvolvimento e constrangimentos se aplicável.

Os testes ao sistema são abordados no capítulo 6. Aqui, será dado a conhecer quais os preparativos efetuados, os locais onde o sistema foi instalado e o motivo que esteve na base dessa escolha, a duração dos testes e os resultados.

A análise de resultados é detalhada no capítulo 7, onde são caracterizadas as principais funcionalidades do novo sistema quanto ao seu sucesso e impacto em todo o processo de assiduidade, no contexto da desmaterialização da mesma, da poupança de tempo atingida no registo e também no controlo e, acima de tudo, na melhoria das próprias condições de trabalho.

A conclusão deste projeto será discutida no capítulo 8, onde serão abordados todos os conhecimentos adquiridos na realização deste projeto, o sucesso ou insucesso, pontos fortes e fracos da solução encontrada.

2. REVISÃO DA LITERATURA

Neste capítulo é fundamental dar a conhecer as tecnologias de identificação e recolha de dados usadas atualmente para os mais diversos controlos. Serão analisadas as suas vantagens e desvantagens, assim como, a sua aplicabilidade. Existem muitas opções e estão presentes na nossa vida quotidiana. Por exemplo os código de barras que vemos todos os dias nas embalagens de produtos que usamos, os cartões magnéticos e *smartcards* que usamos para levantar dinheiro ou pagar compras nos supermercados, os sistemas que usam o *Radio Frequency Identification* (RFID) como a Via Verde e os sistemas de autenticação com base nos dados biométricos, tão em voga nos dias de hoje, como o reconhecimento facial ou a impressão digital em equipamentos como os computadores ou telemóveis. No caso do controlo e registo de assiduidade serão focados

apenas as tecnologias mais usadas, ou seja, a banda magnética, os *smartcards*, o RFID e os sistemas biométricos.

2.1. CARTÕES DE BANDA MAGNÉTICA

No início dos anos 60, Forrest Parry, Engenheiro da IBM, foi o primeiro homem a tentar fixar uma fita magnética num cartão de plástico de forma a criar um cartão de identificação para os trabalhadores da CIA. Curiosamente a sua maior dificuldade residia na forma de fixar a fita ao cartão de plástico. Mais curioso ainda foi o facto de ter sido a sua esposa a dar a ideia de fixá-la através do calor libertado por um ferro de engomar. Assim fez e a IBM tornou-se pioneira na fabricação de cartões de banda magnética [6]. Um exemplo de um cartão com este tipo de tecnologia pode ser visto na figura 2.



Figura 2. Cartão de banda magnética [7]

Após esta invenção e depois do estabelecimento de normas em 1970, foi a vez dos cartões de crédito adoptarem esta tecnologia. Ainda em voga atualmente, estes cartões são responsáveis por milhões de transações comerciais em todo o mundo, ou seja, hoje em dia, este tipo de cartões financeiros constituem a maior parte de utilização dos cartões de banda magnética. No entanto podemos ver esta tecnologia aplicada a cartões de controlo de acesso, cartas de condução, entre outros. Estes cartões seguem as normas ISO/IEC 7810 e ISO/IEC 7811 [8] que definem as características físicas e as várias opções para codificação interoperável da informação da banda magnética.

- Cartões

Os cartões de banda magnética, como já foi referido, estão sujeitos a normas no que diz respeito às suas características físicas e da banda magnética.

Quanto às características físicas os cartões deverão ter 53.795mm de largura, 85,725mm de comprimento e 0.762mm de espessura como mostra a figura 3.

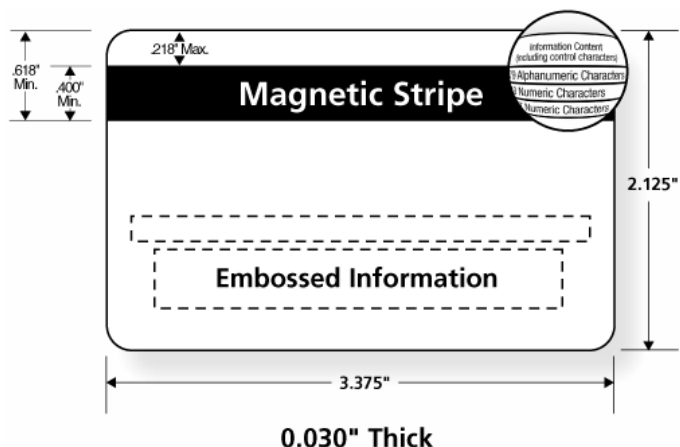


Figura 3. Características físicas dos cartões de banda magnética – ISO 7810 [8]

Relativamente à banda magnética, a sua distância à extremidade superior do cartão será de 5.664mm. É constituída por 3 pistas de dados independentes e têm cada uma 2.794mm de largura. O posicionamento de cada uma, da sua mediatriz em relação à extremidade superior do cartão é de 7.061mm, 10.363mm e 13,919 respectivamente. Esta informação pode ser vista na figura 4.

	0.223"		Recording Density (bits per inch)	Character Configuration (including parity bit)	Information Content (including control characters)
0.110"	Track 1	IATA	210 BPI	7 Bits per Character	79 Alphanumeric Characters
0.110"	Track 2	ABA	75 BPI	5 Bits per Character	40 Numeric Characters
0.110"	Track 3	THRIFT	210 BPI	5 Bits per Character	107 Numeric Characters

Figura 4. Características da banda magnética – ISO 7811 [8]

A pista 1 foi desenvolvida pela International Air Transport Association (IATA). Tem uma capacidade para 79 caracteres alfanuméricos e é apenas de leitura. Na figura 5 é possível ver representado o possível conteúdo da pista 1 de um cartão magnético.

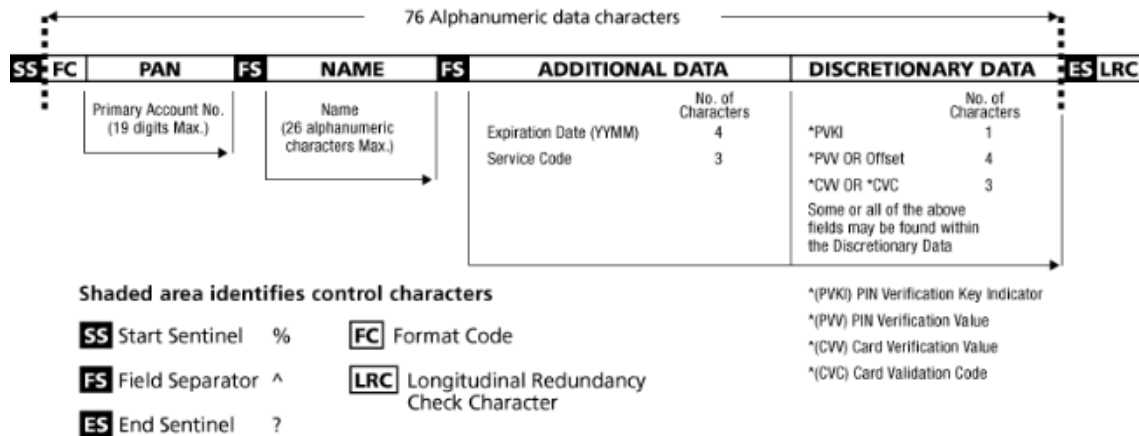


Figura 5. Representação do conteúdo da pista 1 – ISO 7811 [8]

A figura 6 representa o possível conteúdo da pista 2 de um cartão magnético. Foi desenvolvida pela American Bankers Association (ABA), tem 40 caracteres alfanuméricos e é apenas de leitura.

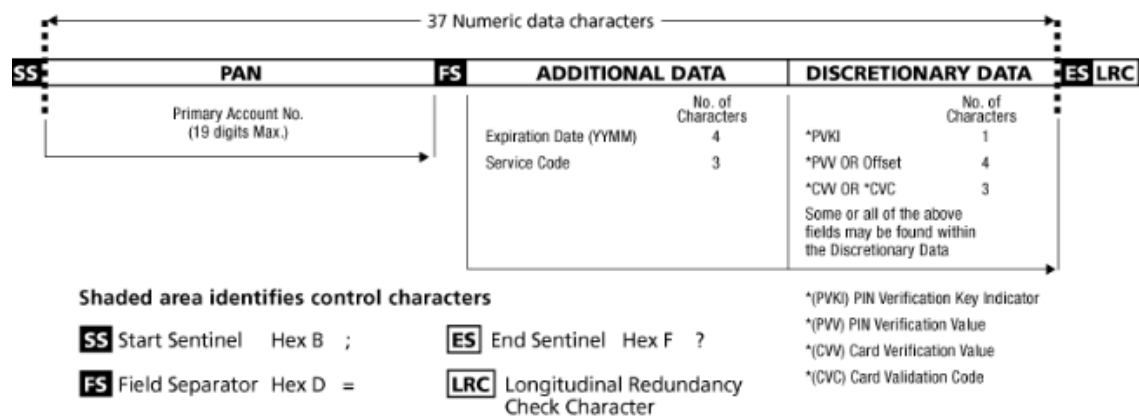


Figura 6. Representação do conteúdo da pista 2 – ISO 7811 [8]

Por último, a pista 3, menos usada que as outras pista é a única com a capacidade de leitura e escrita, onde se pode, por exemplo, gravar um PIN diferente do original.

Diferencia-se também por ter 40 caracteres mas apenas numéricos. A sua estrutura pode ser observada na figura 7.

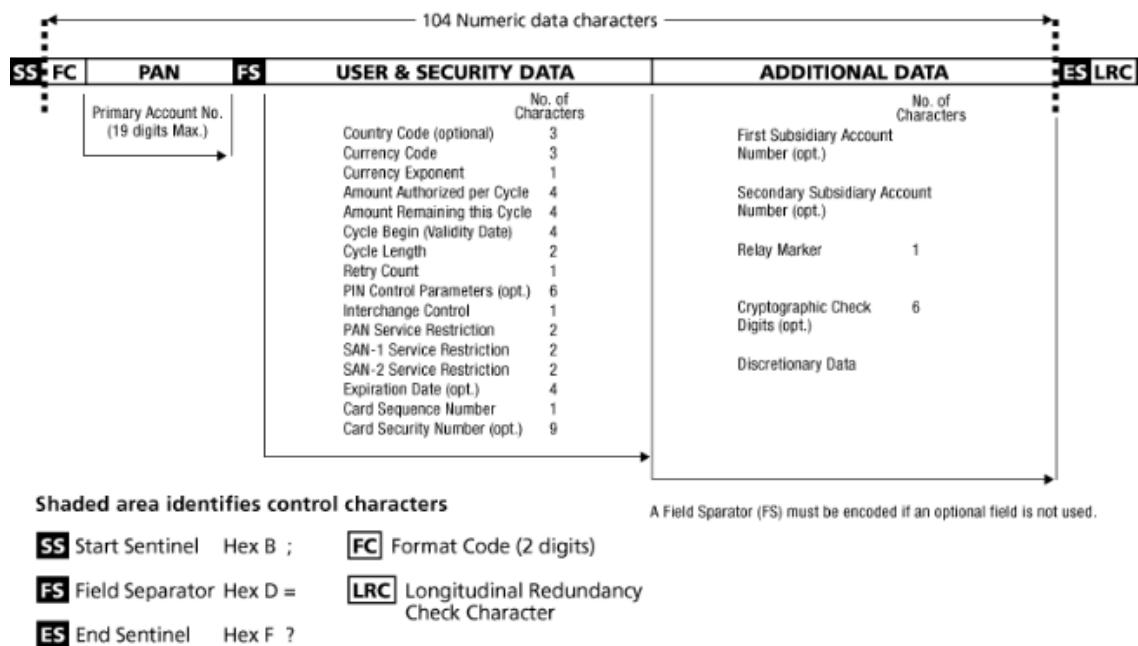


Figura 7. Representação do conteúdo da pista 3 – ISO 7811 [8]

- Leitores e gravadores

Na figura 8 é possível ver um exemplo de leitor/gravador de cartões de banda magnética. Têm a função de ler e/ou escrever dados na banda magnética. Ressalvando-se apenas o facto de que os gravadores de alta coercividade apenas podem gravar bandas magnéticas de alta coercividade e o mesmo se aplica para os gravadores de baixa coercividade de forma análoga. Já os leitores tanto lêem um tipo, como outro.



Figura 8. Leitor e gravador de cartões de banda magnética [9]

- Princípio de funcionamento

A banda magnética é composta por partículas magnéticas com cerca de 500nm e cada uma atua como um ímã. Na produção da banda magnética estas partículas são orientadas com os eixos Norte-Sul todos paralelos à própria banda através de um campo magnético externo aplicado enquanto um aglutinador (resina) as fixa no devido lugar. Ou seja, uma banda magnética que nunca tenha sido codificada funciona como um ímã com um pólo em cada extremo o que faz com que as partículas estejam todas orientadas no mesmo sentido cardinal. Ora, a codificação não é mais do que a aplicação de magnetismo através de um solenóide que invertendo a sua polaridade em conjunto com o movimento da banda magnética altera a orientação dessas partículas de forma a alcançar a codificação desejada.

É também importante perceber o conceito de coercividade nesta tecnologia apresentada. Existem bandas magnéticas de alta e baixa coercividade e isto significa que no primeiro caso é necessário aplicar um campo magnético muito mais forte do que no segundo caso para que as partículas se orientem na posição desejada [10].

De forma a sintetizar a informação, é possível ver na tabela 2 as várias vantagens e desvantagens dos cartões de banda magnética.

Tabela 2. Vantagens e desvantagens dos cartões de banda magnética

Vantagens	Desvantagens
<ul style="list-style-type: none"> • Tecnologia de baixo custo • Tecnologia usada em todo o mundo • Tecnologia madura • Recodificação 	<ul style="list-style-type: none"> • Durabilidade (desgaste e danos físicos) • Vulnerável a campos magnéticos • Leitura por contacto

2.2. CARTÕES RFID

A Identificação por RFID surgiu no período da II Guerra Mundial. Não é novidade que muitas das invenções ou avanços tecnológicos surjam em tempos de guerra. Normalmente são nestes tempos sombrios que se fazem descobertas tecnológicas e que, por consequência, muitas vezes mudam o curso da história. Neste contexto a RFID foi criada com o objectivo de identificar as diferentes aeronaves como amigas ou inimigas. As aeronaves estavam munidas de um rádio emissor, *transponder*, que transmitia constantemente um código de identificação. Se em terra, a leitura destes códigos de identificação não coincidissem com as listagens de aeronaves aliadas eram por exclusão de partes consideradas inimigas.

Esta tecnologia foi evoluindo, e entre 1970 e 1980, começa a ser usada comercialmente na identificação de gado e na automação industrial. Atualmente é aplicada em variadíssimas áreas como logística ou segurança, por exemplo. Hoje em dia é possível identificar esta tecnologia de forma massiva nos comércios de venda a retalho. Fala-se obviamente nos sistemas anti-roubo em que é colocado em cada produto (ou por amostragem) uma *tag* (etiqueta) RFID que é lida pelos leitores à saída dos estabelecimentos e que se o produto não tiver sido removido do inventário imediatamente emite um alarme, processo este exemplificado na figura 9.



Figura 9. Ciclo de um sistema anti-roubo através da tecnologia RFID [11]

Esta tecnologia é também muito utilizada para a identificação de pessoas. Existem muitas organizações que usam cartões com tecnologia RFID para identificar os seus funcionários, gerir a sua assiduidade e até controlar o seu acesso a determinados lugares da sua infraestrutura.

- *Tags*

As *tags*, etiquetas ou *transponders* têm como função guardar informações ou dados sobre o objeto ou pessoa sobre o qual se relaciona. É constituído por um circuito integrado, onde estão guardados os dados e uma antena que serve como meio de comunicação e alimentação. Podem ter vários tamanhos e formatos, como comprova a figura 10.



Figura 10. Diferentes tipos de etiquetas [12]

Podem ser classificados como ativos ou passivos. Os ativos possuem bateria interna e são normalmente de leitura e escrita. Têm um maior alcance de leitura, no entanto têm como principal desvantagem o facto de terem um tempo de vida limitado de acordo com as características da bateria que os alimentam. Já os passivos, sem bateria interna, utilizam o campo electromagnético gerado pelo leitor como fonte de energia. Têm por isso um tempo de vida virtualmente ilimitado, mas por outro lado têm um menor alcance e exige leitores bem mais potentes.

- Leitores RFID

As funções dos leitores RFID são de leitura e escrita na etiqueta. São também dotados de uma antena interna ou externa e não funciona apenas como transmissão de dados. A antena serve também para transferência de energia às etiquetas (passivas) através da criação de um campo electromagnético que induz uma corrente pela lei de Faraday na etiqueta. Existem variadíssimos modelos de leitores e um exemplo pode ser visto na figura 11.



Figura 11. Diversos tipos de leitores RFID [13]

- Princípio de funcionamento

Relativamente às etiquetas de 1 bit, as mais simples e vastamente utilizadas em sistemas anti-roubo de artigos, a comunicação pode ser feita através de várias técnicas diferentes como a rádio frequência, microondas, divisor de frequência, electromagnética e acústico-

magnética. O princípio de funcionamento destas etiquetas consiste na activação ou desactivação do bit. Ou seja, se o bit estiver ativo o leitor deteta a etiqueta, se não, esta não será detetada.

As etiquetas com maior capacidade de memória de dados já funcionam de maneira diferente uma vez que é necessária haver troca de dados. Para isso, a comunicação pode ser feita no modo *half duplex* ou *full duplex*. No caso do modo *half duplex*, em primeiro lugar, são transmitidos os dados do leitor para a etiqueta e só depois deste terminar é que se dá o inverso do processo. Já no caso do modo *full duplex*, os dados são trocados ao mesmo tempo entre leitor e etiqueta mas para isso a frequência de um tem de ser obrigatoriamente diferente da do outro. A tecnologia mais usada para a transmissão de dados entre leitor e etiqueta é o acoplamento indutivo, conforme esquematizado na figura 12 [14], [15], [16].

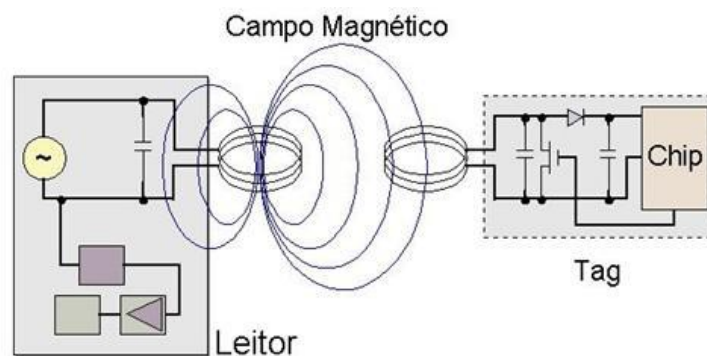


Figura 12. Princípio de acoplamento indutivo [17]

2.3. SMARTCARDS DE CONTACTO

A ideia e primeiras patentes de *smartcard* apareceram no fim da década de 60 e início da década de 70 pelas mãos de um cientista Japonês, Dr. Kunitaka Arimura e dois Engenheiros Alemães Jurgen Dethloff e Helmut Grotrupp que tiveram a ideia de embutir circuitos integrados em cartões de identificação.

Já em 1974, Roland Moreno, inventor Francês, criou e patenteou o primeiro cartão com circuito integrado embutido e ainda o dispositivo para leitura deste cartão. Foi nesta altura que este cartão foi finalmente apelidado de *smartcard*.

Em 1979, foi desenvolvido pela Motorola, o primeiro microcontrolador de um único chip para uso no sistema bancário Francês e a partir daí os *smartcards* têm-se massificado em todo mundo precisamente na área da banca, sendo normal hoje em dia, o mesmo cartão ser dotado com a tecnologia *smartcard* e banda magnética [18].

- Cartões

Os *smartcards* são cartões de plástico, cujas características físicas, à semelhança dos cartões magnéticos são definidas pela norma ISO-7810, que possuem um circuito integrado incorporado. Este circuito integrado tem a capacidade de processar dados, realizar cálculos e armazenar dados denominando-se microcontrolador. O Cartão de Cidadão Português enquadra-se neste tipo de *smartcard* e pode ser observado um exemplo na figura 13.



Figura 13. Cartão de Cidadão Português – Cartão com microcontrolador embutido [19]

Muitas vezes estão associados à nomenclatura *smartcard* os cartões com um circuito integrado apenas de memória, ou seja, que apenas é capaz de guardar dados, não estando apto a realizar operações complexas de processamento. Neste caso concreto é errado categorizá-los como *smartcards* devendo sim integrar-se na categoria de *memory cards*. Têm um custo mais reduzido mas possuem uma menor segurança. Comparativamente aos cartões de banda magnética têm uma maior capacidade de armazenamento e são relativamente mais seguros. A figura 14 apresenta um exemplo de cartão de memória.



Figura 14. Cartão de memória com capacidade de armazenamento de 525Kbits [20]

- Leitores e gravadores

Existem variadíssimas opções no que toca a leitores/gravadores de *smartcards* e podem ser encontrados em quase todas as lojas de venda de aparelhos electrónicos, sendo o seu preço de aquisição bastante baixo. Pode ser visto um exemplo na figura 15. Estes leitores/gravadores são constituídos por uma carcaça de plástico com uma ranhura que permite a entrada de cartões que serão lidos através do contacto do *microchip* com um conjunto de pinos salientes existentes no interior da carcaça de plástico [21].



Figura 15. Leitor/gravador de *smartcards* [22]

- Princípio de funcionamento

Como já foi referido um *smartcard* tem embutido um microprocessador de 16-bit. Aloja-se debaixo de um contacto de ouro e pode ter até 8 kilobytes de RAM, 346 kilobytes de ROM e 256 kilobytes de ROM programável. Usa uma interface *serial* e a energia necessária para o seu funcionamento é normalmente oriunda dos leitores/gravadores [21]. É possível ver os diferentes contactos da área dourada que comunicam com o microprocessador na figura 16.

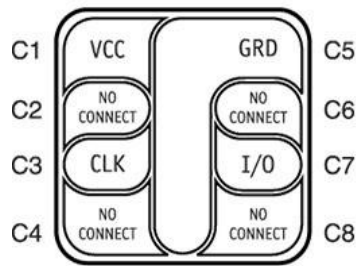


Figura 16. Contactos da área dourada de um *smartcard* [23]

Este, de forma análoga a um computador, possui um CPU (*Central Processing Unit*), um sistema de memórias e barramentos de entrada e saída, um sistema operativo gravado no cartão que permite que comunicações de alto nível possam ser estabelecidas com o leitor a que está conectado. Esta estrutura está representada na figura 17.

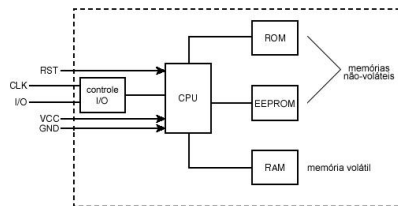


Figura 17. Elementos de um *smartcard* [23]

Um *smartcard* possui três tipos de memórias: ROM (*Read-Only Memory*), EEPROM (*Electrically Erasable Programmable Read-Only Memory*) e RAM (*Random Access Memory*). A ROM é onde o sistema operativo está armazenado e não pode ser alterada. É na EEPROM que são armazenados os dados da aplicação, isto é, pode ser lida e escrita por aplicações.

O canal de entrada e saída de um *smartcard* utiliza uma interface *serial* e unidirecional. Isto é, os bits passam um a um num único sentido de fluxo por vez. A comunicação entre o cartão e o *software* de aplicação é do tipo *master (software)* e *slave (cartão)*. O *software* envia comandos ao cartão e é esperada uma resposta. Nunca de forma inversa, exceto em resposta a um comando.

Os sistemas operativos dos *smartcards* suportam dois tipos de transferência: por caractere ou por bloco. Por caractere ocorre quando os dados são transferidos caractere a caractere, até formar uma palavra. No que diz respeito à transferência por bloco, são transmitidos quadros inteiros de uma vez, o que faz deste tipo de transferência mais complexo que o outro [23].

2.4. BIOMETRIA

A autenticação biométrica baseia-se no reconhecimento das características, fisiológicas e comportamentais, específicas de uma dada pessoa [24]. Desde há muito que tem sido objeto de interesse em questões de segurança, no entanto, até há pouco tempo atrás, a sua adopção restringia-se, apenas, quer por razões de natureza económica, quer tecnológica, a ambientes que exigiam níveis de segurança muito elevados (de alta segurança) e a casos de identificação criminal. Com o aperfeiçoamento contínuo desta tecnologia e, conseqüente, redução do custo dos seus dispositivos, que se tem verificado recentemente, cada vez mais, a biometria tem vindo a generalizar-se, sendo, atualmente, apontada como uma solução promissora para os processos de autenticação, nomeadamente, em sistemas de controlo de acessos físico [25]. Nas subsecções seguintes, apresenta-se uma visão geral desta tecnologia de identificação e autenticação, em que, inicialmente, se começa por abordar os principais tipos de tecnologias biométricas e sua classificação, para depois se caracterizar a que mais é usada atualmente, a impressão digital.

- Classificação e tipo de biometrias

As diversas tecnologias biométricas existentes podem ser classificadas em dois grupos, tendo por base as suas características: [26]

- Fisiológicas – grupo de tecnologias biométricas baseado em características fisiológicas ou estáticas. Traços fisiológicos que têm origem no potencial genético de cada pessoa e que variam pouco, ou nada, ao longo do tempo.

- Comportamentais – grupo de tecnologias biométricas baseado em características comportamentais ou dinâmicas. Características apreendidas ou desenvolvidas com a prática ou utilização contínuas e que podem variar, acentuadamente, com o passar do tempo. Também podem ser, facilmente, alteradas pela vontade ou estado físico e/ou emocional do utilizador.

Dentro das biometrias fisiológicas as mais utilizadas são a impressão digital, a geometria da mão, a íris e a face. Já no que diz respeito às biometrias comportamentais as mais em uso são a voz e a assinatura.

- Impressão digital

O reconhecimento da impressão digital é a biometria mais antiga e conhecida. As impressões digitais formam-se, ainda, no útero materno e, salvo lesões, continuam única e consistentemente ao longo da vida. As impressões digitais, desde há 150 anos atrás, têm constituído uma parte importante nas investigações criminais e na identificação de pessoas.

O reconhecimento da impressão digital consiste na aquisição da formação dos sulcos da pele dos dedos. Estes sulcos possuem determinadas terminações e divisões que diferem de pessoa para pessoa, figura 18.



Figura 18. Reconhecimento da impressão digital [27]

Neste tipo de identificação, a aquisição de imagens da impressão digital faz-se, basicamente, por quatro tipos de tecnologias:

- Óptica (com base na reflexão de um feixe de luz);
- Capacitiva (baseada na criação de diferentes acumulações de carga);
- Térmica (baseada na alteração da temperatura provocada na superfície do sensor);
- Ultra-sónica (com base na reflexão de radiações ultra-som).

A utilização da impressão digital é uma das formas ou tecnologias de identificação e autenticação mais utilizadas.

- Vantagens

- Cada pessoa dispõe de impressões digitais únicas. A impressão digital é uma característica distintiva inata, não existindo, praticamente, possibilidade de falsa correspondência [28]. Possui, ainda, elevada fiabilidade, pois nunca foram encontradas duas pessoas com a mesma impressão digital;
- Existe uma longa tradição legal na utilização da impressão digital como identificador imutável. As impressões digitais não se modificam de modo natural e esta tecnologia é aceite pela maioria das autoridades institucionais;
- É um método mais estável do que outras formas de identificação e autenticação, (por ex., o reconhecimento de íris, da voz, da face) e uma tecnologia já bastante comprovada;
- Os dispositivos utilizados para o reconhecimento são pouco dispendiosos (baixo custo), quando comparados com outros, utilizados noutras tecnologias e sistemas biométricos, tendo vindo o custo dos sensores de impressão digital a diminuir, significativamente, nos últimos anos. Os dispositivos de impressão digital em comparação com os de outras tecnologias biométricas, são bastante pequenos, requerem menos energia e menos vulneráveis às alterações ambientais (por ex., à luz e temperatura), proporcionando que esta tecnologia, cujos investimentos em

I&D têm sido elevados, seja flexível e adequada para operar em ambientes difíceis e massivos;

- Tecnologia que é bastante precisa e muito comprovada. A história, já longa, desta tecnologia tem apurado a precisão técnica e proporcionada confiança pública;
- Existem um elevado número de bases de dados de impressões digitais;
- Conveniente para os utilizadores, sendo os sistemas de impressão digital fáceis de manusear.

- Desvantagens

- Vulnerabilidade a distorções provocadas por sujidade e poeiras, com aumento de falhas (rejeição) pela utilização de dedos bastante sujos ou danificados. Pode ser difícil efectuar o reconhecimento de dedos poeirentos, secos, molhados ou oleosos, para além doutras dificuldades dependentes do ângulo e pressão de colocação do dedo [28];
- As pessoas, algumas pertencentes a determinadas culturas, podem ser relutantes a colocarem os seus dedos sobre sítios, onde muitas outras o fizeram. Nalgumas culturas as impressões digitais não são bem aceites, uma vez que são conotadas a criminosos e questões penais (estigma criminal), pessoas iletradas ou, então, por questões higiénicas;
- Algumas pessoas não têm impressões digitais ou têm-nas danificadas. As linhas do dedo são frequentemente influenciadas pelas atividades manuais (por ex., na construção civil, cujas pessoas utilizam intensivamente as mãos), que podem dar origem a sujidade, escoriações ou lesões das linhas do dedo. A qualidade das impressões digitais varia muito no interior duma população, sendo sabido que as linhas do dedo das pessoas mais idosas se tornam mais confusas e, conseqüentemente, mais difíceis de reconhecer pelo sensor. Estima-se que cerca de 2% das pessoas têm deficientes “linhas” no dedo;
- Os sensores mais baratos podem ser comprovadamente objeto de fraude.

2.5. LINGUAGEM .NET

A linguagem de programação .NET, pode-se dizer que esta é, agora, totalmente orientada a objetos, mais conhecida por *Object-Oriented Programming* (OOP). Este tipo de programação surgiu como uma tentativa para criar programas que se aproximem de como as pessoas pensam e lidam com o mundo. Nos estilos de programação mais antigos, quando é apresentado um problema ao programador, este deve identificar uma tarefa de computação que deve ser executada de maneira a resolver o problema. Ou seja, programar consiste em encontrar uma sequência de instruções que execute determinada tarefa. Mas os princípios da programação orientada a objetos, em vez de tarefas, encontram-se objetos – entidades que possuem comportamentos, que guardam informação e que podem interagir umas com as outras. Falta, portanto, definir OOP. O que é a programação orientada a objetos? É um conjunto de ferramentas e métodos que possibilita aos programadores criar aplicações sólidas, amigas do utilizador, sustentáveis, bem documentadas, que preencham os requisitos dos utilizadores. Algumas das grandes diferenças entre o Visual Basic 6.0 e o Visual Basic.NET é que a programação em Visual Basic 6 era orientada a eventos e tinha várias limitações no código, como o número de variáveis no mesmo código, número de arquivos abertos no mesmo código, entre outras, as quais não existem (virtualmente) nas novas versões .NET. Há muitas outras diferenças técnicas que existem e que foram melhoradas, no entanto, o desenvolvimento deste tema seria extenso e não é objectivo do presente trabalho. O principal aspeto a reter é que esta nova plataforma foi concebida para ser ainda mais produtiva no trabalho diário de desenvolvimento, especialmente se houver necessidade de utilizar informações em bases de dados ou criar soluções para a Internet [29].

3. CARACTERIZAÇÃO DO PROBLEMA

A Direção Geral de Alimentação e Veterinária (DGAV), por força das suas múltiplas atribuições, é um serviço com muitas unidades orgânicas desconcentradas pelo território Nacional: cinco Direções de Serviços que se desdobram em dezanove Divisões e vinte e quatro Núcleos. Sendo autoridade sanitária veterinária nacional, o seu Corpo de Inspeção Sanitária desempenha as suas funções nas mais de cem unidades de abate, quase sempre a tempo inteiro. Ainda que algumas destas unidades de abate sejam pertencentes aos Municípios, a grande maioria é do domínio privado, situação esta que por si só já é problemática uma vez que coloca funcionários públicos a trabalhar diariamente em empresas privadas. O facto dos funcionários públicos se regerem pela Lei Geral do Trabalho em Funções Públicas (LGTFP) e os restantes pelo Código do Trabalho, coloca, por si só,

alguns obstáculos ao harmonioso funcionamento laboral dessas unidades de abate como por exemplo os tempos de trabalho.

A súmula destas características é adensada pelo problema capital que é a forma como o Corpo de Inspeção Sanitária pode realizar o registo idóneo e rigoroso da sua assiduidade dentro de uma organização que não é a sua entidade empregadora. Mesmo que se pensasse em popular os registos numa base de dados da DGAV era, no mínimo, irresponsável dar acesso a uma rede interna do Estado Português a cerca de cem organizações privadas sem uma *Virtual Private Network* (VPN), que quer essas quer a DGAV não têm capacidade financeira de suportar.

Ou seja, por um lado é impensável e incomportável financeiramente colocar-se um sistema de registo de assiduidade local em todas as unidades orgânicas desconcentradas da DGAV, sendo que algumas delas são, não raras vezes, ocupadas apenas por duas ou três pessoas e por outro lado a impossibilidade da DGAV instalá-lo nas unidades de abate privadas, por segurança e porque estas empresas privadas não são, por lei, obrigadas a acomodar tal sistema.

No ponto 3.1 será feita uma caracterização modelo existente de registo de assiduidade, que foi implementado há muito tempo tendo em conta as problemáticas acima descritas.

3.1. SISTEMA DE REGISTO DE ASSIDUIDADE AS-IS

À data do início do projeto, o sistema de registo de assiduidade era feito inteiramente de forma manual. Nas unidades orgânicas desconcentradas da DGAV eram usados os livros de ponto conforme é exemplo a figura 19.

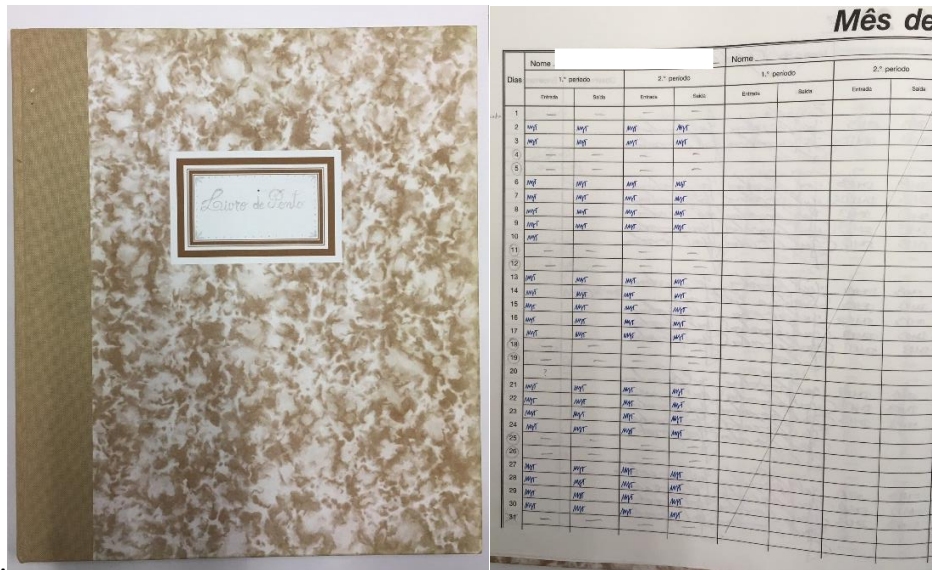


Figura 19. Livro de ponto usado para registo de assiduidade diário

O exemplo da figura 19 demonstra claramente a falta de rigor no registo das horas de trabalho. Não são colocadas horas nas entradas e saídas, sendo exigido apenas a assinatura de presença ao serviço em determinado dia. Muitas das vezes, estes registos não eram feitos dia-a-dia, mas sim no final da semana tudo de uma vez, ora assinando apenas, ora colocando as horas o que demonstra a completa discricionariedade quanto à forma e quanto ao conteúdo retirando a este método toda e qualquer confiança na veracidade dos registos.

Nas unidades de abate era usado outro modelo para registo das horas trabalhadas. Trata-se do Mod.75/DGAV. É possível ver um exemplo do modelo original preenchido na figura 20.

Unidade Orgânica **DAV - PORTO - DSAVRN**

 Mês/Ano **DEZEMBRO/2017**

Trabalhador _____

DIAS	MANHÃ		TARDE		DESCANSOS	Observações / Local de Trabalho
	ENTRADA	SÁTIMA (até às 12:00)	ENTRADA	SÁTIMA (até às 12:00)		
1						FÉRIADO
2						
3						
4						FÉRIAS
5						"
6						"
7						"
8						FÉRIADO
9						
10						
11						
12			13.00	18.15		Maltempo Escuteiro D42
13			13.00	19.00		Maltempo Escuteiro D42
14			12.45	19.30		Maltempo Escuteiro D 52
15			12.50	19.30		Maltempo AVIG D44
16			13.00	17.30		Maltempo Escuteiro D42
17						
18	06.45			13.30		Maltempo Escuteiro D42
19	06.45	13.00				Maltempo Escuteiro D42
20						Abitudo médico
21						Abitudo médico
22						Abitudo médico
23						Abitudo médico
24						
25						Ferado
26	06.45	13.00				Maltempo Escuteiro D42
27			12.50	19.00		Maltempo Escuteiro D42
28	08.30	12.30	13.45	17.15		DAV - PORTO
29	06.45			14.30	14.00-14.30	Maltempo Escuteiro D44
30						
31						

Superior Hierárquico

Data: **31/12/2017** Ass: _____

ASSINATURA DO FUNCIONÁRIO _____

Figura 20. Mod.75/DGAV preenchido com registos de assiduidade

Neste caso, a introdução dos registos era também manuscrita. Existia uma recomendação interna que solicita o preenchimento deste modelo através da ferramenta *Microsoft Excel* mas não era obrigatório pelo que na Divisão de RH aparecem os modelos preenchidos das duas maneiras. Muitos deles, preenchidos de forma manuscrita, são ilegíveis pelo que são necessários uma série de contactos com o trabalhador para que possam ser dissipadas as dúvidas quanto ao conteúdo dos modelos enviados a fim de não se enviarem registos errados que impactam diretamente na assiduidade dos trabalhadores.

Após todas as dúvidas estarem esclarecidas, era necessário fazer a verificação do número de horas trabalhadas. Uma vez que o modelo em questão não permitia um cálculo directo, era necessário que a equipa dos RH transpusesse os dados constantes no modelo de cada trabalhador para uma tabela feita em *Excel* para que esta, configurada com algumas fórmulas, retornasse o número de horas trabalhadas. Só assim era possível validar potenciais horas extraordinárias ou apurar horas em défice aos trabalhadores. Na figura 21, é possível ver um exemplo dessa tabela de aferição.

Após o preenchimento do Mod.62/DGAV, cada unidade orgânica envia o seu para os serviços centrais – Divisão de RH, Formação e Expediente, responsável pelo processamento de vencimentos.

Não é difícil perceber a tarefa hercúlea necessária para fazer o lançamento da assiduidade mensal neste serviço. A título de exemplo, no caso da DSAVRN, para uma população de cerca de cem trabalhadores, havia apenas uma pessoa dos RH responsável pelo lançamento da assiduidade mensal. Muitas vezes, as incidências da assiduidade do mês “n” só eram lançadas no mapa mensal (Mod.62/DGSV) no mês “n+2” e efetivadas no recibo de vencimento no mês “n+4”.

Além deste método, como já vimos, ser muito moroso, com demasiadas etapas e sem qualquer tipo de padronização dava origem a muitos erros de processamento, principalmente devido à transcrição de dados. E mesmo, quando processados corretamente, a sua veracidade era por vezes questionada pois, como já foi referido anteriormente, as incidências das assiduidades mensais, por regra, só surtiam efeito no recibo de vencimento passados alguns meses gerando muitas dúvidas. Segue-se, na figura 23, para um melhor entendimento o fluxograma do processo AS-IS.

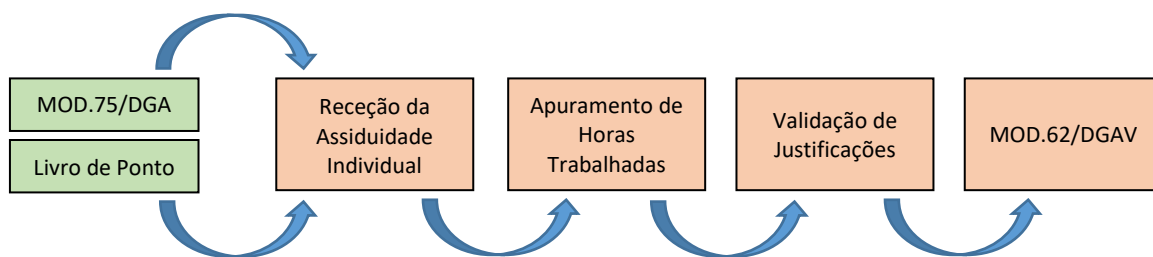


Figura 23. Fluxograma do processo AS-IS

Era esta a forma de registo e controlo da assiduidade, processo completamente retrógrado, desprovido de qualquer método fiável e transparente de marcação de horas de trabalho e que não se espelha de maneira nenhuma com as soluções tecnológicas disponíveis nos dias de hoje.

3.2. SISTEMA DE REGISTO DE ASSIDUIDADE TO-BE

Desde o início de funções do autor do presente trabalho na DGAV, mais propriamente à DSAVRN, foi possível constatar a existência de algum desconforto por parte da Diretora de Serviços em funções no método como eram feitas as marcações de assiduidade diárias pelos trabalhadores e o pouco à vontade que sentia em validar as assiduidades mensais com base apenas e só na boa-fé de quem muitas vezes colocava marcações de assiduidade diárias completamente *ha doc* ou até não raras vezes colocadas todas em catadupa apenas no fecho do mês sem menção de horas de entradas e saídas.

Após algumas trocas de ideias sobre o assunto, foi questionada a possibilidade de se implementar um sistema informático que permitisse desmaterializar o procedimento de registo de assiduidade, em primeiro lugar, mas que, ao mesmo tempo, garantisse o mais possível a veracidade da assiduidade feita pelos trabalhadores.

Um dos pontos essenciais e condição *sine qua non* seria o seu desenvolvimento ter de ser idealizado, desenvolvido, testado, atualizado e mantido internamente, apenas e só com os RH existentes na casa, claro está, por força da contenção orçamental que ainda se faz sentir na AP. É neste momento, que, o candidato se coloca à total disposição para abraçar o projeto e aplicar todo o seu conhecimento académico adquirido na prossecução do objetivo de criar um sistema informático de registo e controlo de assiduidade para os serviços desconcentrados da DGAV.

Assim, e de acordo com aquilo que era a vontade que fosse o modelo de registo de assiduidade no futuro seguirem as linhas orientadoras ou requisitos não funcionais que me foram transmitidas:

- a) Registo e controlo de assiduidade totalmente desmaterializado;
- b) Padronizado e repetível;
- c) Fiável e transparente;
- d) Desenvolvido, e sempre atualizado internamente;
- e) Manutenção interna;
- f) Mesma plataforma informática para trabalhadores afetos às unidades orgânicas e unidades de abate;

- g) Um posto de registo por unidade orgânica;
- h) Implementação de forma gradual e estruturada nas diferentes DSAVR começando pela implementação e testes na DSAVRN;
- i) Minimização dos custos de desenvolvimento e implementação;
- j) Respeitar o plasmado no Regulamento Geral de Protecção de Dados (RGPD) [30].

Aqui são focadas as ideias base para um novo modelo de registo e controlo de assiduidade que serão transformadas em requisitos e funcionalidade concretas e mais detalhadas no próximo capítulo com a proposta do protótipo.

4. ANÁLISE DE REQUISITOS E PROPOSTA DE PROTÓTIPO

A perspetiva abrangente do que deveria ser este sistema de assiduidade que foi apresentada no capítulo anterior serve como farol mas é necessário apresentar ideias concretas de forma a que o programador consiga transformar um sequência de linhas de código numa ferramenta funcional e que agrade a todos os utilizadores. Desta maneira, é absolutamente essencial obter todos os requisitos, funcionalidades, limitações tecnológicas, administrativas ou legais a serem consideradas para o desenvolvimento de um sistema informático no âmbito do registo e controlo de assiduidade. E para isso, foram realizadas várias reuniões com os vários *Stakeholders* de forma a obter de um grupo multidisciplinar todos os valiosos *inputs* para o desenho e futuro desenvolvimento do referido sistema.

Com esta informação e com base na Revisão da Literatura foi possível decidir, em primeiro lugar qual a tecnologia de registo de assiduidade a adotar. Biometria, cartão magnético, *smartcard* entre outros abordados. Em segundo lugar a escolha da linguagem a ser utilizada na programação do sistema e, por último, qual o método e local de gravação dos dados produzidos pelo sistema.

Todos estes pontos serão de seguida desenvolvidos, reservando-se para o fim deste capítulo um fluxograma que resume de forma estruturada o funcionamento geral da do sistema informático de registo e controlo de assiduidade.

4.1. REQUISITOS FUNCIONAIS

Tabela 3. Requisitos funcionais a considerar no desenvolvimento do SRACC

ID	Requisito	Descrição
000	Hardware e compatibilidade SO	O sistema deverá funcionar em máquinas com SO <i>Windows 7</i> ou superior. Arquitectura x86. Deverá comportar-se com fluidez em máquinas com CPU Core2Duo ou equivalente e 2GB de memória RAM DDR2. O sistema não deverá ocupar mais de 100MB de armazenamento.
001	Aquisição de nome	O sistema deverá poder adquirir, através de um sistema de armazenamento de dados, e devolver o nome completo do trabalhador, automaticamente, na credenciação do próprio perante o sistema por biometria, cartão magnético, <i>username/password</i> , etc.
002	Aquisição de imagens	O sistema deverá poder adquirir, através de um sistema de armazenamento de dados, e devolver uma fotografia do trabalhador, automaticamente, na credenciação do próprio perante o sistema por biometria, cartão magnético, <i>username/password</i> , etc.
003	Aquisição de local	O sistema deverá, através de um ficheiro de

		configuração (*.txt, *.config, outro) ler o seu conteúdo e dar a conhecer ao utilizador qual a Direção de Serviços Regional e UO/Unidade de Abate em que está a registar a assiduidade.
004	Aquisição de data via <i>webservice</i>	O sistema deverá, no seu modo de registo automático privilegiar a obtenção da data da ação de registo através da invocação de um <i>webservice</i> de um servidor fiável mitigando a possibilidade de viciar este dado com a manipulação da data local.
005	Aquisição de hora via <i>webservice</i>	O sistema deverá, no seu modo de registo automático privilegiar a obtenção da hora da ação de registo através da invocação de um <i>webservice</i> de um servidor fiável mitigando a possibilidade de viciar este dado com a manipulação da hora local.
006	Aquisição de data local	O sistema deverá, se não obtiver uma resposta do <i>webservice</i> (<i>Timeout</i>), considerar a data do registo de acordo com a data local, gravando esta falha de rede em ficheiro log ou outro método semelhante.
007	Aquisição de hora local	O sistema deverá, se não obtiver uma resposta do <i>webservice</i> (<i>Timeout</i>), considerar a hora do registo de acordo com a hora local, gravando esta falha de rede em ficheiro log ou outro método semelhante.
008	Introdução de data manual	Perante inevitabilidades como o esquecimento de registo por parte do utilizador ou problemas alheios ao mesmo (falha elétrica, falha de hardware), o sistema deverá ter uma componente que permita ao utilizador, <i>a posteriori</i> , fazer os registos colocando ele as datas. Este método deverá sempre subscrever o automático (correção de erros).
009	Introdução de hora manual	Perante inevitabilidades como o esquecimento de registo por parte do utilizador ou problemas alheios ao mesmo (falha elétrica, falha de hardware), o sistema deverá ter uma componente que permita ao utilizador, <i>a posteriori</i> , fazer os registos colocando ele as horas.

		Este método deverá sempre subscrever o automático (correção de erros).
010	Aquisição de IP (<i>Internet Protocol</i>)	O sistema deverá adquirir o IP do terminal de registo e guardar a informação aquando da sua utilização de forma a garantir a rastreabilidade e evitar potenciais instalações da aplicação em locais não autorizados. (Método apenas possível para terminais instalados nas UO, uma vez que têm IP fixo)
011	Aquisição do <i>macaddress</i>	O sistema deverá adquirir o <i>macaddress</i> do terminal de registo e guardar a informação aquando da sua utilização de forma a garantir a rastreabilidade e evitar potenciais instalações da aplicação em locais não autorizados. (Complemento à aquisição do IP uma vez que nas unidades de abate os IP são dinâmicos e apenas o <i>macaddress</i> atesta com certeza absoluta que o terminal onde é feito o registo é o autorizado).
012	Aquisição da data de validade	Seja qual for o método de credenciação à área privada do utilizador o sistema deverá adquirir o limite de validade dessa credenciação e avisar o utilizador antecipadamente “X” tempo antes deste expirar de forma a poder renovar as credenciais.
013	Adição de observações excecionais	O sistema deverá apresentar ao utilizador um número limitado e pré-determinado de observações que pode adicionar ao seu registo de assiduidade de forma a complementá-la ou a justificar determinada ausência de registo adequando o respetivo diferencial horário. (Férias, atestados, consultas, parentalidade, entre outros).
014	Validação de observações excecionais	O sistema deverá validar se para o mesmo dia já existe uma observação que justique o dia todo de trabalho não permitindo adicionar outro. Deverá, ainda, permitir mais do que uma observação se o tempo total de cada uma não for igual a 1 dia de trabalho (7H).
015	Validação de horários	O sistema deverá validar se o valor do registo $Y > X$, $Z > Y$

		e assim sucessivamente, além de que deverá validar o intervalo obrigatório de pausa do dia de trabalho ser no mínimo de 01:00H. Se a validação for negativa deverá ser apresentado ao utilizador a causa do erro e invalidar o registo.
016	Cálculo de diferenciais horários	O sistema calculará e deverá registar e apresentar o tempo em falta de forma a completar um dia de trabalho ou o tempo trabalhado a mais para lá do legalmente exigido. (7H)
017	Validação de número de registos diários	O sistema deverá permitir no máximo, por dia, 4 registos.
018	Gravação dos dados do registo	O sistema deverá ser capaz de editar e guardar os dados do registo efetuado pelo utilizador em ficheiro <i>Excel</i> (Mod.75/DGAV). O seu conteúdo deve ser protegido com <i>password</i> e respeitar o RGPD.
019	Geração de documento PDF (<i>Portable Document Format</i>)	Se o utilizador assim o desejar, o sistema deverá ser capaz de devolver um documento com todos os seus registos no formato PDF para posterior gravação ou impressão.

4.2. MÉTODO DE IDENTIFICAÇÃO E ACESSO AO SISTEMA

Como foi possível compreender através da Revisão da Literatura, existe uma panóplia alargada de escolhas no que diz respeito ao método que pode ser usado pelo utilizador para se identificar e autenticar perante o sistema. No entanto, à partida existem alguns que devem ser excluídos pois exigem um investimento inicial e de manutenção bastante elevado que vai contra um dos requisitos não funcionais que é a minimização de custos. A identificação por biometria ou por leitura de cartões RFID ficaram imediatamente excluídos da proposta de protótipo.

A identificação por leitura de cartões magnéticos é das tecnologias mais baratas no que toca à compra de leitores/gravadores e cartões de identificação. Foi a primeira escolha, mas o

autor desta Tese, após alguma reflexão deu a ideia de utilização do CC para autenticação do utilizador perante o sistema. A escolha do CC enquadra-se na categoria de *smartcard* como método de autenticação, e traz cinco vantagens muito importantes:

1. Todos os trabalhadores são portadores de CC, não havendo necessidade de aquisição de *smartcards*;
2. O CC já tem, em si, guardados os dados necessários para a autenticação, não sendo necessário programar um novo *smartcard*;
3. Muitos dos teclados existentes na DGAV já têm leitor de CC incorporado, o que mais uma vez evita a necessidade de aquisição de leitores, reduzindo potenciais custos;
4. Mitiga a hipótese de fraude na autenticação através da troca de cartões entre trabalhadores, uma vez que, a responsabilidade de troca de CC é muito maior do que se for um cartão *smartcard* indiferenciado.
5. O *Software Development Kit* (SDK) associado ao CC é Português e disponibilizado pela Agência para a Modernização Administrativa I.P. (AMA) sendo que o acesso ao suporte deverá ser muito mais fácil do que a maioria dos fabricantes internacionais de *smartcards*.

Antes da ideia ter sido aceite encetou-se uma análise à Legislação sobre a possibilidade de uso do CC para este efeito.

O trabalhador em funções públicas está sujeito aos deveres gerais previstos na LTFP¹, entre os quais dever de assiduidade e de pontualidade, que consistem em comparecer ao serviço regular e continuamente e nas horas que estejam designadas.

Por sua vez, o empregador público deve manter um registo que permita apurar o número de horas de trabalho prestadas pelo trabalhador, por dia e por semana, com indicação da hora de início e de termo do trabalho, bem como dos intervalos efetuados. Nos órgãos ou

¹ Aprovada pela Lei n.º 35/2014, de 20 de junho, alterada pelas Leis n.ºs 84/2015, de 7 de agosto, 18/2016, de 20 de junho, 42/2016, de 28 de dezembro, e 25/2017, de 30 de maio.

serviços com mais de cinquenta trabalhadores, o registo deve ser efetuado por sistemas automáticos ou mecânicos².

A Lei n.º 7/2007, de 5 de Fevereiro, na redação dada pela Lei n.º 32/2017, de 1 de junho, cria o Cartão de Cidadão (CC) e rege a sua emissão e utilização. O CC constitui título bastante para provar a identidade do titular perante quaisquer autoridades e entidades públicas ou privadas, sendo válido em todo o território nacional (cf. Art.º 4.º da referida Lei). O CC é um documento de identificação múltipla, que inclui uma zona específica destinada a leitura ótica e incorpora um ou mais circuitos integrados. Permite ao respetivo titular, entre outros, provar a sua identidade perante terceiros através da leitura de elementos visíveis, coadjuvada pela leitura ótica de uma zona específica. É ainda de ressaltar que, a leitura ótica da zona específica do cartão está reservada a entidades ou serviços do Estado e da AP³.

Em suma, a DGAV enquanto serviço da AP tem habilitação legal para uso dos dados do CC para autenticação no sistema de registo de assiduidade a implementar, o que tornou a ideia viável e por isso aceite.

4.3. LINGUAGEM DE PROGRAMAÇÃO E FORMA DE ARMAZENAMENTO A ADOPTAR

Começando pela linguagem de programação a escolher, aliando a necessidade de rapidez de desenvolvimento e de implementação do sistema só poderia ser a linguagem .NET. Pela sua flexibilidade de programação multiplataformas, pela experiência alargada do autor nesta linguagem adquirida ao longo dos anos a nível profissional e académico e pelo suporte dado pelo SDK do CC a linguagem de programação escolhida foi C#. O ambiente de

² Cf. n.ºs 1 e 2 do artigo 104.º da LTFP.

³ N.ºs 1, 2 e 3 do artigo 6º da Lei n.º 7/2007, de 5 de Fevereiro, na redação dada pela Lei n.º 32/2017, de 1 de junho.

desenvolvimento escolhido foi o Visual Studio 2019 cuja licença é oferecida aos alunos do ISEP.

Já a forma de armazenamento foi uma escolha mais complexa. A forma de guardar os registos recorrendo a uma base de dados foi descartada desde o início do projecto, uma vez que, o sistema seria implementado também nas unidades de abate, que sendo de direito privado, não havia permissão para que se abrisse um canal de comunicação com a rede interna da DGAV por questões de segurança. Poderia ser adoptado o uso de uma Virtual Private Network (VPN), mas daria origem a gasto de capital que nem os operadores privados nem a DGAV estavam dispostos a ter.

Sendo assim, o autor considerou como única solução, e de forma a ter os dados centralizados, a gravação dos ficheiros *Excel* com o conteúdo da assiduidade numa nuvem afeta a uma conta criada para o efeito e que só os administradores do sistema teriam acesso.

Relativamente à oferta de mercado para esta solução de armazenamento existe muita escolha, e quase todas com um excelente nível de segurança (no acesso e na fiabilidade de manutenção dos dados), com subscrições *free* e com disponibilização de bastante espaço. No entanto, pela forma como se posicionam no mercado, pelo suporte técnico existente na comunidade e pela experiência de utilização do autor com esta solução de armazenamento, as duas soluções em vista eram a *Google Drive* ou a *OneDrive*.

A escolha acabou por recair sobre a solução da “Microsoft”, “OneDrive”, pelo simples facto de que é uma solução que vem já pré-instalada no sistema operativo da *Microsoft* mais recente, o *Windows 10*. E, sendo assim, já que muitas das máquinas da DGAV já têm esse sistema operativo, assim como as máquinas afetas à IS nas unidades de abate, não há necessidade de mais uma aplicação instalada nas máquinas e gasto de tempo na sua instalação, reduzindo, assim, o tempo de implementação. É claro, que aliado a este facto, complementa-se a esta decisão o facto desta solução ter várias *Application Programming Interface* (API) que se integram na perfeição à linguagem e ambiente de programação escolhidos.

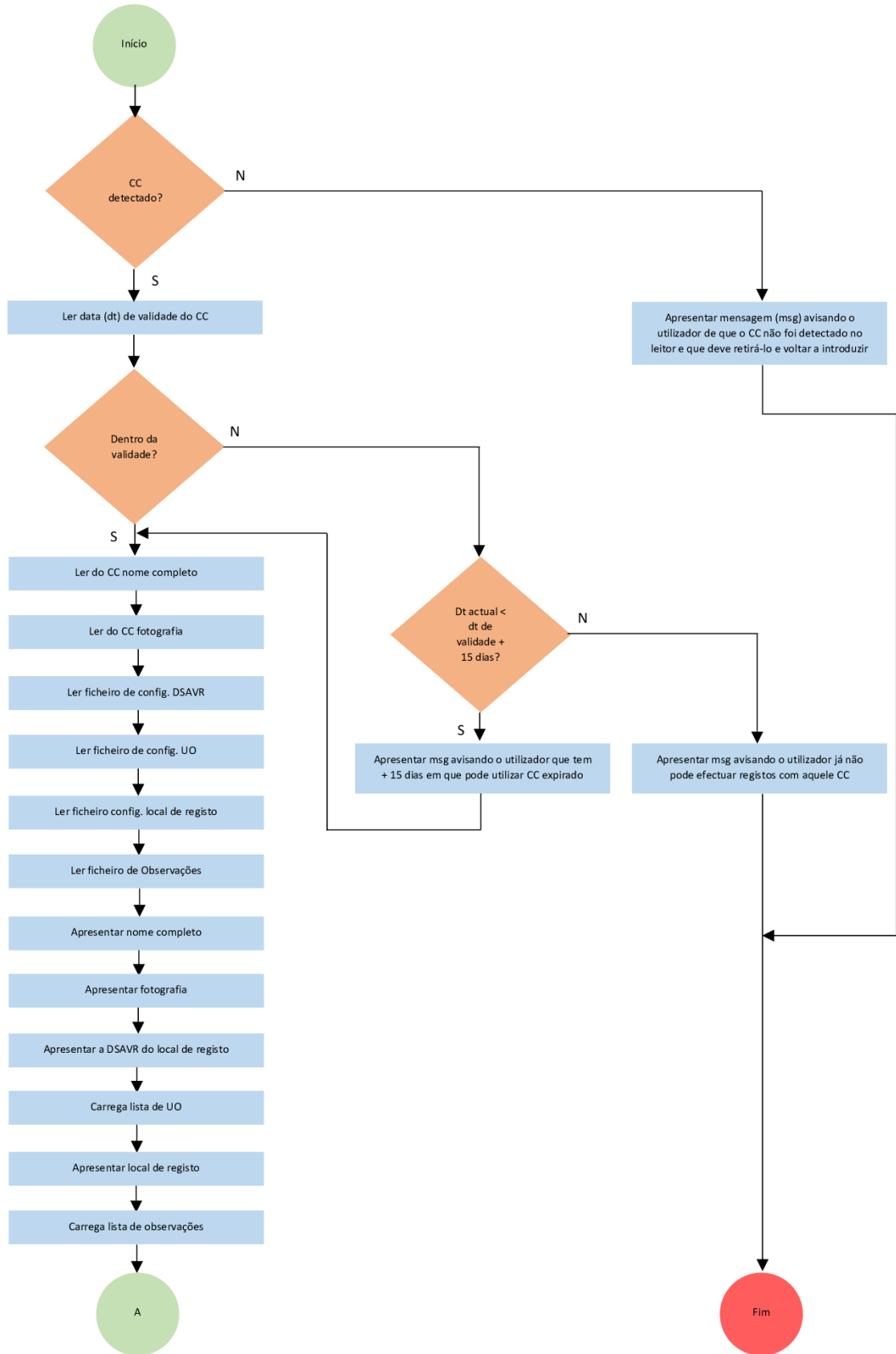
4.4. FLUXOGRAMA DO PROTÓTIPO

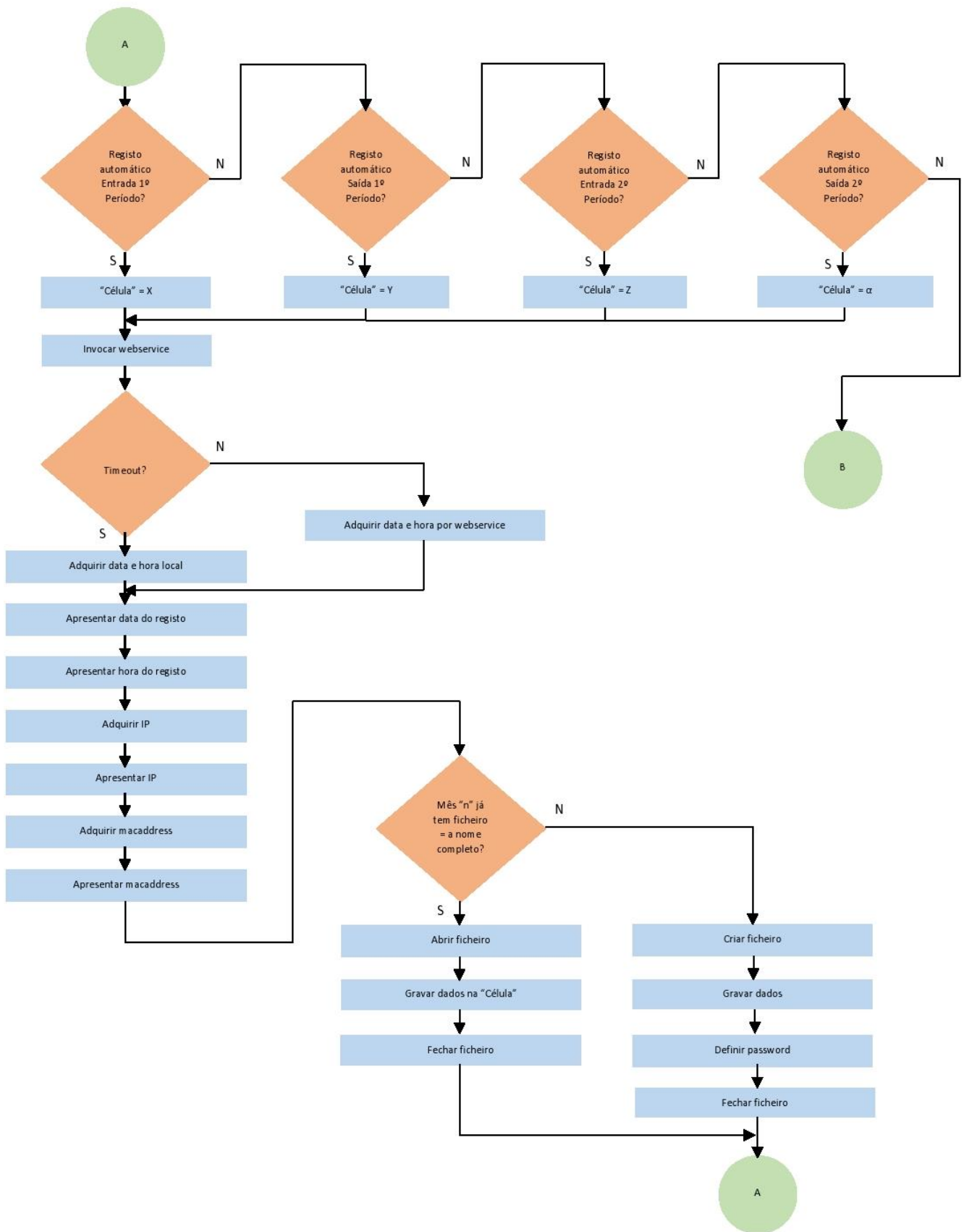
Dizem as boas práticas, que na criação ou melhoramento de qualquer processo devem ser usados fluxogramas para que se possa obter uma representação visual clara do processo, com descrições sucintas, precisas e claras de todas as etapas assim como a sua sequência. Permite ter uma visão global de todo o processo e é uma ótima ferramenta para se conseguir detetar facilmente etapas que não acrescentam valor, outras que conduzem a redundâncias indesejadas e até oportunidades de melhoria.

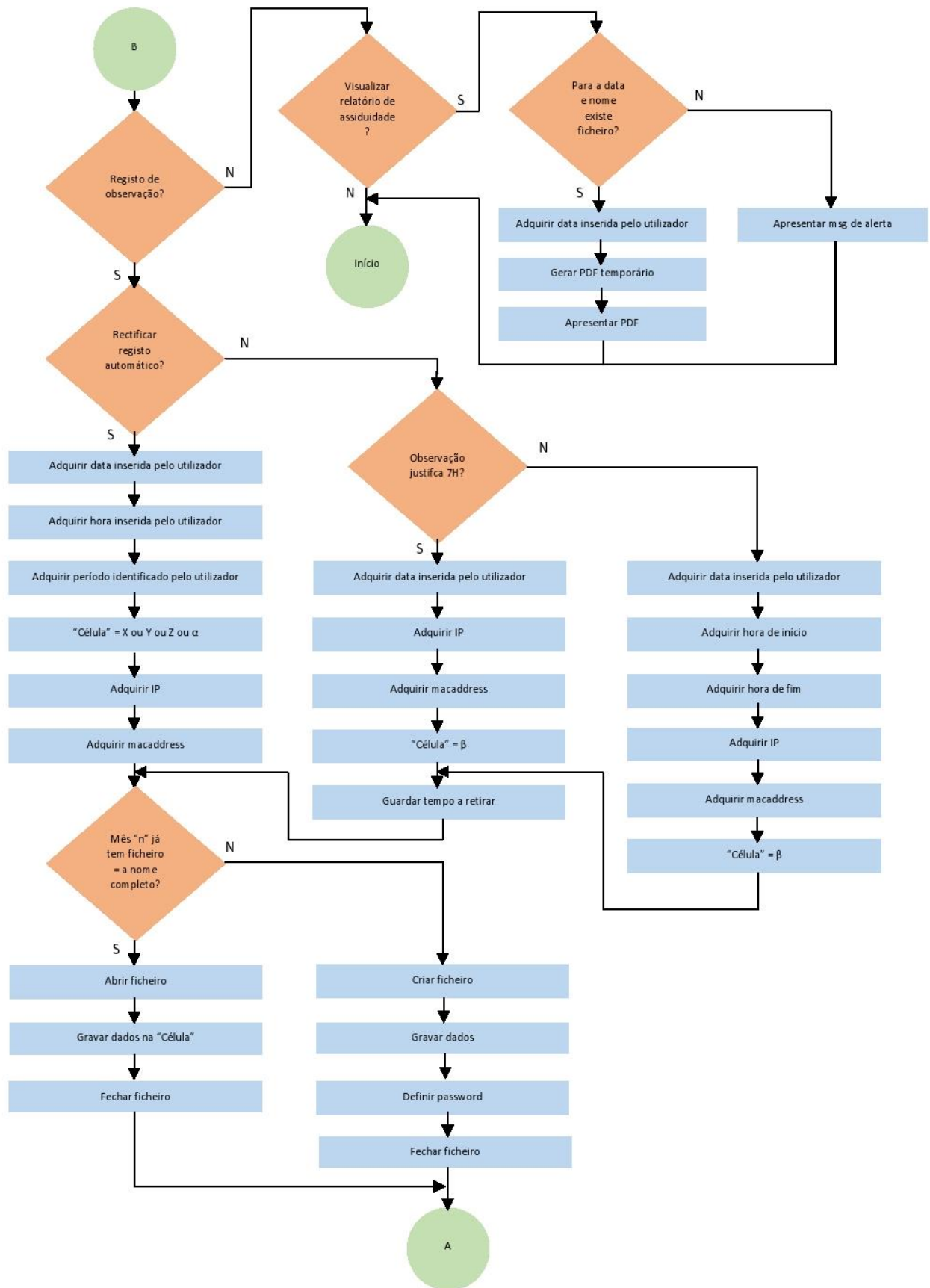
Efetivamente, esta boa prática, que não sendo obrigatória, ajuda muito na idealização e elaboração de aplicações informáticas, e no caso do SRACC não foi diferente. Depois de recolhidos e analisados os requisitos foram elaborados os fluxogramas referentes às funcionalidades principais do sistema de assiduidade a ser desenvolvido. É sempre um procedimento de tentativa e erro porque depois da construção de um fluxograma, imediatamente é vislumbrado algum erro, ou possível melhoria que é exactamente o que se deve esperar de uma ferramenta destas.

Assim sendo, depois de algumas tentativas, correcções e ajustes, foi possível juntar todos os fluxogramas das principais funcionalidades e elaborar uma versão global e final.

A elaboração deste fluxograma foi fundamental e sempre um farol no desenvolvimento deste novo sistema de assiduidade. Sempre que surgia alguma dúvida, a sua consulta era uma ajuda valiosa para manter o foco e as ideias sempre bem estruturadas. Este fluxograma pode ser visto imediatamente abaixo.







5. DESENVOLVIMENTO E APRESENTAÇÃO DO SISTEMA

Neste capítulo será dada a conhecer a programação das várias funcionalidades do sistema, demonstrando sempre com trechos de código e a respetiva explicação. É evidente que só serão mencionadas as mais importantes para o desenvolvimento deste sistema, querendo com isto dizer que, na programação de um sistema desta complexidade existem imensas rotinas de leitura, escrita, validação, entre outros, que, fazendo parte do código, não são o que verdadeiramente destringam este sistema de outros.

No entanto, e devido à sua extensão, o código poderá ser consultado, na sua totalidade, em formato digital que acompanhará a versão impressa.

5.1. REQUISITOS PRÉVIOS

5.1.1. Criação de conta *OneDrive* e estrutura de pastas

Como já tinha sido referido, e por várias razões, o local de armazenamento da aplicação, das suas dependências e dos ficheiros de assiduidade é numa nuvem, neste caso concreto a *OneDrive*.

Esta opção permite, por exemplo, que a versão seja sempre a mesma e o seu *deploy* possa ser feito apenas uma vez para todos os terminais.

Foi por isso criada uma conta *OneDrive* que, gratuitamente, disponibiliza 5GB, com a estrutura na sua raiz como indica a figura 24.

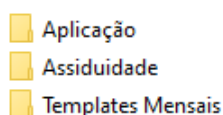


Figura 24. Estrutura de pastas da *OneDrive*

Depois, de forma automática pelo sistema, são criadas subpastas em “Assiduidade” onde são guardados os ficheiros de assiduidade dos trabalhadores, por Ano e Mês como exemplifica a figura 25.

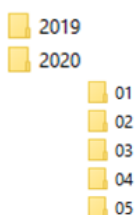


Figura 25. Estrutura de pastas criadas pelo sistema de assiduidade

É também necessário, no momento da instalação, copiar para a raiz do disco local a pasta “DSAVRN” (localizada na *OneDrive* em “Aplicação”) onde estarão ficheiros que serão lidos pelo sistema e que contêm informações específicas de cada posto onde é instalada a

aplicação, como, por exemplo, o endereço onde está instalado a *OneDrive* ou o local onde foi instalada a aplicação. Mais à frente, será detalhada a importância destes ficheiros, a forma como são lidos e como podem ser livremente editados.

5.1.2. Adição de bibliotecas

Inicialmente, foi necessário recorrer ao Manual SDK que a AMA disponibiliza para saber quais os requisitos, como as bibliotecas a serem importadas e os métodos a serem utilizados para a deteção do CC e para a leitura dos seus dados [31].

Em primeiro lugar, é necessário instalar a aplicação oficial do CC em autenticação.gov.pt de acordo com a arquitectura x86 ou x64.

De seguida, é necessário adicionar a biblioteca “*pteidlib_dotnet.dll*” às *references* do projecto *Visual Studio*, como demonstra a figura 26. Este ficheiro encontra-se na pasta “*sdk*” no local onde foi instalado a aplicação oficial do CC.

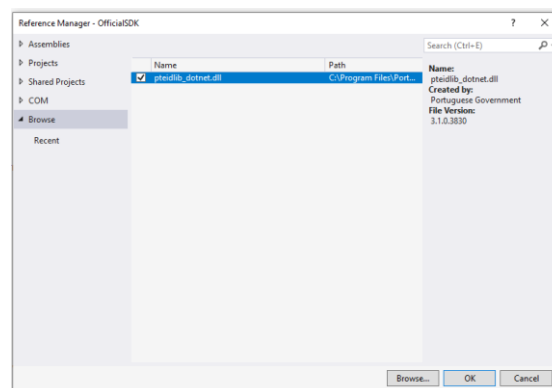


Figura 26. Adição da biblioteca *pteidlib_dotnet.dll*

Agora, de forma a chamar a biblioteca adicionada, devemos no início do código acrescentar o *namespace*:

```
using eidpt;
```

Adicionou-se também às *references* do projeto a biblioteca relativa ao *Microsoft Excel* uma vez que o sistema irá manipular ficheiros deste processador de cálculo. É uma biblioteca interna, “Microsoft.Office.Interop.Excel” e foi adicionada ao código com o *namespace*:

```
using Excel = Microsoft.Office.Interop.Excel;
```

Por último, e uma vez que as fotografias estão guardadas em formato “JPEG200”, foi usada outra biblioteca, “CSJ2K.dll”, que pode ser descarregada em <https://www.nuget.org/packages/CSJ2K/>. Foi esta também adicionada às *references* do projeto e ao código através do *namespace*:

```
using CSJ2K.Util;
```

Resumidamente, são estes os requisitos prévios a ter em conta para o desenvolvimento e instalação do sistema. Pode ser visto no Manual Técnico – TimeSet, no anexo C, em detalhe, todos os requisitos técnicos deste sistema.

5.2. LOGIN NO SISTEMA

Para fazer o login neste sistema de assiduidade é condição bastante e essencial o trabalhador inserir o seu CC válido no leitor antes de iniciar a aplicação. Ou seja, o sistema aceita qualquer CC válido como autenticação, mesmo que não seja trabalhador da DGAV. O sistema não faz validação se aquele CC em particular é de um trabalhador da DGAV ou não. Este método foi equacionado assim, uma vez que a elevada dinâmica de entradas e saídas de trabalhadores da DGAV, e frequente falta de comunicação entre Sede e DSAVR traria muitos problemas e um trabalho acrescido a quem administraria o sistema. Teria de ter, a todo o tempo, uma base de dados atualizada dos trabalhadores ativos para que fosse feita a validação sempre que fosse introduzido um CC. A pessoa identificada em determinado CC só tem acesso aos dados dessa mesma pessoa estando, por isso, sempre salvaguardados os dados alheios.

O Diagrama de Casos de Uso (DCU) desta funcionalidade foi elaborado de acordo com a figura 27. O *Login* está presente em todas as outras funcionalidades, uma vez que não é possível acedê-las sem primeiro validar a entrada no sistema.

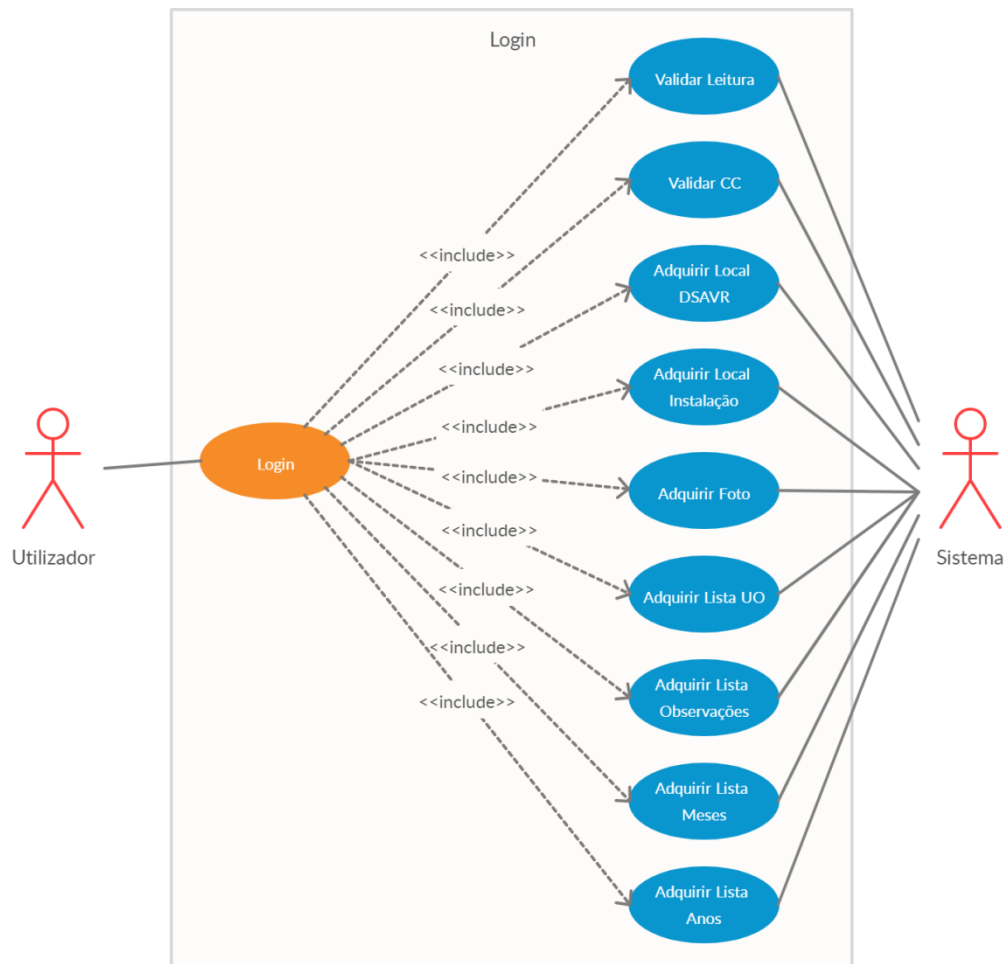


Figura 27. DCU para a funcionalidade *Login*

5.2.1. Validação de leitura do CC

Como já foi referenciado, o *login* será a simples detecção de um CC válido no leitor. Para isso foram escritas as seguintes linhas de código:

```

public Form1()
{
//VALIDA SE OS DADOS DO CARTÃO DE CIDADÃO (CC) PODEM SER LIDOS OU
NÃO
    InitializeComponent();
    try
    {
        Pteid.Init(null);
    }
    catch
    {
        MessageBox.Show("CARTÃO DE CIDADÃO NÃO DETETADO.
RETIRE O CARTÃO E INTRODUZA NOVAMENTE E VOLTE A
INICIAR APLICAÇÃO");
        Environment.Exit(0);
    }
//FIM
}

```

Se nada for detetado, o sistema não inicia e apresenta uma mensagem de erro ao utilizador conforme figura 28.

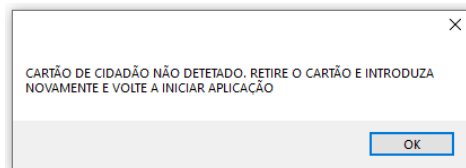


Figura 28. Mensagem ao utilizador de CC não detetado

5.2.2. Aquisição de data e hora atual/local

Sendo certo que o sistema de assiduidade será também instalado nas unidades de abate onde a DGAV não controla de todo a forma de acesso às definições do Sistema Operativo (SO) foi necessário equacionar a forma de aquisição das informações data e hora. Sem este controlo, seria possível ao utilizador definir uma data e hora local a gosto através das definições de data e hora do SO.

Assim a aquisição de data e hora, por defeito, será através da invocação de um *webservice* aos servidores da *Microsoft*.

No entanto, é necessário que o sistema funcione mesmo sem o acesso à internet (fatores externos) e aplicar uma redundância. Neste caso, se o sistema não conseguir obter resposta à invocação, a data e hora considerada será a local. O código implementado para o efeito foi o seguinte:

```
//CAPTURA DO MÊS E DO ANO LOCAL/WEB
string sDay7= DateTime.Now.ToString("dd");
string sMonth7 = DateTime.Now.ToString("MM");
string sYear7 = DateTime.Now.ToString("yyyy");
DateTime dateTime7 = DateTime.MinValue;
System.Net.HttpWebRequest request7 =
(System.Net.HttpWebRequest) System.Net.WebRequest.Create("http://www.
microsoft.com");
request7.Method = "GET";
request7.Accept = "text/html, application/xhtml+xml, */*";
request7.UserAgent = "Mozilla/5.0 (compatible; MSIE 10.0; Windows NT
6.1; Trident/6.0)";
request7.ContentType = "application/x-www-form-urlencoded";
request7.CachePolicy = new
System.Net.Cache.RequestCachePolicy(System.Net.Cache.RequestCacheLev
el.NoCacheNoStore);
    try
    {
        request7.Timeout = 2000;
        System.Net.HttpWebResponse response
        =(System.Net.HttpWebResponse) request7.GetResponse();
        response.Close();
        if (response.StatusCode ==
        System.Net.HttpStatusCode.OK)
        {
            string todaysDates = response.Headers["date"];

            dateTime7 = DateTime.ParseExact(todaysDates,
            "ddd, dd MMM yyyy HH:mm:ss 'GMT'",

            System.Globalization.CultureInfo.InvariantCulture,
            System.Globalization.DateTimeStyles.AssumeUniver
            sal);
            sDay7= (dateTime7.ToString("dd"));
            sMonth7 = (dateTime7.ToString("MM"));
            sYear7 = (dateTime7.ToString("yyyy"));
        }
    }
    catch
    {
        Console.WriteLine("Sem acesso à internet");
    }
//FIM
```

5.2.3. Aquisição e validação da data de validade do CC

Uma vez que o CC deve ser válido, como já anteriormente mencionado, o sistema valida a data de validade do mesmo. O sistema dá uma tolerância de quinze dias sobre a data de validade para possibilitar que o utilizador tenha acesso ao sistema enquanto faz a renovação do seu CC. A aquisição da data de validade do CC e a validação é feita através do seguinte código:

```
//VERIFICA A VALIDADE DO CARTÃO DE CIDADÃO
CardData i = getCardData();
string validade = i.IdInformation.validityDate.ToString();
string validade1 = validade.Remove(2, 8);
string validade2 = validade.Remove(0, 6);
var validade3 = validade.Remove(0, 3);
var validade4 = validade3.Remove(2, 5);
var validade5 = validade1 + "/" + validade4 + "/" + validade2+
"+"00:00:00";
var validade6 = sDay7 + "/" + sMonth7 + "/" + sYear7 + " " +
"00:00:00";
DateTime dateTime8 = DateTime.ParseExact(validade5, "dd/MM/yyyy
HH:mm:ss",
System.Globalization.CultureInfo.InvariantCulture.DateTimeFormat,
System.Globalization.DateTimeStyles.AssumeUniversal);
DateTime dateTime9 = DateTime.ParseExact(validade6, "dd/MM/yyyy
HH:mm:ss",
System.Globalization.CultureInfo.InvariantCulture.DateTimeFormat,
System.Globalization.DateTimeStyles.AssumeUniversal);
DateTime dateTime10 = dateTime8.AddDays(15);
    if (dateTime9 > dateTime10)
    {
        MessageBox.Show("A VALIDADE DO SEU CARTÃO DE CIDADÃO
EXPIROU. A TOLERÂNCIA TAMBÉM EXPIROU. NÃO É POSSÍVEL
FAZER PICAGENS. O PROGRAMA IRÁ ENCERRAR!");
        Environment.Exit(0);
    }
    if (dateTime9>dateTime8)
    {
        MessageBox.Show("A VALIDADE DO SEU CARTÃO DE CIDADÃO
EXPIROU. EXISTE UMA TOLERÂNCIA DE 15 DIAS CORRIDOS PARA
CONTINUAR A PICAR COM O CARTÃO EXPIRADO! DEPOIS DISSO DEIXA
DE SER POSSÍVEL PICAR COM ESTE CARTÃO! CARREGAR EM 'OK'
PARA CONTINUAR!");
    }
//FIM
```

Sendo assim, se o utilizador ultrapassar os 15 dias de tolerância irá receber um aviso sob a forma de mensagem que pode ser visto na figura 29, e o programa irá encerrar.

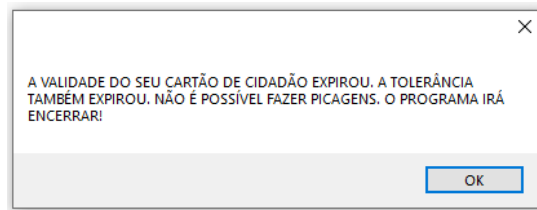


Figura 29. Mensagem ao utilizador de validade e tolerância expirada

Na tentativa de aceder ao programa, com o CC expirado mas dentro dos quinze dias de tolerância, o utilizador será informado que o seu CC expirou e que ainda conseguirá aceder ao sistema de assiduidade por mais 15 dias a contar da data de expiração. A informação é dada através de uma caixa de mensagens como mostra a figura 30.

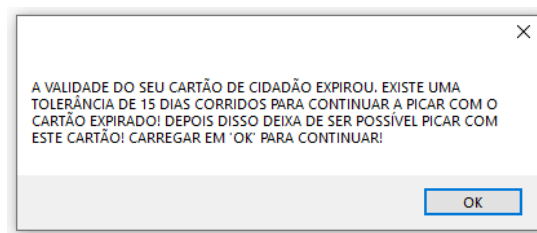


Figura 30. Mensagem ao utilizador de validade do CC expirada mas dentro da tolerância

5.2.4. Aquisição do nome completo do CC

É através do nome completo presente no CC inserido no leitor que o utilizador se autentica e apresenta perante o sistema. É certo que o nome completo não é um dado com características únicas, uma vez que, pode haver mais do que um nome completo igual. Não é de todo o ideal e seria importante complementá-lo com o número de CC ou Número de Identificação Fiscal (NIF), por exemplo, de forma a atestar que não haveria dois utilizadores iguais. No entanto, por parte da DGAV, houve sempre uma prerrogativa de não usar qualquer dado do CC do trabalhador que não apenas o seu nome completo. Colocada a questão da possibilidade de vários utilizadores com o mesmo nome, situação totalmente

indesejável para qualquer sistema, foi referido ao autor que neste Serviço não havia dois trabalhadores com o mesmo nome e que essa possibilidade dispersa pelas várias DSAVR era ainda menos provável no futuro. Deixou claro que, se isso viesse, a acontecer, só aí se discutiria outras opções.

Sendo assim, é de seguida apresentado o código usado para leitura e formatação do nome completo:

```
//LÊ, CODIFICA PARA UTF8 E APRESENTA O NOME COMPLETO DO UTILIZADOR A PARTIR DO CC
string space = i.IdInformation.firstname.ToString();
string paddedParam = space + " ";
int space2 = space.Length;
textBox1.Text = paddedParam + i.IdInformation.name.ToString();
byte[] bytes1 = Encoding.Default.GetBytes(textBox1.Text);
textBox1.Text = Encoding.UTF8.GetString(bytes1);
//FIM
```

5.2.5. Aquisição da fotografia do CC

Embora não fosse um requisito essencial ler e apresentar este dado, foi opinião de todos os elementos ligados ao projeto que faria sentido dotar o sistema de um ambiente gráfico mais inclusivo e personalizado de forma a torná-lo mais *user-friendly*. Sendo assim, foi utilizado o seguinte código para obter a fotografia do utilizador gravado no CC:

```
//CAPTURA DA FOTOGRAFIA DO CARTÃO DE CIDADÃO
Pteid.SetSODChecking(0);
var pictureData = Pteid.GetPic();
var pictureBytes = pictureData.picture;
BitmapImageCreator.Register();
var j2kImg = CSJ2K.J2kImage.FromBytes(pictureBytes);
var bmap = j2kImg.As<Bitmap>();
pictureBox1.Image = bmap;
CardData i = getCardData();
//FIM
```

5.2.6. Aquisição da DSAVR na qual está instalado o sistema

Sempre que é feito o *login* no sistema, este apresenta uma série de informações ao utilizador, uma das quais, qual a DSAVR onde está afeto aquele terminal de registo de

assiduidade em questão. Sendo que se pretendia que o sistema fosse instalado em todas as DSAVR, foi necessário criar um método de maneira a que fosse possível editar o nome da região sem ser necessário fazê-lo directamente no código (*hardcoding*). Para que assim fosse, codificou-se o sistema para que obtivesse este dado através de uma fonte externa, neste caso optou-se por um simples ficheiro *.txt. Pode ser aberto com qualquer processador de texto e editado livremente. Este ficheiro está localizado na pasta “Aplicação” da *OneDrive* e tem o nome de “Direção de Serviços”. Não é necessário estar na pasta local “DSAVRN” uma vez que cada DSAVR terá uma conta *OneDrive* própria. O ficheiro e o seu conteúdo pode ser visto na figura 31.

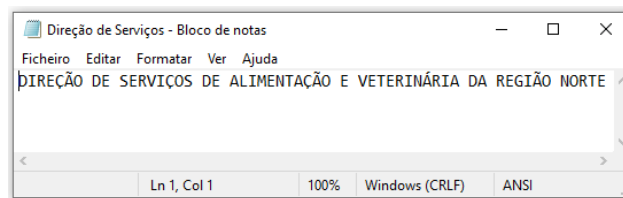


Figura 31. Ficheiro *.txt com a informação do nome da DSAVR

A codificação desenvolvida para a leitura da informação contida neste ficheiro foi a seguinte:

```
//LEITURA E APRESENTAÇÃO DA DIREÇÃO DE SERVIÇOS REGIONAL A PARTIR DO  
CAMINHO RESPECTIVO  
string[] unidade2 = File.ReadAllLines(text +  
@"\Aplicação\Debug\Direção de Serviços.txt",  
Encoding.GetEncoding("windows-1254"));  
string array = String.Concat(unidade2);  
label15.Text = array;  
//FIM
```

5.2.7. Aquisição da lista de unidades orgânicas da respetiva DSAVR

A lógica para aquisição da lista de unidades orgânicas é a mesma da demonstrada no ponto anterior. De forma a evitar o *hardcoding* foi criado, também, um ficheiro na pasta “Aplicação” da *OneDrive* o ficheiro *.txt com o nome “Unidade Orgânica”. O conteúdo do ficheiro pode ser visto na figura 32.

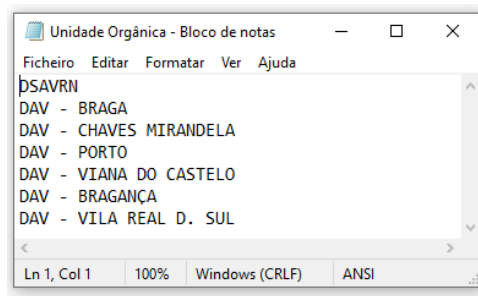


Figura 32. Ficheiro *.txt com a informação das unidades orgânicas da DSAVRN

O código responsável pela leitura destes dados constantes no ficheiro pode ser visto de seguida:

```
//LEITURA DE TODAS AS UO EXISTENTES NO CAMINHO IDENTIFICADO E POPULA  
A RESPECTIVA COMBOBOX  
string[] unidade = File.ReadAllLines(text +  
@"\Aplicação\Debug\Unidade Orgânica.txt",  
Encoding.GetEncoding("windows-1254"));  
    foreach (var line2 in unidade)  
    {  
        comboBox9.Items.Add(line2);  
    }  
//FIM
```

5.2.8. Aquisição de lista de observações

Qualquer sistema de assiduidade deve permitir ao utilizador a possibilidade de justificar ausências ao serviço, leia-se ausência de registos de assiduidade. Sendo assim, de forma análoga ao desenvolvido nos pontos anteriores, foi criado um ficheiro externo (*.txt) com vários tipos de observações e justificações que podem, a qualquer momento, ser criadas ou removidas. Neste ficheiro foram incluídas as várias possibilidades de faltas justificadas que a LGTFP prevê, como por exemplo, greve, trabalhador-estudante, consulta médica, assim como outras que são próprias e específicas enquadradas na missão da DGAV a saber, serviço externo, folga por compensação, representação DGAV entre outras. É também neste ficheiro que estão localizadas as observações relativas a erros de/e alheios ao sistema.

Este ficheiro encontra-se também localizado na pasta “Aplicação” da *OneDrive* com o nome “Observações”. Parte do conteúdo do ficheiro pode ser visto na figura 33.

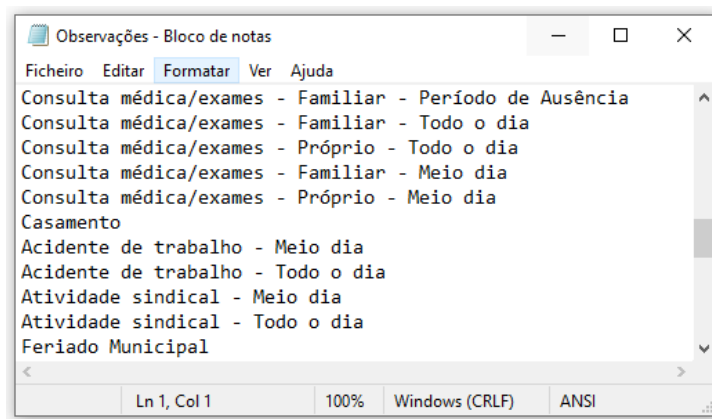


Figura 33. Ficheiro *.txt com algumas observações/justificações existentes

O código desenvolvido para a leitura destes dados constantes no ficheiro pode ser visto de seguida:

```
//LEITURA DE TODAS AS OBSERVAÇÕES EXISTENTES NO CAMINHO IDENTIFICADO
E POPULA A RESPECTIVA COMBOBOX
string[] lineOfContents = File.ReadAllLines(text +
@"\Aplicação\Debug\Observações.txt", Encoding.GetEncoding("windows-
1254"));
    foreach (var line in lineOfContents)
    {
        comboBox1.Items.Add(line);
    }
//FIM
```

5.2.9. *Graphic User Interface (GUI)* do SRACC

Finalmente, depois da validação de todos os requisitos e da aquisição de todos os dados necessários para *login*, é apresentado ao utilizador a janela principal do SRACC conforme se pode comprovar na figura 34.

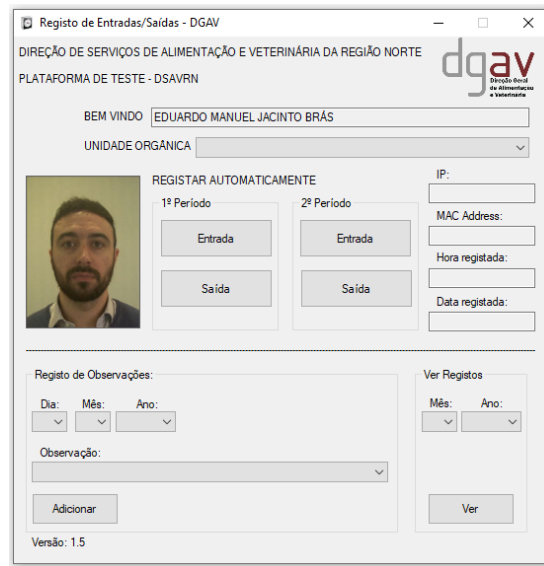


Figura 34. Página principal do sistema de registo de assiduidade

De referir que, as caixas de seleção referentes a Dias, Meses e Anos são populadas automaticamente pelo sistema, pelo que não há necessidade de recurso à leitura de um ficheiro externo. Estas caixas de seleção têm todas um sistema de verificação, ou seja, se o utilizador para determinada funcionalidade que as requeira não selecionar um valor, o sistema devolverá uma mensagem de aviso interrompendo o processo.

No próximo ponto serão apresentadas as várias possibilidades de registo de assiduidade, a sua visualização/monitorização, assim como, o respetivo código desenvolvido.

5.3. REGISTO DE ASSIDUIDADE AUTOMÁTICA

Após o login efetuado e apresentada a janela principal do sistema ao utilizador, como é possível ver na figura imediatamente acima é possível ao mesmo, a partir daqui, fazer o seu registo de assiduidade. Esta janela foi dividida em quatro botões. Dois botões, um de Entrada e outro de Saída para o 1º Período de trabalho do dia e outros dois botões também de Entrada e Saída para o 2º Período de trabalho do dia.

Assim, o utilizador deve, em primeiro lugar, e, obrigatoriamente seleccionar a sua Unidade Orgânica e depois carregar no botão “Entrada” ou “Saída” do 1º ou 2º Período conforme o caso. Neste tipo de registos são feitas quatro validações.

1. Valida a selecção da Unidade Orgânica. Se nada for seleccionado o processo de registo de assiduidade é interrompido e é mostrada uma mensagem de aviso, como se pode ver na figura 35.

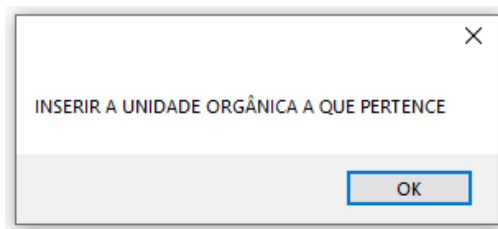


Figura 35. Mensagem de aviso de não selecção da Unidade Orgânica

2. O sistema verifica se, para cada ação de registo, já existe ou não uma igual feita anteriormente. Ou seja, o utilizador não consegue, para o mesmo dia, registar a assiduidade carregando uma segunda vez no mesmo botão. Se o utilizador fizer esta tentativa será mostrada a mensagem espelhada na figura 36 e o registo é interrompido.

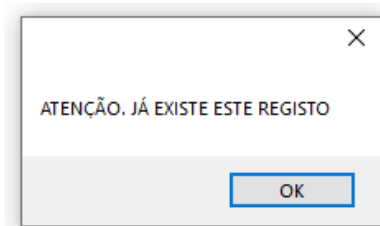


Figura 36. Mensagem de aviso de registo de assiduidade já existente

3. Outra verificação prende-se, com a sequência de registos, ou seja, o utilizador não consegue, por exemplo, fazer um registo de “Saída” do 1º Período sem antes ter feito

o registo de “Entrada” do mesmo período, como se pode comprovar pela figura 37 na mensagem que é apresentada ao utilizador. Também aqui o registo fica suspenso.

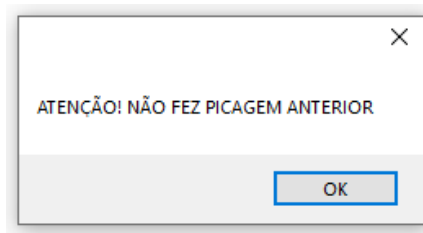


Figura 37. Mensagem de aviso de registo anterior não efectuado

4. A validação final tem a ver com a hora de pausa (vulgarmente de almoço) que por Lei, na AP, é de, no mínimo, obrigatoriamente uma hora. Assim, o sistema só permite ao utilizador fazer um registo de “Entrada” do 2º Período após uma hora relativamente ao registo anterior. O não respeito desta regra interrompe o registo avisando o utilizador com a mensagem como mostra a figura 38.

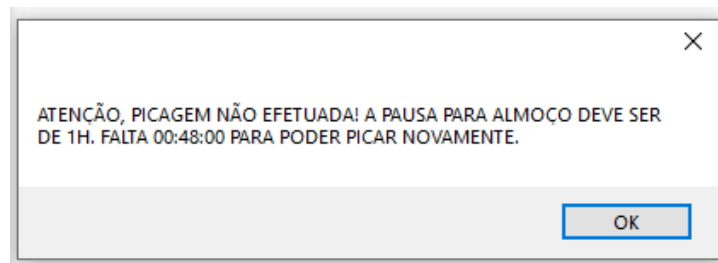


Figura 38. Mensagem de aviso de não respeito pela pausa de almoço

De forma a compreender, ainda melhor, o comportamento da funcionalidade “Registo de Assiduidade Automática”, todas as interações e cenários possíveis entre utilizador e sistema, podem ser observados na figura 39 através do respectivo DCU.

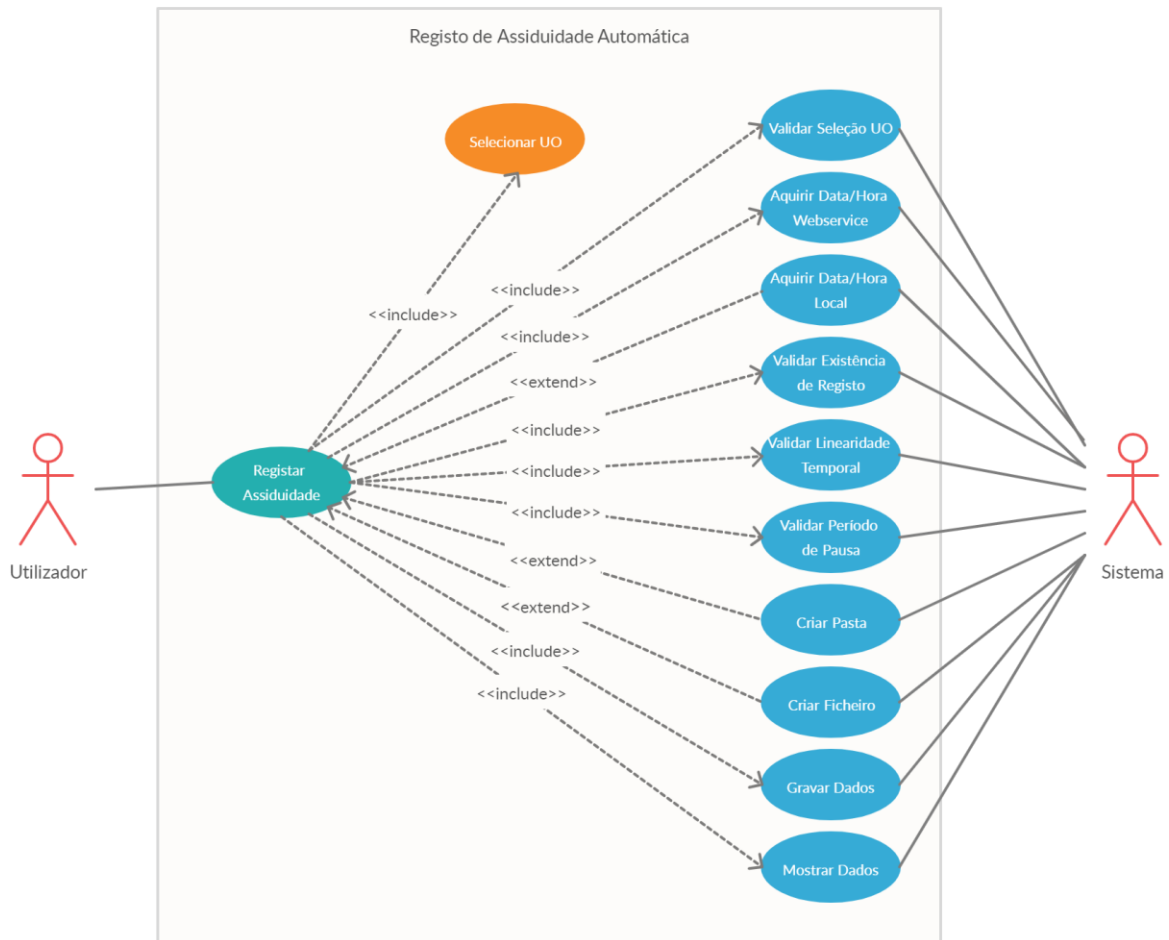


Figura 39. DCU para a funcionalidade “Registo de Assiduidade Automática”

5.3.1. Aquisição da unidade orgânica ou Unidade de Abate onde é efectuado o registo de assiduidade

Como foi descrito no capítulo “Requisitos prévios”, é necessário copiar a pasta “DSAVRN” que se encontra na pasta da “OneDrive” “Aplicação” para a raiz do disco local. Esta pasta contém um ficheiro *.txt com o nome “Matadouro_DAV_NAV_DSAVRN” cujo conteúdo é o nome do local onde está instalado o sistema usado para fazer o registo de assiduidade naquele momento, vide figura 40. Esse ficheiro é lido pelo sistema e coloca essa informação apenas ao registo de assiduidade. É bem verdade que este ficheiro sendo de edição livre e localizado no disco local significa um fácil acesso e alteração do seu conteúdo a não administradores. No entanto, o conteúdo, ou seja, o nome do local onde foi instalado o

sistema deve ser sempre coincidente com o *macaddress* apontado na altura da sua instalação. Para isso existe uma tabela na posse dos administradores que relaciona o *macaddress* com o nome do local para que se possam identificar possíveis adulterações. Embora não essencial, o nome do local torna a supervisão dos registos, por quem de direito, muito mais expedita.

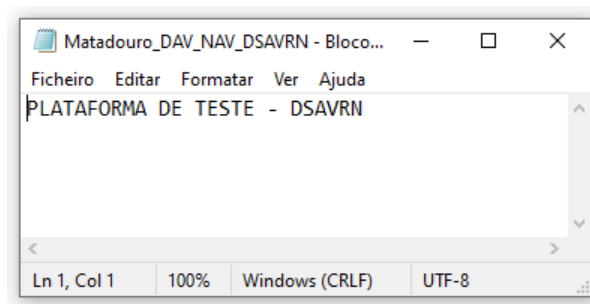


Figura 40. Ficheiro *.txt com o nome do local onde está instalado o sistema

Neste caso o conteúdo é lido, apresentado na janela principal do sistema (vide figura 34) e guardado temporariamente numa variável que será depois descarregada como informação suplementar no registo de assiduidade do utilizador. O código desenvolvido foi o seguinte:

```
//LÊ E APRESENTA O NOME DO MATADOURO OU UNIDADE ORGÂNICA (UO) A  
PARTIR DO CAMINHO ESPECIFICADO  
string matadouro =  
System.IO.File.ReadAllText(@"C:\DSAVRN\Matadouro_DAV_NAV_DSAVRN.txt"  
, Encoding.GetEncoding("windows-1254"));  
label13.Text = matadouro;  
//FIM
```

5.3.2. Aquisição dos dados “Data”, “Hora”, “IP” e “*macaddress*”

Quanto à aquisição da Data e Hora a lógica e a rotina usada é a mesma da demonstrada no ponto 5.2.2.

A aquisição do IP prendeu-se essencialmente com a perspetiva de os utilizadores poderem, de forma mais rápida, saber e transmitir ao administrador este dado, para que se possa aceder remotamente a determinada máquina de forma a resolver alguma problemática relacionada com os registos de assiduidade daquele posto de trabalho em particular. Não é de todo uma forma de identificação inequívoca desse posto, uma vez, que nem as unidades de abate nem as unidades orgânicas da DGAV têm como política a fixação dos endereços IP. É neste contexto que é adquirido também o *macaddress* aquando de qualquer registo de assiduidade. É este dado que identifica inequivocamente a máquina onde é feito determinado registo. Quando é feita a instalação do sistema, o técnico deve sempre relacionar o *macaddress* da máquina com o local onde ela se encontra numa tabela *Excel*. Na figura 41 pode ser visto um extracto desta tabela. A tabela está constantemente atualizada pelos administradores e as unidades de abate devem obrigatoriamente comunicar à DGAV sempre que substituam o módulo de rede por onde acedem à internet de forma a ser atualizado na tabela o novo *macaddress*.

TIMESET - GESTÃO E CONTROLO DE MACADDRESS			
Código	Localização	macaddress	Tipo
000001	DAV - Porto	00-1E-0B-63-F3-61	UO
000002	DAV - Vila Real	E8-39-35-50-2E-BF	UO
000003	Izicar	E8-39-35-54-C8-2E	UA
000004	Landeiro	F0-4D-AL-0E-BA-43	UA
000005	Linda Rosa	00-1E-0B-82-61-D3	UA
000006	Seara	00-19-99-48-48-E0	UA
000007	Sorares e Santos	00-1E-0B-64-8F-CB	UA

Figura 41. Extrato da tabela usada na gestão de *macaddress*

Todos os dados referidos neste ponto, Data, Hora, IP e *macaddress*, são apresentados ao utilizador na página principal do sistema como se pode atestar na figura 42. Assim, o utilizador sabe que dados foram inseridos na sua folha de assiduidade e com isto dá também a possibilidade de deteção de algum erro.

IP:	127.0.0.1
MAC Address:	90:11:20:52:41:53
Hora registada:	09:30
Data registada:	01 jun 2020

Figura 42. Informações apresentadas ao utilizador aquando do registo de assiduidade

O código desenvolvido para a aquisição IP e do *macaddress* foi o seguinte:

```
// LER O IP
string sHostName = Dns.GetHostName();
string IPfinal = "0";
IPHostEntry ipE = Dns.GetHostEntry(sHostName);
IPAddress[] IpA = ipE.AddressList;
    for (int a = 0; a < IpA.Length; a++)
    {
        IPfinal = IpA[a].ToString();
    }

//FIM

//LER MACADDRESS
System.Management.ManagementClass objMgmtCls = new
System.Management.ManagementClass("Win32_NetworkAdapter");
List<string> colors = new List<string>();
    foreach (System.Management.ManagementObject objMgmt in
objMgmtCls.GetInstances())
    {
        string result;
        result = Convert.ToString(objMgmt["MACAddress"]);
        if (String.IsNullOrEmpty(result))
            Console.WriteLine("is null or empty");
        else
            listBox1.Items.Add(result);
    }
string[] arr = new string[listBox1.Items.Count];
    for (int s = 0; s < listBox1.Items.Count; s++)
    {
        arr[s] = listBox1.Items[s].ToString();
    }
string address = (arr[0]);
string macIP = "IP:" + " " + IPfinal + " " + "MACAddress:" + " " +
address + " " + matadouro + " " + redeok;
//FIM
```

5.3.3. Geração e gravação de uma folha de cálculo com os dados adquiridos

O processo de registo de assiduidade culmina com transferência dos vários dados adquiridos para a folha de registos de assiduidade mensal do trabalhador. Foi referido nos “Requisitos técnicos” a necessidade da *OneDrive* possuir na sua estrutura uma pasta essencial. Trata-se da pasta “Templates Mensais”. Nesta pasta estão doze ficheiros *Excel* correspondentes a cada mês do ano que são nada mais nada menos do que o Mod.75/DGAV adaptado. Ou seja, para cada mês existe um Mod.75/DGAV com os respectivos dias de trabalho, fins-de-semana e feriados existentes cujo conteúdo das células onde são registadas as horas de Entrada e Saída está totalmente em branco.

Neste ponto do processo, o sistema verifica se já existem as pastas “Ano” e “Mês” referentes ao momento do registo. Se assim não for, são automaticamente criadas na pasta “Assiduidade” da *OneDrive*. De seguida, verifica, finalmente, se o utilizador já fez algum registo para aquele “Ano” e “Mês”. Se ainda não, copia o *template* (ficheiro *Excel* Mod. 75/DGAV) para a pasta “Assiduidade”>”Ano”>”Mês” e atribui-lhe como nome, o nome completo do utilizador.

Feitas estas duas validações o sistema abre o ficheiro *Excel* e insere na célula correspondente ao dia, ao tipo de registo (Entrada/Saída) e ao período de trabalho, as informações anteriormente adquiridas e referidas nos pontos anteriores.

Logo de seguida, é atribuída uma *password* aos ficheiros, que só os administradores têm conhecimento, e é feita a gravação dos dados no ficheiro *Excel*. Esta é a etapa final do processo de registo de assiduidade e o seu término é dado a conhecer ao utilizador exibindo os dados do registo e uma mensagem de sucesso como se pode ver na figura 43.

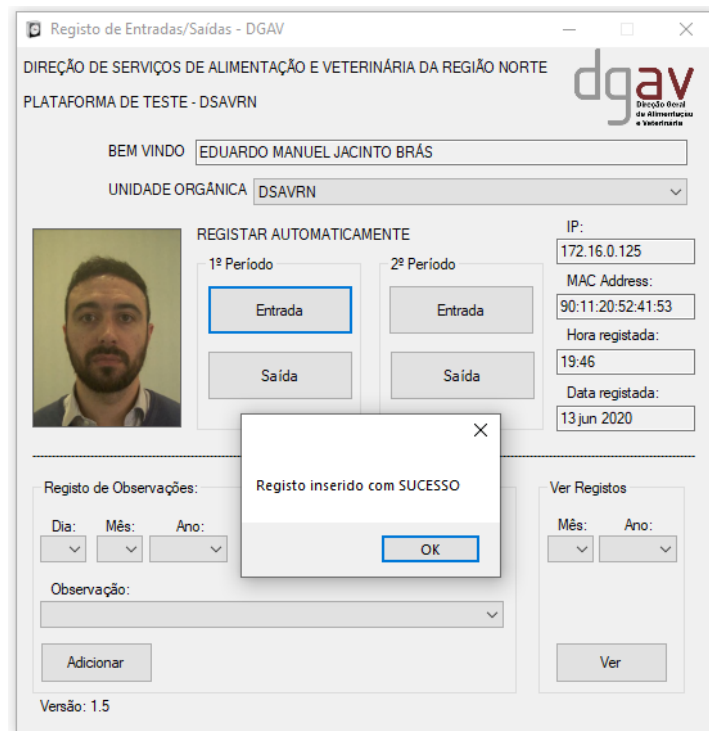


Figura 43. Informação dos dados gravados e mensagem de sucesso do registo

De seguida é dado a conhecer o extracto do código desenvolvido para a geração e gravação dos dados adquiridos numa folha de *Excel*.

```
//SE FICHEIRO DO TRABALHADOR AINDA NÃO EXISTE
if (!File.Exists(caminho))
{
//COPIA O TEMPLATE DO MÊS RESPECTIVO PARA A PASTA A SINCRONIZAR COM
A NUVEM
string caminho3 = text + @"\" + "Templates Mensais";
    if (sMonth=="01")
    {
        System.IO.File.Copy(caminho3 + @"\" + "Mod 75-A-DGAV
        Registo de Assiduidade Janeiro.xlsx", caminho);
    }
    if (sMonth == "02")
    {
        System.IO.File.Copy(caminho3 + @"\" + "Mod 75-A-DGAV
        Registo de Assiduidade Fevereiro.xlsx", caminho);
    }
    if (sMonth == "03")
    {
        System.IO.File.Copy(caminho3 + @"\" + "Mod 75-A-DGAV
        Registo de Assiduidade Março.xlsx", caminho);
    }
    if (sMonth == "04")
```

```

    {
        System.IO.File.Copy(caminho3 + @"\" + "Mod 75-A-DGAV
        Registo de Assiduidade Abril.xlsx", caminho);
    }
    if (sMonth == "05")
    {
        System.IO.File.Copy(caminho3 + @"\" + "Mod 75-A-DGAV
        Registo de Assiduidade Maio.xlsx", caminho);
    }
    if (sMonth == "06")
    {
        System.IO.File.Copy(caminho3 + @"\" + "Mod 75-A-DGAV
        Registo de Assiduidade Junho.xlsx", caminho);
    }
    if (sMonth == "07")
    {
        System.IO.File.Copy(caminho3 + @"\" + "Mod 75-A-DGAV
        Registo de Assiduidade Julho.xlsx", caminho);
    }
    if (sMonth == "08")
    {
        System.IO.File.Copy(caminho3 + @"\" + "Mod 75-A-DGAV
        Registo de Assiduidade Agosto.xlsx", caminho);
    }
    if (sMonth == "09")
    {
        System.IO.File.Copy(caminho3 + @"\" + "Mod 75-A-DGAV
        Registo de Assiduidade Setembro.xlsx", caminho);
    }
    if (sMonth == "10")
    {
        System.IO.File.Copy(caminho3 + @"\" + "Mod 75-A-DGAV
        Registo de Assiduidade Outubro.xlsx", caminho);
    }
    if (sMonth == "11")
    {
        System.IO.File.Copy(caminho3 + @"\" + "Mod 75-A-DGAV
        Registo de Assiduidade Novembro.xlsx", caminho);
    }
    if (sMonth == "12")
    {
        System.IO.File.Copy(caminho3 + @"\" + "Mod 75-A-DGAV
        Registo de Assiduidade Dezembro.xlsx", caminho);
    }
}
//FIM

//ABRE A INSTÂNCIA EXCEL, O FICHEIRO RESPECTIVO E CONFIGURA AS
PROPRIEDADES
Excel.Workbook MyBook = null;
Excel.Application MyApp = null;
Excel.Worksheet MySheet = null;
MyApp = new Excel.Application();
MyApp.Visible = false;
MyBook = MyApp.Workbooks.Open(caminho);
MySheet = (Excel.Worksheet)MyBook.Sheets[1];
//FIM

```

```
//SE TUDO VALIDADO GRAVA A HORA DA PICAGEM E AS RESTANTES
INFORMAÇÕES
```

```
MySheet.Cells[6, 3].Value = comboBox9.Text;
MySheet.Cells[8, 3].Value = space4;
MySheet.Cells[6, 9].Value = sMonth + "/" + sYear;
MySheet.Cells[13 + diasn2, 2].Value = today3;
MySheet.Cells[13 + diasn2, 2].AddComment(macIP);
MyBook.Password = "ceagp2016";
MyBook.Save();
MyBook.Close();
MyApp.Quit();
textBox9.Text = dia2;
textBox10.Text = today3;
textBox11.Text = IPfinal;
textBox12.Text = address;
Pteid.Exit(2);
MessageBox.Show("Registo inserido com SUCESSO");
//FIM
```

Na figura 44 pode ser visto um exemplo das informações gravadas numa célula do ficheiro em apreço quando é feito um registo de assiduidade. No anexo D é possível ver um registo mensal completo introduzido no Mod.75/DGAV adaptado ao SRACC.

REGISTO MENSAL DE ASSIDUIDADE

Unidade Orgânica: DSAVRN Mês/Ano: 06/2020

Trabalhador: EDUARDO MANUEL JACINTO BRÁS

IP: 127.0.0.1
 MACAddress: 90:11:20:52:41:53
 PLATAFORMA DE TESTE - DSAVRN
 Rede OK

DIAS	1º Período		2º Período		DIFERENCIAL	OBSERVAÇÕES
	ENTRADA	SAIDA	ENTRADA	SAIDA		
1	9:30	12:30			-4:00	
2					-7:00	

Figura 44. Extracto da assiduidade mensal de um utilizador com as informações registadas

Como se pode verificar pela figura acima, o horário registado aparece nas células de Entrada ou Saída e cada uma delas é complementada com um comentário onde se tem acesso a outras informações como o IP, *macaddress*, local de registo e, ainda, muito importante se a invocação do *webservice* de aquisição da data e hora foi feita com sucesso, aparecendo a informação de “Rede OK”. Caso contrário a informação que apareceria seria de “Rede NOK – Atenção: Data e Hora Local” para que se possam detetar possíveis abusos.

Este é o registo padrão, aquele que deve ser a regra e que naturalmente é isso que se verifica na prática.

5.4. REGISTO DE OBSERVAÇÕES/JUSTIFICAÇÕES

Se o ponto anterior, retrata o típico registo de assiduidade, feito na maioria dos casos, este ponto, embora absolutamente necessário, serve apenas para situações pontuais que exijam uma observação ou justificação para uma falha de registo de assiduidade por parte do trabalhador. No ponto 5.2.8. já foram abordadas algumas das justificações permitidas pelo sistema que podem ser inseridas pelo trabalhador.

Neste âmbito, e deixando um pouco de lado a especificidade de cada justificação/observação, o sistema divide-as em quatro grupos.

Grupo 1

O primeiro grupo são as justificações de ausência de registo por imprevistos naturais do dia-a-dia ou por força da missão da DGAV como, por exemplo, a realização de vistorias a operadores externos onde não existe a possibilidade de fazer o registo de assiduidade. Por imprevistos do dia-a-dia, leia-se falha do sistema, ou falha alheia ao sistema ou falha do utilizador. Neste último caso, considera-se por exemplo, esquecimento do CC ou esquecimento inadvertido de realização de registo. Nestes casos a aplicação permite selecionar as opções “Serviço externo” ou “Erro de Entrada/Saída” em que é dada a possibilidade ao utilizador selecionar o tipo de registo, o período de trabalho, a data e hora. Este tipo de justificação é considerado um “Registo de Assiduidade Manual”.

Na figura 45 é possível ver uma dessas opções selecionadas na janela principal do sistema.

Registro de Observações:

Dia: 02 ▼ Mês: 06 ▼ Ano: 2020 ▼ Horas: 09 ▼ Minutos: 30 ▼

Observação:
Erro Entrada - 1º Período ▼

Adicionar

Versão: 1.5

Figura 45. Adição de justificação/observação Erro de Entrada – 1º Período

Esta opção deve ser usada com responsabilidade e um exagero na adopção desta opção é imediatamente analisada pelos administradores e chefias diretas do trabalhador. Neste caso são feitas as mesmas validações que foram apresentadas até ao ponto 5.3 excetuando a validação número dois que é substituída por outra que verifica se a hora do registo atual é inferior ao registo anterior de forma a haver linearidade no tempo. Se assim não for, aparecerá uma mensagem ao utilizador de acordo com a figura 46.

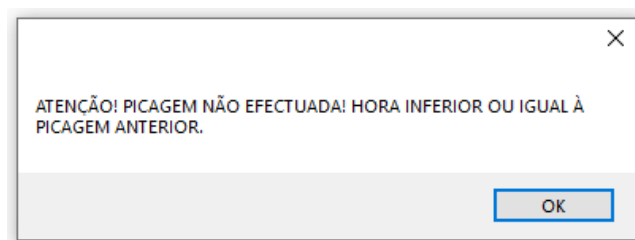


Figura 46. Mensagem de aviso de registo com hora inferior ao registo anterior

Se tudo for validado, o sistema regista a justificação apresentando-a no campo observações como pode ser demonstrado na figura 47.

REGISTO MENSAL DE ASSIDUIDADE

Unidade Orgânica

Mês/Ano

Trabalhador

DIAS	1º Período		2º Período		DIFERENCIAL	OBSERVAÇÕES
	ENTRADA	SAIDA	ENTRADA	SAIDA		
1	9:30	12:30	13:30	17:30	0:00	Erro Entrada/Saída
2	9:30	13:30	14:30	17:30	0:00	Serviço Externo
3					-7:00	
4					-7:00	
5					-7:00	
6						Sábado
7						Domingo
8					-7:00	
9					-7:00	
10						Feriado

Figura 47. Extrato da assiduidade mensal de um trabalhador com observações do primeiro grupo

Na figura 48, o DCU explica de forma sucinta e estruturada todas as relações e cenários possíveis desta funcionalidade, “Registo de Observações/Justificações”, mais concretamente, todos os casos que se inserem no grupo um, que permitem que os trabalhadores possam fazer um registo indicando a hora e data de forma manual.

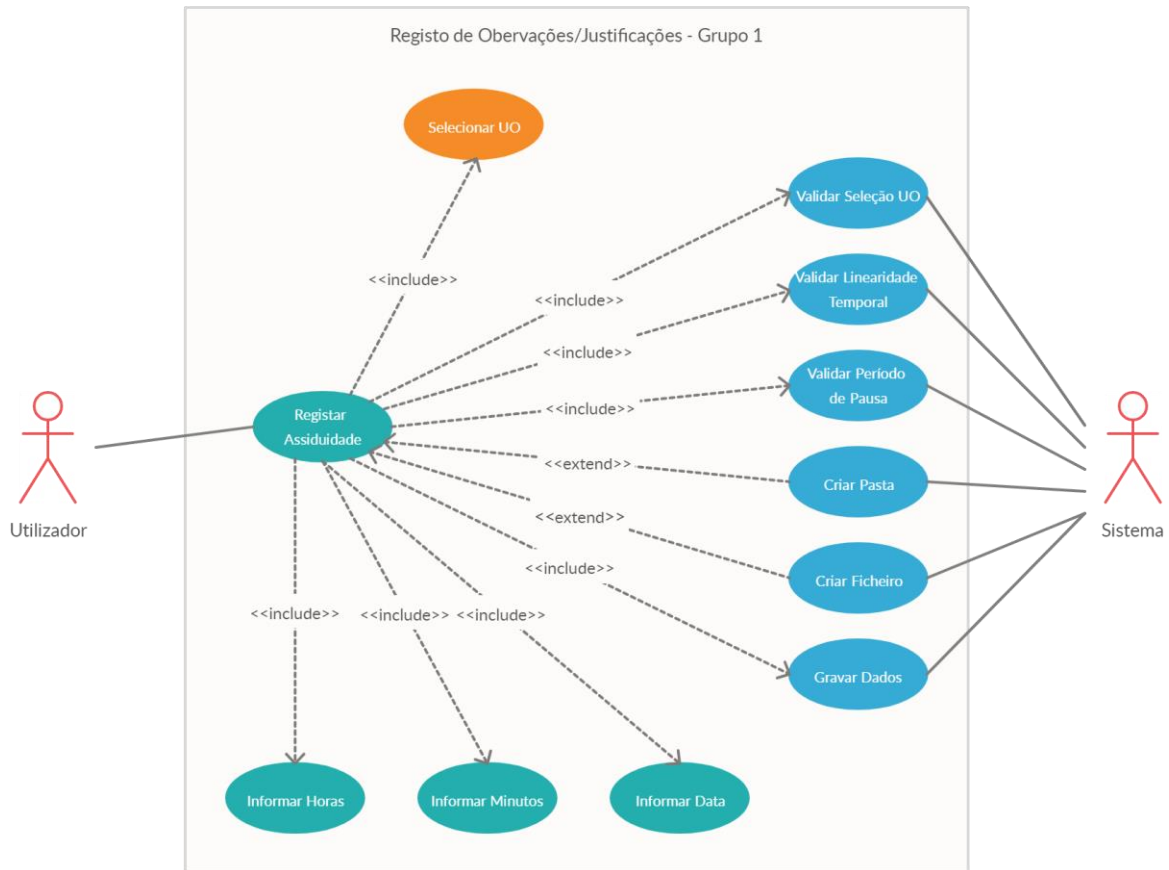


Figura 48. DCU para a funcionalidade “Registo de Observações/Justificações” - grupo 1

De seguida pode ser visto o código desenvolvido para um típico registo de observação enquadrada neste grupo:

```

//ABRIR APP EXCEL (APLICÁVEL A TODOS OS GRUPOS
Excel.Workbook MyBook = null;
Excel.Application MyApp = null;
Excel.Worksheet MySheet = null;
MyApp = new Excel.Application();
MyApp.Visible = false;
MySheet = (Excel.Worksheet)MyBook.Sheets[1];
string diaobs = comboBox2.Text;
string mesobs = comboBox3.Text;
string anoobs = comboBox4.Text;
var horasobs = comboBox5.Text;
var minutosobs = comboBox6.Text;
var horaseminutos = horasobs + ":" + minutosobs;
int diasn = 0;
//FIM
    
```

```

//REGISTO DE ERRO DE ENTRADA PRIMEIRO PERÍODO
if (comboBox1.Text == "Erro Entrada - 1º Período")
{
    if (MySheet.Cells[13 + diasn, 2].Comment != null)
    {
        MySheet.Cells[13 + diasn, 2].Comment.Delete();
    }
    MySheet.Cells[13 + diasn, 2].Value = horaseminutos;
    MySheet.Cells[13 + diasn, 2].AddComment("Erro Entrada - 1º
    Período" + ";" + dia2 + ";" + today3 + ";" + matadouro2);
    var conter = MySheet.Cells[13 + diasn, 7].Value;
    if (conter == null)
    {
        Console.WriteLine("Não contém nada");
        MySheet.Cells[13 + diasn, 7].Value = "Erro
        Entrada/Saída";
    }
else
{
    if (conter.Contains("Erro"))
    {
        Console.WriteLine("Contém erro");
    }
else
{
        MySheet.Cells[13 + diasn, 7].Value = "Erro
        Entrada/Saída" + " + " + conter;
    }
}
}
//FIM

```

Grupo 2

O segundo grupo, são as observações que justificam um dia de trabalho completo, como por exemplo Férias, Feriado Municipal ou Greve. Neste caso, basta seleccionar na caixa de escolha uma destas opções, por exemplo, e indicar a data dessa observação, vide figura 49.

Registo de Observações:

Dia: 01 ▼ Mês: 06 ▼ Ano: 2020 ▼

Observação: Férias ▼

Adicionar

Figura 49. Adição de observação "Férias"

No caso deste grupo, são usadas as validações até ao ponto 5.3, não inclusive. A única validação que é feita neste caso concreto é a verificação se, para aquela data, já existe outra observação. Uma vez que se trata de uma observação de um dia inteiro não faz sentido para o mesmo dia ter mais do que uma, seja ela de que grupo for. A não validação desta regra despoleta uma mensagem de aviso confirmada pela figura 50.

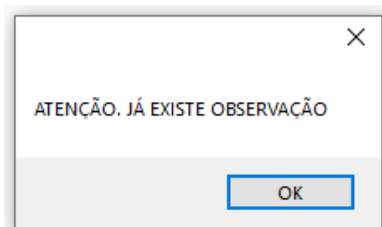


Figura 50. Mensagem de aviso de observação já existente

No caso das validações estarem todas corretas, a observação é adicionada ao registo e o diferencial de horas de trabalho para aquele dia é colocado a zero como ilustra a figura 51.



REGISTO MENSAL DE ASSIDUIDADE

Unidade Orgânica Mês/Ano

Trabalhador

DIAS	1º Período		2º Período		DIFERENCIAL	OBSERVAÇÕES
	ENTRADA	SAIDA	ENTRADA	SAIDA		
1					0:00	Férias

Figura 51. Extrato da assiduidade mensal de um trabalhador com observação do segundo grupo

O código desenvolvido para o registo de uma justificação deste grupo é a demonstrada imediatamente abaixo:

```
//REGISTO DA OBSERVAÇÃO FÉRIAS
if (comboBox1.Text == "Férias")
{
    var calculo = diasn + 13;
```

```

var calculo1 = "(C" + calculo + "-B" + calculo + ")" + "+" +
"(E" + calculo + "-D" + calculo + ")" + "-" +
"TIMEVALUE(\"00:00\")";
MySheet.Cells[13 + diasn, 6].Formula = calculo1;
var conteudo = MySheet.Cells[13 + diasn, 7].Value;
MySheet.Cells[13 + diasn, 7].Value = "Férias";
var conteudo2 = MySheet.Cells[1, 3].Value;
}
//FIM

```

Grupo 3

As observações que justifiquem apenas metade do dia de trabalho (03:30H) são o terceiro grupo. Tal como as justificações/observações do segundo grupo as validações são as mesmas até ao ponto 5.3, não inclusive. Neste terceiro grupo são adicionadas da mesma maneira que as do segundo grupo e existe também uma validação específica. Apenas permite registo de observações ou justificações que na sua soma não ultrapassem um dia de trabalho completo, que, se for o caso, avisa o utilizador através de uma mensagem como se pode confirmar na figura 52.

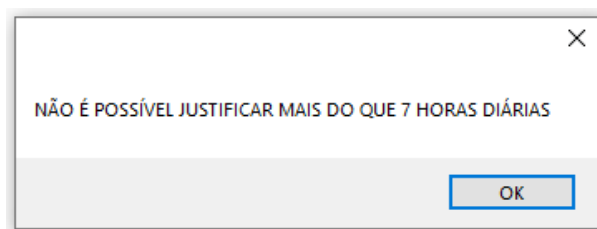


Figura 52. Mensagem de aviso de tentativa de justificação de mais de sete horas

Cada justificação/observação de meio-dia fica registado no campo observações do Mod. 75/DGAV no dia pretendido e são subtraídas três horas e trinta minutos até um máximo de sete horas. A figura 53, mostra duas justificações de meio-dia e o respetivo diferencial colocado a zero, uma vez que as duas justificações somam as sete horas de trabalho obrigatórias.

REGISTO MENSAL DE ASSIDUIDADE

Unidade Orgânica Mês/Ano
 Trabalhador

DIAS	1º Período		2º Período		DIFERENCIAL	OBSERVAÇÕES
	ENTRADA	SAIDA	ENTRADA	SAIDA		
1					0:00	Formação - Meio dia + Consulta médica/exames - Próprio - Meio dia

Figura 53. Extrato da assiduidade mensal de um trabalhador com observação do terceiro grupo

O código desenvolvido foi o seguinte:

```
//REGISTO DA OBSERVAÇÃO FORMAÇÃO - MEIO DIA
if (comboBox1.Text == "Formação - Meio dia")
{
  var calculo = diasn + 13;
  string calculo1 = "=(C" + calculo + "-B" + calculo + ")" + "+"
  + "(E" + calculo + "-D" + calculo + ")" + "-" +
  "TIMEVALUE(\"03:30\")";
  string calculo3 = "=(C" + calculo + "-B" + calculo + ")" + "+"
  + "(E" + calculo + "-D" + calculo + ")" + "-" +
  "TIMEVALUE(\"00:00\")";
  string meio = MySheet.Cells[13 + diasn, 6].Formula;

  if (meio == calculo1 || meio == calculo3)
  {
    var calculo2 = "=(C" + calculo + "-B" + calculo + ")" +
    "+" + "(E" + calculo + "-D" + calculo + ")" + "-" +
    "TIMEVALUE(\"00:00\")";
    MySheet.Cells[13 + diasn, 6].Formula = calculo2;
  }
  else
  {
    var ausencial = "03:30:00";
    var causencial = TimeSpan.Parse(ausencial);
    var calculocelula = diasn + 13;
    string formcelula = MySheet.Cells[13 + diasn, 6].Formula;
    string formcelula1 = formcelula.Substring(34, 5);
    var ausencia3 = formcelula1 + ":" + "00";
    var causencia3 = TimeSpan.Parse(ausencia3);
    var diferencialausencial = (causencia3) - (causencial);
    string calculofinal = "=(C" + calculocelula + "-B" +
    calculocelula + ")" + "+" + "(E" + calculocelula + "-D" +
    calculocelula + ")" + "-" + "TIMEVALUE(\"" +
    diferencialausencial + "\")";
    MySheet.Cells[13 + diasn, 6].Formula = calculofinal;
  }
}
```

```

var testetempo = MySheet.Cells[13 + diasn, 6].Value;
string menos = calculofinal.Substring(34, 1);
    if (menos == "-")
    {
        MySheet.Cells[13 + diasn, 6].Formula =
            formcelula;
        MessageBox.Show("NÃO É POSSÍVEL JUSTIFICAR
            MAIS DO QUE 7 HORAS DIÁRIAS");
        MyBook.Save();
        MyBook.Close();
        MyApp.Quit();
        return;
    }
}
var conteudo = MySheet.Cells[13 + diasn, 7].Value;
if (conteudo == null)
{
    Console.WriteLine("Não contém nada");
    MySheet.Cells[13 + diasn, 7].Value = "Formação - Meio
        dia";
}
else
{
    if (conteudo.Contains("Formação"))
    {
        Console.WriteLine("Contém Formação");
    }
    else
    {
        MySheet.Cells[13 + diasn, 7].Value = "Formação -
            Meio dia" + " + " + conteudo;
    }
}
}
//FIM

```

Grupo 4

Por último, o quarto grupo são justificações/observações que apenas justificam um período de tempo, inferior a meio-dia de trabalho. Em termos de validações são exatamente as mesmas usadas para as observações do terceiro grupo. Já o processo de adição ao registo de assiduidade tem uma particularidade diferente dos outros grupos, a saber, uma nova janela de introdução do período de ausência a justificar. É possível ver o aspeto desta janela na figura 54.

Figura 54. Janela de introdução do período de ausência

Nesta janela o utilizador coloca o período de ausência a justificar e se este período for impeditivo de um registo de Entrada ou Saída assinala também a caixa “Descontar Tempo” o que vai fazer com que aquele tempo seja descontado no diferencial de horas trabalhadas naquele dia como demonstra a figura 55.



REGISTO MENSAL DE ASSIDUIDADE

Unidade Orgânica Mês/Ano

Trabalhador EDUARDO MANUEL JACINTO BRÁS

DIAS	1º Período		2º Período		DIFERENCIAL	OBSERVAÇÕES
	ENTRADA	SAIDA	ENTRADA	SAIDA		
1					-5:00	Consulta médica/exames - Familiar das 09:00 às 11:00

Figura 55. Extrato da assiduidade mensal de um trabalhador com observação do quarto grupo

Quanto ao DCU, é possível, na figura 56, ver resumidas todas as interações possíveis entre utilizador e sistema assim como todos os cenários possíveis da funcionalidade “Registo de Observações/Justificações”, mas, neste caso, para todas as observações/justificações no âmbito dos grupos dois, três e quatro.

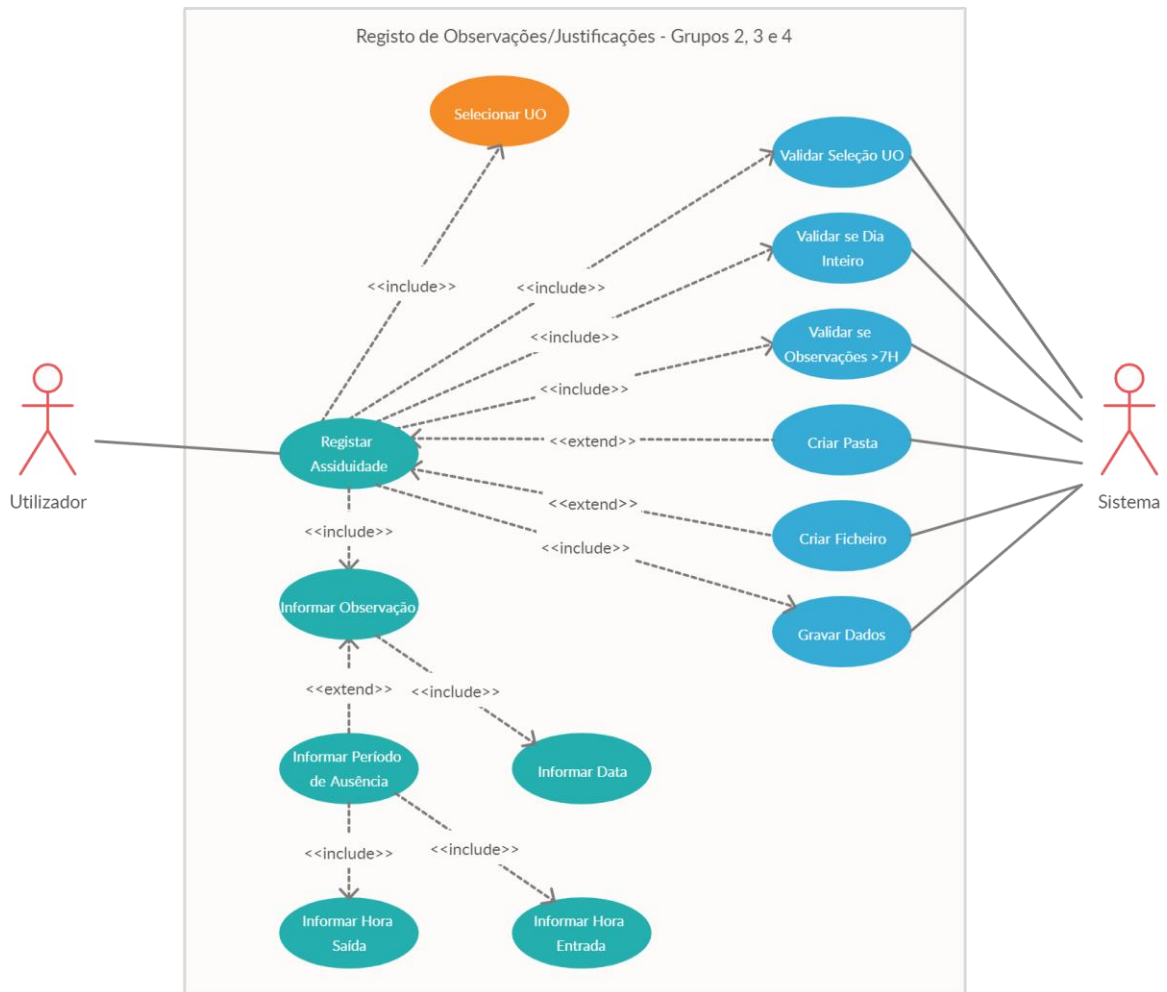


Figura 56. DCU para a funcionalidade “Registo de Observações/Justificações” – grupos 2,3 e 4

No caso deste tipo de observações o código desenvolvido foi o demonstrado abaixo:

```

//REGISTO DE UM PERÍODO DE AUSÊNCIA PARA CONSULTA MÉDICA PRÓPRIO
if (comboBox1.Text == "Consulta médica/exames - Próprio - Período de Ausência")
{
    frm2 = new Form2();
    frm2.ShowDialog();
    if (frm2.label3.Text == "0")
    {
        MyBook.Save();
        MyBook.Close();
        MyApp.Quit();
        return;
    }
    if (frm2.checkBox1.Checked == true)

```

```

{
    var ausencia1 = frm2.comboBox5.Text + ":" +
    frm2.comboBox6.Text + ":" + "00";
    var causencial1 = TimeSpan.Parse(ausencia1);
    var ausencia2 = frm2.comboBox2.Text + ":" +
    frm2.comboBox1.Text + ":" + "00";
    var causencia2 = TimeSpan.Parse(ausencia2);
    var diferencialausencia = (causencia2) - (causencial1);
    var calculocelula = diasn + 13;
    string formcelula = MySheet.Cells[13 + diasn, 6].Formula;
    string formcelula1 = formcelula.Substring(34, 5);
    var ausencia3 = formcelula1 + ":" + "00";
    var causencia3 = TimeSpan.Parse(ausencia3);
    var diferencialausencial = (causencia3) -
    (diferencialausencia);
    string calculofinal = "=" + "(" + calculocelula + "-B" +
    calculocelula + ")" + "+" + "(E" + calculocelula + "-D" +
    calculocelula + ")" + "-" + "TIMEVALUE(\"" +
    diferencialausencial + "\")";
    MySheet.Cells[13 + diasn, 6].Formula = calculofinal;
    var testetempo = MySheet.Cells[13 + diasn, 6].Value;
    string menos = calculofinal.Substring(34, 1);
    if (menos == "-")
    {
        MySheet.Cells[13 + diasn, 6].Formula = formcelula;
        MessageBox.Show("NÃO É POSSÍVEL JUSTIFICAR MAIS DO
        QUE 7 HORAS DIÁRIAS");
        MyBook.Save();
        MyBook.Close();
        MyApp.Quit();
        return;
    }
}
var saidaservico = "das" + " " + frm2.comboBox5.Text + ":" +
frm2.comboBox6.Text + " " + "às" + " " + frm2.comboBox2.Text +
":" + frm2.comboBox1.Text;
var conteudo = MySheet.Cells[13 + diasn, 7].Value;
var conter = MySheet.Cells[13 + diasn, 7].Value;
if (conter == null)
{
    Console.WriteLine("Não contém nada");
    MySheet.Cells[13 + diasn, 7].Value = "Consulta
    médica/exames - Próprio" + " " + saidaservico;
}
else
{
    MySheet.Cells[13 + diasn, 7].Value = "Consulta
    médica/exames - Próprio" + " " + saidaservico + " + " +
    conter;
}
}
//FIM

```

Existem muitas observações que podem ser utilizadas mas, todas elas a nível de lógica e codificação seguem os mesmos princípios dos quatro grupos acima apresentados. Assim, de forma a tornar esta dissertação mais prática e de mais fácil leitura e compreensão não serão aqui abordadas todas as hipóteses.

5.5. VISUALIZAÇÃO DE ASSIDUIDADE

Um sistema de assiduidade deve ter como funcionalidade a possibilidade de permitir que o utilizador consiga a qualquer momento verificar e analisar os seus registos até então. Deste modo, o utilizador seleciona o Mês e o Ano que pretende nas caixas de seleção presentes na área “Ver Registos” na janela principal do sistema, vide figura 57.

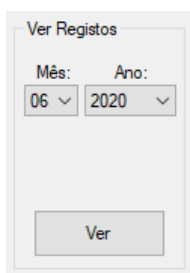


Figura 57. Área reservada à visualização de registos.

A única validação que é feita, além da verificação de presença do CC no leitor e da sua validade, é a análise da existência ou não de algum registo para a data indicada. No caso, se não existir o sistema avisa o utilizador através da mensagem que aparece na figura 58.

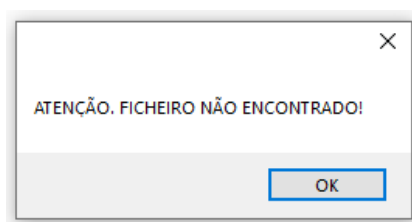


Figura 58. Mensagem de aviso de ficheiro não encontrado

Existindo, o sistema devolve um ficheiro *.PDF temporário que desaparece quando o utilizador fecha a respetiva janela. Se este quiser pode gravar o PDF para seu arquivo. Convém esclarecer que este PDF não tem o nível de detalhe que tem o ficheiro *Excel*, e nem fazia sentido o utilizador saber os IP ou *macaddress* de cada registo.

Para a funcionalidade “Visualização de Registos”, foi também elaborado um DCU, como se pode comprovar na figura 59.

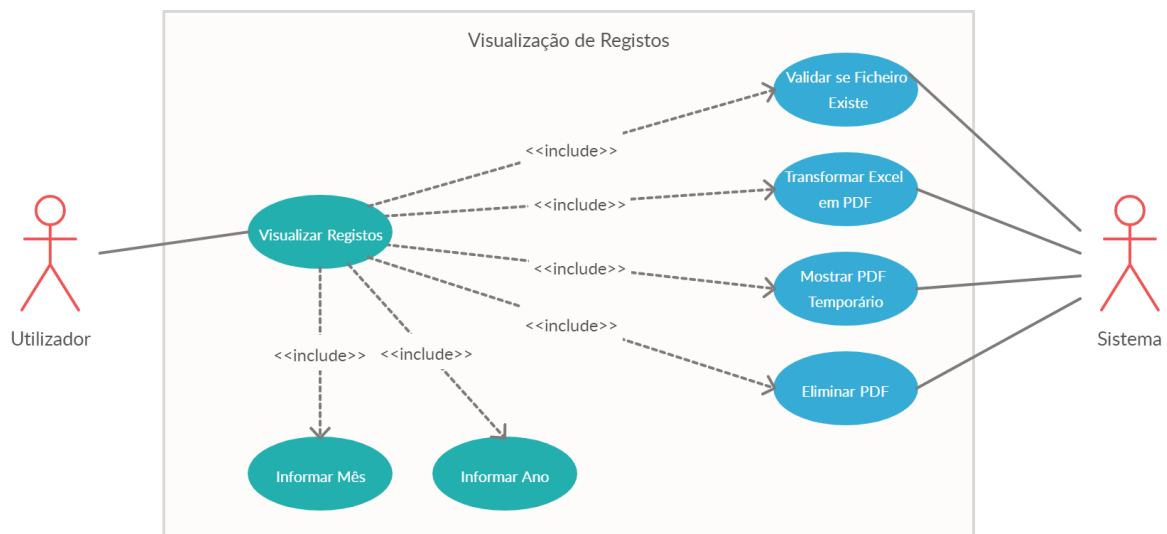


Figura 59. DCU para a funcionalidade “Visualização de Registos”

Para a transformação do ficheiro *Excel* em ficheiro PDF foi desenvolvido o seguinte código:

```

//EXPORTAÇÃO DE EXCEL PARA PDF
Excel.Workbook MyBook = null;
Excel.Application MyApp = null;
MyApp = new Excel.Application();
MyApp.Visible = false;
MyBook.ExportAsFixedFormat(Excel.XlFixedFormatType.xlTypePDF,
Filename: @"C:\DSAVRN\" + space4 + ".PDF", OpenAfterPublish: true);
MyBook.Save();
MyBook.Close();
MyApp.Quit();
comboBox7.SelectedIndex = -1;
comboBox8.SelectedIndex = -1;
Pteid.Exit(2);
//FIM
  
```

```
//ELIMINAÇÃO DO FICHEIRO TEMPORÁRIO
var nomedapessoa = textBox1.Text;
foreach (var process in Process.GetProcesses())
{
    if (process.MainWindowTitle.Contains(nomedapessoa + ".PDF"))
    {
        process.Kill();
        Thread.Sleep(1000);
        break;
    }
}
try
{
    File.Delete(@"C:\DSAVRN\" + nomedapessoa + ".PDF");
}
catch
{
    Console.WriteLine("NÃO FOI POSSÍVEL ELIMINAR");
}
//FIM
```


6. TESTES DO SISTEMA E ANÁLISE DE RESULTADOS

6.1. TESTES DO SISTEMA

Após o término do desenvolvimento do SRACC foram novamente reunidos os Stakeholders e dado a conhecer o novo processo e o seu modo de funcionamento, de modo a obter a aprovação das chefias para o início da sua implementação. A reunião correu favoravelmente e o novo sistema informático de registo de assiduidade obteve a aprovação para iniciar a etapa de testes.

Nesta etapa, como locais de teste foram definidos apenas duas Unidades de Abate, Matadouro Linda Rosa e Matadouro Landeiro e duas unidades orgânicas, DAV de Braga e

DSAVRN. O número de trabalhadores eram quatro, dois, dez e sete respetivamente. Foram escolhidos estes locais, por não terem muitos trabalhadores comparativamente a outros e pela proximidade geográfica com as instalações da DSAVRN, onde o autor deste trabalho exerce funções.

Após a instalação do SRACC nestes locais, foi dada formação e um exemplar do “Manual de Utilização – TimeSet” (vide anexo E) a estes trabalhadores sobre o seu funcionamento. Foi-lhes informado que deveriam começar a utilizar este sistema e, em paralelo, fazer o registo de assiduidade que estava instituído até então de forma a acautelar a assiduidade de cada trabalhador para a eventualidade dos testes serem mal sucedidos. Foi-lhes também pedido para que comunicassem ao responsável pelo desenvolvimento do sistema todos os erros que surgissem e opiniões/sugestões de melhoria.

Embora a opinião do autor desta tese fosse de que o tempo de testes não deveria ser inferior a dois meses, a vontade por parte das chefias era a introdução o mais rápida possível do novo sistema informático de assiduidade, pelo que o período de testes foi reduzido para um mês.

Durante esse mês, não foi reportado qualquer falha ou erro no funcionamento do sistema. Apenas foram apontadas algumas falhas no que diz respeito à nomenclatura de algumas observações que o sistema dá hipótese de escolher, não tendo sido dada nenhuma sugestão de melhoria por parte desses trabalhadores durante essa fase de testes.

Terminado o período de testes sem sobressaltos e com as falhas identificadas corrigidas deu-se então início à instalação em massa do novo sistema nos restantes locais. Numa primeira fase nas unidades de abate e numa segunda fase as unidades orgânicas das DSAVR.

6.2. ANÁLISE DE RESULTADOS

Para a análise de resultados foram tidos em conta indicadores de desempenho que demonstrassem o sucesso, ou não, da implementação deste sistema de assiduidade. Para isso, foi pedido a colaboração dos RH para que contribuíssem na conceção destes indicadores, de forma a serem esclarecedores e concretos.

É evidente que, a implementação trouxe outras mais-valias que não são quantificáveis como, por exemplo, a melhoria das condições de trabalho ou mesmo o impacto ambiental pelo simples facto de todo o processo de assiduidade ter sido desmaterializado.

No entanto, tendo em conta o esclarecimento cabal do sucesso ou não da implementação deste sistema, foram criados quatro indicadores que serão abordados nos pontos abaixo. Ressalva-se que, para efeitos de pressupostos que o tempo é sempre medido em minutos e que X="Entre outubro e dezembro de 2019" e Y="Entre março e maio de 2020".

6.2.1. Redução de tempo na elaboração dos mapas de assiduidade mensal

Como referido no capítulo de "Caracterização do problema", um dos principais problemas identificados era o tempo excessivo gasto na elaboração dos mapas de assiduidade mensal, ou seja, todo o tempo gasto entre o momento em que o trabalhador elaborava e enviava os seus registos mensais de assiduidade até ao momento de envio para a Sede, através do Mod.62/DGAV, para efeitos de processamento salarial.

Assim, foi elaborado o indicador que se segue e recolhidos os dados que permitem perceber a melhoria que a implementação deste novo sistema de assiduidade trouxe neste âmbito.

$(\text{Tempo médio de elaboração dos Mapas de Assiduidade em Y} / \text{Tempo médio de elaboração dos Mapas de Assiduidade no ano X}) * 100$

Antes da implementação do SRACC, o tempo médio de cada etapa era:

- a) Preenchimento e envio do Mod.75/DGAV pelos trabalhadores = 4200 minutos ou 70 horas;
- b) Apuramento dos tempos de trabalho de todos os trabalhadores = 2100 minutos ou 35 horas;
- c) Validação e lançamento das observações com impacto salarial no Mod.62/DGAV = 840 minutos ou 14 horas.

Após a implementação do SRACC, as etapas identificadas a) e b) deixaram de existir, uma vez, que a assiduidade individual de cada trabalhador está imediatamente acessível aos RH através da nuvem.

- c) Validação e lançamento das observações com impacto salarial no Mod.62/DGAV = 780 minutos ou 13 horas.

Voltando a lembrar o indicador:

(Tempo médio de elaboração dos Mapas de Assiduidade em Y / Tempo médio de elaboração dos Mapas de Assiduidade no ano X)*100

$$= (780/7140)*100 = 10.92\%$$

Este indicador permite concluir que, com a implementação deste novo sistema de assiduidade, foi possível alcançar uma redução de 89.08% na elaboração dos mapas de assiduidade mensal. Transformando estes números em horas, passou-se de um tempo gasto de aproximadamente cento e dezanove horas para aproximadamente treze horas.

Este indicador é revelador da melhoria de eficiência no trabalho trazido pelo novo sistema.

6.2.2. Redução do n.º de erros de processamento salarial

Antes deste novo sistema de assiduidade ser implementado havia algumas queixas sobre erros de processamento salarial reportados à equipa dos RH. Era e continua a ser um tema sensível e compreensivelmente um dos mais importantes aos olhos tanto do trabalhador, como da própria entidade patronal. Não é um assunto agradável e, por vezes, gera grandes dissabores. Naturalmente, analisando esses casos, a grande maioria das vezes tratavam-se de erros de transcrição devido à passagem dos registos manuscritos dos trabalhadores para formato digital.

Com o propósito de avaliar o desempenho do novo sistema de assiduidade, concebeu-se o indicador abaixo demonstrado e recolheram-se os dados para a sua aferição.

$(N.º \text{ de erros de processamento salarial em Y} / N.º \text{ de erros de processamento salarial em X}) * 100$

$$= (1/10) * 100 = 10\%$$

Para este indicador, os dados recolhidos revelam que em Y foi identificado um erro reportado à equipa dos RH, enquanto em X houve dez reclamações sobre erros de processamento salarial. Ora, este cenário demonstra, para este período de tempo em causa, uma redução de 90% no número de casos de erros no processamento de salários o que prova que o facto de este novo sistema de assiduidade ser totalmente digital, permitiu mitigar, em grande parte, os erros de transcrição.

6.2.3. Redução do n.º de trabalhadores com tempo efetivo de trabalho mensal inferior ao tempo legalmente estabelecido

Devido à falta de rigor no modelo anterior de registo da assiduidade, havia, todos os meses, trabalhadores que não cumpriam o número de horas de trabalho legalmente exigidas. É evidente que a falta de rigor desse sistema, era sabido de todos e infelizmente isso gerava um aproveitamento por parte de alguns trabalhadores, resultando em menos horas trabalhadas. Nesse sentido, de forma a analisar e avaliar a frequência deste tipo de irregularidades foi idealizado um indicador, se seguida revelado, que com a ajuda dos dados recolhidos foi possível chegar a uma conclusão.

$(N.º \text{ de trabalhadores com tempo efetivo mensal de trabalho inferior ao tempo legalmente estabelecido em Y} / N.º \text{ de trabalhadores com tempo efetivo mensal de trabalho inferior ao tempo legalmente estabelecido em X}) * 100.$

$$= (1/7) * 100 = 14.29\%$$

A partir do momento em que o novo sistema foi implementado, o rigor com que são tratados os registos de assiduidade foi imediatamente sentido e desde logo desencorajou possíveis abusos. É notória a redução do número de trabalhadores com horas mensais efetivas inferiores ao exigido por lei. Traduzindo em números, neste momento com os dados disponíveis, verifica-se uma redução de 85.71%.

7. CONCLUSÃO

Terminado este projecto, é com grande satisfação que se lhe reconhece uma ajuda importantíssima na melhoria na qualidade e do ambiente de trabalho dentro da DGAV. Foi extremamente benéfica a implementação sistema informático de assiduidade, tanto para quem regista como para quem controla.

Sendo um âmbito tão sensível, assiduidade, era esperada uma grande resistência à mudança uma vez que o anterior modelo foi sempre aquele que vigorou. Era um sistema pouco rigoroso, suscetível a muitas falhas e erros, que ao mesmo tempo permitia, por isso mesmo, alguma flexibilidade para prevaricar. Ao introduzir algo novo que maximizasse o controlo seria realmente de esperar bastante resistência.

Na verdade, não foi o que aconteceu. Houve sempre muita curiosidade sobre o que viria e os trabalhadores sempre se mostraram bastante recetivos nesta mudança. Os grandes motivos para isto ter acontecido foram em primeiro lugar a tentativa de inclusão de todos nas várias etapas do projeto dando sempre a conhecer o estado da implementação e ouvindo as várias opiniões sobre funcionalidades a incluir no sistema de assiduidade. Em segundo lugar, porque a grande maioria dos trabalhadores, que fazem vezes sem conta horas a mais do que aquelas que são obrigatórias por Lei tinham agora um sistema credível que contabilizava ao minuto as horas trabalhadas e que este facto não mais poderia ser negado pelos seus superiores.

Relativamente ao desenvolvimento do sistema, foi idealizado de forma bastante rápida e as funcionalidades a incluir foram facilmente identificadas e aprovadas. Já no que à elaboração do código diz respeito, registaram-se alguns altos e baixos.

Contrariamente ao esperado, o suporte técnico que a AMA I.P. fornece a quem quer aproveitar as potencialidades do CC é quase inexistente. Existem manuais que ajudam, mas muitas vezes não estão atualizados com as bibliotecas que disponibilizam, o que faz com que quem desenvolve determinada aplicação tenha de fazer por tentativa e erro. No entanto, depois de algumas experiências foi possível tirar total partido dessas potencialidades.

O ponto mais negativo foi sem dúvida a partilha de tempo entre as tarefas quotidianas e o desenvolvimento do código. Muitas paragens, perda de raciocínio, que levaram a que o desenvolvimento não tenha sido sempre contínuo dando origem a erros de programação e horas a mais passadas no processo de *debugging*. Também por este motivo, há algumas rotinas que se repetem, causando redundâncias e que poderiam ser escritas uma vez e depois chamadas sempre que necessário. Poupava-se em tempo gasto de codificação e tornava o código um pouco menos extenso.

Não obstante este ponto mais negativo, a experiência de poder colocar toda a experiência adquirida no âmbito académico e profissional, num projeto começado do zero, vivenciar todas as suas etapas culminando na sua implementação e utilização a cem por cento foi sem dúvida excecional e o grande ponto positivo.

Neste momento, é já um sistema que faz parte do dia-a-dia dos trabalhadores e chefias e de certa maneira é perceptível que já ninguém se imagina sem um sistema de assiduidade desta natureza, quanto mais se houvesse regressão para o antigo modelo de registo de assiduidade.

Salienta-se também a mais-valia do sistema apresentado em termos de redução de tempo na elaboração dos mapas de assiduidade mensal, redução do número de erros de processamento salarial e redução do número de trabalhadores com tempo efetivo de trabalho mensal inferior ao tempo legalmente estabelecido.

Uma grande vantagem a assinalar prende-se com a possibilidade do sistema ser completamente parametrizável para outra realidade. Ou seja, é bastante fácil a reprodução deste mesmo modelo de registo de assiduidade para qualquer outra entidade ou serviço da AP, sem que sejam necessárias grandes adaptações ou esforço financeiro.

Como melhorias futuras, sem dúvida que a criação de um *Backoffice* para complementar o SRACC seria a grande mais-valia. Ou seja, criar uma aplicação para acesso apenas pelos Administradores, RH e Chefias que conseguisse agregar e mostrar num GUI os ficheiros de todos os trabalhadores, apenas de alguns, de acordo com determinado filtro aplicado, ou fazer uma pesquisa pelo nome de um trabalhador em particular. A funcionalidade de filtro seria de uma importância enorme, basta imaginar-se a possibilidade, por exemplo, do utilizador apenas querer ver o nome dos trabalhadores que no fim do mês tivessem horas inferiores de trabalho relativamente ao legalmente exigido. Eliminava ao processo ter de se abrir os ficheiros um a um para verificar esse dado. O mesmo se aplica ao inverso, pesquisar apenas pelos trabalhadores com horas efetivas de trabalho superiores ao exigido e dessa forma saber imediatamente quem teria direito a remuneração de horas extra. Para além destes exemplos, há um grande número de outros filtros que podiam ser aplicados, como visualizar apenas trabalhadores de atestado, com registos em falta, de licença de paternidade etc.

Poderia também ser implementado um *Dashboard* que mostrasse em tempo real quais os trabalhadores em serviço e em que locais, dando a possibilidade de poder agregar por local e também quais os trabalhadores não ao serviço e qual o motivo também com a possibilidade de agregar por motivo.

Por último, uma melhoria futura, que é transversal a todos os projectos de desenvolvimento de aplicações informáticas, é a revisão do código, na tentativa de o otimizar o mais possível de forma a tornar a aplicação mais leve e por conseguinte aumentar o seu desempenho e estabilidade.

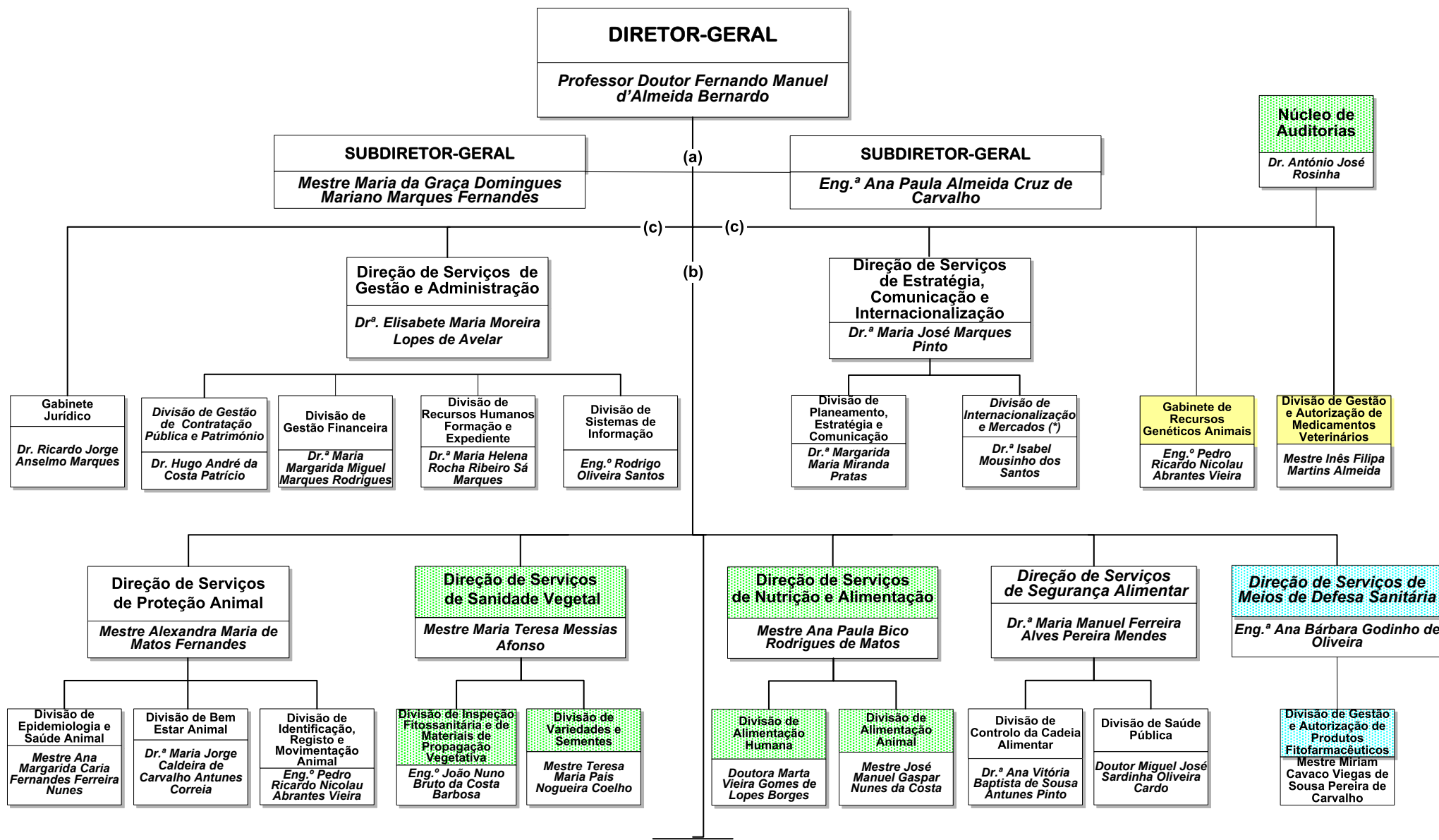
8. REFERÊNCIAS DOCUMENTAIS

- [1] <https://www.infopedia.pt/dicionarios/lingua-portuguesa/assiduidade>, acedido em 23/12/2019.
- [2] <https://www.cnpd.pt/bin/orientacoes/principiosbiometricos.htm>, acedido em 24/12/2019.
- [3] Decreto Lei n.º 7/2012 de 17 de janeiro, pp.218 a 219.
- [4] Decreto Regulamentar n.º 31/2012 de 13 de março, pp.1128 a 1129.
- [5] Portaria n.º 282/2012 de 17 de setembro, p.5236 a 5239.
- [6] <https://www.ibm.com/ibm/history/ibm100/us/en/icons/magnetic/>, acedido em 28/01/2020.
- [7] <https://www.thecardnetwork.co.uk/mifare-desfire-ev2-4k-cards-hi-co.html>, acedido em 28/01/2020.

- [8] <https://www.q-card.com/about-us/iso-magnetic-stripe-card-standards/page.aspx?id=1457>, acessado em 28/01/2020.
- [9] <https://www.amazon.com/OSAYDE-Magnetic-Encoder-Scanner-Software/dp/B019IKBYMA>, acessado em 28/01/2020.
- [10] Security Science – The Theory and Practice of Security, 2013, p. 153-175.
- [11] https://www.researchgate.net/figure/RFID-setup-in-the-library-Source-wwwrfid-librarycom_fig7_236147932, acessado em 28/01/2020.
- [12] https://www.researchgate.net/figure/Type-of-an-RFID-tags_fig3_332319065, acessado em 28/01/2020.
- [13] https://www.researchgate.net/figure/There-are-many-kinds-of-RFID-readers-The-image-shows-a-small-collection-taken-from-the_fig5_289345018, acessado em 28/01/2020.
- [14] "Fundamental Operating Principles," in Fundamentals and Applications in Contactless Smart Cards and Identification, 2nd edition: Wiley & Sons LTD, 2003, pp. 29-59.
- [15] S. Preradovic and N. C. Karmakar, "RFID Transponders - A Review," in Electrical and Computer Engineering, 2006. ICECE '06. International Conference on, 2006, pp. 96-99.
- [16] G. Whyte, "Evaluating the Effect of EMI in Transceivers (Readers) Used to Identify Animals Electronically," 2008.
- [17] <https://www.ncontrol.com.pt/o-que-e-rfid.html>, acessado em 28/01/2020.
- [18] https://www.gta.ufrj.br/ensino/eel879/trabalhos_vf_2015_2/Seguranca/conteudo/Smartcard-e-Javacard/Historico.html, acessado em 04/02/2020.
- [19] https://conferencia.morenet.ac.mz/2018/wp-content/uploads/INCMoalorda%20seguran%C3%A7a_VitorCastro.pdf, acessado em 04/02/2020.
- [20] <https://www.cardlogix.com/product/atmel-at24c256c-memory-smart-card-512k/>, acessado em 04/02/2020.
- [21] Smart Cards and Parking. Smart Card Alliance, 2006.
- [22] <https://www.walmart.com/ip/Multi-Function-CAC-Card-Reader-EEEkit-Can-Read-DOD-Military-Common-Access-Smart-Card-ID-Card-support-with-Windows-Mac-OS-10-6-10-10-and-Linux/835317367>, acessado em 04/02/2020.
- [23] https://www.gta.ufrj.br/grad/04_2/smartcard/, acessado em 04/02/2020.
- [24] www.ibia.org/biometrics/, 2008.
- [25] Physical Access Control System Migration Options for Using FIPS 201-1 Compliant Credentials. PAC-07002. Smart Card Alliance Physical Access Council in collaboration with Open Security Exchange, Security Industry Association, and International Biometric Industry Association, 2007.

- [26] Costa, L. R., Obelheiro, R. R., and Fraga, J. S. Introdução à Biometria. Departamento de Automação e Sistemas, Universidade Federal de Santa Catarina, 2007.
- [27] Kuhn, M. Security - Biometric identification. Michaelmas 2003 - Part II. University of Cambridge, 2003.
- [28] German Biometric Strategy Platform - Biometrics State of the Art, Industry Strategy Development, and Platform Conception. Study. BITKOM, 2005.
- [29] Curso Prático de C#, Paulo Capela Marques, FCA – Editora de Informática, Lda, 1ª Edição, 2016, pp. 1-4.
- [30] Regulamento (UE) 2016/679 do Parlamento Europeu e do Conselho de 27 de abril de 2016 relativo à proteção das pessoas singulares no que diz respeito ao tratamento de dados pessoais e à livre circulação desses.
- [31] https://kitcc.pt/ccidadao/documentos/cc_manual_middleware_technical_v121.pdf, acedido em 04/02/2020.

ANEXO A



DIREÇÃO-GERAL DE ALIMENTAÇÃO E VETERINÁRIA
(Decreto-Regulamentar n.º 31/2012, de 13 de março)

Legenda dos serviços Centrais

- (a) **Orgãos**
- (b) **Serviços Operativos Centrais**
- (c) **Serviços de Apoio Técnico e Administrativo**
- (d) **Serviços Operativos Regionais**

- Campo Grande, 50, 1700-093 Lisboa - Tel.: 21 323 95 00
- DGAV - Tapada da Ajuda, 1349-017 Lisboa – Tel.: 21 361 32 00
- Rua António Serpa, n.º 26, 1º, 1050-027 Lisboa - Tel.: 21 780 82 00
- DGAV – Quinta do Marquês, Av. da República, 2780-157 Oeiras – Tel.: 21 446 40 00

ANEXO B

ANEXO C

DIREÇÃO DE SERVIÇOS DE ALIMENTAÇÃO E VETERINÁRIA DA REGIÃO NORTE

Manual	14/2019
Plano:	Relatórios Técnicos
Nome do Autor:	Eduardo Manuel Jacinto Brás
Última Revisão:	2020-01-07
Título:	MANUAL DE TÉCNICO - TIMESET

1. ENQUADRAMENTO

As tecnologias da informação e da comunicação têm assumido um papel progressivamente mais importante na Administração Pública, tanto nas relações interadministrativas, como nas relações da Administração com os particulares. Elas tornaram-se num instrumento essencial para a inovação e modernização administrativa, permitindo melhorar a eficiência, integrar e disponibilizar serviços, e antecipar necessidades.

Esta importância crescente tem sido plasmada nos diplomas legais que regulamentam a atividade administrativa, nas orientações estratégicas dos Governos para a Administração Pública, bem como nos objetivos estratégicos dos próprios órgãos e serviços.

O novo Código do Procedimento Administrativo, aprovado pelo Decreto-Lei n.º 4/2015, de 7 de janeiro, determina os princípios aplicáveis à administração eletrónica, preceituando que os órgãos e serviços da Administração Pública devem utilizar meios eletrónicos no desempenho da sua atividade, de modo a promover a eficiência e a transparência administrativas e a proximidade com os interessados.

O XXI Governo Constitucional, no seu programa, definiu uma aposta na modernização do Estado. Trata-se de um compromisso de mandato do Governo, que investe na implementação de medidas de eliminação de burocracias desnecessárias, de simplificação legislativa e de procedimentos, de atualização organizacional, de recursos e soluções tecnológicas. Constituem exemplo desta orientação estratégica a Estratégia TIC 2020¹ - Estratégia Para a Transformação Digital na Administração Pública – e o recuperar do programa SIMPLEX.

A DGAV tem assumido também no seu Quadro de Avaliação e Responsabilização, QUAR, como objetivo estratégico do Serviço *Promover a utilização das Tecnologias de Informação e Comunicação*, que se desdobra em diferentes objetivos operacionais, nomeadamente, implementar novos sistemas de gestão de informação e comunicação e promover a reengenharia de processos internos.

Neste âmbito, tem-se destacado como fundamental para a Administração a desmaterialização de processos, que procura otimizar os circuitos de informação através de aplicações tecnológicas, sendo a informação comunicada e arquivada no formato digital, no sentido de agilizar e facilitar a monitorização dos processos.

A desmaterialização de processos surge como uma ferramenta fundamental, nomeadamente devido aos problemas associados à existência de documentação em formato papel, permitindo, assim, uma definição clara, estruturada e mais transparente dos processos. O

¹ Aprovada pela Resolução do Conselho de Ministros n.º 108/2017

aumento da produtividade, que geralmente resulta da desmaterialização, provém da diminuição do tempo de realização de tarefas, da rentabilização de recursos humanos afetos àquelas tarefas, do aumento da velocidade de acesso à informação e, ainda, da melhoria da comunicação entre as unidades orgânicas de um serviço e/ou entre diferentes serviços.

O trabalhador em funções públicas está sujeito aos deveres gerais previstos na Lei Geral do Trabalho em Funções Públicas², doravante LTFP, entre os quais dever de assiduidade e de pontualidade, que consistem em comparecer ao serviço regular e continuamente e nas horas que estejam designadas.

Por sua vez, o empregador público deve manter um registo que permita apurar o número de horas de trabalho prestadas pelo trabalhador, por dia e por semana, com indicação da hora de início e de termo do trabalho, bem como dos intervalos efetuados. Nos órgãos ou serviços com mais de 50 trabalhadores, o registo deve ser efetuado por sistemas automáticos ou mecânicos³.

Tendo como objetivo geral a desmaterialização do processo de registo de assiduidade dos trabalhadores da DGAV, e conseqüente melhoria da eficiência dos respetivos processos de gestão de recursos humanos, procedeu-se ao desenvolvimento de uma aplicação. Esta aplicação recorre ao registo de entradas e saídas com base na identificação do trabalhador mediante leitura ótica do seu Cartão de Cidadão.

A Lei n.º 7/2007, de 5 de Fevereiro, na redação dada pela Lei n.º32/2017, de 1 de junho, cria o Cartão de Cidadão (CC) e rege a sua emissão e utilização. O CC constitui título bastante para provar a identidade do titular perante quaisquer autoridades e entidades públicas ou privadas, sendo válido em todo o território nacional (cf. Art.º 4.º da referida Lei). O CC é um documento de identificação múltipla, que inclui uma zona específica destinada a leitura ótica e incorpora um ou mais circuitos integrados. Permite ao respetivo titular, entre outros, provar a sua identidade perante terceiros através da leitura de elementos visíveis, coadjuvada pela leitura ótica de uma zona específica. É ainda de ressaltar que, a leitura ótica da zona específica do cartão está reservada a entidades ou serviços do Estado e da Administração Pública⁴.

² Aprovada pela Lei n.º 35/2014, de 20 de junho, alterada pelas Leis n.ºs 84/2015, de 7 de agosto, 18/2016, de 20 de junho, 42/2016, de 28 de dezembro, e 25/2017, de 30 de maio.

³ Cf. n.ºs 1 e 2 do artigo 104.º da LTFP.

⁴N.ºs 1, 2 e 3 do artigo 6º da Lei n.º 7/2007, de 5 de Fevereiro, na redação dada pela Lei n.º32/2017, de 1 de junho.

2. MANUAL TÉCNICO

2.1. LINGUAGEM DE PROGRAMAÇÃO:

dot.NET - C#

2.2. REQUISITOS MÍNIMOS:

- 2.2.1. CPU: Pentium 4
- 2.2.2. Memória RAM: 1GB
- 2.2.3. Disco Rígido: 10MB de espaço livre
- 2.2.4. Leitor ou teclado com leitor de Cartão de Cidadão (CC)
- 2.2.5. Sistema Operativo: Windows 7 ou superior
- 2.2.6. Arquitetura: 32 ou 64 Bits
- 2.2.7. Rede: Acesso à internet
- 2.2.8. Aplicações de Terceiros: Microsoft OneDrive, Microsoft Excel 2007⁵ ou superior e Aplicação do Cartão de Cidadão 1.62.0⁶.

2.3. OBJETIVO DA APLICAÇÃO

Esta aplicação tem como objectivo desmaterializar o processo de registo de assiduidade no âmbito das Unidades Desconcentradas e Unidades de Abate.

2.4. INSTALAÇÃO:

- a) Instalar a aplicação do CC na versão já mencionada;
- b) Instalar a aplicação Microsoft OneDrive (se necessário, uma vez que o Windows 10 já traz de raiz esta aplicação);
- c) Instalar a aplicação Microsoft Excel (se necessário);
- d) **(Válido apenas para a primeira instalação)** Inscrição na página da “Cloud” Microsoft OneDrive; Se já existir conta criada (<https://onedrive.live.com/about/pt-pt/>)
- e) Fazer o “login” na aplicação OneDrive com as credenciais criadas no passo d);

⁵ No caso da versão do Microsoft Excel ser a de 2007 deverá ser instalado um suplemente fornecido pela própria Microsoft: Suplemento para o Microsoft Office 2007: Guardar como PDF ou XPS da Microsoft - <https://www.microsoft.com/pt-pt/download/details.aspx?id=7>; Aplicação também disponível na pasta “Aplicações de Terceiros” que acompanha este manual na sua versão digital.

⁶ É obrigatória ser esta a versão. Aplicação disponível na pasta “Aplicações de Terceiros” que acompanha este manual na sua versão digital.

- f) **(Válido apenas para a primeira instalação)** Copiar para a pasta local associada à “Cloud” OneDrive (ex. C:\Users*****\OneDrive\) as seguintes pastas (que seguem juntamente com este documento no seu formato digital):
1. “Aplicação”;
 2. “Assiduidade”;
 3. “Templates Mensais”;

NOTA: A estrutura local das pastas da “OneDrive” deverá ser a mesma da figura 1.

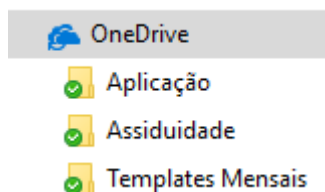


Figura 1. Estrutura de pastas da "Cloud" local

- g) Copiar para a raiz do disco de Sistema a pasta “DSAVRN” (ex. C:\DSAVRN\), que segue juntamente com este documento no seu formato digital;
- h) Nessa pasta editar o ficheiro “Caminho OneDrive.txt” substituindo o endereço existente pelo endereço relativo ao computador onde é instalada a aplicação “TimeSet” e gravar;
- i) Também nessa pasta editar o ficheiro “Matadouro_DAV_NAV_DSAVRN.txt”, substituindo o nome já existente por um de acordo com a identificação que se queira dar consoante o local onde é instalado a aplicação “TimeSet” e gravar;
- j) Criar um atalho da aplicação “TimeSet” no Ambiente de Trabalho da estação de trabalho em causa.
- k) Na pasta “Aplicação” da OneDrive, editar o ficheiro “Direção de Serviços.txt” substituindo o nome já existente pelo da Direção de Serviços Regional onde é feita a instalação.

NOTA: Por cada máquina em que é instalada a aplicação é necessário recolher o respetivo macaddress e registar num ficheiro para o efeito (ex. Excel) que deverá ser gerido por um administrador nomeado superiormente. As entidades externas à DGAV deverão, sempre que mudarem de máquina, ou de equipamento de rede da própria máquina, comunicar ao referido administrador.

Quanto às contas da “OneDrive”, poderão existir de forma repartida por departamentos, organismos, regiões, etc de forma a ter mais espaço (5 GB por conta – OneDrive Freeware) ou apenas uma conta. De uma maneira ou de outra deverá também ser um administrador a gerir os espaços livres e a possuir as credenciais de acesso.

Não foi contemplado um backup aos ficheiros de registo de assiduidade uma vez que esta aplicação está vocacionada para entidades extra DGAV e não é possível associar por exemplo um servidor.

Não é propósito deste manual demonstrar e abordar todo o código presente na aplicação. São mais de 10.000 linhas de código. Pretende-se apenas que se perceba a estrutura de codificação e o seu raciocínio. Os códigos exemplificados neste manual são apenas excertos.

2.5. ASPETO GERAL DA APLICAÇÃO

Figura 2. Aspeto geral da aplicação "TimeSet"

2.6. CARACTERÍSTICAS TÉCNICAS – ÁREAS

2.6.1. Nota prévia:

A aplicação valida sempre, tanto na sua inicialização como na acção sobre qualquer um dos botões, se o CC se encontra presente no leitor de cartões através do código seguinte;

```
//INICIAR LEITURA DO CARTÃO
try
{
Pteid.Init(null);
}
catch
{
MessageBox.Show("Por favor insira o Cartão de Cidadão no leitor");
Environment.Exit(0);
}
}
```

É importante referir também, que, qualquer ação que o utilizador exerça sobre a aplicação é –lhe sempre exibida uma mensagem de sucesso ou insucesso após o término da operação.

2.6.2. Área 1 – Identificação:

Esta área tem como função identificar a Direção de Serviços (hardcoded⁷), o local de trabalho e a unidade orgânica a que o trabalhador pertence assim como o seu nome completo e a sua fotografia.

O **local de trabalho** é possível mudar editando o ficheiro “Matadouro_DAV_NAV_DSAVRN.txt” presente na pasta “DSAVRN” (C:\DSAVRN\) de acordo com a figura 3;

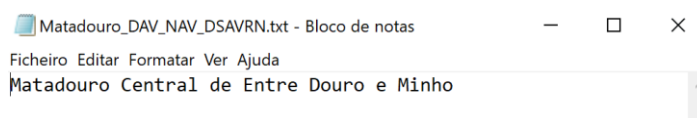


Figura 3. Ficheiro de identificação do local de trabalho

A leitura do ficheiro “txt” é feita da seguinte maneira:

```
string matadouro =
System.IO.File.ReadAllText(@"C:\DSAVRN\Matadouro_DAV_NAV_DSAVRN.txt",
Encoding.GetEncoding("windows-1254"));
label13.Text = matadouro;
```

⁷ Dados ou parâmetros de uma aplicação que não podem ser alterados sem modificar a codificação da mesma. Neste caso apenas se pode modificar editando a propriedade “Text” da “label15”

Os itens do campo “Unidade Orgânica” podem ser alterados (adicionados, removidos, editados) através da edição do ficheiro “Unidade Orgânica.txt” presente na pasta “Aplicação” (C:\users\????\OneDrive\Aplicação) conforme figura 4;

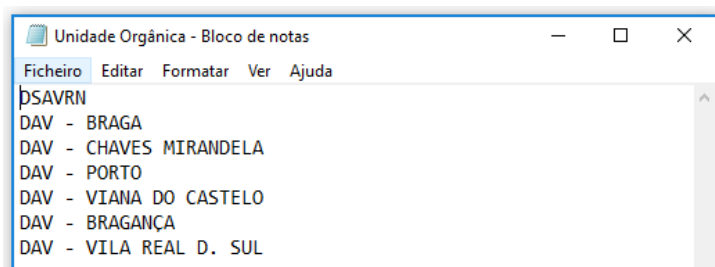


Figura 4. Editor de "strings" da Unidade Orgânica

A leitura e população dos itens na caixa de seleção respetiva é feita da seguinte forma:

```
string[] unidade = File.ReadAllLines(@"C:\DSAVRN\Unidade Orgânica.txt",
Encoding.GetEncoding("windows-1254"));
foreach (var line2 in unidade)
{
    comboBox9.Items.Add(line2);
}
```

O nome do utilizador e a fotografia são lidos a partir do CC. O nome do utilizador é a concatenação dos primeiros nomes e dos apelidos cuja codificação é apresentada de seguida:

```
CardData i = getCardData();
string space2 = i.IdInformation.firstname.ToString() + " " +
i.IdInformation.name.ToString();
```

Já a fotografia é obtida usando o seguinte código:

```
var pictureData = Pteid.GetPic();
var pictureBytes = pictureData.picture;
BitmapImageCreator.Register();
var j2kImg = CSJ2K.J2kImage.FromBytes(pictureBytes);
var bmap = j2kImg.As<Bitmap>();
pictureBox1.Image = bmap;
```

NOTA IMPORTANTE: De se referir que os únicos dados que a aplicação lê são o nome, apelido e fotografia. E aqueles que são efetivamente guardados são apenas o nome e apelido e só para identificação do colaborador no Mod-75-A.

Ou seja, a aplicação não se socorre de dados únicos nem é necessária qualquer introdução de PINs para ler estes dados do CC.

2.6.3. Área 2 – Registo de assiduidade:

É neste espaço que o colaborador deve registar a sua assiduidade acionando os botões “Entrada” e “Saída” referentes ao 1º ou 2º períodos conforme o caso.

Todos os botões têm uma estrutura de codificação igual. Ou seja, só muda o código que diga respeito às células no Excel onde irá registar a assiduidade. Sendo assim, este manual abordará as funcionalidades de apenas um botão, porque estruturalmente o código é igual nos restantes.

Criação de pastas e ficheiros – Após o utilizador carregar num dos botões, a aplicação em primeiro lugar, adquire a data local e de seguida a data e hora através da invocação de um “webservice” disponibilizado livremente por parte da Microsoft. A invocação deste webservice será feita sempre que uma funcionalidade envolva a questão tempo de forma a que a data e hora não dependa localmente, isto porque a manutenção da máquina não é da responsabilidade da DGAV e por força disso mesmo possa haver problemas com a hora e data local (ex. pilha de lítio da BIOS gasta). A outra razão para a invocação deste “webservice” é a certeza que a data e hora não é alterada localmente de forma maliciosa por parte do utilizador.

Posto isto, se a invocação for bem sucedida (ligação à internet existente e webservice em funcionamento) é essa data e hora utilizadas, adicionando um comentário (“Rede OK”)na célula correspondente. Caso contrário a data e hora usada é a local mas é também adicionado um comentário na célula correspondente com essa informação (“Rede NOK. Data e Hora local”). O código usado é o seguinte:

```
string today3 = DateTime.Now.ToString("HH:mm");
string dia = DateTime.Now.ToString("dd");
string dia2 = DateTime.Now.ToString("dd MMM yyyy");
string redeok = "Rede OK";
DateTime dateTime = DateTime.MinValue;
```

```

        System.Net.HttpWebRequest request =
(System.Net.HttpWebRequest)System.Net.WebRequest.Create("http://www.
microsoft.com");
        request.Method = "GET";
        request.Accept = "text/html, application/xhtml+xml, */*";
        request.UserAgent = "Mozilla/5.0 (compatible; MSIE 10.0; Windows
NT 6.1; Trident/6.0)";
        request.ContentType = "application/x-www-form-urlencoded";
        request.CachePolicy = new
System.Net.Cache.RequestCachePolicy(System.Net.Cache.RequestCacheLevel.
NoCacheNoStore);
        try
        {
            System.Net.HttpWebResponse response =
(System.Net.HttpWebResponse)request.GetResponse();

            if (response.StatusCode == System.Net.HttpStatusCode.OK)
            {
                string todaysDates = response.Headers["date"];

                dateTime = DateTime.ParseExact(todaysDates, "ddd, dd MMM
yyyy HH:mm:ss 'GMT'",

System.Globalization.CultureInfo.InvariantCulture.DateTimeFormat,
System.Globalization.DateTimeStyles.AssumeUniversal);
                today3 = (dateTime.ToString("HH:mm"));
                dia = (dateTime.ToString("dd"));
                dia2 = (dateTime.ToString("dd MMM yyyy"));
            }
        }
        catch
        {
            Console.WriteLine("Sem acesso à internet");
            redeok = "Rede NOK. Data e Hora local. ATENÇÃO";
        }
    
```

Em segundo lugar a aplicação valida se a pasta referente ao mês e ano já existem e se o ficheiro Excel também já existe para o atual utilizador.

A estrutura de pastas e ficheiros que a aplicação cria é:

...\OneDrive\Assiduidade\Ano\Mês\Nome completo constante do cartão de cidadão.xlsx".

Se nada existir, a aplicação cria todas as pastas e respectivo ficheiro. Se já existir a pasta do “Ano”, cria a pasta do “Mês” e o ficheiro Excel. Se já existir todas as pastas, cria apenas o Excel e se este já existir é feita a sua edição. A imagem 5 mostra a estrutura. Código usado:

//VERIFICAÇÃO DE EXISTÊNCIA E CRIAÇÃO DA PASTA NA ONEDRIVE

```
System.IO.Directory.CreateDirectory(text + @"\" + "Assiduidade" + @"\" + sYear + @"\" + sMonth);
string caminho = text + @"\" + "Assiduidade" + @"\" + sYear + @"\" + sMonth + @"\" + space4 + ".xlsx";
//FIM
```

//VERIFICAÇÃO SE EXISTE O FICHEIRO

//SE NÃO EXISTE

```
if (!File.Exists(caminho))
```

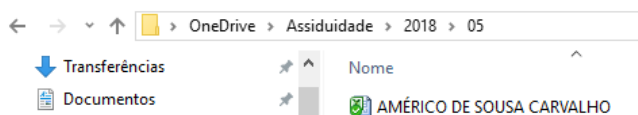


Figura 5. Ficheiro Excel de registo de assiduidade

O ficheiro Excel em boa verdade não é criado, mas sim copiado partir da pasta “Templates Mensais” guardada também na “Cloud”. Isto porque os ficheiros Excel já estão criado e formatados para cada mês de cada ano em particular, estando já contemplados os fins-de-semana e feriados nacionais. Ou seja, a aplicação copia o ficheiro Excel referente ao mês em que regista a assiduidade. O código usado segue abaixo:

```
string caminho3 = text + @"\" + "Templates Mensais";
if (sMonth=="01")
{
    System.IO.File.Copy(caminho3 + @"\" + "Mod 75-A-DGAV Registo de Assiduidade Janeiro.xlsx", caminho);
}
if (sMonth == "02")
{
    System.IO.File.Copy(caminho3 + @"\" + "Mod 75-A-DGAV Registo de Assiduidade Fevereiro.xlsx", caminho);
}
if (sMonth == "03")
{
```

```

        System.IO.File.Copy(caminho3 + @"\" + "Mod 75-A-DGAV Registo
de Assiduidade Março.xlsx", caminho);
    }
    if (sMonth == "04")
    {
        System.IO.File.Copy(caminho3 + @"\" + "Mod 75-A-DGAV Registo
de Assiduidade Abril.xlsx", caminho);
    }
    if (sMonth == "05")
    {
        System.IO.File.Copy(caminho3 + @"\" + "Mod 75-A-DGAV Registo
de Assiduidade Maio.xlsx", caminho);
    }
    if (sMonth == "06")
    {
        System.IO.File.Copy(caminho3 + @"\" + "Mod 75-A-DGAV Registo
de Assiduidade Junho.xlsx", caminho);
    }
    if (sMonth == "07")
    {
        System.IO.File.Copy(caminho3 + @"\" + "Mod 75-A-DGAV Registo
de Assiduidade Julho.xlsx", caminho);
    }
    if (sMonth == "08")
    {
        System.IO.File.Copy(caminho3 + @"\" + "Mod 75-A-DGAV Registo
de Assiduidade Agosto.xlsx", caminho);
    }
    if (sMonth == "09")
    {
        System.IO.File.Copy(caminho3 + @"\" + "Mod 75-A-DGAV Registo
de Assiduidade Setembro.xlsx", caminho);
    }
    if (sMonth == "10")
    {
        System.IO.File.Copy(caminho3 + @"\" + "Mod 75-A-DGAV Registo
de Assiduidade Outubro.xlsx", caminho);
    }
    if (sMonth == "11")
    {
        System.IO.File.Copy(caminho3 + @"\" + "Mod 75-A-DGAV Registo
de Assiduidade Novembro.xlsx", caminho);
    }
    if (sMonth == "12")
    
```

```

    {
        System.IO.File.Copy(caminho3 + @"\" + "Mod 75-A-DGAV Registo
de Assiduidade Dezembro.xlsx", caminho);
    }

```

NOTA: É necessário, de forma manual, todos os anos se editarem os ficheiros Excel referente a cada mês mantendo sempre o nome dos ficheiros já existentes.

Aquisição dos principais dados do registo de assiduidade – Os dados adquiridos usados para o registo de assiduidade após se carregar nos botões de ação para o efeito são a “Data”, a “Hora”, o “IP” e o “MACAddress”.

- a) Data e Hora – São obtidos da mesma forma descrita mais acima no caso da criação de pastas e ficheiros. Adquire-se a hora e data local e invoca-se o “webservice” para o mesmo efeito. A data e hora local só prevalecem se não for possível adquiri-la por “webservice”.
- b) IP – Embora não seja um dado que se possa controlar fora da DGAV, e não forneça nenhuma mais valia para o registo de assiduidade em si, é um dado que se pode tornar relevante, no contexto de segurança/redundância no caso de se usar a aplicação em instalações da DGAV.

```

// LER O IP
string sHostName = Dns.GetHostName();
string IPfinal = "0";
IPHostEntry ipE = Dns.GetHostByName(sHostName);
IPAddress[] IpA = ipE.AddressList;
for (int a = 0; a < IpA.Length; a++)
{
    IPfinal = IpA[a].ToString();
}
//FIM

```

- c) MACAddress – Este é o dado mais importante para assegurar que utilizador fez o registo no local devido. É um código alfa-numérico único.

```

//LER MACADDRESS
System.Management.ManagementClass objMgmtCls = new
System.Management.ManagementClass("Win32_NetworkAdapter");
List<string> colors = new List<string>();
foreach (System.Management.ManagementObject objMgmt in
objMgmtCls.GetInstances())

```

```

{
    string result;
    result = Convert.ToString(objMgmt["MACAddress"]);
    if (String.IsNullOrEmpty(result))
        Console.WriteLine("is null or empty");
    else
        listBox1.Items.Add(result);
}
string[] arr = new string[listBox1.Items.Count];
for (int s = 0; s < listBox1.Items.Count; s++)
{
    arr[s] = listBox1.Items[s].ToString();
}
string address = (arr[0]);
textBox12.Text = address;
string matadouro =
System.IO.File.ReadAllText(@"C:\DSAVRN\Matadouro_DAV_NAV_DSAVRN.txt",
Encoding.GetEncoding("windows-1254"));
string maIP = "IP:" + " " + IPfinal + " " + "MACAddress:" + " " + address
+ " " + matadouro + " " + redeok;
//FIM
    
```

Preenchimento dos dados adquiridos no ficheiro Excel (Mod-75-A) – Após todos os dados adquiridos, uns carregando nos botões de ação de registo de assiduidade automática e outros logo no “load” do “form”, estes são preenchidos no ficheiro Excel supra identificado conforme comprova a figura 6.

Unidade Orgânica Mês/Ano

Trabalhador

DIAS	MANHÃ		TARDE		DIFERENCIAL	OBSERVAÇÕES
	ENTRADA	SAIDA	ENTRADA	SAIDA		
1						Feriado
2	6:50	9:42	12:44	17:57	1:05	Erro Entrada/Saída
3	9:10	15:59			-0:11	
4	9:33	12:30	13:29		-17:32	
5						Sábado
6						Domingo
7	7:04	10:11	12:57	17:39	0:49	
8	7:14	9:22	12:54	16:34	-1:12	
9					-7:00	
10	8:58	12:49	13:07	17:27		IP: 192.168.1.79 MACAddress: B4:85:2F:AB:4A:BC Matadouro e Carnes Linda Rosa Rede OK
11						
12						
13						
14	6:50	10:20	12:45		-16:15	
15					-7:00	

Figura 6. Ficheiro Excel de registo de assiduidade criado pela aplicação "TimeSet"

Relativamente a validações durante esta etapa a aplicação verifica se já existe um registo para o período em que o utilizador pretende registar que em caso afirmativo avisa o utilizador através de uma caixa de mensagem não fazendo qualquer registo no ficheiro Excel. Este caso é válido para todos os períodos. Verifica também, exceto na Entrada do 1º Período, se existem picagens nos períodos imediatamente anteriores que em caso negativo também avisa o utilizador não fazendo nenhum registo no ficheiro Excel.

No caso, especial, da Entrada do 2º Período a aplicação valida se há um intervalo de “1 Hora” entre o registo anterior e este dando a informação ao utilizador se for o caso de que não pode fazer o registo e que deverá esperar “x tempo”.

Se as validações estiverem corretas o utilizador é informado através de uma caixa de mensagem de que o registo foi inserido com sucesso.

Todos os ficheiros Excel são gravados com uma “password” para que haja confidencialidade dos registos e não seja possível manipular ou editar os dados. Só os Recursos Humanos (RH) e o administrador de sistema têm acesso à “password”. A sua transmissão depende de autorização superior. Esclarece-se ainda que esta está definida no código (“hardcoded”).

O código aqui espelhado refere-se apenas ao dia “1” dos vários períodos uma vez que nos restantes 29 dias é usado a mesma estrutura de código mudando apenas as coordenadas das células a preencher que dependem do dia.

```
Excel.Workbook MyBook = null;
Excel.Application MyApp = null;
Excel.Worksheet MySheet = null;
MyApp = new Excel.Application();
MyApp.Visible = false;
MyBook = MyApp.Workbooks.Open(caminho);
MySheet = (Excel.Worksheet)MyBook.Sheets[1];
MySheet.Cells[6, 5].Value = comboBox9.Text;
MySheet.Cells[8, 5].Value = space4;
MySheet.Cells[6, 9].Value = sMonth + "/" + sYear;

if (dia == "01")
{
    if (MySheet.Cells[13, 2].Value != null)
    {
        MessageBox.Show("ATENÇÃO. JÁ EXISTE ESTE REGISTO");
        MyBook.Password = "?????";
    }
}
```

```

        MyBook.Save();
        MyBook.Close();
        MyApp.Quit();
        return;
    }
    MySheet.Cells[13, 2].Value = today3;
    MySheet.Cells[13, 2].AddComment(maclP);
}

if (dia == "01")
{
    if (MySheet.Cells[13, 3].Value != null)
    {
        MessageBox.Show("ATENÇÃO. JÁ EXISTE ESTE REGISTO");
        MyBook.Password = "??????";
        MyBook.Save();
        MyBook.Close();
        MyApp.Quit();
        return;
    }
    if (MySheet.Cells[13, 2].Value == null)
    {
        MessageBox.Show("ATENÇÃO! NÃO FEZ PICAGEM ANTERIOR");
        MyBook.Password = "??????";
        MyBook.Save();
        MyBook.Close();
        MyApp.Quit();
        return;
    }
    MySheet.Cells[13, 3].Value = today3;
    MySheet.Cells[13, 3].AddComment(maclP);
}

if (dia == "01")
{
    if (MySheet.Cells[13, 4].Value != null)
    {
        MessageBox.Show("ATENÇÃO. JÁ EXISTE ESTE REGISTO");
        MyBook.Password = "??????";
        MyBook.Save();
        MyBook.Close();
        MyApp.Quit();
        return;
    }
}

```

```

if (MySheet.Cells[13, 3].Value == null)
{
    MessageBox.Show("ATENÇÃO! NÃO FEZ PICAGEM ANTERIOR");
    MyBook.Password = "?????";
    MyBook.Save();
    MyBook.Close();
    MyApp.Quit();
    return;
}
var almoco = DateTime.FromOADate(MySheet.Cells[13, 3].Value);
var almoco2 = (almoco.ToString("HH:mm:ss"));
var almoco3 = TimeSpan.Parse(almoco2);
var resultadoalmoco = TimeSpan.Parse(almoco2) +
TimeSpan.Parse("01:00");
var todayteste = TimeSpan.Parse(today3);
var diferencial = (resultadoalmoco) - (todayteste);
if ((todayteste) < (resultadoalmoco))
{
    MessageBox.Show("ATENÇÃO, PICAGEM NÃO EFETUADA! A
PAUSA PARA ALMOÇO DEVE SER DE 1H. FALTA" + " " + diferencial + " " + "PARA
PODER PICAR NOVAMENTE.");
    MyBook.Password = "?????";
    MyBook.Save();
    MyBook.Close();
    MyApp.Quit();
    return;
}
MySheet.Cells[13, 4].Value = today3;
MySheet.Cells[13, 4].AddComment(maclP);
}
    
```

```

if (dia == "01")
{
    if (MySheet.Cells[13, 5].Value != null)
    {
        MessageBox.Show("ATENÇÃO. JÁ EXISTE ESTE REGISTO");
        MyBook.Password = "?????";
        MyBook.Save();
        MyBook.Close();
        MyApp.Quit();
        return;
    }
    if (MySheet.Cells[13, 4].Value == null)
    {
    
```

```

MessageBox.Show("ATENÇÃO! NÃO FEZ PICAGEM ANTERIOR");
MyBook.Password = "?????";
MyBook.Save();
MyBook.Close();
MyApp.Quit();
return;
}
MySheet.Cells[13, 5].Value = today3;
MySheet.Cells[13, 5].AddComment(macIP);
}

```

2.6.4. Área 3 – Registo de observações:

Esta área de registo de observações está destinada a ser usada apenas quando existe a impossibilidade de fazer o registo automaticamente conforme o ponto 2.6.3. (ex. mau funcionamento da máquina, erro da aplicação, falha de energia, etc) ou então existe a necessidade de assinalar uma observação pertinente para a falta de registo de assiduidade de determinado período ou mesmo dia. As observações podem ser adicionadas, removidas ou editadas no ficheiro “Observações.txt” guardado na pasta “C:\users\???\OneDrive\Aplicação”. A figura 7 mostra o referido ficheiro.

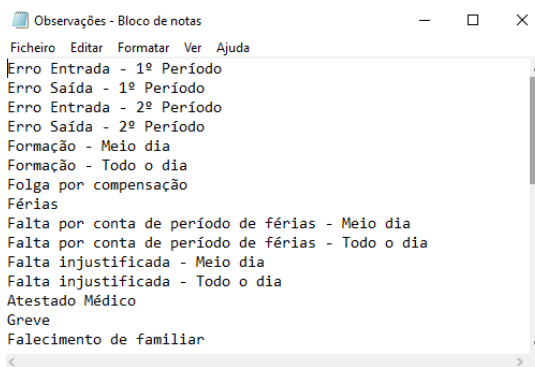


Figura 7. Ficheiro onde estão inseridas as observações possíveis de serem utilizadas pela aplicação

As observações existentes são:

Erro Entrada - 1º Período; Erro Saída - 1º Período; Erro Entrada - 2º Período; Erro Saída - 2º Período; Formação - Meio dia; Formação - Todo o dia; Folga por compensação; Férias; Falta por conta de período de férias - Meio dia; Falta por

conta de período de férias - Todo o dia, Falta injustificada - Meio dia; Falta injustificada - Todo o dia, Atestado Médico; Greve; Falecimento de familiar, Parentalidade; Trabalhador-Estudante, Serviço externo - Entrada 1º Período; Serviço externo - Saída 1º Período, Serviço externo - Entrada 2º Período; Serviço externo - Saída 2º Período; Assistência a familiares - Entrada 1º Período; Assistência a familiares - Saída 1º Período; Assistência a familiares - Entrada 2º Período; Assistência a familiares - Saída 2º Período; Consulta médica - Entrada 1º Período; Consulta médica - Saída 1º Período; Consulta médica - Entrada 2º Período; Consulta médica - Saída 2º Período; Casamento; Acidente de trabalho - Meio dia; Acidente de trabalho - Todo o dia; Atividade sindical - Entrada 1º Período; Atividade sindical - Saída 1º Período; Atividade sindical - Entrada 2º Período; Atividade sindical - Saída 2º Período.

Relativamente a validações, quando se aciona o botão “Adicionar”, a aplicação verifica se as caixas de seleção “Dia”, “Mês”, “Ano” e “Observação” estão preenchidas avisando o utilizador se houver alguma que não esteja. Verifica também as caixas de texto “Horas” e “Minutos” no caso de alguns tipos de observações como se poderá ver mais abaixo.

```

if (comboBox1.Text == "" || comboBox2.Text == "" || comboBox3.Text == "" ||
    comboBox4.Text == "")
    {
        MessageBox.Show("ATENÇÃO! É OBRIGATÓRIO O PREENCHIMENTO
        DOS CAMPOS 'Dia', 'Mês', 'Ano' e 'Observação'");
        return;
    }
    
```

```

if (comboBox1.Text == "Erro Entrada - 1º Período" || comboBox1.Text
    == "Erro Saída - 1º Período" || comboBox1.Text == "Erro Entrada - 2º Período"
    || comboBox1.Text == "Erro Saída - 2º Período" || comboBox1.Text ==
    "Serviço externo - Entrada 1º Período" || comboBox1.Text == "Serviço externo
    - Saída 1º Período" || comboBox1.Text == "Serviço externo - Entrada 2º
    Período" || comboBox1.Text == "Serviço externo - Saída 2º Período" ||
    comboBox1.Text == "Assistência a familiares - Entrada 1º Período" ||
    comboBox1.Text == "Assistência a familiares - Saída 1º Período" ||
    comboBox1.Text == "Assistência a familiares - Entrada 2º Período" ||
    comboBox1.Text == "Assistência a familiares - Saída 2º Período" ||
    comboBox1.Text == "Consulta médica - Entrada 1º Período" ||
    comboBox1.Text == "Consulta médica - Saída 1º Período" || comboBox1.Text
    == "Consulta médica - Entrada 2º Período" || comboBox1.Text == "Consulta
    
```

```

médica - Saída 2º Período" || comboBox1.Text == "Atividade sindical - Entrada
1º Período" || comboBox1.Text == "Atividade sindical - Saída 1º Período" ||
comboBox1.Text == "Atividade sindical - Entrada 2º Período" ||
comboBox1.Text == "Atividade sindical- Saída 2º Período")
    {
        if (comboBox5.Text == "" || comboBox6.Text == "")
        {
            MessageBox.Show("ATENÇÃO! É OBRIGATÓRIO O PREENCHIMENTO
DOS CAMPOS 'Horas' e 'Minutos'");
            return;
        }
    }
    
```

Outra verificação que é feita é se para a data selecionada existe ou não já um ficheiro criado. Se não houver não é possível adicionar uma observação. Isto porque o registo de observações pressupõe sempre que existe um registo automático criado à priori. Logo, o primeiro registo mensal de um utilizador nunca pode ser feito de forma manual através da área de registo de observações. O utilizador também é avisado desse facto através de uma caixa de mensagem.

```

string caminho = text + @"\" + "Assiduidade" + @"\" + anomanual + @"\" +
mesmanual + @"\" + space4 + ".xlsx";
if (!File.Exists(caminho))
    {
        MessageBox.Show("ATENÇÃO. FICHEIRO NÃO ENCONTRADO!");
        MessageBox.Show("A PRIMEIRA PICAGEM DO MÊS NÃO PODE SER
MANUAL! SENDO ASSIM FAÇA A PICAGEM AUTOMÁTICA DE ENTRADA DO 1º
PERÍODO E DEPOIS JÁ A PODE ALTERAR NOVAMENTE NA ÁREA DE
OBSERVAÇÕES.");
        comboBox1.SelectedIndex = -1;
        comboBox2.SelectedIndex = -1;
        comboBox3.SelectedIndex = -1;
        comboBox4.SelectedIndex = -1;
        comboBox5.SelectedIndex = -1;
        comboBox6.SelectedIndex = -1;
    }
    
```

Para analisar as validações inerentes a cada tipo de observação, podem-se dividi-las em três grupos para facilitar a compreensão do código e de que maneira este foi pensado.

1º Grupo – As observações que causam o desconto de um dia inteiro (ex. Férias, Parentalidade, Greve, Falta injustificada – Todo o dia);

Uma vez que o não registo de assiduidade se deve a uma situação que preenche todo o dia de trabalho a fórmula existente no Excel é re-escrita assumindo as horas de trabalho como zero em vez de sete. Se posteriormente o utilizador quiser acrescentar outra observação, neste caso não é possível, pois a observação contempla o dia todo. O seguinte código refere-se à observação “Férias” mas é estruturalmente igual para as restantes observações que contemplam o dia todo.

```
if (comboBox1.Text == "Férias")
{
    var calculo = diasn + 13;
    var calculo1 = "=(C" + calculo + "-B" + calculo + ")" + "+" + "(E" +
    calculo + "-D" + calculo + ")" + "-" + "TIMEVALUE(\"00:00\")";
    MySheet.Cells[13 + diasn, 6].Formula = calculo1;
    var conteudo = MySheet.Cells[13 + diasn, 7].Value;
    MySheet.Cells[13 + diasn, 7].Value = "Férias";
}
```

2º Grupo – As observações que causam o desconto de meio dia (ex. Acidente de trabalho – Meio dia, Falta injustificada – Meio dia, etc);

Uma vez que o não registo de assiduidade se deve a uma situação que preenche meio dia de trabalho a fórmula existente no Excel é re-escrita assumindo as horas de trabalho como “03:30H” em vez de sete. Se posteriormente o utilizador quiser acrescentar outra observação, neste caso é possível, pois a observação só contempla meio dia. O seguinte código refere-se à observação “Falta injustificada – Meio dia” mas é estruturalmente igual para as restantes observações que contemplam meio dia.

```
if (comboBox1.Text == "Falta injustificada - Meio dia")
{
    var calculo = diasn + 13;
    var calculo1 = "=(C" + calculo + "-B" + calculo + ")" + "+" + "(E" +
    calculo + "-D" + calculo + ")" + "-" + "TIMEVALUE(\"03:30\")";
    MySheet.Cells[13 + diasn, 6].Formula = calculo1;
    var conteudo = MySheet.Cells[13 + diasn, 7].Value;

    if (conteudo == null)
    {
        Console.WriteLine("Não contém nada");
    }
}
```

```

        MySheet.Cells[13 + diasn, 7].Value = "Falta injustificada - Meio
dia";
    }
    else
    {
        if (conteudo.Contains("injustificada"))
        {
            Console.WriteLine("Contém injustificada");
        }
        else
        {
            MySheet.Cells[13 + diasn, 7].Value = "Falta injustificada - Meio
dia" + " + " + conteudo;
        }
    }
}
    
```

3º Grupo – As observações que podem ser usadas por períodos de tempo não definido (ex. Consulta médica – Entrada 1º Período, Saída 2º Período, Erro Entrada – 1º Período, etc)

Este caso é o mais complexo, pois englobam situações que muitas vezes não preenchem sequer meios dias de trabalho, ou então referem-se a erros de entrada ou saída que como já se viu podem acontecer devido a erros de “hardware”, “software” ou do próprio utilizador que este tem de ter a possibilidade de alterar.

Sendo assim, tem de haver a possibilidade de se marcar manualmente todos os períodos existentes. Quando selecionam este tipo de observações tornam-se visíveis as caixas de texto “Horas” e “Minutos”.

```

if (comboBox1.Text == "Erro Entrada - 1º Período" || comboBox1.Text == "Erro Saída - 1º Período" || comboBox1.Text == "Erro Entrada - 2º Período" || comboBox1.Text == "Erro Saída - 2º Período" || comboBox1.Text == "Serviço externo - Entrada 1º Período" || comboBox1.Text == "Serviço externo - Saída 1º Período" || comboBox1.Text == "Serviço externo - Entrada 2º Período" || comboBox1.Text == "Serviço externo - Saída 2º Período" || comboBox1.Text == "Assistência a familiares - Entrada 1º Período" || comboBox1.Text == "Assistência a familiares - Saída 1º Período" || comboBox1.Text == "Assistência a familiares - Entrada 2º Período" || comboBox1.Text == "Assistência a familiares - Saída 2º Período" || comboBox1.Text == "Consulta médica -
    
```

```

Entrada 1º Período" || comboBox1.Text == "Consulta médica - Saída 1º
Período" || comboBox1.Text == "Consulta médica - Entrada 2º Período" ||
comboBox1.Text == "Consulta médica - Saída 2º Período" || comboBox1.Text
== "Atividade sindical - Entrada 1º Período" || comboBox1.Text == "Atividade
sindical - Saída 1º Período" || comboBox1.Text == "Atividade sindical - Entrada
2º Período" || comboBox1.Text == "Atividade sindical - Saída 2º Período")
    {
        label21.Visible = true;
        label22.Visible = true;
        label23.Visible = true;
        comboBox5.Visible = true;
        comboBox6.Visible = true;
    }
    
```

Nestes casos concretos, além do registo da observação no campo das observações no ficheiro Excel, também é adicionada uma nota em cada célula em que o utilizador registou manualmente.

Isto acontece, porque ao contrário dos restantes grupos é possível ter em cada período uma observação diferente e, quem faz a verificação dos registos de assiduidade (RH) tem a possibilidade de ver que justificação teve o utilizador para marcar manualmente cada período do dia.

Desta maneira, é óbvio que como já vimos é possível ao utilizador registar várias observações deste grupo. O código mostrado abaixo refere-se ao caso da "Consulta Médica", sendo que estruturalmente as outras observações deste grupo é exactamente igual.

//CONSULTA MÉDICA-----

```

if (comboBox1.Text == "Consulta médica - Entrada 1º Período")
    {
        if (MySheet.Cells[13 + diasn, 2].Comment != null)
            {
                MySheet.Cells[13 + diasn, 2].Comment.Delete();
            }

        MySheet.Cells[13 + diasn, 2].Value = horaseminutos;
        MySheet.Cells[13 + diasn, 2].AddComment("Consulta médica -
Entrada 1º Período");
        var conter = MySheet.Cells[13 + diasn, 7].Value;
        if (conter == null)
    
```

```

    {
        Console.WriteLine("Não contém nada");
        MySheet.Cells[13 + diasn, 7].Value = "Consulta médica";
    }
    else
    {
        if (conter.Contains("Consulta"))
        {
            Console.WriteLine("Contém Consulta");
        }
        else
        {
            MySheet.Cells[13 + diasn, 7].Value = "Consulta médica" + " + "
+ conter;
        }
    }
}
if (comboBox1.Text == "Consulta médica - Saída 1º Período")
{
    if (MySheet.Cells[13 + diasn, 2].Value == null)
    {
        MessageBox.Show("ATENÇÃO! NÃO FEZ PICAGEM ANTERIOR");
        MyBook.Password = "?????";
        MyBook.Save();
        MyBook.Close();
        MyApp.Quit();
        return;
    }
    var tempomenor = DateTime.FromOADate(MySheet.Cells[13 +
diasn, 2].Value);
    var tempomenor1 = (tempomenor.ToString("HH:mm:ss"));
    var tempomenor2 = TimeSpan.Parse(tempomenor1);
    var horaseminutos2 = horaseminutos + ":" + "00";
    var horaseminutos3 = TimeSpan.Parse(horaseminutos2);
    if (horaseminutos3 <= tempomenor2)
    {
        MessageBox.Show("ATENÇÃO! PICAGEM NÃO EFECTUADA!
HORA INFERIOR OU IGUAL À PICAGEM ANTERIOR.");
        MyBook.Password = "?????";
        MyBook.Save();
        MyBook.Close();
        MyApp.Quit();
        return;
    }
}

```

```

if (MySheet.Cells[13 + diasn, 3].Comment != null)
{
    MySheet.Cells[13 + diasn, 3].Comment.Delete();
}
MySheet.Cells[13 + diasn, 3].Value = horaseminutos;
MySheet.Cells[13 + diasn, 3].AddComment("Consulta médica -
Saída 1º Período");
var conter = MySheet.Cells[13 + diasn, 7].Value;
if (conter == null)
{
    Console.WriteLine("Não contém nada");
    MySheet.Cells[13 + diasn, 7].Value = "Consulta médica";
}
else
{
    if (conter.Contains("Consulta"))
    {
        Console.WriteLine("Contém Consulta");
    }
    else
    {
        MySheet.Cells[13 + diasn, 7].Value = "Consulta médica" + " + "
+ conter;
    }
}
}
if (comboBox1.Text == "Consulta médica - Entrada 2º Período")
{
    if (MySheet.Cells[13 + diasn, 3].Value == null)
    {
        MessageBox.Show("ATENÇÃO! NÃO FEZ PICAGEM ANTERIOR");
        MyBook.Password = "?????";
        MyBook.Save();
        MyBook.Close();
        MyApp.Quit();
        return;
    }
    var almoco = DateTime.FromOADate(MySheet.Cells[13 + diasn,
3].Value);
    var almoco2 = (almoco.ToString("HH:mm:ss"));
    var almoco3 = TimeSpan.Parse(almoco2);
    var resultadoalmoco = TimeSpan.Parse(almoco2) +
TimeSpan.Parse("01:00");
    var todayteste = TimeSpan.Parse(today3);

```

```

var horaseminutos2 = horaseminutos + ":" + "00";
var horaseminutos3 = TimeSpan.Parse(horaseminutos2);
var diferencial = (resultadoalmoco) - (horaseminutos3);
if ((horaseminutos3) < (resultadoalmoco))
{
    MessageBox.Show("ATENÇÃO, PICAGEM NÃO EFETUADA! A
PAUSA PARA ALMOÇO DEVE SER DE 1H. FALTA" + " " + diferencial + " " + "PARA
PODER PICAR NOVAMENTE.");
    MyBook.Password = "?????";
    MyBook.Save();
    MyBook.Close();
    MyApp.Quit();
    return;
}
if (MySheet.Cells[13 + diasn, 4].Comment != null)
{
    MySheet.Cells[13 + diasn, 4].Comment.Delete();
}
MySheet.Cells[13 + diasn, 4].Value = horaseminutos;
MySheet.Cells[13 + diasn, 4].AddComment("Consulta médica -
Entrada 2º Período");
var conter = MySheet.Cells[13 + diasn, 7].Value;
if (conter == null)
{
    Console.WriteLine("Não contém nada");
    MySheet.Cells[13 + diasn, 7].Value = "Consulta médica";
}
else
{
    if (conter.Contains("Consulta"))
    {
        Console.WriteLine("Contém Consulta");
    }
    else
    {
        MySheet.Cells[13 + diasn, 7].Value = "Consulta médica" + " " +
+ conter;
    }
}
}
if (comboBox1.Text == "Consulta médica - Saída 2º Período")
{
    if (MySheet.Cells[13 + diasn, 4].Value == null)
    {

```

```

        MessageBox.Show("ATENÇÃO! NÃO FEZ PICAGEM ANTERIOR");
        MyBook.Password = "?????";
        MyBook.Save();
        MyBook.Close();
        MyApp.Quit();
        return;
    }
    var tempomenor = DateTime.FromOADate(MySheet.Cells[13 +
diasn, 4].Value);
    var tempomenor1 = (tempomenor.ToString("HH:mm:ss"));
    var tempomenor2 = TimeSpan.Parse(tempomenor1);
    var horaseminutos2 = horaseminutos + ":" + "00";
    var horaseminutos3 = TimeSpan.Parse(horaseminutos2);
    if (horaseminutos3 <= tempomenor2)
    {
        MessageBox.Show("ATENÇÃO! PICAGEM NÃO EFECTUADA!
HORA INFERIOR OU IGUAL À PICAGEM ANTERIOR.");
        MyBook.Password = "?????";
        MyBook.Save();
        MyBook.Close();
        MyApp.Quit();
        return;
    }
    if (MySheet.Cells[13 + diasn, 5].Comment != null)
    {
        MySheet.Cells[13 + diasn, 5].Comment.Delete();
    }
    MySheet.Cells[13 + diasn, 5].Value = horaseminutos;
    MySheet.Cells[13 + diasn, 5].AddComment("Consulta médica -
Saída 2º Período");
    var conter = MySheet.Cells[13 + diasn, 7].Value;
    if (conter == null)
    {
        Console.WriteLine("Não contém nada");
        MySheet.Cells[13 + diasn, 7].Value = "Consulta médica";
    }
    else
    {
        if (conter.Contains("Consulta"))
        {
            Console.WriteLine("Contém Consulta");
        }
        else
        {

```

```

        MySheet.Cells[13 + diasn, 7].Value = "Consulta médica" + " + "
    + conter;
    }
}
}
    
```

2.6.5. Área 4 – Visualização e validação de registos

Nesta área o utilizador tem a oportunidade de ir verificando a sua assiduidade ao longo do tempo e no final do mês se for o seu entendimento de que a folha de registos está correta pode validar o registo.

No caso da visualização apenas, a aplicação valida o preenchimento das caixas de texto “Mês” e “Ano”, que se não for o caso, informa o utilizador através de uma caixa de mensagem fazendo com que o código não avance.

```

if (comboBox8.Text == "" || comboBox7.Text == "")
{
    MessageBox.Show("ATENÇÃO! É OBRIGATÓRIO O PREENCHIMENTO
    DOS CAMPOS 'Mês' e 'Ano'");
    return;
}
    
```

Após esta validação, após o utilizador carregar no botão “Ver”, a aplicação verifica se existe algum ficheiro para o mês, ano, e nome do utilizador, que caso contrário tem o mesmo comportamento descrito imediatamente acima.

A visualização é feita através de um PDF temporário que é criado na pasta “C:\\DSAVRN\” e aberto automaticamente pela aplicação. Este ficheiro mantém-se guardado até o utilizador gerar outro.

```

string caminho = text + @"\" + "Assiduidade" + @"\" + anomanual + @"\" +
mesmanual + @"\" + space4 + ".xlsx";
    
```

```

string caminho2 = @"C:\\DSAVRN" + @"\" + space4 + ".PDF";
if (File.Exists(caminho2))
{
    File.Delete(caminho2);
}
if (!File.Exists(caminho))
{
    MessageBox.Show("ATENÇÃO. FICHEIRO NÃO ENCONTRADO!");
}
    
```

```

Excel.Workbook MyBook = null;
    
```

```

Excel.Application MyApp = null;
Excel.Worksheet MySheet = null;
MyApp = new Excel.Application();
MyApp.Visible = false;
MyBook = MyApp.Workbooks.Open(caminho, Password: "?????");
    
```

```

MyBook.ExportAsFixedFormat(Excel.XlFixedFormatType.xlTypePDF,
Filename: @"C:\DSAVRN\" + space4 + ".PDF",OpenAfterPublish:true);
MyBook.Save();
MyBook.Close();
MyApp.Quit();
comboBox7.SelectedIndex = -1;
comboBox8.SelectedIndex = -1;
    
```

Olhando para o caso da validação, caso o utilizador esteja pronto para validar o seu registo de assiduidade mensal deverá carregar no botão “Validar”. A aplicação valida o preenchimento das mesmas caixas de texto que se abordou no caso da visualização assim como a verificação de existência ou não do ficheiro para a o mês, ano e utilizador pretendido.

Feitas as validações, a aplicação pergunta ao utilizador se pretende assinar o documento validado digitalmente ou não.

Em caso afirmativo é pedido o PIN de Assinatura e se este for introduzido corretamente é guardado na pasta de assiduidade da “Cloud” um ficheiro PDF assinado digitalmente cujo nome é o do utilizador mais um sufixo “Assinado”. É apresentado ao utilizador uma caixa de mensagem a informar do sucesso da operação.

```

string caminho = text + @"\\" + "Assiduidade" + @"\\" + anomanual + @"\\" +
mesmanual + @"\\" + space4 + ".xlsx";
string caminho1 = text + @"\\" + "Assiduidade" + @"\\" + anomanual +
@"\" + mesmanual + @"\\" + space4;
string caminho2 = @"C:\DSAVRN" + @"\\" + space4 + ".PDF";
DialogResult dialogResult = MessageBox.Show("Deseja validar o registo
escolhido com a sua assinatura digital?", "Assinatura Digital",
MessageBoxButtons.YesNo);
if (dialogResult == DialogResult.Yes)
{
Excel.Workbook MyBook = null;
Excel.Application MyApp = null;
Excel.Worksheet MySheet = null;
MyApp = new Excel.Application();
MyApp.Visible = false;
MyBook = MyApp.Workbooks.Open(caminho, Password: "?????");
MySheet = (Excel.Worksheet)MyBook.Sheets[1];
    
```

```

MyBook.ExportAsFixedFormat(Excel.XlFixedFormatType.xlTypePDF,
caminho1, Excel.XlFixedFormatQuality.xlQualityMinimum, OpenAfterPublish:
true);

MyBook.Password = "?????";
MyBook.Save();
MyBook.Close();
MyApp.Quit();
PdfSignature ps = new PdfSignature("serial number");
ps.LoadPdfDocument(caminho1+".PDF");
ps.SignaturePosition = SignaturePosition.BottomRight;
ps.SigningReason = "Aprovo o documento";
ps.SignaturePosition = SignaturePosition.BottomMiddle;
ps.DigitalSignatureCertificate = DigitalCertificate.LoadCertificate(false,
"",
"Selecione o certificado do Cartão de Cidadão", "");
File.WriteAllBytes(text + @"\ " + "Assiduidade" + @"\ " + anomanual +
@"\ " + mesmanual + @"\ " + space4 + " _ " + "Assinado" + ".PDF",
ps.ApplyDigitalSignature());
File.Delete(caminho1 + ".PDF");
MessageBox.Show("Mês de" + " " + mesmanual + " " + "do Ano de" + "
" + anomanual + " " + "VALIDADO E ASSINADO");
comboBox7.SelectedIndex = -1;
comboBox8.SelectedIndex = -1;
    
```

Caso o utilizador apenas queira validar e não assinar digitalmente; à semelhança do caso imediatamente anterior é guardado também um ficheiro PDF na “Cloud” cujo nome é o do utilizador, sem o sufixo “Assinado”. É apresentado ao utilizador uma caixa de mensagem a informar do sucesso da operação.

```

Excel.Workbook MyBook = null;
Excel.Application MyApp = null;
Excel.Worksheet MySheet = null;
MyApp = new Excel.Application();
MyApp.Visible = false;
MyBook = MyApp.Workbooks.Open(caminho, Password:
"eduardo");

MySheet = (Excel.Worksheet)MyBook.Sheets[1];
MyBook.ExportAsFixedFormat(Excel.XlFixedFormatType.xlTypePDF,
caminho1, Excel.XlFixedFormatQuality.xlQualityMinimum, OpenAfterPublish:
true);

MyBook.Password = "eduardo";
MyBook.Save();
    
```

```

MyBook.Close();
MyApp.Quit();
MessageBox.Show("Mês de" + " " + mesmanual + " " + "do Ano de"
+ " " + anomanual + " " + "VALIDADO");
comboBox7.SelectedIndex = -1;
comboBox8.SelectedIndex = -1;

```

2.6.6. Referências usadas

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;
using eidpt;
using System.IO;
using OfficialSDK;
using System.Net;
using CSJ2K.Util;
using Microsoft.VisualBasic;
using Excel = Microsoft.Office.Interop.Excel;
using SignLib.Certificates;
using SignLib.Pdf;

```

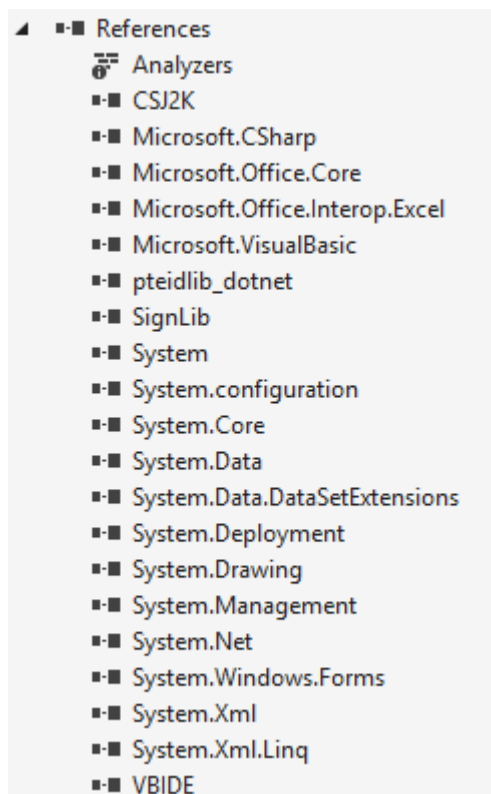


Figura 8. Referências carregadas no projeto - Visual Studio

3. RESOLUÇÃO DE PROBLEMAS – CONTACTOS

É sempre preferível o contato por correio electrónico com o maior número de detalhes possível para o problema encontrado (descrição pormenorizada, imagens, etc...).

Eng.º Eduardo Brás – DGAV:

Telefone: 253 783 000

Correio electrónico: eduardo.bras@dgav.pt

4. ANEXOS

ANEXO 1 – Registo de Assiduidade Mod.75 preenchido



IP: 10.100.2.66
MACAddress:
00:1E:0B:64:90:0F DAV
- Braga Rede OK

REGISTO MENSAL DE ASSIDUIDADE

Unidade Orgânica

Mês/Ano

Trabalhador

DIAS	MANHÃ		TARDE		DIFERENCIAL	OBSERVAÇÕES
	ENTRADA	SAIDA	ENTRADA	SAIDA		
1						Feriado
2	6:50	9:42	12:44	17:57	1:05	Erro Entrada/Saída
3	9:10	15:59			-0:11	
4	9:33	12:30	13:29	17:23	-0:09	Erro Entrada/Saída
5						Sábado
6						Domingo
7	7:04	10:11	12:57	17:39	0:49	Erro Entrada - Tarde
8	7:14	9:22	12:54	16:34	-1:12	
9					-7:00	
10	8:58	12:49	13:07	17:27	1:11	
11					-7:00	
12						Sábado
13						Domingo
14	6:50	10:20	12:45	16:33	0:18	IP: 192.168.1.79 MACAddress: B4:B5:2F:AB:4A:BC Matadouro e Carnes Linda Rosa Rede OK
15	6:47	9:45	12:46	17:53	1:05	
16	8:36	12:30	13:01	16:08	0:01	
17	8:40	12:35	12:54	14:50	-1:09	
18	9:01	12:31	13:52	17:20	-0:02	
19						Sábado
20						Domingo
21	6:55	10:11	12:44	17:07	0:39	
22	6:56	12:00	12:59	15:04	0:09	
23	8:52	12:30	12:55	17:02	0:45	
24	9:03	12:33	13:19	13:59	-2:50	
25	8:51	12:26	13:25	17:24	0:34	
26						Sábado
27						Domingo
28	6:50	10:01	12:54	17:51	1:08	Erro Entrada/Saída
29	9:10	12:14	13:47	17:28	-0:15	
30	9:07			17:21	1:14	
31						Feriado
Total					-10:50	

MOD. 75-A/DGAV

Superior Hierárquico

Data Ass. _____

ASSINATURA DO FUNCIONÁRIO

CAMPO GRANDE, 50 - 1700-093 LISBOA

ANEXO D

REGISTO MENSAL DE ASSIDUIDADE

Unidade Orgânica

Mês/Ano

Trabalhador

DIAS	1º Período		2º Período		DIFERENCIAL	OBSERVAÇÕES
	ENTRADA	SAIDA	ENTRADA	SAIDA		
1						Sábado
2						Domingo
3	6:13	12:32	14:00	15:00	0:19	Trabalhador - Estudante das 14:00 às 15:00
4	6:15	12:15	14:30	15:30	0:00	Trabalhador - Estudante das 14:30 às 15:30
5	6:15	10:00			0:15	Atividade sindical - Meio dia
6	6:15	10:30	13:15	16:00	0:00	Trabalhador - Estudante das 15:00 às 16:00
7	8:45	13:00	14:00	15:00	-1:45	Trabalhador - Estudante das 14:00 às 15:00 + Erro Entrada/Saída
8						Sábado
9						Domingo
10	6:15	12:15	14:00	15:00	0:00	Serviço externo + Trabalhador - Estudante das 14:00 às 15:00
11	6:15	9:15	14:30	15:30	0:30	Trabalhador - Estudante das 14:30 às 15:30 + Atividade sindical - Meio dia
12	6:15	13:15	14:15	15:00	0:45	Serviço externo + Trabalhador - Estudante das 14:00 às 15:00
13	6:15	9:45	11:00	16:00	1:30	
14	8:30	13:00	14:00	15:00	-1:30	
15						Sábado
16						Domingo
17	6:15	10:15	12:00	15:00	0:00	
18	6:15	12:01			2:16	Formação - Meio dia
19	6:15	13:15	14:15	15:15	1:00	Trabalhador - Estudante das 14:15 às 15:00
20	6:15	12:30	15:00	16:00	0:15	
21	8:45	12:20	13:20	16:30	-0:15	Erro Entrada/Saída
22						Sábado
23						Domingo
24	6:15	12:20	14:00	15:00	0:05	
25					0:00	Tolerância de ponto - Todo o dia
26	8:47	13:35	14:35	15:35	-1:12	
27	8:58	13:00	14:00	16:00	-0:58	
28	8:30	12:30	14:00	16:00	-1:00	
29						Sábado
30						
31						
Total					0:15	

MOD. 75-A/DGAV

Superior Hierárquico	
Data <input type="text" value="/ /"/>	Ass. _____
ASSINATURA DO FUNCIONÁRIO	

ANEXO E

DIREÇÃO DE SERVIÇOS DE ALIMENTAÇÃO E VETERINÁRIA DA REGIÃO NORTE

Manual	13/2019
Plano:	Relatórios Técnicos
Nome do Autor:	Eduardo Manuel Jacinto Brás
Última Revisão:	2020-01-07
Título:	MANUAL DE UTILIZAÇÃO - TIMESET

1. Enquadramento

As tecnologias da informação e da comunicação têm assumido um papel progressivamente mais importante na Administração Pública, tanto nas relações interadministrativas, como nas relações da Administração com os particulares. Elas tornaram-se num instrumento essencial para a inovação e modernização administrativa, permitindo melhorar a eficiência, integrar e disponibilizar serviços, e antecipar necessidades.

Esta importância crescente tem sido plasmada nos diplomas legais que regulamentam a atividade administrativa, nas orientações estratégicas dos Governos para a Administração Pública, bem como nos objetivos estratégicos dos próprios órgãos e serviços.

O novo Código do Procedimento Administrativo, aprovado pelo Decreto-Lei n.º 4/2015, de 7 de janeiro, determina os princípios aplicáveis à administração eletrónica, preceituando que os órgãos e serviços da Administração Pública devem utilizar meios eletrónicos no desempenho da sua atividade, de modo a promover a eficiência e a transparência administrativas e a proximidade com os interessados.

O XXI Governo Constitucional, no seu programa, definiu uma aposta na modernização do Estado. Trata-se de um compromisso de mandato do Governo, que investe na implementação de medidas de eliminação de burocracias desnecessárias, de simplificação legislativa e de procedimentos, de atualização organizacional, de recursos e soluções tecnológicas. Constituem exemplo desta orientação estratégica a Estratégia TIC 2020¹ - Estratégia Para a Transformação Digital na Administração Pública – e o recuperar do programa SIMPLEX.

A DGAV tem assumido também no seu Quadro de Avaliação e Responsabilização, QUAR, como objetivo estratégico do Serviço *Promover a utilização das Tecnologias de Informação e Comunicação*, que se desdobra em diferentes objetivos operacionais, nomeadamente, implementar novos sistemas de gestão de informação e comunicação e promover a reengenharia de processos internos.

Neste âmbito, tem-se destacado como fundamental para a Administração a desmaterialização de processos, que procura otimizar os circuitos de informação através de aplicações tecnológicas, sendo a informação comunicada e arquivada no formato digital, no sentido de agilizar e facilitar a monitorização dos processos.

A desmaterialização de processos surge como uma ferramenta fundamental, nomeadamente devido aos problemas associados à existência de documentação em formato papel, permitindo, assim, uma definição clara, estruturada e mais transparente dos processos. O

¹ Aprovada pela Resolução do Conselho de Ministros n.º 108/2017

aumento da produtividade, que geralmente resulta da desmaterialização, provém da diminuição do tempo de realização de tarefas, da rentabilização de recursos humanos afetos àquelas tarefas, do aumento da velocidade de acesso à informação e, ainda, da melhoria da comunicação entre as unidades orgânicas de um serviço e/ou entre diferentes serviços.

O trabalhador em funções públicas está sujeito aos deveres gerais previstos na Lei Geral do Trabalho em Funções Públicas², doravante LTFP, entre os quais dever de assiduidade e de pontualidade, que consistem em comparecer ao serviço regular e continuamente e nas horas que estejam designadas.

Por sua vez, o empregador público deve manter um registo que permita apurar o número de horas de trabalho prestadas pelo trabalhador, por dia e por semana, com indicação da hora de início e de termo do trabalho, bem como dos intervalos efetuados. Nos órgãos ou serviços com mais de 50 trabalhadores, o registo deve ser efetuado por sistemas automáticos ou mecânicos³.

Tendo como objetivo geral a desmaterialização do processo de registo de assiduidade dos trabalhadores da DGAV, e conseqüente melhoria da eficiência dos respetivos processos de gestão de recursos humanos, procedeu-se ao desenvolvimento de uma aplicação. Esta aplicação recorre ao registo de entradas e saídas com base na identificação do trabalhador mediante leitura ótica do seu Cartão de Cidadão.

A Lei n.º 7/2007, de 5 de Fevereiro, na redação dada pela Lei n.º32/2017, de 1 de junho, cria o Cartão de Cidadão (CC) e rege a sua emissão e utilização. O CC constitui título bastante para provar a identidade do titular perante quaisquer autoridades e entidades públicas ou privadas, sendo válido em todo o território nacional (cf. Art.º 4.º da referida Lei). O CC é um documento de identificação múltipla, que inclui uma zona específica destinada a leitura ótica e incorpora um ou mais circuitos integrados. Permite ao respetivo titular, entre outros, provar a sua identidade perante terceiros através da leitura de elementos visíveis, coadjuvada pela leitura ótica de uma zona específica. É ainda de ressaltar que, a leitura ótica da zona específica do cartão está reservada a entidades ou serviços do Estado e da Administração Pública⁴.

² Aprovada pela Lei n.º 35/2014, de 20 de junho, alterada pelas Leis n.ºs 84/2015, de 7 de agosto, 18/2016, de 20 de junho, 42/2016, de 28 de dezembro, e 25/2017, de 30 de maio.

³ Cf. n.ºs 1 e 2 do artigo 104.º da LTFP.

⁴N.ºs 1, 2 e 3 do artigo 6º da Lei n.º 7/2007, de 5 de Fevereiro, na redação dada pela Lei n.º32/2017, de 1 de junho.

2. Manual de utilização

2.1. Iniciar a aplicação – TimeSet

A aplicação pode ser encontrada no Ambiente de Trabalho do computador com o nome “TimeSet” e com o ícone:



Antes de iniciar a aplicação deverá introduzir o Cartão de Cidadão (CC) no leitor e esperar que seja lido. Normalmente, a leitura correta faz aparecer no canto inferior direito a fotografia contida no CC referente ao utilizador.

Se não tiver introduzido o CC no leitor ou houver um problema na leitura, a aplicação não se inicia apresentando uma mensagem de erro como mostra a figura 1.

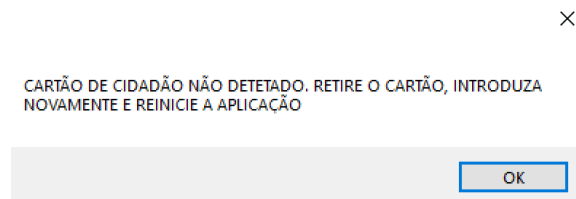


Figura 1. Mensagem de erro por incorreta leitura do CC;

NOTA: Todas as funcionalidades presentes nesta aplicação necessitam da leitura do CC. Ou seja, o CC deve estar sempre inserido no leitor enquanto a aplicação estiver aberta e em funcionamento.

Após iniciar a aplicação com sucesso irá surgir o ecrã como demonstra a figura 2.

Figura 2. Aspeto geral da aplicação "TimeSet"

2.2. Registo de assiduidade

Em primeiro lugar deve seleccionar a Unidade Orgânica a que pertence na caixa de seleção conforme figura 3.

Figura 3. Caixa de seleção com as várias unidades orgânicas

O registo de assiduidade faz-se carregando num dos botões conforme figura 4.

Figura 4. Botões de ação para registo de assiduidade

O registo segue uma lógica sequencial e cronológica e que deve ser respeitada, ou seja, o primeiro registo deve ser “Entrada” e o último deve ser “Saída”. A não aplicação desta regra origina registos inválidos.

Devem ser registados os quatro períodos salvo algumas exceções, de acordo com autorização superior.

A picagem deve ser feita sem saltar períodos. Ou seja, não é possível fazer a picagem de Saída – 1º Período sem primeiro fazer a picagem Entrada – 1º Período e assim sucessivamente. O erro que pode surgir é o apresentado na figura 5.

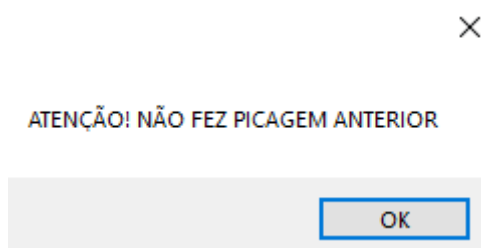


Figura 5. Mensagem de erro por falta de picagem anterior

Nestes casos deve ser registado apenas o 1º período, com uma picagem de “Entrada” e outra de “Saída”.

Pausa para almoço:

Embora a pausa para almoço seja obrigatoriamente de 1 hora é possível realizar o registo de Entrada - 2º Período depois de passados 50 minutos do registo anterior (Saída – 1º Período), no entanto o sistema adiciona automaticamente os minutos restantes até perfazer 1 hora. Se for tentado um registo antes de passados 50 minutos relativamente ao registo anterior (Saída – 1º Período) aparecerá a caixa de mensagem com a informação mostrada na figura 6.

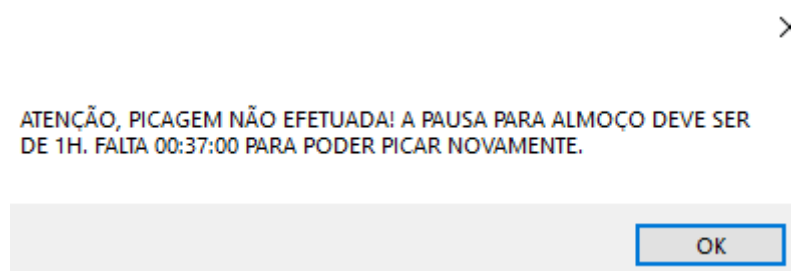


Figura 6. Pausa inferior a uma hora - mensagem de erro

Após pressionado o botão de registo pretendido (1 clique apenas), deve esperar pela mensagem que aparece na figura 7.

NOTA: O processo entre carregar no botão e o aparecimento da mensagem supra indicada pode demorar alguns segundos, pelo que não deve, de maneira nenhuma, carregar noutros botões enquanto esta não aparecer. Pode haver situações, muito raras, que podem demorar até 5 minutos. Deve-se sempre esperar pela caixa de mensagem de sucesso ou insucesso para encerrar a aplicação. Embora possa demorar algum tempo a aparecer as referidas caixas a hora que é gravada é a imediatamente adquirida após carregar no respetivo botão.

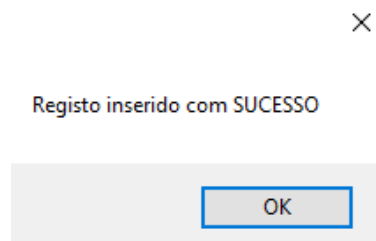


Figura 5. Mensagem com a indicação de sucesso no registo de assiduidade

É possível observar no lado direito da aplicação, como mostra a figura 8, os dados que foram registados.

Figura 6. Dados de assiduidade registados pela aplicação

2.3. Registo de observações

Uma vez que o quotidiano representa uma série de variáveis e nada é imutável, esta aplicação contempla também a introdução de observações. É possível registar manualmente horas de entrada e saída no caso de avaria do sistema, esquecimento de picagem, etc., e, ainda outras observações como “Consultas/Exames”, “Formação”, etc.

O registo de observações pode ser feito na área representada pela figura 9.

Figura 7. Área da aplicação dedicada ao registo de observações

Deve preencher o dia, o mês e o ano, assim como a correspondente observação antes de carregar no botão “Adicionar”.

Se não o fizer aparecerá uma mensagem de erro, figura 10.

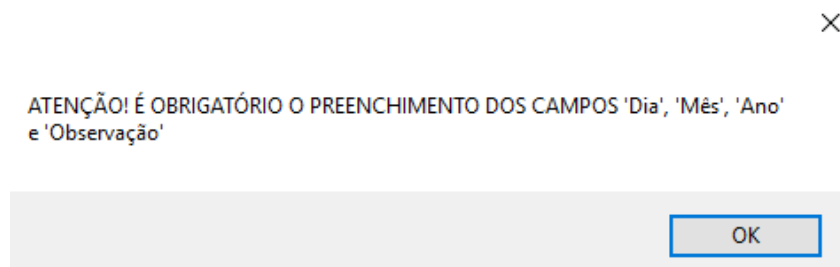


Figura 8. Mensagem de erro por não introdução dos campos obrigatórios, “Dia”, “Mês”, “Ano” e “Observação”

NOTA: Se o ficheiro (Mod 75-A-DGAV Registo de Assiduidade) referente ao mês pretendido não existir a aplicação irá devolver um erro conforme figura 11.

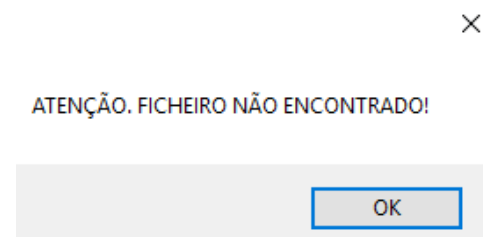


Figura 9. Mensagem de alerta para ficheiro não encontrado

No caso de se querer registar um erro de entrada ou saída deve-se seleccionar nas observações o período do dia relacionado como exemplificado pela figura 12. Ou seja, “Erro Entrada – 1º Período”, “Erro Saída – 1º Período”, “Erro Entrada – 2º Período” e “Erro Saída – 2º Período” conformem pretendido.



Figura 10. Algumas das opções presentes na caixa de selecção "Observações"

Existem outras observações com este comportamento em que é possível seleccionar manualmente a hora dos diversos períodos como por exemplo, os “Serviços externos”. Ao seleccionar este tipo de observações irão surgir dois campos para introdução, um relativo às horas e outro relativo aos minutos. Estes campos também são de preenchimento obrigatório. Caso contrário irá aparecer uma mensagem como demonstrado pela figura 13.

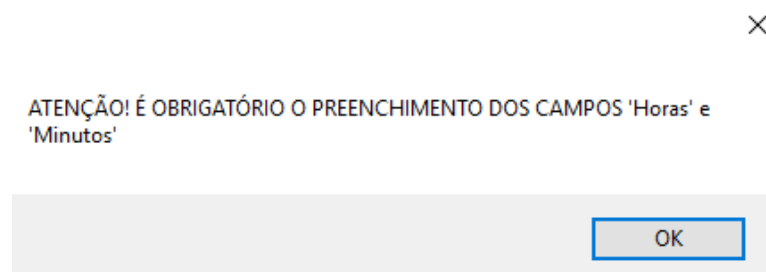


Figura 11. Mensagem de erro por não introdução dos campos obrigatórios, “Horas” e “Minutos”

Tal como acontece no registo automático de assiduidade do ponto 2.2. as picagens não devem saltar períodos e devem respeitar a hora de pausa para almoço (uma hora). Neste caso existe mais uma validação que é a cronológica, ou seja, a hora da picagem que se pretende inserir não deve ser inferior à picagem feita imediatamente antes. O erro que aparece se não for respeitado é o demonstrado na figura 14.

ATENÇÃO! PICAGEM NÃO EFECTUADA! HORA INFERIOR OU IGUAL À PICAGEM ANTERIOR.

OK

Figura 12. Erro ao inserir uma picagem com um a hora inferior à picagem anterior

No caso de seleccionar uma observação do tipo “Férias” ou “Feriado Municipal” que pressuponha o dia completo, são retiradas as sete horas previstas para aquele dia.

Se for seleccionada uma observação que contenha a expressão “Meio dia” só serão retiradas três horas e trinta minutos às sete previstas para aquele dia.

NOTA: Só é possível adicionar uma observação por dia, exceto nos casos em que há introdução manual dos vários períodos como já referenciado (ex. Erros de Entrada/Saída, Serviço Externo, etc). Por exemplo, se em determinado dia tiver uma “Formação” da parte da manhã e se esquecer também de realizar um registo de entrada/saída é possível ter essas duas observações. No entanto, nestes casos deve ser feito em **primeiro lugar a picagem manual e só depois o registo de observação “Formação”**. O não cumprimento destas condições resulta no erro referido na figura 13.

ATENÇÃO. JÁ EXISTE OBSERVAÇÃO

OK

Figura 13. Mensagem de aviso de observação já existente

Se todas as regras forem cumpridas, após carregar-se no botão “Adicionar” irá aparecer a mensagem de êxito, figura 14.

×

Registo inserido com SUCESSO

OK

Figura 14. Mensagem com a indicação de sucesso no registo de observações

2.3.1. Seguem, de seguida, todas as opções possíveis de selecionar no campo “Observações, assim como um breve descritivo do seu efeito e ainda, se relevante, um exemplo prático.

2.3.2.

Observações que justificam todo o dia:

Acidente de trabalho - Todo o dia
 Assistência a familiares < 10 - Todo o dia
 Assistência a familiares > 10 - Todo o dia
 Atestado Médico – Doença
 Atividade sindical - Todo o dia
 Candidatos a eleições - Todo o dia
 Casamento
 Consulta médica/exames - Familiar - Todo o dia
 Consulta médica/exames - Próprio - Todo o dia
 Cumprimento de obrigações legais
 Doação de sangue/Socorrismo - Todo o dia
 Falecimento de familiar
 Falta injustificada - Todo o dia
 Falta por conta de período de férias - Corrente - Todo o dia
 Falta por conta de período de férias - Seguinte - Todo o dia
 Feriado Municipal
 Férias
 Folga por compensação - Todo o dia
 Formação - Todo o dia
 Greve
 Isolamento profilático
 Parentalidade
 Provas de procedimento concursal - Todo o dia
 Tolerância de ponto
 Trabalhador - Estudante - Todo o dia

NOTA: Só é possível adicionar uma destas opções para cada dia. A adição destas observações não eliminam a necessidade de apresentação de justificação conforme procedimento definido para cada caso. **Se for possível ao utilizador registar os seus atestados, alerta-se que deve**

registá-los também nos fins de semana e feriados nacionais. Se calhar numa tolerância de ponto ou feriado municipal, apenas devem lançar o atestado e não os dois últimos referenciados imediatamente antes.

Observações que justificam meio dia:

Acidente de trabalho - Meio dia
 Atividade sindical - Meio dia
 Candidatos a eleições - Meio dia
 Consulta médica/exames - Familiar - Meio dia
 Consulta médica/exames - Próprio - Meio dia
 Desloc. escola de filho p/reunião - Meio dia
 Doação de sangue/Socorrismo - Meio dia
 Falta injustificada - Meio dia
 Falta por conta de período de férias - Corrente - Meio dia
 Falta por conta de período de férias - Seguinte - Meio dia
 Formação - Meio dia
 Provas de procedimento concursal - Meio dia

NOTA: Só é possível adicionar duas destas opções para cada dia. A adição destas observações não elimina a necessidade de apresentação de justificação conforme procedimento definido para cada caso. Nestes casos, a picagem automática ou manual só deverá ser feita para o 1º Período.

Observações que apenas justificam um breve período de ausência:

Assistência a familiares < 10 - Período de Ausência
 Assistência a familiares > 10 - Período de Ausência
 Consulta médica/exames - Familiar - Período de Ausência
 Consulta médica/exames - Próprio - Período de Ausência
 Desloc. escola de filho p/reunião - Período de Ausência
 Serviço externo - Período de Ausência
 Trabalhador - Estudante - Período de Ausência

NOTA: A partir da versão 1.5 do sistema é possível descontar/justificar o tempo de período de ausência através da selecção ou não da opção “Descontar Tempo”, figura 15. Será demonstrada esta funcionalidade nos exemplos abaixo.

Figura 15. Formulário para introdução do período de ausência

NOTAS/EXEMPLOS:

Suponhamos que determinado funcionário (cujo horário é o de atendimento) entrou ao serviço normalmente às 09:00H, teve uma consulta médica às 10:00H e conseguiu regressar ao serviço às 11:30H. O procedimento a adotar é:

- Picar a entrada ao serviço de forma automática;
- Quando regressar da consulta, para a data corrente, escolher a opção “Consulta médica/exames - Próprio - Período de Ausência” e carregar em “Adicionar”;
- Irá aparecer uma nova janela onde deverá introduzir o período de ausência, neste caso das 10:00H às 11:30H, e por fim confirmar no botão “OK” (não colocar o visto na caixa “Descontar Tempo”);
- Efetuar as restantes picagens do dia normalmente.

Suponhamos que determinado funcionário (cujo horário é o de atendimento) teve uma consulta médica às 08:30, não conseguindo picar a entrada ao serviço, e regressou ao serviço às 11:30H. O procedimento a adotar é:

- Quando regressar da consulta, para a data corrente, picar a entrada ao serviço de forma automática;
- De seguida, para a data corrente, escolher a opção “Consulta médica/exames - Próprio - Período de Ausência” e carregar em “Adicionar”;
- Irá aparecer uma nova janela onde deverá introduzir o período de ausência, neste caso das 09:00H (hora de início de atendimento) às 11:30H, colocar um visto na caixa “Descontar Tempo” e por fim confirmar no botão “OK”;
- Efetuar as restantes picagens do dia normalmente.

Suponhamos que determinado funcionário entrou ao serviço, picando normalmente, teve de sair em Serviço Externo às 10:00H e conseguiu regressar ao serviço às 11:30H. O procedimento a adotar é:

- Picar a entrada ao serviço de forma automática;
- Quando regressar do Serviço Externo, para a data corrente, escolher a opção “Serviço externo - Período de Ausência e carregar em “Adicionar”;
- Irá aparecer uma nova janela onde deverá introduzir o período de ausência, neste caso das 10:00H às 11:30H, e por fim confirmar no botão “OK” (não colocar o visto na caixa “Descontar Tempo”);
- Efectuar as restantes picagens do dia normalmente.

Suponhamos que determinado funcionário teve de sair em Serviço Externo às 08:00 e regressou ao serviço às 15:30H. Falhou, portanto, três picagens. O procedimento a adotar é:

- Quando regressar do Serviço Externo, para a data corrente, escolher a opção “Serviço externo - Entrada 1º Período” selecionar 08:00H e carregar em “Adicionar”;
- Para efetuar a outra picagem em falta, para a data corrente, escolher a opção “Serviço externo - Saída 1º Período” selecionar a hora que saiu para almoço durante o período de Serviço Externo e carregar em “Adicionar”;
- Para efetuar a terceira picagem, para a data corrente, escolher a opção “Serviço externo - Entrada 2º Período” selecionar a hora que voltou de almoço durante o período de Serviço Externo e carregar em “Adicionar”;
- De seguida, para a data corrente, escolher a opção “Serviço externo - Período de Ausência” e carregar em “Adicionar”;
- Irá aparecer uma nova janela onde deverá introduzir o período de ausência, neste caso das 08:00H às 15:30H, e por fim confirmar no botão “OK”.
- Efetuar as restantes picagens do dia normalmente.

2.4. Outras validações:

2.4.1. Expiração da validade do CC

O sistema verifica se a validade do CC do utilizador já expirou. Se for esse o caso o sistema avisa o utilizador que poderá continuar a registar as entradas/saídas com o CC expirador por mais 15 dias corridos relativamente à data de expiração de forma a dar tempo para que seja pedida uma renovação de CC. A mensagem de aviso será a relatada na figura 16.

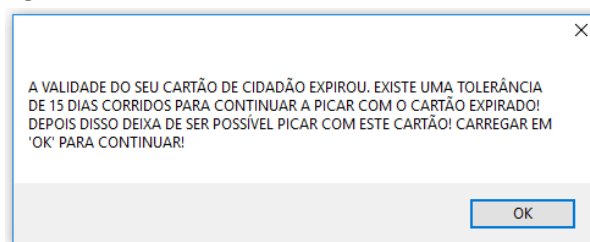


Figura 16. Caixa de mensagem de alerta sobre cartão expirado;

Se houver tentativa de registo de assiduidade para além da tolerância de 15 dias após expiração do CC o sistema não permite apresentando a mensagem que aparece na figura 17.

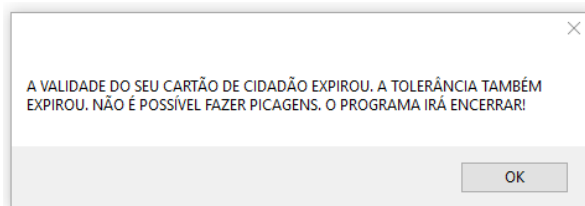


Figura 17. Caixa de mensagem de alerta de tolerância expirada;

2.4.2. Validação de observações realizadas após dia do fecho do mês decretado em procedimento

Verifica se o utilizador está a tentar inserir registo manual depois de passado o prazo de fecho do mês. Se for o caso o sistema alerta o utilizador de que o registo não foi efectuado devido a esta validação, conforme ilustra a figura 18.

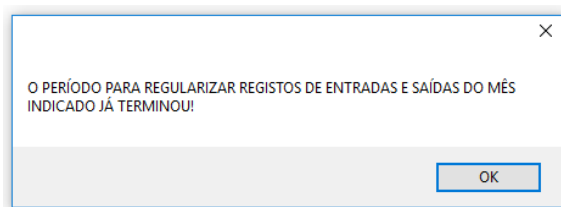


Figura 18. Caixa de mensagem com erro de validação relativa ao registo de observações para lá do fecho do mês

2.5. Visualização de registos

2.5.1. O utilizador pode a qualquer momento visualizar os seus registos de assiduidade, mês a mês. A área referente à visualização de registos é a explanada na figura 19.

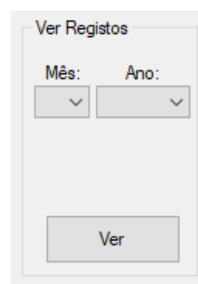


Figura 19. Área da aplicação reservada à visualização de registos

Para proceder à visualização deve seleccionar o “Mês” e “Ano” pretendidos e carregar de seguida no botão “Ver”.

De forma semelhante ao que acontece com as funcionalidades já acima descritas, os campos “Mês” e “Ano” são de preenchimento obrigatório e não introdução dos campos resulta numa mensagem de erro, figura 20.

×

ATENÇÃO! É OBRIGATÓRIO O PREENCHIMENTO DOS CAMPOS 'Mês' e 'Ano'

OK

Figura 20. Mensagem de erro por não introdução dos campos obrigatórios, “Mês” e “Ano”

Será gerado um ficheiro “*.PDF” que abrirá automaticamente.

NOTA: Se o ficheiro (Mod 75-A-DGAV Registo de Assiduidade) referente ao mês pretendido não existir a aplicação irá devolver um erro conforme figura 21.

×

ATENÇÃO. FICHEIRO NÃO ENCONTRADO!

OK

Figura 21. Mensagem de alerta para ficheiro não encontrado

3. Resolução de problemas – Contactos

É sempre preferível o contato por correio electrónico com o maior número de detalhes possível para o problema encontrado (descrição pormenorizada, imagens, etc...).

Eng.º Eduardo Brás – DGAV:

Telefone: 253 783 000

Correio electrónico: eduardo.bras@dgav.pt

4. Histórico

Versão	Autor	Descrição	Data
1.0	Eduardo Brás	Elaboração do Manual	26-12-2019
1.1	Eduardo Brás	Acrescentadas as justificações “Descanso compensatório”, “Trabalhador-Estudante” e “Feriado Municipal”	03-01-2020
1.2	Eduardo Brás	Verificação da validade do Cartão de Cidadão (CC), não deixando a aplicação fazer registos de entradas/saídas com CC expirado.	07-01-2020
1.3	Eduardo Brás	Adição da possibilidade de seleccionar horário nocturno.	29-05-2019
1.4	Eduardo Brás	Possibilidade de picar num período de 10 minutos antes de perfazer 1H para almoço; Adicionada tolerância de 15 dias para se poder registar entradas/saídas com o CC expirado; Resolução de erro que não permitia a adição de serviço externo em dia com justificação de meio-dia de folga por compensação.	10-01-2020
1.5	Eduardo Brás	Adição da possibilidade da aplicação reconhecer diferentes modalidades de horário; Adição da possibilidade de justificar tempo com as opções “... - Período de Ausência”; Valida se o utilizador está a tentar inserir algum registo manual depois de passado o prazo de fecho do mês; (>dia 4 do mês seguinte).	16-06-2020