



## Telemetria em Máquinas de Venda de Tabaco / Caso Prático ? Vodafone WinVending

TELMO EMANUEL SILVA RODRIGUES

novembro de 2017

# **Telemetria em Máquinas de Venda de Tabaco**

## **Caso Prático – Vodafone WinVending**

**Telmo Emanuel Silva Rodrigues**

**Dissertação para obtenção do Grau de Mestre em  
Engenharia Informática, Área de Especialização em  
Gestão do Conhecimento e Decisão**

**Orientador: Paulo Maio**

**Júri:**

Presidente:

[Nome do Presidente, Categoria, Escola]

Vogais:

[Nome do Vogal1, Categoria, Escola]

[Nome do Vogal2, Categoria, Escola] (até 4 vogais)

Porto, novembro de 2017



# Resumo

No contexto de mudança que o setor de Vending atravessa e consciente da nova realidade do mercado e do valor que adiciona às empresas gestoras de máquinas de tabaco, a ABS Informática, empresa sobre a qual incidirá o foco desta dissertação, alia uma componente de telemetria à sua atual solução de gestão de máquinas de Vending. Com esse objetivo, estabelece com a Vodafone, reconhecida operadora de telecomunicações e a KTS, produtora de dispositivos eletrônicos, um acordo de colaboração, com o objetivo de reunir o conhecimento especializado de cada parceiro, na respetiva área de atuação. Dessa parceria resulta o desenvolvimento de uma nova solução (Vodafone WinVending), alojada no cliente, em ambiente *desktop* e assente no sistema gestor de base de dados *SQL Server*.

Como complemento à solução de telemetria, pretende-se colmatar uma necessidade de segurança relacionada com frequentes casos de atos de roubo e vandalismo às máquinas de tabaco, causadores de prejuízos e transtornos no negócio do setor. Conhecedores dessa realidade, a ABS Informática propõe, aos seus clientes, uma solução dedicada, exclusivamente, à deteção e visualização de alarmes com características suspeitas, denominada: WinGuard.

O primeiro ponto deste projeto consiste na introdução e contextualização do caso de estudo. Segue-se uma análise e levantamento de requisitos (funcionais e não funcionais).

Posteriormente, apresenta-se o desenho da solução, a arquitetura e a sua implementação. Idealiza-se uma arquitetura, segundo o padrão três camadas: apresentação, lógica de negócio e acesso a dados. Elege-se, como modo de comunicação entre camadas, o padrão: entidades de negócio. A comunicação com sistemas externos estabelece-se pela implementação dos padrões: *Service Interface* e *Service Gateway*. Tais padrões partilham, entre si, uma estrutura comum que segue o padrão: objetos de transferência de dados.

Na apresentação de cada uma das camadas, salientam-se técnicas e pormenores de implementação. Fundamentam-se e argumentam-se as decisões tomadas, com base no caso em concreto e nas alternativas possíveis.

Finalmente, realizam-se testes de verificação da solução e experiências com utilizadores finais, de forma a validar a solução e o seu impacto nos vários *stakeholders*.

**Palavras-chave:** Telemetria, M2M, Máquinas *Vending*, Tabaco, Sensores, Comunicação, Alarmes, GPS, Inovação, Análise de Valor, Inovação, Vandalismo, API, *Middleware*, Filas de Mensagens, Arquitetura

# Abstract

In the context of the change that the Vending sector is going through and aware of the new reality of the market and of the value it adds to tobacco machinery management companies, ABS Informática, the company in which this dissertation centers around, combines a telemetry component with its current Vending machine management solution. To this end, it establishes with Vodafone, a renowned telecommunications operator and KTS, a provider of electronic devices, a collaboration agreement with the objective of bringing together the specialized knowledge of each partner in its respective area of activity. This partnership resulted in the development of a new solution (Vodafone WinVending), hosted on the client in a desktop environment and based on the SQL Server database manager system.

As a complement to the telemetry solution, it is intended to fill a need for security related to frequent cases of theft and vandalism to tobacco machines, causing damage and inconvenience in the business sector. Knowing this reality, ABS Informática proposes to its customers a solution dedicated exclusively to the detection and visualization of alarms with suspicious characteristics, named: WinGuard.

The first point of this project is the introduction and contextualization of the case study. The following is an analysis and survey of requirements (functional and non-functional).

Afterwards, the design solution, the architecture and its implementation are presented. An architecture is idealized according to the three layers pattern: presentation, business logic and data access. It is chosen as the mode of communication between layers, the pattern: business entities. Communication with external systems is established by the implementation of the patterns: Service Interface and Service Gateway. These patterns share a common structure that follows the pattern: data transfer objects.

The presentation of each of the layers emphasizes techniques and details of implementation. The decisions taken on the basis of this particular case and the possible alternatives are based and argued.

Finally, solution verification tests and end-user experiences are carried out in order to validate the solution and its impact on the various stakeholders.

**Keywords:**

Telemetry, M2M, Vending Machines, Tobacco, Sensors, Communication, Alarms, GPS, Innovation, Value Analysis, Innovation, Vandalism, API, Middleware, Message Queues, Architecture

# Agradecimentos

Aproveito este espaço para agradecer a todas as pessoas e entidades que me apoiaram, incentivaram e possibilitaram a aquisição de conhecimento que tenho adquirido e que tornaram possível a realização deste projeto.

Assim, agradeço:

À ABS Informática, na pessoa do Eng.º Paulo Almeida, pela oportunidade, confiança e autonomia depositada no desenvolvimento do projeto.

Ao Instituto Superior de Engenharia do Porto, pela inovação, organização, rigor e seriedade que o tornam uma instituição reconhecida e de prestígio nacional e internacional, ao qual me orgulho de pretender. Em especial, aos docentes do Departamento de Engenharia Informática, pelas técnicas e metodologias de ensino que resultam numa formação académica rica, com uma forte integração no mercado de trabalho, cujos ensinamentos procurei absorver, ao longo dos anos, e colocá-los em prática, neste trabalho.

Ao professor Paulo Maio, por ter aceite ser meu orientador, pela paciência, empenho, disponibilidade e sentido prático com que me orientou e pela visão crítica e construtiva inculcada na revisão e estrutura da dissertação.

Ao parceiro KTS, nas pessoas do Eng.º Filipe Biléu, Eng.º Pedro Maria e Dr.º Renato Caldeira, pela disponibilidade e troca de ideias e conhecimento, que permite uma evolução contínua da solução.

Ao parceiro Vodafone, na pessoa do Dr.º Nuno Barrau, pela defesa e apresentação do projeto junto dos clientes e pelo *feedback* e elo de ligação entre as partes envolvidas.

Aos colegas de trabalho, em especial, ao Dr.º Marcelo Carvalho e Bernardo Simões pela defesa, suporte e sugestões que enriqueceram a solução, ao Dr.º Nelson Gomes pela troca de ideias de integração com a aplicação móvel, ao Eng.º Renato Silva pela implementação do WinGuard, ao Eng.º Ricardo Matos, atual programador do WinVending BackOffice, pelo rigor, dedicação e iniciativa dadas à continuidade da implementação do projeto e ao Eng.º Hermano Costa, pela implementação do conceito de migrações na API do WinVending e pelo espírito simplificador, que torna a experiência de manutenção mais ágil.

A todos aqueles que dedicam o seu tempo e partilham o seu conhecimento e experiência na Internet, os quais me ajudaram a resolver problemas técnicos, a selecionar as ferramentas e tecnologias de trabalho e a ponderar decisões de desenho e arquitetura.

Aos clientes da solução, pela compreensão do, nem sempre ótimo, funcionamento das aplicações e pelas críticas e sugestões de requisitos e usabilidade, que tornam a solução mais robusta e próxima dos seus utilizadores.

Ao colega de mestrado Hugo Pinto, pela partilha de conhecimentos, experiência e motivação, que nos momentos de maior preguiça me ajudaram a desbloquear e a retomar o escrito.

À Professora Selda Soares pela incansável disponibilidade na revisão ortográfica e gramatical do projeto.

Ao meu pai, irmã, família e amigos, pelo constante apoio que me deram, pelo suporte que são na minha vida e pelos momentos de lazer que desfruto com eles e que se tornam imprescindíveis para o meu bem estar e harmonia entre trabalho e diversão.

A todos, o meu muito obrigado.

# Índice

<b>1</b>	<b>Introdução</b>	<b>1</b>
1.1	Contexto Social e Tecnológico	1
1.2	Apresentação do projeto	2
1.3	Objetivos	5
1.4	Metodologia de Trabalho	5
1.5	Estrutura	6
<b>2</b>	<b>Contextualização</b>	<b>9</b>
2.1	Evolução da Solução	9
2.2	Entidades Envolvidas	11
2.2.1	ABS Informática	11
2.2.2	KTS	12
2.2.3	Vodafone	12
2.3	Solução Legada	13
2.4	Análise de Mercado	17
2.4.1	Clientes	17
2.4.2	Mercado	18
2.4.3	Processo Operacional	20
2.4.4	Concorrência	22
2.5	Soluções Concorrentes	22
2.5.1	GMBOS	22
2.5.2	UPVending	23
2.5.3	Vendon	23
2.5.4	Comparação dos Produtos	24
<b>3</b>	<b>Visão da Solução</b>	<b>27</b>
3.1	Abordagens Possíveis	27
3.1.1	Solução atual Vs nova	27
3.1.2	Alojamento dos dados	30
3.1.3	Ambiente de desenvolvimento	30
3.1.4	Notificação e comunicação de alarmes	31
3.2	Elementos do Sistema	31

3.2.1	Máquina de Vending .....	32
3.2.2	Módulo.....	34
3.2.3	Cartão SIM Global da Vodafone .....	35
3.2.4	GDSP .....	35
3.2.5	Middleware .....	35
3.2.6	WinVending.....	37
3.2.7	WinVending Mobile.....	37
3.2.8	WinGuard .....	38
3.3	Ferramentas .....	38
3.3.1	IDE & Linguagem de Programação.....	38
3.3.2	Sistema Gestor de Base de Dados .....	40
3.3.3	Prototipagem.....	40
3.3.4	Ferramentas UML.....	41
<b>4</b>	<b>Análise do Valor do Negócio.....</b>	<b>43</b>
4.1	Modelo de Desenvolvimento de Novo Conceito .....	43
4.1.1	Processo de inovação .....	44
4.1.2	Motor .....	45
4.1.3	Fatores de influência .....	45
4.1.4	Identificação das oportunidades.....	46
4.1.5	Análise de oportunidades .....	47
4.1.6	Geração de ideia .....	48
4.1.7	Seleção da ideia .....	49
4.1.8	Definição do conceito .....	49
4.2	Modelo de Negócio de Canvas.....	50
4.3	Benefícios e Sacrifícios .....	54
4.4	Proposta de Valor .....	55
4.5	Cenários de Negócio .....	57
4.6	Rede de Valor .....	58
4.7	Medição da Criação de Valor .....	59
4.7.1	Abordagem Prática do Método AHP .....	61
4.7.2	Definição de Prioridades .....	62
4.7.3	Análise de Consistência dos Julgamentos .....	65
4.7.4	Análise de Resultados .....	67
<b>5</b>	<b>Requisitos &amp; Análise .....</b>	<b>69</b>

5.1	Conceitos de Negócio .....	69
5.1.1	Comunicação .....	69
5.1.2	Programação .....	72
5.1.3	Ocorrência .....	73
5.1.4	Rota .....	74
5.1.5	Categorização de Alarmes .....	75
5.2	Atores .....	76
5.2.1	Utilizador .....	77
5.2.2	Administrador .....	77
5.2.3	Administrativo .....	77
5.2.4	WinVending Backoffice .....	77
5.2.5	WinVending Mobile .....	77
5.2.6	Middleware .....	78
5.2.7	ERP .....	78
5.2.1	Operador .....	78
5.2.2	WinGuard .....	78
5.3	Requisitos Funcionais .....	78
5.3.1	WinVending BackOffice .....	78
5.3.2	WinGuard .....	92
5.4	Requisitos Não Funcionais .....	96
5.4.1	Usabilidade (USB) .....	97
5.4.2	Confiabilidade (CNF) .....	98
5.4.3	Desempenho (DSP) .....	99
5.4.4	Suporte (SPR) .....	100
5.4.5	Desenho (DSN) .....	102
5.4.6	Implementação (MPL) .....	103
5.4.7	Interface (NTR) .....	104
<b>6</b>	<b>Desenho .....</b>	<b>107</b>
6.1	Visão Geral .....	107
6.2	Visão WinVending .....	109
6.3	Visão WinGuard .....	111
6.4	Diagramas conceptuais de classes .....	112
6.4.1	WinVending .....	112
6.4.1	WinGuard .....	116
6.5	Modelo de Dados .....	117

6.5.1	Regras & Práticas.....	117
6.5.2	WinVending.....	120
6.5.3	WinGuard .....	123
<b>7</b>	<b>Implementação.....</b>	<b>125</b>
7.1	Tecnologias.....	125
7.1.1	Serviços de Internet .....	125
7.1.2	Mapeamento relacional de objetos .....	127
7.1.3	Ferramentas de registo.....	130
7.1.4	Filas de Mensagens.....	132
7.1.5	Ferramentas de Integração Empresarial (EAI).....	135
7.2	Modelo Arquitetural.....	137
7.2.1	Camada de Apresentação .....	139
7.2.2	Camada de Regras de Negócio .....	142
7.2.3	Camada de Serviço de Dados.....	145
7.2.4	Objetos de Negócio .....	150
7.2.5	Objetos de Transferência de Dados.....	151
7.2.6	Comunicação Entre API's .....	152
7.2.7	Controllers / Interface .....	153
7.2.8	Gateway.....	155
7.3	Integração de Aplicações Empresariais.....	156
7.3.1	Middleware .....	157
7.3.2	Sistema BackOffice.....	158
7.3.3	Sistema Móvel.....	160
7.3.4	Sistema WinGuard.....	161
7.3.5	Sistema ERP .....	161
<b>8</b>	<b>Implantação.....</b>	<b>165</b>
8.1	Representação da Localização da Máquina .....	165
8.1.1	Mapa .....	166
8.1.2	Marcadores .....	168
8.1.3	Camadas .....	170
8.2	Instalação .....	171
8.2.1	Setup (InstallShield) .....	171
8.2.2	Publicação ( <i>ClickOnce</i> ).....	177
8.3	Gestão da Mudança .....	181
8.3.1	Arquitetura da aplicação.....	181

8.3.2	Atualização do modelo de dados .....	182
8.3.3	Atualização da aplicação .....	183
<b>9</b>	<b>Avaliação do Sistema .....</b>	<b>185</b>
9.1	Grandezas.....	185
9.1.1	Métricas da solução .....	185
9.1.2	Métricas de Utilização da Aplicação .....	186
9.1.3	Métricas de Suporte da Aplicação .....	187
9.2	Testes .....	188
9.2.1	Testes Unidade .....	188
9.2.2	Testes de Integração .....	189
9.2.3	Testes de Sistema .....	190
9.2.4	Testes de Aceitação .....	190
9.3	Testes de Hipóteses.....	191
9.3.1	Nível de Stock Circulante .....	191
9.3.2	Duração de Visita.....	193
9.4	Inquéritos .....	194
<b>10</b>	<b>Conclusões.....</b>	<b>195</b>
10.1	Objetivos Realizados.....	195
10.2	Limitações e Trabalho Futuro .....	199
10.3	Apreciação Final .....	202
<b>11</b>	<b>Anexos.....</b>	<b>211</b>
11.1	Casos de uso.....	211
11.1.1	<i>WinVending Backoffice</i> .....	211
11.1.2	<i>WinGuard</i> .....	292
11.2	Modelo de dados.....	296
11.3	Histórico de alterações.....	298
11.4	Amostras .....	299



# Lista de Figuras

Figura 1 – Diagrama de implantação da solução legada WinVending.....	13
Figura 2 – Caso de uso “Sistema <i>backoffice</i> legado” .....	14
Figura 3 – Caso de uso “Sistema móvel” .....	15
Figura 4 – Diagrama de atividades de abastecimento pelo método Auto Venda .....	21
Figura 5 – Elementos do sistema. ....	32
Figura 6 – Imagem frontal do módulo .....	34
Figura 7 – Visão geral do sistema Middleware .....	36
Figura 8 – Modelo de negócio CANVAS – Vodafone WinVending .....	53
Figura 9 – Diagrama de análise de rede de valor da solução Vodafone WinVending .....	59
Figura 10 – Diagrama de Seleção do Método de Notificação de Alarmes.....	62
Figura 11 – Diagrama de estados de comunicação.....	70
Figura 12 – Tipos de comunicação (ABS Informática 2014).....	72
Figura 13 – Diagrama de estados de programação da máquina .....	73
Figura 14 – Diagrama de atividade de prioridade do alarme .....	76
Figura 15 – Caso de uso do WinVending <i>Backoffice</i> .....	81
Figura 16 – Caso de uso de Análise Personalizável.....	83
Figura 17 – Caso de uso de Comunicar com Máquinas .....	83
Figura 18 – Caso de uso de Manter Tabelas .....	84
Figura 19 –Caso de uso de Manter Clientes.....	85
Figura 20 –Caso de uso de Manter Fornecedores .....	85
Figura 21 - Figura 20 –Caso de uso de Manter Fornecedores .....	86
Figura 22 – Caso de uso de Manter Agendamentos de Comunicação .....	86
Figura 23 - Caso de uso de Manter Produtos.....	87
Figura 24 – Caso de uso de Manter Máquinas.....	88
Figura 25 – Caso de Uso de Assistente de Programação da Máquina .....	89
Figura 26 – Manter Módulo .....	89
Figura 27 – Diagrama de estados – Estados Módulo.....	90
Figura 28 – Diagrama de Sequência de Notificação de Alarmes .....	91
Figura 29 – Diagrama de Sequência de Notificação de Operações .....	92
Figura 30 – Caso de uso “Central de Alarme” .....	94
Figura 31 – Visão geral do sistema Vodafone WinVending .....	109

Figura 32 – Diagrama de componentes do sistema WinVending .....	111
Figura 33 – Diagrama de componentes do sistema WinGuard.....	111
Figura 34 – Diagrama conceptual de classes WinVending .....	112
Figura 35 – Diagrama conceptual de classes da visão de máquina.....	113
Figura 36 – Diagrama conceptual de classes da Visão de <i>Comunicação</i> .....	114
Figura 37 – Diagrama conceptual de classes da visão de plano de rota .....	115
Figura 38 – Diagrama conceptual de classes do sistema WinGuard.....	116
Figura 39 – Modelo de dados do sistema WinVending.....	120
Figura 40 – Modelo de dados da visão da máquina.....	121
Figura 41 – Modelo de dados da visão da comunicação.....	122
Figura 42 – Modelo de dados da visão do plano de rota .....	123
Figura 43 – Modelo de dados WinGuard .....	124
Figura 44 – Comparação do desempenho das ferramentas de <i>log</i> (segundos / 100.000 eventos de registo) (Marsh 2016).....	131
Figura 45 – Arquitetura da Solução Vodafone WinVending .....	138
Figura 46 – Diagrama de pacotes UI.....	139
Figura 47 – Diagrama de classes UI .....	140
Figura 48 – Janela de manutenção de máquina.....	141
Figura 49 – Diagrama de pacotes BL - Aplicação.....	142
Figura 50 – Diagrama de pacotes BL - API.....	143
Figura 51 – Diagrama de classes BL.....	143
Figura 52 – Diagramas de pacotes DA.....	146
Figura 53 – Diagrama de classes DA.....	146
Figura 54 – Diagrama de pacotes comunicação API .....	153
Figura 55 – Diagrama de pacotes <i>Controller</i> .....	153
Figura 56 – Diagrama de classes <i>Controller</i> .....	154
Figura 57 – Diagrama de pacotes <i>gateway</i> .....	155
Figura 58 – Diagrama de classes <i>gateway</i> .....	156
Figura 59 – Mapa Google da aplicação WinVending <i>BackOffice</i> .....	167
Figura 60 – Marcador máquina .....	169
Figura 61 – Marcador rasto da máquina ( <i>zoom out</i> ).....	169
Figura 62 – Marcador rasto da máquina ( <i>zoom in</i> ) .....	170
Figura 63 – Histórico de localização da máquina ( <i>zoom out</i> ) .....	170

Figura 64 - Figura 63 – Histórico de localização da máquina ( <i>zoom in</i> ).....	171
Figura 65 – Criação de projeto <i>InstallShield Express</i> .....	173
Figura 66 – Dados gerais do projeto de instalação.....	173
Figura 67 – Características e sub-características do projeto de instalação.....	174
Figura 68 – Tipos de instalação.....	174
Figura 69 – Ficheiros de instalação.....	175
Figura 70 – Propriedades do ficheiro de saída primária de um serviço.....	175
Figura 71 – Propriedades do ficheiro de conteúdo.....	176
Figura 72 – Ficheiros e características de instalação.....	176
Figura 73 – Preparação do lançamento da instalação.....	177
Figura 74 – Compilação do projeto.....	177
Figura 75 – Assinar o manifesto <i>ClickOnce</i> .....	179
Figura 76 – Propriedades da publicação <i>ClickOnce</i> .....	179
Figura 77 – Política de verificação de atualização do <i>ClickOnce</i> .....	180
Figura 78 – Teste de normalidade das amostras de stock circulante ( <i>Shapiro-Wilk</i> ).....	192
Figura 79 – Teste “ <i>Wilcoxon</i> ” para avaliar as hipóteses de stock circulante.....	192
Figura 80 - Janela de Credenciação.....	212
Figura 81 – Janela de Empresa.....	213
Figura 82 - Janela de Painel de Bordo.....	215
Figura 83 – Janela de Localização de Máquinas.....	216
Figura 84 – Janela de Análises Personalizáveis.....	218
Figura 85 - Janela de Introdução de Parâmetros da Análise Personalizável.....	219
Figura 86 – Janela de Visualização de Análises Personalizáveis.....	220
Figura 87 – Janela Assistente de Comunicação Máquina – Etapa 1 – Selecionar Máquinas...	221
Figura 88 – Janela Assistente de Comunicação Máquina – Etapa 2 – Efetuar Pedidos de Comunicação.....	222
Figura 89 – Notificação de Violação de Raio de Segurança.....	223
Figura 90 – Janela de Tabela de Administração.....	225
Figura 91 – Diagrama conceptual de classes da visão de <i>cliente</i> .....	227
Figura 92 – Modelo de dados da visão do cliente.....	228
Figura 93 – Janela de Cliente.....	229
Figura 94 – Janela Cliente – Separador Estabelecimentos.....	230
Figura 95 – Janela Cliente de Separador Segmentação.....	231

Figura 96 – Janela Fornecedor – Separador Dados Principais.....	232
Figura 97 – Caso de uso de Separador Estabelecimentos Fornecedor .....	233
Figura 98 – Janela Fornecedor – Separador WebSAFT.....	234
Figura 99 – Diagrama conceptual de classes da visão de <i>modelo</i> .....	236
Figura 100 – Modelo de dados da visão do modelo .....	236
Figura 101 – Janela Modelo Máquina – Separador Dados Principais .....	237
Figura 102 – Janela Modelo Máquina – Separador Desenho.....	238
Figura 103 – Janela Modelo Máquina – Separador Máquinas.....	239
Figura 104 – Diagrama conceptual de classes da Visão de <i>Produto</i> .....	241
Figura 105 – Modelo de dados da visão de produto.....	242
Figura 106 – Janela Produto – Separador Principal.....	243
Figura 107 – Janela Produto – Separador Códigos Máquina .....	244
Figura 108 – Diagrama conceptual de classes da Visão de Periodicidade .....	245
Figura 109 – Janela Periodicidade .....	245
Figura 110 – Diagrama conceptual de classes da Visão de Rota.....	247
Figura 111 – Modelo de dados visão rota .....	248
Figura 112 – Janela Rota.....	248
Figura 113 – Diagrama conceptual de classes da Visão de Agendamento .....	250
Figura 114 – Modelo de dados da visão de agendamento .....	250
Figura 115 – Janela Agendamento – Separador Periodicidade.....	251
Figura 116 – Janela Agendamento – Separador Alvo.....	252
Figura 117 – Janela Agendamento – Separador Histórico .....	254
Figura 118 – Janela Agendamento – Separador Parâmetros .....	255
Figura 119 – Janela Máquina – Separador Visão Geral.....	256
Figura 120 – Janela Máquina – Separador Dados Principais.....	260
Figura 121 – Janela Máquina – Separador Visitas.....	262
Figura 122 – Janela Visita .....	264
Figura 123 – Janela Máquina – Separador Seleções / Canais .....	265
Figura 124 – Janela Máquina – Separador Facing.....	266
Figura 125 – Janela Máquina – Separador Histórico Programação .....	267
Figura 126 – Janela Máquina – Separador Comunicações.....	269
Figura 127 – Janela Máquina – Separador Ocorrências.....	270
Figura 128 – Janela Máquina – Separador Localização .....	272

Figura 129 – Janela de Assistente de Programação Máquina – Selecionar Método de Atribuição de Seleção & Canais .....	273
Figura 130 - Janela de Assistente de Programação Máquina – Copiar Seleções & Canais de Outra Máquina .....	274
Figura 131 – Janela de Assistente de Programação Máquina – Manter Seleções & Canais – Visão Tabular .....	276
Figura 132 – Janela de Assistente de Programação Máquina – Manter Seleções & Canais – Visão Facing .....	278
Figura 133 – Janela de Configuração de Produto .....	279
Figura 134 - Janela de Assistente de Programação Máquina – Aplicar Configuração .....	280
Figura 135 – Janela de Assistente de Programação Máquina – Progresso de Configuração ..	281
Figura 136 – Janela de Assistente de Programação Máquina – Programar Máquina .....	283
Figura 137 – Diagrama conceptual de classes da Visão de Programação Máquina .....	284
Figura 138 – Modelo de dados visão programação da máquina .....	285
Figura 139 – Diagrama conceptual de classes da visão de módulo .....	286
Figura 140 – Modelo de dados da visão do módulo .....	287
Figura 141 – Janela de Módulo .....	287
Figura 142 – Janela de Módulo – Atribuir Máquina.....	288
Figura 143 – Janela de Módulo – Atribuir Módulo .....	289
Figura 144 – Janela de Módulo – Histórico.....	290
Figura 145 – Janela Principal WinGuard (Visão Geral).....	293
Figura 146 – Janela Pesquisa Personalizada WinGuard .....	294
Figura 147 - Janela Principal WinGuard (Visão Máquina).....	295



# Lista de Tabelas

Tabela 1 – Resumo mercado Vending em Portugal (2013) .....	19
Tabela 2 – Resumo mercado Vending em Espanha (2014).....	20
Tabela 3 – Comparação das soluções WinVending com a concorrência .....	25
Tabela 4 – Benefícios Telemetria .....	55
Tabela 5 – Níveis de importância de comparações binárias (Nicola 2016) .....	62
Tabela 6 – Comparação paritária entre critérios .....	63
Tabela 7 – Matriz Normalizada e Vetor de Prioridade Relativa dos Critérios.....	63
Tabela 8 – Comparação paritária dos critérios .....	64
Tabela 9 – Matriz Normalizada para os critérios e Prioridade Local .....	65
Tabela 10 – Índice de Inconsistência Aleatória (Marins et al. 2009) .....	66
Tabela 11 – Vetor Ax dos Fatores .....	67
Tabela 12 – Índices de Consistência e Razões de Consistência .....	67
Tabela 13 – Matriz classificação das alternativas e prioridade global .....	68
Tabela 14 – Requisitos não funcionais de “usabilidade” do WinVending <i>Backoffice</i> .....	97
Tabela 15 – Requisitos não funcionais de “usabilidade” do <i>WinGuard</i> .....	98
Tabela 16 – Requisitos não funcionais de “confiabilidade” do WinVending <i>Backoffice</i> .....	98
Tabela 17 – Requisitos não funcionais de “usabilidade” do <i>WinGuard</i> .....	99
Tabela 18 – Requisitos não funcionais de “desempenho” do WinVending <i>Backoffice</i> .....	99
Tabela 19 – Requisitos não funcionais de “desempenho” do <i>WinGuard</i> .....	100
Tabela 20 – Requisitos não funcionais de “suporte” do WinVending <i>Backoffice</i> .....	101
Tabela 21 – Requisitos não funcionais de “suporte” do <i>WinGuard</i> .....	101
Tabela 22 – Requisitos não funcionais de “desenho” do WinVending <i>Backoffice</i> .....	102
Tabela 23 – Requisitos não funcionais de “desenho” do <i>WinGuard</i> .....	102
Tabela 24 – Requisitos não funcionais de “implementação” do WinVending <i>Backoffice</i> .....	103
Tabela 25 – Requisitos não funcionais de “implementação” do <i>WinGuard</i> .....	104
Tabela 26 – Requisitos não funcionais de “interface” do WinVending <i>Backoffice</i> .....	105
Tabela 27 – Requisitos não funcionais de “interface” do <i>WinGuard</i> .....	105
Tabela 28 – Comparação entre ORM’s (dados de novembro de 2013).....	129
Tabela 29 – Comparação entre ferramentas de registo (Gurock Software GmbH 2017) .....	132
Tabela 30 – Comparação de características entre plataformas de integração (Aqeel 2015) ..	136
Tabela 31 – Caso de uso de Autenticar .....	211

Tabela 32 – Caso de uso de Selecionar Empresa .....	212
Tabela 33 – Caso de uso de Consultar Painel de Bordo .....	213
Tabela 34 – Caso de uso de Consultar Localização de Máquinas .....	215
Tabela 35 – Caso de uso de Manter Análises Personalizáveis .....	216
Tabela 36 – Caso de uso de Introduzir Parâmetros da Análise Personalizável .....	218
Tabela 37 – Caso de uso de Visualizar Análise Personalizável .....	219
Tabela 38 – Caso de uso de Comunicar com Máquinas .....	220
Tabela 39 – Caso de uso de Selecionar Máquina a Comunicar .....	221
Tabela 40 – Caso de uso de Executar Pedidos de Comunicação .....	222
Tabela 41 – Caso de uso de Notificar Ocorrências .....	223
Tabela 42 – Caso de uso de Consultar Tabelas .....	224
Tabela 43 – Caso de uso de Manter Clientes .....	225
Tabela 44 – Caso de uso de Manter Estabelecimentos Cliente .....	229
Tabela 45 – Caso de uso de Manter Segmentação do Cliente .....	230
Tabela 46 – Caso de uso de Manter Fornecedores .....	232
Tabela 47 – Caso de uso de Manter Estabelecimentos Fornecedor .....	233
Tabela 48 – Caso de uso de Manter WebSAFT .....	233
Tabela 49 – Caso de uso de Manter Modelos Máquina .....	234
Tabela 50 – Caso de uso de Consultar <i>Facing</i> Modelo .....	237
Tabela 51 – Caso de uso de Consultar Máquinas Associadas ao Modelo .....	238
Tabela 52 – Caso de uso de Manter Produtos .....	239
Tabela 53 – Caso de uso de Manter Códigos de Produto Máquina .....	243
Tabela 54 – Caso de uso de Manter Periodicidades .....	244
Tabela 55 – Caso de uso de Manter Rotas .....	245
Tabela 56 – Caso de uso de Manter Agendamentos Comunicações .....	248
Tabela 57 – Caso de uso de Manter Alvos de Agendamento .....	251
Tabela 58 – Caso de uso de Consultar Histórico de Agendamento .....	252
Tabela 59 – Caso de uso de Manter Parâmetros de Agendamento .....	254
Tabela 60 – Caso de uso de Consultar Visão Geral Máquina .....	255
Tabela 61 – Caso de uso de Teste da Máquina .....	257
Tabela 62 – Caso de uso de Manter Estado da Máquina .....	258
Tabela 63 – Caso de uso de Manter Dados Principais Máquina .....	259
Tabela 64 – Caso de uso de Manter Código de Operador .....	260

Tabela 65 – Caso de uso de Programar Mensagem Publicitária.....	261
Tabela 66 – Caso de uso de Consultar Visitas Máquinas .....	262
Tabela 67 – Caso de uso de Consultar Visita Máquina .....	263
Tabela 68 – Caso de uso de Manter Consulta Canal.....	264
Tabela 69 – Caso de uso de Consultar Seleções & Canais .....	265
Tabela 70 – Caso de uso de Consultar Histórico Programação.....	266
Tabela 71 – Caso de uso de Consultar Comunicações Máquina.....	267
Tabela 72 – Caso de uso de Consultar Ocorrências Máquina .....	269
Tabela 73 – Caso de uso de Consultar <i>Localização da Máquina</i> .....	270
Tabela 74 – Caso de uso de Programar Máquina Por Assistente .....	272
Tabela 75 – Caso de uso de Copiar Seleções & Canais de Outra Máquina.....	273
Tabela 76 – Caso de uso de Manter Seleções & Canais.....	274
Tabela 77 – Caso de uso de Manter Seleções & Canais – Vista Tabular .....	275
Tabela 78 – Caso de uso de Manter Seleções & Canais – Vista <i>Facing</i> .....	276
Tabela 79 – Caso de uso de Configurar Produto Por Assistente.....	278
Tabela 80 – Caso de uso de Manter Seleções & Canais – Aplicar Configuração .....	279
Tabela 81 – Caso de uso de Atualizar Configuração da Máquina .....	280
Tabela 82 – Caso de uso de Programar Máquina .....	281
Tabela 83 – Caso de uso de Manter Módulo .....	285
Tabela 84 – Caso de uso de Atribuir Máquina ao Módulo.....	287
Tabela 85 – Caso de uso de Atribuir Módulo à Máquina.....	288
Tabela 86 – Consultar Histórico Módulo.....	289
Tabela 87 – Caso de uso de Notificar Alarmes.....	290
Tabela 88 – Caso de uso de Notificar Operações.....	291
Tabela 89 – Caso de uso de Notificar Alarme .....	292
Tabela 90 – Caso de uso de Consultar Alarmes .....	292
Tabela 91 – Caso de uso de Consultar Alarmes Com Vista Personalizada.....	293
Tabela 92 – Caso de uso de Consultar Alarmes Prioritários .....	294
Tabela 93 – Caso de uso de Consultar Alarmes Informativos.....	294
Tabela 94 – Caso de uso de Consultar Alarmes Por Máquina .....	295
Tabela 95 – Caso de uso de Fechar Alarme .....	296
Tabela 96 – Abreviaturas de tabelas e campos .....	297
Tabela 97 – Histórico de alterações (Novas funcionalidades) .....	298

Tabela 98 – Histórico de alterações (Melhoramentos).....	298
Tabela 99 – Histórico de alterações (Correções).....	299
Tabela 100 – Amostra de stock circulante antes e depois da adoção da telemetria.....	299

# Acrónimos e Símbolos

**ADO** *ActiveX Data Objects*

**API** *Application Programming Interface*

**AHP** *Analytic Hierarchy Process*

**AT** *Autoridade Tributária*

**CRUD** *Create, Read, Update e Delete*

**BD** *Base de Dados*

**BDE** *Borland Database Engine*

**BL** *Business Logic*

**BO** *Business Objects*

**CLR** *Common Language Runtime*

**CSV** *Comma-separated values*

**DA** *Data Access*

**DLL** *Dynamic-link Library*

**DTO** *Data Transfer Object*

**EAI** *Enterprise Architecture Integration*

**EDIFACT** *Electronic Data Interchange for Administration, Commerce and Transport*

**EDI** *Electronic Data Interchange*

**ERP** *Enterprise Resource Planning*

**ESB** *Enterprise Service Bus*

**FFE** *Fuzzy Front End*

**FURPS** *Functional, Usability, Reliability, Performance, Supportability*

**GAC** *Global Assembly Cache*

**GDI** *Graphics Device Interface*

**GPS** *Global Positioning System*

**GSM** *Global System for Mobile Communications*

**GSDP** *Global Data Service Platform*

**GPRS** *Global Packet Radio Services*

**GUI** *Graphical User Interface*

**HTML** *HyperText Markup Language*

**IDE** *Integrated Development Environment*

**IC** *Índice Consistência*

**IIA** *Índice Inconsistência Aleatória*

**IoT** *Internet of Things*

**IP** *Internet Protocol*

**ISO** *International Organization for Standardization*

**JMS** *Java Message Service*

**M2M** *Machine to Machine*

**MS** *Microsoft*

**MSDN** *Microsoft Developer Network*

**MSMQ** *Microsoft Message Queuing*

**MSIL** *Microsoft Intermediate Language*

**NCD** *New Concept Development*

**NMS** *Apache .NET Messaging API*

**NPD** *New Product Development*

**ODBC** *Open Database Connectivity*

**ORM** *Object-Relacional Mapping*

**PC** *Personal Computer*

**PDA** *Personal Digital Assistance*

**POCO** *Plain old CLR objects*

**POJO** *Plain old Java objects*

**RAD** *Rapid Application Development*

**RC** *Rácio Consistência*

**SAFT-T** *Standard Audit File for Tax Purposes – Portuguese version*

**SIM** *Subscriber Identity Module*

**SMS** *Short Message Service*

**SOA** *Service-Oriented Architecture*

**SQL** *Structured Query Language*

**TCP** *Transmission Control Protocol*

**UI** *User Interface*

**UDP** *User Datagram Protocol*

**UML** *Unified Modelling Language*

**UTC** *Coordinated Universal Time*

**VB** *Visual Basic*

**XAML** *Extensible Application Markup Language*

**WPF** *Windows Presentation Foundation*

**XLS** *Excel Binary File Format*

# 1 Introdução

Neste capítulo, pretende-se enquadrar a dissertação na realidade do mercado das máquinas de Vending, esclarecer o seu âmbito e levantar um conjunto de questões que durante o documento são respondidas e fundamentadas, sintetizar os seus objetivos, descrever sumariamente a metodologia de trabalho adotada e, por fim, apresentar a estrutura do documento através de uma breve descrição do conteúdo, a abordar em cada capítulo.

## 1.1 Contexto Social e Tecnológico

Atualmente, as máquinas de venda automática, também vulgarmente conhecidas por máquinas de Vending, estão a sofrer uma transformação digital, impulsionadas pela tendência da conexão de dispositivos, através da Internet. A introdução do termo “Internet das Coisas”<sup>1</sup>, no mercado das máquinas de Vending, tem agilizado o processo de modernização das tradicionais máquinas, fornecendo, aos clientes, uma experiência mais cativante e interativa (Intelligent Vending Machines 2015). Assim, as máquinas de Vending têm evoluindo no sentido de, por um lado, oferecerem aos clientes (i.e. pessoas que interagem com as máquinas) uma experiência com uma maior interatividade, personalização e entretenimento e, por outro lado, permitem aos seus proprietários e às marcas dos produtos comercializados, recolherem dados

---

<sup>1</sup> Também conhecida pela abreviatura IoT do termo inglês “Internet of Things”.

estatísticos e usarem serviços na nuvem<sup>2</sup>, para alcançarem eficiência e redução de custos operacionais (Intelligent Vending Machines 2015).

O contexto atual de avanços tecnológicos do setor impulsiona o recurso à telemetria<sup>3</sup> via comunicação máquina a máquina<sup>4</sup>. Esta tecnologia proporciona, aos proprietários das máquinas, uma mudança nos seus processos de negócio. Evoluem de um modelo de visita periódica e fixa para um conceito baseado nas necessidades operacionais e de manutenção observadas em tempo real ou, através de comunicações realizadas à distância, a qualquer momento, garantidas pela característica sempre ativa das máquinas. Esta mudança de paradigma adiciona valor ao negócio por meio de uma gestão eficaz e eficiente do mesmo (Heinkele & Pousttchi 2004).

Nesta nova realidade das máquinas de Vending, nem tudo são vantagens e oportunidades, também se detetam ameaças, das quais resultam necessidades para os proprietários das máquinas. Uma das ameaças é a consecutiva onda de furtos e vandalismos de que as máquinas têm sido alvo, fora do horário de funcionamento dos estabelecimentos onde as mesmas estão localizadas (Market Publishers 2014).

Considera-se que o universo das máquinas de Vending vive num ambiente tecnológico de inovação, de amadurecimento e, por isso, propício à mudança.

## 1.2 Apresentação do projeto

Um dos princípios gerais propostos por David Hooker (“O Primeiro Princípio – A Razão Por que Tudo existe”), é que o sistema de *software* existe por uma razão: fornecer valor aos seus utilizadores (Hooker 1996). Para garantir esse princípio, tipicamente colocam-se duas questões: (i) qual é o valor da solução de *software*? e (ii) qual a proposta de valor para o negócio?

Este princípio é um dos pilares deste trabalho e todas as decisões são tomadas tendo-o em consideração. Desse pressuposto, vingam apenas as decisões que acrescentam valor ao sistema.

---

<sup>2</sup> Tradução para português do termo *cloud*

<sup>3</sup> Telemetria significa medição à distância de origem Grega *tele* significa remoto e *metron* significa medida

<sup>4</sup> Também conhecida pela abreviatura M2M do termo inglês “Machine to Machine”

No contexto de mudança que o setor de Vending atravessa e consciente da nova realidade do mercado e do valor que adiciona às empresas gestoras de máquinas de tabaco, a ABS Informática<sup>5</sup> (ABS Informática 2017), empresa sobre a qual incidirá o foco desta dissertação, procura aliar uma componente de telemetria à sua atual solução de gestão de máquinas de Vending. Para atingir esse fim, estabeleceu um acordo de colaboração com entidades experientes no mercado de comunicações, máquina a máquina, nomeadamente com a: Vodafone<sup>6</sup> (Vodafone 2017) na vertente de telecomunicações e com a KTS<sup>7</sup> (KTS 2017) no desenvolvimento de dispositivos eletrónicos.

Nesta dissertação, o objeto de estudo incide, principalmente, no acrescento à atual solução de Vending, denominada WinVending, da possibilidade de comunicar remotamente com as máquinas. Esta necessidade levanta a seguinte questão: estende-se o desenvolvimento da atual solução, no mesmo ambiente, ou constrói-se uma nova solução? Coloca-se em questão, neste ponto, o modo de disponibilização da solução: na nuvem para todos os clientes ou, nas infraestruturas individuais de cada cliente? Segue-se a questão da forma de apresentação da solução: em ambiente *web* ou *desktop*?

Como complemento à solução de telemetria, pretende-se colmatar a necessidade de segurança, motivado por casos de furto e vandalismo às máquinas de Vending, que ocorrem fora do horário dos estabelecimentos. A proposta de resolução dessa necessidade coloca as próximas questões: como gerir e a quem comunicar os alertas nesse horário? Por que meio devem os alarmes ser comunicados (email, SMS, solução específica)? Ao optar-se por uma solução específica, onde se aloja: num ambiente comum a todos os clientes ou, cada cliente é responsável pelo seu alojamento?

No processo de levantamento de requisitos surge uma nova questão: quais os requisitos funcionais e não-funcionais da solução?

---

<sup>5</sup> Sítio da empresa ABS Informática: <http://www.absinformatica.pt>

<sup>6</sup> Sítio da empresa Vodafone: <http://www.vodafone.pt>

<sup>7</sup> Sítio da empresa KTS: <http://www.kts.pt>

Na definição das principais tecnologias depara-se com um conjunto de questões. Qual a tecnologia de serviços de Internet a utilizar: SOAP ou REST? Qual a ferramenta de log mais indicada para acompanhar os projetos? Que implementação de ORM<sup>8</sup> adotar?

Selecionadas as tecnologias, passa-se à implementação e, nesta fase, define-se o modo de estruturação de cada uma das aplicações, por forma a garantir: a reutilização de componentes, a segurança, o desempenho, a estabilidade e a interoperabilidade.

Passa-se para a implementação e surgem as seguintes dúvidas: qual o estilo arquitetural a implementar em cada um dos componentes? Como implementar o modelo de acesso a dados? Que entidades de negócio usar para a comunicação entre camadas? Como gerir transações?

No desenvolvimento das aplicações surgem dúvidas, questões e problemas que necessitam de uma solução, nem sempre pacífica, imediata e única (Rodrigues 2007). De entre estas questões, destacam-se:

- Localização GPS – que framework, componentes ou tecnologias utilizar para apresentar a localização da máquina no mapa? Que mapa usar (Google, Bing Maps, Yahoo Maps, OpenStreetMap, ArcGIS)?
- Instalação – como disponibilizar os diferentes projetos aos utilizadores?
- Gestão da mudança – que desenvolvimentos são necessários na eventualidade de se alterar o ERP ou o SGBD? O incremento de novas funcionalidades torna-se mais difícil com o crescimento da solução? Como proceder face a uma alteração no modelo de dados?

Concluída a implementação da solução, coloca-se a pergunta: como validar o grau de satisfação do cliente e em que medida os benefícios enumerados foram alcançados?

Do confronto de dúvidas, questões e problemas resultou uma parte da experiência neste trabalho.

---

<sup>8</sup> ORM Object-Relational Mapping

### **1.3 Objetivos**

Pretende-se, nesta secção, identificar e enumerar resumidamente os objetivos a que se propõe esta dissertação:

- Descrever, sistematizar e detalhar o objeto de estudo descrito anteriormente, identificando as suas diversas dimensões, bem como os seus intervenientes e interesses;
- Especificar os requisitos da solução a desenvolver;
- Justificar a decisão entre a evolução da solução anterior ou uma nova solução;
- Definir o seu ambiente e o modo de desenvolvimento;
- Identificar e descrever sumariamente soluções informáticas, nesta área de negócio;
- Estudar e comparar ferramentas / tecnologias para aferir o seu grau de adequação ao objeto de estudo;
- Clarificar e esclarecer a importância de analisar o valor da solução proposta;
- Conceber uma solução para o objeto de estudo, de modo a satisfazer os requisitos especificados, usando boas práticas de engenharia informática;
- Implementar a solução de telemetria de acordo com a conceção realizada anteriormente;
- Realizar teste de verificação da solução (ex. de unidade, funcionais, de carga, de integração e de aceitação) (Pressman 2005);
- Concretizar experiências com utilizadores finais de forma a validar a solução e o seu impacto nos vários interessados.

### **1.4 Metodologia de Trabalho**

A concretização deste escrito é suportada por sete pilares correspondentes aos sete princípios de David Hooker (Hooker 1996): “O Primeiro Princípio – A Razão Por que Tudo existe”, “Segundo Princípio – Manter as coisas simples”, “Terceiro Princípio – Manter a visão”, “Quarto Princípio – O que é produzido alguém consumirá”, “Quinto Princípio – Estar aberto para o futuro”, “Sexto Princípio – Planear com antecedência para reutilizar”, “Sétimo Princípio – Pensar”. Ao longo deste trabalho introduz-se cada um dos pilares e demonstra-se a sua utilidade recuperando-os, uma ou diversas vezes, ao longo do desenvolvimento do escrito. Ocorrido pela

primeira vez, cada princípio em causa é enunciado, a que acresce uma advertência para futuras utilizações, em situações semelhantes.

Na apresentação do projeto são levantadas questões que são esclarecidas, capítulo a capítulo, culminado com uma resposta final, no capítulo de conclusão.

## **1.5 Estrutura**

O presente trabalho divide-se em três partes: introdução, corpo e conclusão, de acordo com (Rodrigues 2013). O conjunto das partes são onze capítulos.

A primeira parte corresponde ao primeiro capítulo. Introduce um breve enunciado da dissertação, resume o estado de arte, descreve os objetivos e contribuições e argumenta a validade e utilidade da solução proposta (Rodrigues 2013).

No capítulo número um, pretende-se enquadrar a dissertação na realidade do mercado das máquinas de Vending, esclarecer o seu âmbito, levantar um conjunto de questões, sistematizar os objetivos e apresentar a metodologia usada. Refere-se também a organização da dissertação e os temas abordados em cada capítulo.

A segunda parte compreende os capítulos dois a dez. Referem-se conceitos e tecnologias relevantes para o trabalho, apresentam-se outros trabalhos e descreve-se a abordagem tomada e a solução proposta (Rodrigues 2013).

No capítulo número dois, contextualizam-se os elementos fundamentais para a concretização deste projeto, sendo eles: a empresa produtora do WinVending, os seus parceiros de negócio e o produto WinVending. Procede-se ao estudo do mercado de Vending, inserido no mercado Ibérico, identificando-se os clientes, a dimensão e evolução do mercado, os concorrentes e o processo operacional.

No capítulo número três, confronta-se e argumenta-se os diferentes cenários de implementação e expõe-se a opção adotada. Sumarizam-se os elementos da solução. Apresentam-se as ferramentas utilizadas para a construção da solução.

No capítulo número quatro, contextualiza-se o modelo de desenvolvimento de novo conceito no processo de inovação da ABS Informática. Identificam-se os benefícios e sacrifícios da

solução apresentada, enquadrados numa perspetiva longitudinal de valor. Explica-se a necessidade de uma proposta de valor bem definida para o negócio de Vending. Expõem-se os possíveis cenários de negócios. Utiliza-se o modelo de Canvas para descrever a ideia de negócio. Apresenta-se a rede de valor do negócio e os seus intervenientes. Por último, exemplifica-se a utilização do modelo AHP, no processo de decisão do método a usar, para notificar / monitorizar os alarmes das máquinas de tabaco.

No capítulo número cinco, descrevem-se os conceitos de negócio, apresentam-se os actores do sistema e enumeram-se os requisitos funcionais e não funcionais.

No capítulo número seis, ilustram-se as opções arquiteturais, o diagrama conceptual de classes e o modelo de dados dos sistemas.

No capítulo número sete, apresentam-se as principais decisões de carácter tecnológico que antecedem a implementação da solução. Identificam-se os modelos arquiteturais usados para as diferentes aplicações que compõem a solução e explicam-se as diferentes partes desses modelos.

No capítulo número oito, descreve-se a tecnologia e o modo de apresentação da localização por GPS. Apresentam-se os métodos de instalação adotados. No final, descreve-se como a solução se prepara para enfrentar o futuro e lidar com as mudanças

No capítulo número nove, realiza-se a avaliação do sistema em termos de: segurança, tolerância a falhas, escalabilidade, gestão da mudança, grau de satisfação dos utilizadores e os resultados do modelo tradicional de Vending comparados com a inclusão da telemetria no negócio.

A terceira parte refere-se ao último capítulo. Concretiza-se um balanço do escrito, através do destaque dos principais aspetos do “que se fez”, formulam-se juízos (positivos e negativos) sobre o que se conseguiu e lançam-se sugestões para trabalho futuro (Rodrigues 2013).

No último capítulo, capítulo número dez, responde-se às questões colocadas no capítulo de introdução, referem-se as limitações do projeto e os métodos previstos para as superar, enumeram-se funcionalidades futuras e termina-se com a apreciação geral, final, do projeto.



## 2 Contextualização

Neste capítulo, descreve-se a evolução da solução da Vending desde a sua aquisição, por parte da ABS Informática, até à incorporação da componente de telemetria. De seguida descreve-se o estado arquitetural e funcional da atual solução (i.e. solução legada) e apresentam-se sumariamente as entidades parceiras envolvidas no desenvolvimento da (nova) solução descrita neste documento. Na última secção, analisa-se o negócio na perspetiva dos clientes, do mercado, da concorrência e do processo operacional atual.

### 2.1 Evolução da Solução

Em 2006, a ABS Informática decide apostar no mercado de Vending adquirindo uma solução de gestão de máquinas de tabaco com as componentes: *backoffice* e móvel. O *backoffice* encontrava-se desenvolvido em Delphi<sup>9</sup> (Delphi 2017), na versão 6, com uma base de dados assente no SGBD Paradox<sup>10</sup> (Paradox 2017) e com integração com o ERP PHC<sup>11</sup> (PHC 2017). A aplicação móvel estava construída em Windows CE 3<sup>12</sup> (Windows CE 3 2017) e, entre outras funcionalidades, comunicava por Bluetooth (Bluetooth 2017) com as máquinas.

---

<sup>9</sup> Sítio do IDE Delphi: <https://www.embarcadero.com/products/delphi>

<sup>10</sup> Sítio do SGBD Paradox: <https://www.spectralcore.com/databases/paradox>

<sup>11</sup> Sítio do ERP PHC: <http://www.phcsoftware.com>

<sup>12</sup> Sítio do sistema operativo Windows CE 3: [https://msdn.microsoft.com/en-US/library/gg162629\(v=winembedded.0\).aspx](https://msdn.microsoft.com/en-US/library/gg162629(v=winembedded.0).aspx)

Um pouco mais tarde, a ABS Informática batiza a aplicação com o nome WinVending e regista como sua propriedade.

Entretanto, vários desafios foram lançados e, todos eles, superados com sucesso, dos quais se destacam: (i) a certificação da AT (Autoridade Tributária) obtida a 15 de novembro de 2010 com o número 242; (ii) a constante atualização da solução para suportar a comunicação com os diferentes protocolos e versões de *firmware*<sup>13</sup> dos fabricantes e (iii) a disponibilização, também em 2010, de uma nova versão da componente móvel com o lançamento do WinVending Mobile em Windows CE 6<sup>14</sup> (Windows CE 6).

A assistência ao cliente e a satisfação das suas necessidades sempre foram prioridade da empresa e o reconhecimento, dessa dedicação, é visível no aumento da quota de mercado da solução, colocando-a, atualmente, como líder no mercado nacional.

Os vários anos de experiência na área de Vending e os notáveis avanços tecnológicos nas comunicações máquina-a-máquina despertam, nos responsáveis da empresa, a vontade de encontrar um parceiro que construísse um dispositivo com a finalidade de comunicar com a máquina e devolver as comunicações remotamente. É também objetivo permitir a invocação remota de comandos da máquina, como por exemplo: programação de seleções e preços, alteração de código de operador e alteração da mensagem publicitária. No ano de 2011, a empresa realiza algumas abordagens junto de potenciais parceiros, mas sem sucesso.

Em meados do ano 2012, a Vodafone contacta a ABS Informática para medir a motivação da empresa relativamente à concretização do projeto, numa parceria a três: a ABS Informática, a Vodafone e a KTS, especialista na construção de dispositivos que comunicam com máquinas, atualmente parceiro da Vodafone noutros projetos de M2M.

Assim, inicia-se esta parceria e a inclusão da componente de telemetria na solução de Vending da empresa passando, a partir desse momento, a denominar-se de Vodafone WinVending.

---

<sup>13</sup> *Firmware* é o conjunto de instruções operacionais programadas diretamente no *hardware* de um equipamento eletrónico (Firmware 2015)

<sup>14</sup> Sítio do sistema operativo Windows CE 6: [https://msdn.microsoft.com/en-us/library/dn600283\(v=winembedded.60\).aspx](https://msdn.microsoft.com/en-us/library/dn600283(v=winembedded.60).aspx)

## 2.2 Entidades Envolvidas

Neste subcapítulo, faz-se a apresentação de cada uma das partes envolvidas na parceria do projeto Vodafone WinVending: ABS Informática, KTS e Vodafone.

### 2.2.1 ABS Informática

Fundada em 1987, a ABS Informática (Almeida Brandão & Salgado, Lda.) tem como missão ser o parceiro privilegiado dos seus clientes, na área de Sistemas Informáticos (ABS Informática 2017).

Os produtos que comercializa assentam nas áreas de *software* de gestão, consultadoria informática, *hardware*, comunicações, integração entre sistemas e desenvolvimento de soluções verticais. Tais produtos complementam-se permitindo à empresa conceber, fornecer e implementar a totalidade de um sistema informático.

A empresa dispõe de acordos de parceria com fabricantes de *software* de gestão (PHC, ETICADATA<sup>15</sup>) (Eticadata 2017), servidores e redes (UNISYS<sup>16</sup>) (UNISYS 2017) e computadores pessoais (TSUNAMI<sup>17</sup>) (TSUNAMI 2017).

O desenvolvimento de soluções verticais, integradas com as soluções PHC e ETICADATA, surge como uma mais-valia a apresentar aos seus clientes, entre as quais se incluem: o controlo de expedições e conferência de mercadoria (WinPicking), a gestão de transitários (Wintrans<sup>18</sup>) (Wintrans 2017) e a gestão de máquinas de venda automática de tabaco (Vodafone WinVending) que será o objeto de estudo nesta dissertação.

---

<sup>15</sup> Sítio do ERP Eticadata: <https://pt.eticadata.com/>

<sup>16</sup> Sítio da empresa UNISYS: <http://www.unisys.com/>

<sup>17</sup> Sítio da empresa TSUNAMI: <http://www.tsunami.pt/>

<sup>18</sup> Sítio da solução Wintrans: <http://wintrans.pt/#inicio>

### **2.2.2 KTS**

Fundada no ano 2000, a KTS é uma empresa multinacional, de capital exclusivamente português, de distribuição e representação de componentes eletrónicos, módulos e terminais (KTS 2017). A sua sede é em Lisboa e tem delegações em Braga e Madrid.

A KTS tem como missão ser o melhor distribuidor e representante técnico nos mercados onde atua. O seu fator de diferenciação é a qualidade do serviço prestado e a competitividade e inovação dos seus produtos. Como empresa focada no serviço ao cliente, tem a capacidade de desenvolver componentes e soluções personalizadas às necessidades do seu cliente (KTS, 2016).

Desde 2007, com a criação de um departamento interno focado em soluções M2M, tem apoiado os seus clientes e parceiros na realização de diversos projetos nesta área. O projeto Vodafone WinVending é um claro exemplo de um desses projetos.

### **2.2.3 Vodafone**

Fundada em 1991, a Vodafone é uma operadora de telecomunicações móveis Britânica que conta, atualmente, com mais de 400 milhões de clientes em mais de 30 países. Em Portugal é uma referência no mercado, líder em inovação, imagem de marca e orientação e satisfação dos seus clientes.

No mercado empresarial, a Vodafone, desde 2009, altura em que entra no mercado de M2M, está determinada a estender e a marcar a sua presença nesta indústria, criando redes de intercomunicação de dispositivos dentro dos seguintes setores: cuidados de saúde, transportes, energético e controlo remoto de máquinas de venda automática, cuja aplicação prática é estudada nesta dissertação.

A área de M2M é das que mais tem crescido na empresa, apresentando um crescimento de 25% em 2014. É considerada uma área estratégica da empresa, porque a Internet das coisas tende a aumentar e a Vodafone pretende estar presente e investir, para aumentar a sua posição.

O investimento na área de M2M foi reconhecido, num estudo da Analysys Mason (James 2015). A consultora considerou, pelo quarto ano consecutivo, como a melhor operadora mundial de fornecimento de serviços M2M. Para o referido estudo, cada operador é avaliado em cinco

critérios: escalabilidade e execução, capacidade tecnológica, apoio ao cliente e pós-venda e parcerias e estrutura organizacional (James 2015).

A Vodafone é atualmente o operador que mais ligações gere na Internet das Coisas.

## 2.3 Solução Legada

A solução WinVending legada divide-se em dois sistemas: o *backoffice* e a aplicação móvel. Na Figura 1 introduz-se ambos através de um diagrama de implantação.

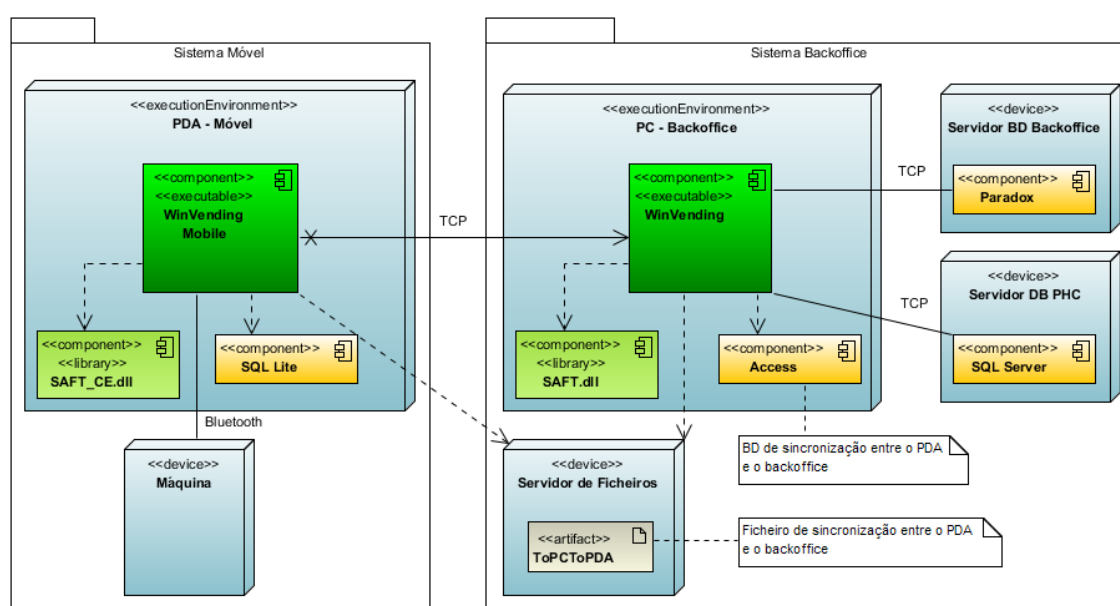


Figura 1 – Diagrama de implantação da solução legada WinVending

De seguida, procede-se à descrição dos componentes dos sistemas, salientando as suas relações e dependências.

O sistema *backoffice* é constituído por uma aplicação executável, desenvolvida em Delphi 7, que integra uma dll<sup>19</sup> “SAFT.dll”, desenvolvida em VB.NET<sup>20</sup> (VB.NET 2017), para assinar os documentos de venda. O executável acede a diferentes fontes de dados:

<sup>19</sup> DLL do termo inglês Dynamic-link Library

<sup>20</sup> Sítio da linguagem de programação VB.NET: <https://msdn.microsoft.com/pt-br/library/cc580665.aspx>

- do Paradox obtém os dados operacionais relacionados com, por exemplo, os casos de uso UC\_WVB\_01, UC\_WVB\_02, UC\_WVB\_03, representados na Figura 2;
- do SQL Server<sup>21</sup> (SQL Server 2017) mantém os dados do ERP PHC relativos aos documentos de vendas e às entidades UC\_WVB\_04, UC\_WVB\_05, UC\_WVB\_06 da Figura 2;
- e através do MS Access<sup>22</sup> (MS Access 2017) gere os dados temporários necessários para a integração com os PDA's<sup>23</sup> (c.f. UC\_WVB\_07 da Figura 2).

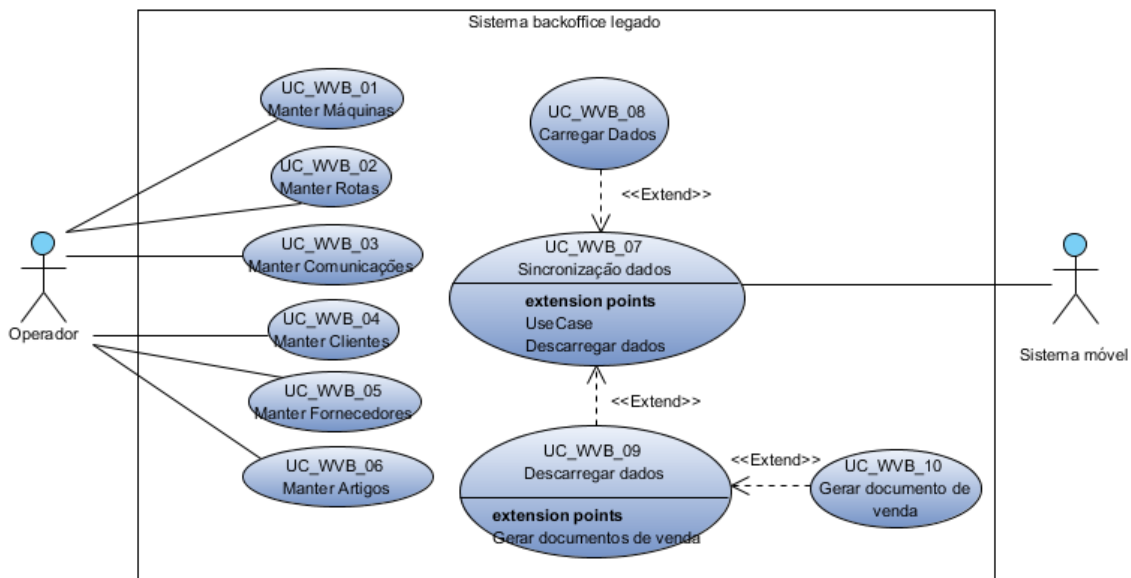


Figura 2 – Caso de uso “Sistema *backoffice* legado”

O sistema móvel é composto por um executável, desenvolvido em Windows CE 6 e dependente de uma biblioteca “SAFT\_CE.dll” que assina os documentos de venda gerados. O PDA comunica com as máquinas através de um dispositivo Bluetooth, que é colocado na máquina com a finalidade de obter os dados contabilísticos (cf. UC\_WVM\_01 da Figura 3) e de efetuar a sua programação (cf. UC\_WVM\_02 da Figura 3). Mantêm-se os dados localmente, numa base de dados em SQL Lite<sup>24</sup>, de modo a proporcionar uma utilização desconectada. Posteriormente,

<sup>21</sup> Sítio do SGBD SQL Server: <https://www.microsoft.com/pt-pt/sql-server>

<sup>22</sup> Sítio do SGBD MS Access: <https://products.office.com/pt-pt/access?legRedir=true&CorrelationId=8538c744-abfd-4699-bb78-f132d2822945>

<sup>23</sup> PDA do termo inglês Personal Digital Assistance

<sup>24</sup> “Considera-se o SQLite como uma biblioteca implementadora de um mecanismo de base de dados SQL transacional autónomo, sem servidor e sem necessidade de configuração.” (tradução nossa) (SQLite 2017)

sincroniza-se com o *backoffice* (cf. UC\_WVB\_07 da Figura 2 e UC\_WVM\_03 da Figura 3) de dois modos distintos: a qualquer momento por GPRS ou, localmente, através da cópia de ficheiros com os dados por sincronizar.

A sincronização por GPRS realiza-se por um canal de comunicação TCP, entre o PDA e um PC possuidor de um IP público e de uma instalação do *backoffice* preparado para a receção dos dados. Durante a transferência é utilizada uma BD em MS Access para auxiliar a integração com a BD operacional.

O segundo modo de sincronização concretiza-se aquando da comunicação entre o PDA e uma máquina local. Socorre-se da ferramenta “Windows Mobile Device Center”<sup>25</sup> como intermediária na comunicação entre os dispositivos e realizam-se os próximos passos consoante a operação pretendida (carga ou descarga). Na operação de carga, o *backoffice* cria um ficheiro (cf. UC\_WVB\_08 da Figura 2) que, posteriormente, é integrado pelo PDA (UC\_WVM\_04 da Figura 3). No mecanismo de descarga, o PDA gera um ficheiro (UC\_WVM\_05 da Figura 3) que é importado pelo *backoffice* (cf. UC\_WVB\_09 da Figura 2) e, caso necessário, é responsável pela integração dos documentos de venda no ERP (cf. UC\_WVB\_10 da Figura 2).

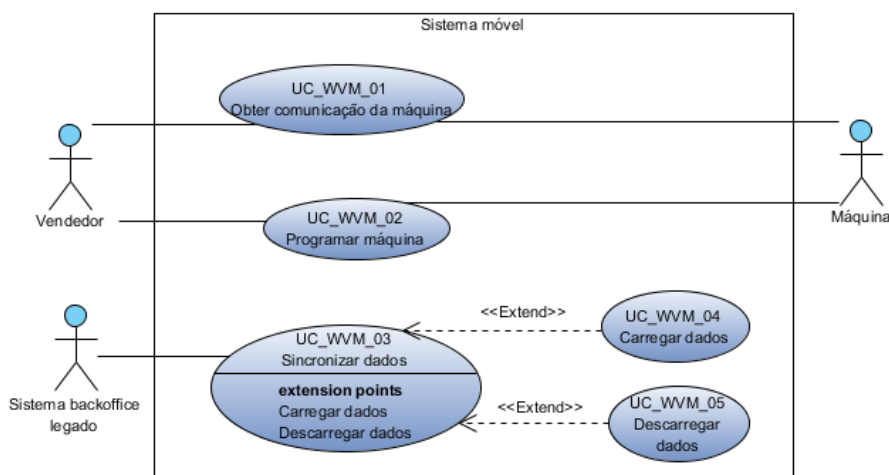


Figura 3 – Caso de uso “Sistema móvel”

<sup>25</sup> “Windows Mobile Device Center oferece a sincronização de dados e de gestão do dispositivo entre um dispositivo baseado no Windows Mobile e um computador.” (Windows Mobile Device Center 2017)

Deste modo, sintetizam-se as funcionalidades principais dos sistemas *backoffice* e móvel e o modo como comunicam e se integram. Da perspetiva do sistema móvel salienta-se, ainda, o relacionamento com a máquina.

Da análise da solução atual (legada) detetam-se limitações, problemas e inconvenientes tanto sob o ponto de vista de negócio, como de engenharia informática.

Sob o ponto de vista de negócio verificam-se limitações, na sua maioria, relacionadas com operações de gestão das máquinas, à distância e em tempo real, das quais se salientam: (i) a comunicação, (ii) a programação e (iii) o controlo alarmístico (porta aberta / fechada, máquina ligada / desligada, localização e movimentação da máquina).

Da perspetiva de engenharia informática, a solução legada apresenta um conjunto de restrições relacionados com: (i) a gestão das bases de dados, (ii) a comunicação com o ERP, (iii) a usabilidade e estética da aplicação de *backoffice* e (iv) a arquitetura da solução.

Por um lado, a fragmentação e heterogeneidade, das fontes de dados (BD's relacionais e ficheiros) e, por outro lado, a utilização de diferentes SGBD's (Paradox, MS Access) dificultam o entendimento e gestão dos dados. Aliam-se problemas de consistência e de desempenho no acesso aos dados comprovados, ao longo do tempo, pela utilização dos SGBD's mencionados.

Encara-se a comunicação com um único ERP (PHC) como um entrave à evolução da solução, em mercados onde se utilizam diferentes ERP's como: Eticadata, SAGE<sup>26</sup> (SAGE 2017), Primavera<sup>27</sup> (Primavera 2017) e Dynamics NAV<sup>28</sup> (Dynamics NAV 2017). Agrava-se o referido entrave pela total dependência da solução relativamente ao ERP, pelo facto de não permitir a utilização da solução, em ambiente desconectado do ERP.

A observação dos utilizadores, no seu ambiente de trabalho e os comentários que tecem apontam um desagrado comum relativamente à estética da aplicação *backoffice*, cujo desenho se considera ultrapassado e com défice de usabilidade, causadora de demora, na execução das tarefas e dificuldade na compreensão de novas funcionalidades.

---

<sup>26</sup> Sítio do ERP SAGE: <http://www.sage.pt/software-e-servicos/erp>

<sup>27</sup> Sítio do ERP Primavera: <http://pt.primaverabss.com/pt/>

<sup>28</sup> Sítio do ERP Dynamics NAV: <https://www.microsoft.com/pt-pt/dynamics365/nav-overview>

A nível arquitetural, a solução carece de uma divisão lógica e física por camadas (apresentação, lógica de negócio e acesso a dados) de modo a flexibilizar e agilizar a alteração de diferentes componentes como: acesso a um novo SGBD, comunicação com um novo ERP e alteração da interface com o utilizador. Verifica-se também a existência de: (i) uma baixa coesão caracterizada pela execução de tarefas com responsabilidades distintas na mesma classe (e.g. a janela de consulta da máquina acede diretamente à tabela da base de dados referente aos clientes) e (ii) um alto acoplamento constatado pelo elevado número de dependências que uma classe pode ter (e.g. uma alteração de um campo no cliente implica modificações em múltiplas classes da aplicação).

A constatação das limitações, problemas e inconvenientes da atual solução (legada) motivam a reflexão de possíveis abordagens para as resolver.

## **2.4 Análise de Mercado**

Neste subcapítulo, procede-se ao estudo do mercado de Vending, com o foco principal no mercado Ibérico, identificando-se os clientes, a dimensão e evolução do mercado, os concorrentes e o processo operacional geral e o potencial de otimização.

### **2.4.1 Clientes**

Em Portugal, as máquinas de venda automática de tabaco encontram-se, nos mais diversos locais: restaurantes, cafés, bares, discotecas, postos de abastecimentos, entre outros. Estes estabelecimentos, na maioria das vezes, optam por criar acordos de exploração com os proprietários das máquinas, a troco de uma comissão sobre o valor das vendas, em vez de comprar as máquinas

Comumente denominadas de depositários de tabaco, estas entidades, possuem um parque de máquinas que necessitam de uma gestão rigorosa das suas atividades: disponibilização, manutenção, reparação e abastecimento das máquinas, a todo o momento.

Em Espanha, com a exceção das Ilhas Canárias, a venda a retalho de tabaco é um monopólio propriedade do Estado que coloca, sob concurso público, a exploração de determinadas áreas

geográficas onde podem ser construídos estabelecimentos conhecidos por estancos<sup>29</sup>. Um estanco, embora assuma uma forma diferente, partilha as mesmas atividades de um depositário de tabaco.

A solução apresentada, nesta dissertação, tem como objetivo satisfazer as necessidades do seu cliente-alvo o “proprietário de máquinas de tabaco”, em todas as suas dimensões.

### **2.4.2 Mercado**

A perceção da dimensão e evolução do mercado é de extrema importância para o conhecimento da sustentabilidade e crescimento da solução. Partindo desse pressuposto, procede-se, de seguida, ao estudo do mercado português, onde a empresa ABS Informática é, atualmente, líder de soluções para máquinas de Vending de tabaco com o WinVending.

“A Informa D&B publicou um estudo, no qual refere que o mercado das máquinas de venda automática gerou, no mercado nacional, em 2013, um volume de negócios de 520 milhões de euros. O referido estudo, concluiu que as máquinas de tabaco representaram 66% das vendas totais do setor, com 345 milhões de euros” (Informa D&B 2014).

Segundo este estudo (Informa D&B 2014), no final de 2013, operavam em Portugal cerca de 90.000 máquinas de venda, das quais, 45% eram máquinas de tabaco, 29% de bebidas quentes, 16% de bebidas frias e 11% de alimentos sólidos.

De acordo com a Informa D&B, as previsões do mercado de exploração de máquinas de venda automática apontam um ligeiro aumento em 2014, sendo que, o mercado de tabaco perderá peso, a curto e médio prazo, no volume de negócios total. Na Tabela 1, é apresentado um resumo do mercado português de Vending focado no segmento de tabaco (Informa D&B 2014).

---

<sup>29</sup> Local de venda de produtos de acesso restrito como: o tabaco e os selos.

Tabela 1 – Resumo mercado Vending em Portugal (2013)

<b>Mercado Vending em Portugal (2013)</b>	
<b>Parque de máquinas</b>	90.000
<b>Parque de máquinas tabaco</b>	40.500
<b>Mercado de exploração (milhões de euros)</b>	520
<b>Mercado de exploração tabaco (milhões de euros)</b>	345
<b>Crescimento do mercado</b>	-3,7%

Limitada pela dimensão do mercado nacional, é intenção da solução proposta consolidar a posição de líder no mercado nacional e expandir-se, por meio da internacionalização.

Numa primeira fase, procede-se ao incremento gradual no mercado espanhol onde, ao momento, o WinVending tem uma representação nas Ilhas Canárias. Por essa via, apresenta-se, de seguida, uma análise do mercado espanhol.

A principal conclusão do estudo de Informa D&B observa que o mercado de Vending cresce, em 2014, contrariando a tendência descendente dos cinco anos anteriores. Assim, a faturação do mercado situa-se em 2.000 milhões de euros, mais 1,3% do que no ano transato. Em paralelo, o parque de máquinas experimentou um aumento moderado, situando-se em 555.000 unidades (Informa D&B 2015).

Relativamente ao mercado específico de máquinas de tabaco que representa 52% do total, manteve o ritmo de descida (-1,4%), penalizado pelo prolongar da contração do consumo (Informa D&B 2015).

Na Tabela 2, mostra o resumo do mercado de Vending, em Espanha, com ênfase para o segmento específico do tabaco.

Tabela 2 – Resumo mercado Vending em Espanha (2014)

<b>Mercado Vending em Espanha (2014)</b>	
<b>Parque de máquinas</b>	555.000
<b>Parque de máquinas de tabaco (extrapolado)</b>	292.842
<b>Mercado de exploração (milhões de euros)</b>	2.000
<b>Mercado de exploração de tabaco (milhões de euros) (extrapolado)</b>	1.055,2
<b>Crescimento do mercado de tabaco</b>	-1,4%

O processo de internacionalização tem potencial de crescimento para outros países, em particular, no mercado europeu (França, Itália, Alemanha) e sul-americano.

### **2.4.3 Processo Operacional**

Na sua essência, o negócio do depositário concede, aos seus clientes, máquinas de Vending funcionais e abastecidas de produtos e dinheiro (moedas) para efetuar trocos. Para estas tarefas, um depositário emprega operadores para abastecer e técnicos para reparar. Nos abastecimentos as rotas são fixas e as máquinas são inspecionadas por ciclos temporais definidos (e.g. todas as terças-feiras). Na perspetiva das reparações os técnicos são solicitados, apenas, na ocorrência de falhas ou avarias (Heinkele & Pousttchi 2004). Neste ponto, pretende-se descrever os métodos de abastecimento.

O processo convencional de abastecimento, também conhecido por Auto Venda, consiste nos seguintes passos ilustrados na Figura 4. O operador inicia a sua rota. De seguida e para cada visita: sai da carrinha, abre a máquina, efetua a verificação do stock, recolhe o dinheiro, fecha a máquina, regressa ao veículo, conta o dinheiro recolhido, reúne as mercadorias em falta, volta à máquina, abre-a, enche com produto e trocos em falta, fecha a máquina e regressa ao veículo, onde atualiza o formulário de stock. No final do dia de trabalho, termina a rota, entrega o dinheiro recolhido e o stock do veículo à administração.

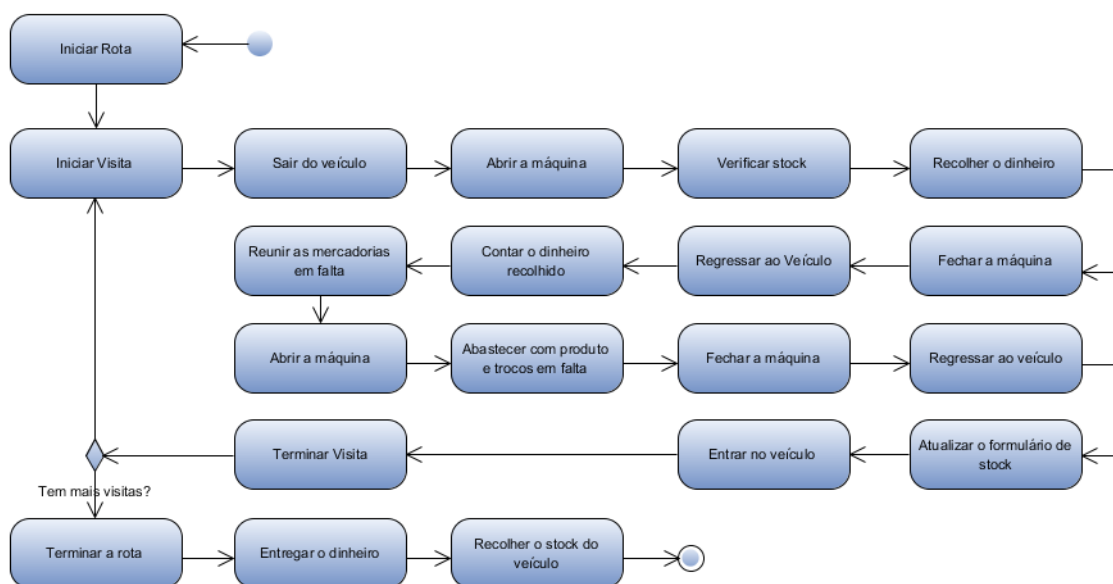


Figura 4 – Diagrama de atividades de abastecimento pelo método Auto Venda

Este método tem como inconvenientes: excesso de tempo numa visita pela duplicação do percurso máquina-veículo e demasiado stock circulante resultado de uma compensação pelo desconhecimento das quantidades vendidas.

Para contrariar essas desvantagens, alguns depositários usam o método de Pré-Venda. Este processo de abastecimento consiste em duas visitas. Na primeira, o operador recolhe o dinheiro e verifica as necessidades da máquina. Na segunda, é concretizado o abastecimento da máquina com base na análise de necessidades, anteriormente, levantadas. A desvantagem deste método é a necessidade de se visitar a máquinas duas vezes.

De modo a reduzir o número de visitas à máquina para uma, alguns depositários, criam o modelo de Pré-Venda baseado em estimativas estatísticas. A desvantagem deste método reside na pouca fiabilidade das estimativas que resultam da oscilação aleatória do consumo de tabaco. Para compensar a falta de precisão das estimativas, as carrinhas são carregadas com mais produtos.

A telemetria sugere o método de Pré-Venda com base nos valores obtidos nas comunicações remotas com a máquina. Com o intuito de atingir maiores níveis de precisão, as comunicações devem ser efetuadas o mais próximo possível da visita. No capítulo análise de negócio, são apresentadas as vantagens deste modelo.

#### **2.4.4 Concorrência**

A análise da concorrência permite às organizações conhecerem as suas estratégias, pontos fortes e pontos fracos. A reunião desse conhecimento aumenta a confiança, na tomada de decisões (Costa 2010).

Com a consciência da importância do conhecimento das soluções concorrentes à solução Vodafone WinVending analisam-se três soluções concorrentes e estabelece-se um comparativo entre elas detalhado no subcapítulo 0.

### **2.5 Soluções Concorrentes**

Neste subcapítulo, identificam-se e descrevem-se sumariamente três soluções, na área de Vending: GMBOS (GM), UPVending (UV) e Vendon (V). No final, realiza-se uma confrontação das características da concorrência com as soluções: WinVending legada (WVL) e Vodafone WinVending (VWV) concebida no âmbito desta dissertação.

#### **2.5.1 GMBOS**

A GM é um fabricante de máquinas de tabaco que complementa as suas máquinas com uma solução de telemetria por modem ou GSM<sup>30</sup> denominada “GMBOS”<sup>31</sup> (GMBOS 2017).

O facto de a solução ser desenvolvida por um fabricante pode ser encarado, por um lado, como uma vantagem, pois tem acesso a todas as funcionalidades da máquina e, na sua essência, implementa acesso à informação, em tempo real. Por outro lado, tem a limitação de não comunicar com outros fabricantes, o que, da perspectiva do depositário, é uma desvantagem, pois, normalmente, para não dependerem de um único fabricante, tornam o seu parque de máquinas eclético.

---

<sup>30</sup> GSM do termo inglês Global System for Mobile Communications

<sup>31</sup> Sítio da solução de telemetria GMBOS: <http://www.gmvending.com/gm-vending-/productos/maquinas-de-tabaco/sistema-telemetria-gmbos.html>

Deteta-se as próximas carências: integração com múltiplos ERP's, gestão da localização da máquina e acompanhamento do vendedor com uma aplicação móvel.

### **2.5.2 UPVending**

O UpVending é um sistema português de telemetria, para gestão remota de máquinas de Vending. A solução inclui hardware, conectividade (via cartão M2M) e um software de gestão.

Como pontos a favor, apresenta uma acessibilidade multiplataforma, uma interface gráfica apelativa, a localização de máquinas e análises nativas abrangentes às diferentes áreas de negócio.

As limitações são: a integração com um único ERP proprietário, a carência de análises personalizadas, a possibilidade de comunicação dos alarmes a outros aplicativos, a falta de sensores de porta aberta/fechada e a ausência de uma aplicação móvel que acompanhe o trabalho dos vendedores.

### **2.5.3 Vendon**

Vendon<sup>32</sup> (Vendon 2017) é uma solução completa de telemetria para gestão remota de máquinas de Vending. O produto é modular e composto por duas componentes: um dispositivo físico e um serviço de análise de dados na nuvem.

A modularidade oferecida por este produto é uma mais-valia, ao permitir a sua integração com sistemas terceiros. Além desta valência, apresenta: uma interface gráfica apelativa, a capacidade de integração com os principais sistemas ERP europeus de venda automática, análises nativas diversas (e.g. análise de rentabilidade por cliente/local/máquina/produto) e a integração com plataformas de pagamento por cartão de crédito.

Uma outra vantagem é a facilidade de instalação do módulo físico, no entanto, essa mais valia é conseguida com o sacrifício da possibilidade de ligação a qualquer tipo de máquinas e da não

---

<sup>32</sup> Sítio da solução de telemetria Vendon: <http://www.vendon.net/>

utilização de sensores de porta aberta/fechada. Como outras desvantagens, nota-se a falta de análises personalizadas criadas pelos próprios utilizadores (e.g. vendas de produto entre visitas) e a não existência de uma aplicação móvel que acompanhe as funções dos vendedores.

#### **2.5.4 Comparação dos Produtos**

Nesta secção, pretende-se concretizar, sob a forma da Tabela 3, uma comparação das características de cada um dos produtos existentes com as soluções WinVending: legada (WVL) e nova (VWV). Destacam-se as próximas características:

- acessibilidade total – o sistema pode ser acedido em qualquer dispositivo (Sim – acessível em todos os dispositivos / Não – somente em dispositivos específicos);
- comunicação em tempo real – o sistema permite a comunicação com as máquinas à distância e em tempo real (Sim – comunicação no momento da venda ao cliente final / Não – não permite comunicações à distância / Parcial – comunicação das vendas à distância através de pedidos executados por utilizadores ou rotinas automáticas);
- comunicação com todas as máquinas – o sistema permite comunicar com diferentes fabricantes de máquinas (Parcial – mais do que um / Limitada – apenas um);
- presença de sensor – indica a presença de determinado sensor na máquina (Sim – presente / Não – ausente);
- integração com múltiplos ERP's – capacidade do sistema para integrar diferentes ERP's (Sim – integra com múltiplos ERP's / Não – comunica com um ERP proprietário);
- Plataformas de pagamento – capacidade do sistema para comunicar com plataformas de pagamento (Sim – comunica / Não – não);
- API de comunicação com as máquinas – indica se o sistema disponibiliza uma API para que terceiros possam integrar, nas suas soluções, a comunicação com as máquinas (Sim – disponibiliza / Não – não disponibiliza);
- análises nativas – o sistema disponibiliza um conjunto diversificado e completo de análises (Sim – disponibiliza um conjunto abrangente de análises / Não – não disponibiliza qualquer análise / Parcial – implementa um conjunto restrito de análises);
- análises personalizáveis – indica se o sistema permite a criação de análises pelos utilizadores (Sim – permite / Não – não permite);
- aplicação móvel – indica se o sistema integra uma solução que acompanha os vendedores na visita à máquina (Sim – disponibiliza / Não – não disponibiliza);
- aplicação de controlo de alarmes – indica se o sistema integra uma solução de monitorização de alarmes das máquinas (Sim – disponibiliza / Não – não disponibiliza).
- rapidez de instalação do dispositivo – tempo de instalação do dispositivo e sensores (Não – não implica instalação / Embutido – o dispositivo e sensores estão incorporados na máquina / Rápida – menos de 5 minutos / Razoável – de 5 a 10 minutos / Demorado – de 10 a 30 minutos).

Tabela 3 – Comparação das soluções WinVending com a concorrência

Característica	GM	UV	V	WVL	VWV
Acessibilidade total	Sim	Sim	Sim	Não	Não
Comunicação em tempo real	Sim	Parcial	Parcial	Não	Parcial
Comunicação com todas as máquinas	Limitado	Parcial	Parcial	Parcial	Parcial
Sensor de movimento	Não	Não	Não	Não	Sim
Sensor de porta aberta/fechada	Não	Não	Não	Não	Sim
Sensor de GPS	Não	Sim	Sim	Não	Sim
Integração com múltiplos ERP's	Não	Não	Sim	Não	Sim
Plataforma de pagamento	Não	Não	Sim	Não	Não
API de comunicação com as máquinas	Sim	Não	Sim	Não	Sim
Análises nativas	Sim	Sim	Sim	Sim	Parcial
Análises personalizáveis	Não	Não	Não	Sim	Sim
Aplicação móvel	Não	Não	Não	Sim	Sim
Aplicação de controlo de alarmes	Não	Não	Não	Não	Sim
Rapidez de instalação do dispositivo	Embutida	Razoável	Rápida	Não	Demorada



## 3 Visão da Solução

Neste capítulo, justifica-se a decisão do desenvolvimento de uma nova solução suportada pela comparação e argumentação de possíveis abordagens alternativas. Apresentam-se os vários elementos do sistema e fundamenta-se a decisão das ferramentas adotadas no desenvolvimento da solução.

### 3.1 Abordagens Possíveis

A incorporação das funcionalidades de telemetria na atual solução de Vending coloca em aberto um conjunto de possíveis cenários: (i) implementar as novas características no ambiente atual, ou numa nova solução; (ii) alojar os dados localmente, ou na nuvem; (iii) desenvolver a nova solução na *web*, ou em *desktop*; (iv) comunicar os alarmes das máquinas por SMS, ou através de uma aplicação *web* centralizada, ou por meio de uma aplicação *desktop*.

#### 3.1.1 Solução atual Vs nova

O primeiro cenário, relaciona-se com a possibilidade de incorporar as novas funcionalidades no ambiente atual, ou num ambiente totalmente distinto. O desenvolvimento sob a atual versão é uma possibilidade, desde que as modificações necessárias não envolvam significativos esforços de reengenharia. Recuperam-se as limitações de engenharia informática salientadas na secção “2.3 Solução Legada”, de modo a verificar-se a viabilidade de as ultrapassar na atual solução.

Começa-se por analisar a limitação detetada no atual SGBD (Paradox), a nível das recentes especificações de negócio e agravadas pela necessidade de lidar com um maior volume de dados previsto com a adição da componente de telemetria (e.g. comunicações e alarmística). Desse estudo<sup>33</sup>, resulta a decisão de substituir o SGBD Paradox pelo SQL Server. Tal substituição, implica a verificação da compatibilidade entre o SGBD eleito e o motor de acesso a dados BDE<sup>34</sup> (Borland Database Engine) utilizado no desenvolvimento da atual solução. Constatou-se que, embora possível, a ligação do BDE ao SQL Server tem algumas restrições (e.g. não é possível a ligação a tipos de dados `nvarchar`) (Olso Stirling 2003) que afetam o usufruto das potencialidades do SGBD selecionado (e.g. pretende-se utilizar o tipo de dados “`nvarchar`” de modo a garantir o armazenamento de texto no padrão Unicode<sup>35</sup>). Em simultâneo, o fabricante Embarcadero<sup>36</sup> lança a RAD Studio XE7<sup>37</sup> (Ultimate Delphi 2016) e descontinua a tecnologia BDE. Neste ponto, pondera-se a substituição do motor de acesso a dados por um vigente que possibilite a utilização de todas as potencialidades do SQL Server (e.g. FireDAC<sup>38</sup>). A migração de motor de acesso a dados salienta uma limitação da arquitetura da aplicação que se caracteriza pela inexistência da divisão por camadas, mais concretamente, pela ausência da camada de acesso a dados que envolve uma alteração massificada e global de todos os componentes que fazem acesso aos dados.

De seguida, procede-se ao estudo do impacto, na atual solução, do acesso a múltiplos ERP's de um modo desconectado. Surge, novamente, a limitação arquitetural como entrave à

---

<sup>33</sup> A escolha do SGBD SQL Server é fundamentada em detalhe na seção “3.3.2 Sistema Gestor de Base de Dados”.

<sup>34</sup> Borland Database Engine (BDE) é uma camada de acesso a dados, partilhada por todos os produtos Borland's (e.g. Delphi) (tradução nossa) (Embarcadero 2015). BDE tem um conjunto de *drivers* que permite às aplicações comunicar com diversos tipos de SGBD's como: Paradox, dBASE, MS Access ou qualquer fonte de dados ODBC. É utilizado com método de acesso aos dados na solução legada.

<sup>35</sup> “Unicode é um padrão que permite aos computadores representar e manipular, de forma consistente, texto de qualquer sistema de escrita existente”. (Unicode 2017)

<sup>36</sup> “Embarcadero é uma empresa norte-americana de *software* que desenvolve, produz, licencia e dá suporte a uma ampla variedade de produtos e serviços relacionados a software através das suas várias divisões.” (Embarcadero Technologies 2016)

<sup>37</sup> RAD Studio XE7 é uma versão do Delphi lançada pela Embarcadero em 2 de setembro de 2014. (Delphi 2017)

<sup>38</sup> FireDAC é um conjunto único de componentes de acesso a dados universal para Delphi e C++ Builder que possibilita o desenvolvimento de soluções a acederem a diferentes SGBD's, tais como: InterBase, SQLite, MySQL, SQL Server, Oracle, PostgreSQL, IBM DB2, SQL Anywhere, MS Access, Firebird, Informix, entre outros. (tradução nossa) (FireDAC 2015)

incorporação ágil e segura de novas funcionalidades, ao não separar por camadas o acesso a dados e também as regras de negócio, de modo a, por um lado, tornar possível o acesso transparente aos diferentes ERP's, por outro lado, permitir incorporar regras de negócio específicas do ERP, com o menor impacto para os seus componentes dependentes. Para que seja uma realidade o modo desconectado, é necessária uma alteração substancial, no modelo de dados da solução WinVending, de modo a contemplar todas as tabelas necessárias (e.g. clientes, produtos, vendedores, entre outros) e todos os campos pertinentes do ERP.

Finalmente, equaciona-se o modo de alterar o aspeto visual e funcional da atual aplicação e, uma vez mais, considera-se a ausência da divisão por camadas uma lacuna arquitetural de resolução complexa, devido ao alto grau de acoplamento e ao baixo grau de coesão que a aplicação enfrenta. Assim, uma alteração a nível de apresentação tem impacto a nível de negócio ou acesso a dados.

De acordo com as referidas limitações de engenharia informática mencionadas, considera-se a atual solução (legada) obsoleta, face às necessidades tecnológicas que o crescimento e amplitude do projeto o exige. Usa-se o termo obsoleta para descrever uma característica, a má qualidade, normalmente associada aos sistemas legados, isto é, pouca extensibilidade, deficiente estrutura do código e escassez de documentação, o que torna a aplicação de compreensão e manutenção difícil (Pressman 2005). Com base nesses argumentos, opta-se pela construção de uma nova solução que, numa primeira fase, conterá as entidades e processos de negócios da área de telemetria e da sua integração com as componentes: móvel<sup>39</sup>, ERP e API de comunicação com o Middleware do parceiro KTS. Posteriormente, mas não no âmbito desta dissertação, proceder-se-á à migração gradual dos diferentes módulos da atual solução, até à sua extinção.

Nesta fase, sugere-se à empresa a construção de uma nova aplicação assente numa arquitetura possuidora de várias definições e componentes que refletem opções tecnológicas e de desenho de *software*, como:

---

<sup>39</sup> O módulo móvel é conhecido por WinVending Mobile

- Ferramentas de desenvolvimento – linguagens de programação, gestor de versões, desenho UML, prototipagem;
- Padrões de desenho – representa um padrão de desenho que define a melhor prática para resolver problemas concretos e recorrentes, permitindo utilizar padrões já testados pela comunidade de *software* e pensando no desenvolvimento como um processo de engenharia;
- Infraestrutura de comunicações entre componentes – considera-se a forma como se acede às aplicações ERP's<sup>40</sup>, como se estabelece a comunicação entre a aplicação móvel e o *backoffice* e como se obtém os dados das diversas máquinas.

### 3.1.2 Alojamento dos dados

Segue-se a decisão do local de alojamento dos dados: na nuvem, para todos os clientes, ou cada cliente é responsável pelo seu alojamento. No segmento de Vending, a posição dos principais clientes da ABS Informática é perentória e notoriamente contra a possibilidade dos seus dados não estarem “dentro de portas”, em servidores controlados pelos seus administradores de sistemas. O ponto de vista dos clientes, a que se juntam os curtos prazos de implementação, determinam a opção de uma instalação por cliente.

### 3.1.3 Ambiente de desenvolvimento

A próxima decisão prende-se com o ambiente de desenvolvimento: web ou, desktop. Para tal, os fatores tempo, experiência da equipa de programação e reutilização de componentes já existentes foram fulcrais para a predileção do ambiente *desktop*.

Tomam-se as duas últimas decisões, em coincidência e com a premissa da arquitetura definida, para a solução contemplar a mudança de posição relativa a cada uma delas, individualmente, ou em conjunto.

---

<sup>40</sup> Inicialmente, a solução comunicará com o ERP (do termo inglês Enterprise Resource Planning) PHC; não obstante, deverá estar preparada para a integração com outros ERP's do mercado.

### **3.1.4 Notificação e comunicação de alarmes**

Num outro patamar de decisão, situa-se a definição das entidades a notificar e o modo de comunicação dos alarmes de potenciais furtos ou vandalismos das máquinas, em horário pós-laboral dos estabelecimentos. Para fundamentar esta decisão, contactam-se os clientes de modo a conhecer a forma como esperam ser notificados. O modo de notificação por correio eletrónico é uma hipótese descartada à partida, porque os utilizadores não estarão atentos ao correio, nesse horário. A possibilidade do envio de uma SMS<sup>41</sup>, também não é vista com agrado, pelos mesmos motivos, com a agravante de falsos alarmes causarem desconforto e frustração a quem os receciona. Deste modo, decide-se pelo desenvolvimento de uma solução específica e disponibilizada a uma ou mais centrais de alarmes<sup>42</sup>, para ser possível consultar, analisar e atuar perante a receção de alarmes que indiciem situações de furto, ou vandalismo.

Esta decisão implica a tomada de uma outra, que diz respeito ao modo de disponibilização da aplicação: num local comum para todas as centrais, ou num local da responsabilidade de cada central. A manutenção da aplicação é mais ágil numa distribuição centralizada. Contudo, como os custos de armazenamento, comunicações e manutenção dos servidores necessários para esse efeito não estavam orçamentados para contemplar esta aplicação, opta-se por colocar a responsabilidade do alojamento em cada central de alarmes. Esta decisão é suportada pela utilização do método AHP (c.f. secção 4.7.1 Abordagem Prática do Método AHP).

Da ponderação de todos os fatores e das respetivas tomadas de decisões, nascem as soluções Vodafone WinVending e WinGuard, objeto de estudo nesta dissertação.

## **3.2 Elementos do Sistema**

A concretização deste trabalho depende da reunião de um conjunto de elementos desenvolvidos, fabricados e cedidos pelos seus intervenientes. Nesta secção, pretende-se

---

<sup>41</sup> Do termo inglês Short Message Service.

<sup>42</sup> Mediante a abrangência de localizações, pode fazer sentido que determinada máquina reporte os alarmes a uma central de alarmes e outra máquina a outra central.

descrever os principais elementos do sistema: máquina, módulo, cartão SIM Global da Vodafone, GDSP, Middleware, WinVending, WinVending Mobile e WinGuard (cf. Figura 5).

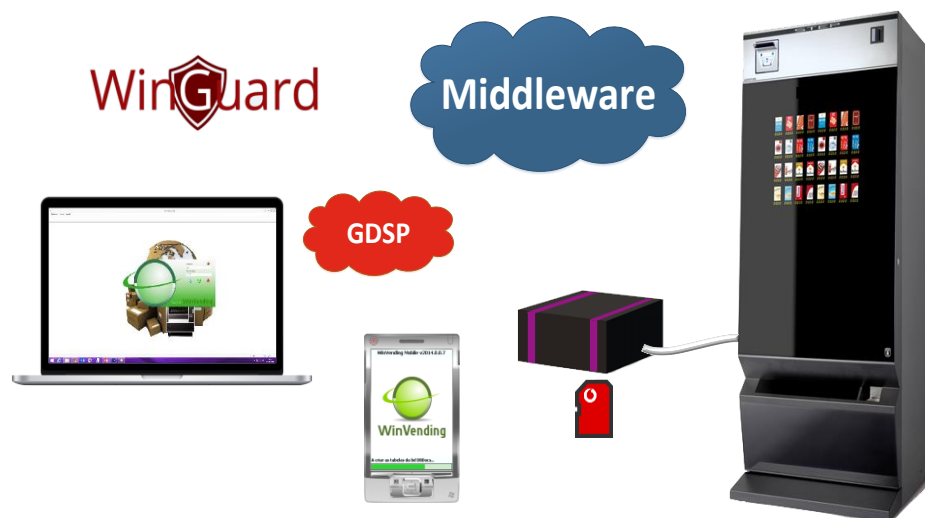


Figura 5 – Elementos do sistema

### 3.2.1 Máquina de Vending

As máquinas de Vending, alvo de estudo nesta dissertação, são equipamentos eletrónicos que dispensam, automaticamente, produtos (bebidas, comidas, selos, tabacos, entre outros) aos clientes, mediante a introdução de dinheiro, sob a forma de moeda, nota ou, mais recentemente, cartões de débito e crédito (Vending machine 2016).

Na perspetiva do cliente, observam-se, tipicamente, as seguintes partes da máquina: seleções<sup>43</sup>, ranhura de entrada de dinheiro (moedas, ou notas, ou cartão de débito e crédito), tecla de recuperação, caixa de recuperação e coletor de produto.

No processo de compra, habitualmente adotado no mercado de Vending de tabaco, o cliente introduz um valor monetário na ranhura de entrega e o ecrã informativo apresenta a quantia aceite. Os valores rejeitados caem diretamente para dentro da caixa de recuperação. Quando tiver atingido, ou superado, o preço da seleção desejada, pressiona-se a tecla correspondente

---

<sup>43</sup> Por seleção entende-se cada uma das opções que o utilizador tem para escolher um produto.

da seleção pretendida e a máquina finaliza o serviço, através da disponibilização do produto no coletor de saída e o troco, se necessário, na caixa de recuperação.

Na ótica do operador ou técnico visualizam-se, normalmente, os seguintes componentes da máquina: porta, onde é instalado o sensor de abertura / fecho, canais<sup>44</sup>, devolvedores a partir dos quais é enviado o troco ao cliente (sistema de tubos<sup>45</sup> ou sistema de hopper<sup>46</sup>), cofre, onde é acumulado o dinheiro, fonte de alimentação e a placa principal da máquina.

As máquinas disponibilizam um interface de programação que pode ser usado pelos operadores para determinar o seu funcionamento, consultar o seu estado, as suas avarias e os valores totais de vendas, devolvedores e vendas perdidas<sup>47</sup>.

Além da consulta direta, os valores das máquinas podem ser acedidos por meio externo de comunicação, por exemplo, Bluetooth. A estrutura dos dados comunicados assume diferentes formas: modelos específicos de impressão (ticket), protocolos de comunicação proprietários. A tendência é o recurso ao protocolo EVA/DTS, recomendação e standard da associação europeia de Vending (EVA<sup>48</sup>) (EVA DTS 2015).

O presente trabalho tem como intenção a comunicação com os modelos dos seguintes fabricantes de máquinas de tabaco: Azkoyen, Jofemar e GM. As máquinas Azkoyen possuem um protocolo de comunicação proprietário VTM nativo e, opcionalmente, permite o protocolo EVA/DTS. A Jofemar e a GM utilizam o sistema de tickets que, por vezes, varia de modelo para modelo.

Com o recurso à telemetria, o componente Middleware abstrai os protocolos de comunicação com a máquina da solução Vodafone WinVending; não obstante, o WinVending Mobile

---

<sup>44</sup> Os canais são formados por tabiques independentes e com uma largura regulável. A regulação dos tabiques depende da largura do maço a colocar.

<sup>45</sup> O sistema de tubos guarda as moedas em recipientes verticais que efetuam a devolução de moedas mediante um sistema de comportas.

<sup>46</sup> O sistema hopper ou rotativo possui um funil de armazenamento com mais capacidade que o sistema de tubos.

<sup>47</sup> Uma venda perdida acontece quando um cliente coloca o dinheiro para comprar o produto e após pressionar a seleção, dá-se a indicação de produto esgotado.

<sup>48</sup> EVA do termo inglês "European Vending Association"

continua a conceder a possibilidade de comunicação por Bluetooth, como mecanismo redundante de comunicação.

### 3.2.2 Módulo

O módulo é um equipamento desenvolvido pela KTS, instalado na máquina de Vending, com a finalidade de comunicar os seus resultados (vendas, contadores de moedas e notas, vendas perdidas, incidências), enviar pedidos de comunicação e programação (seleções, preços, código de operador, mensagem publicitária), bem como, transmitir os dados recolhidos dos sensores.

A comunicação é assegurada por um cartão SIM, alojado no interior do módulo e que permite a comunicação remota com o Middleware.

O conjunto de sensores que equipam o módulo é um fator diferenciador da solução. Distinguem-se os sensores: porta aberta / fechada, máquina ligada / desligada, cabo de dados ligado / desligado, GPS e movimento.

Na Figura 6, visualiza-se a parte frontal do módulo onde se destaca a ficha de ligação e os leds representativos dos diferentes estados, respetivamente, da esquerda para a direita: ligado, GPS e rede. Mais à direita, na imagem, observa-se uma patilha que permite reiniciar o módulo.



Figura 6 – Imagem frontal do módulo

A característica de possuir uma ficha de ligação do módulo concede flexibilidade de conexão a diferentes fabricantes e modelos, sem a sua alteração, derivado da possibilidade de diferenciação da cabelagem de ligação à máquina. Por exemplo, em determinados modelos de certos fabricantes, o ponto de obtenção de energia é o mesmo da comunicação com a máquina. No entanto, noutros, tal não é possível e obriga à divisão do cabo em dois: um para a energia e outro para os dados. Este último cenário justifica a utilidade do sensor de cabo de dados.

A flexibilidade e a diversidade de sensores do módulo, principalmente, o sensor de porta, implica algum conhecimento técnico, no ato da montagem. Dependendo do tipo de máquina e

da experiência do técnico, o tempo de montagem total do módulo pode variar entre 10 a 30 minutos.

### **3.2.3 Cartão SIM Global da Vodafone**

No âmbito da solução de telemetria, qualidade, robustez e fiabilidade das comunicações são fatores vitais para o sucesso do projeto. Se as comunicações falharem, a solução perde o valor proposto: redução de tempo de visita, notificação de alarmística e comunicações à distância.

Com o cartão SIM Global da Vodafone assegura-se a máxima cobertura e qualidade de rede, em todas as situações, através da capacidade de seleção automática do melhor operador para a localização específica (Conectividade M2M 2016). Acrescenta o caráter vitalício e o comportamento eternamente local, independentemente do país onde se encontre.

O cartão SIM encontra-se inserido dentro do módulo e permite a comunicação entre a máquina e o Middleware via GPRS e 3G.

### **3.2.4 GDSP**

A GDSP (Global Data Service Platform) é a plataforma global M2M da Vodafone, desenvolvida especificamente para o modelo de negócio M2M. Permite o controlo centralizado das comunicações, a gestão do tráfego e a suspensão e desativação de cartões SIM (Conectividade M2M 2016). Disponibiliza, ainda, uma API de integração com plataformas externas para a execução das operações atrás referidas.

O relacionamento entre o Middleware e as máquinas de Vending é gerido através da GDSP, que assegura a gestão de todas as comunicações e infraestruturas de rede, bem como, a manutenção dos cartões SIM via API.

### **3.2.5 Middleware**

A Figura 7 apresenta uma visão geral do sistema Middleware, descrito, posteriormente, de um modo resumido.

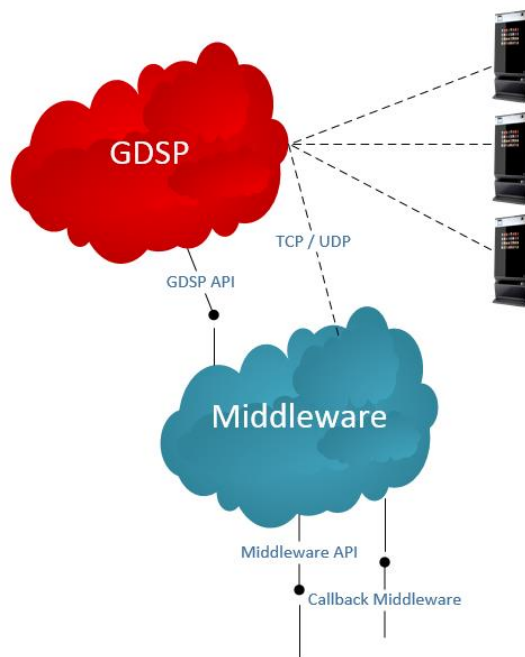


Figura 7 – Visão geral do sistema Middleware

O Middleware é um componente de *software*, desenvolvido pela KTS, que funciona como mediador entre sistemas terceiros, *backoffice* no caso em concreto, e as máquinas de Vending. Abstrai todas as especificidades de hardware das máquinas e integra, de um modo transparente, a plataforma GDSP.

A comunicação entre o Middleware e o GDSP realiza-se de três modos distintos: API, UDP<sup>49</sup> e TCP<sup>50</sup>. Via API, gere-se os cartões SIM's e respetiva ativação/desativação. Pela simplicidade e rapidez do protocolo UDP, elege-se este canal para o envio de alarmes. Utiliza-se o canal TCP pela sua robustez e fiabilidade, na execução de operações complexas, como a comunicação e a programação da máquina.

O Middleware permite a integração com outras soluções, por meio de uma API, que disponibiliza a execução de operações, como: solicitar uma comunicação, efetuar a

---

<sup>49</sup> UDP do termo inglês "User Datagram Protocol" é um protocolo simples da camada de transporte que, pelas suas características, não assegura a entrega dos pacotes enviados e, por esse motivo, não é considerado confiável (User Datagram Protocol 2015).

<sup>50</sup> TCP do termo inglês "Transmission Control Protocol" é um protocolo robusto que garante a entrega dos dados da forma correta, na sequência apropriada e sem erros, pela rede (Transmission Control Protocol 2015).

programação, alterar a mensagem publicitária, alterar o código de operador, criar uma máquina, manter o módulo, obter os alarmes, entre outros.

O envio de alarmes para aplicações terceiras concretiza-se pela utilização da técnica de *callback*. Esta técnica consiste na disponibilização, por parte de soluções terceiras, de um conjunto de serviços, através de um IP público, com uma assinatura definida pelo Middleware. No momento em que o alarme é despoletado, o Middleware invoca o serviço da entidade terceira.

### **3.2.6 WinVending**

O principal foco de incidência deste trabalho é a solução WinVending, por esse motivo, de seguida, é sintetizada a sua função e composição.

A solução é composta por um conjunto de componentes dos quais se destacam: o *backoffice* e a API. O *backoffice* é uma aplicação *desktop* que permite a configuração, programação, comunicação e monitorização das máquinas de Vending, assim como, o acompanhamento das rotas e visitas às mesmas. A API é um componente de *software* que abstrai a lógica de negócio e o acesso a dados bem como, a interligação com sistemas ERP's externos.

As suas especificidades, componentes e relacionamentos são descritos detalhadamente neste trabalho.

### **3.2.7 WinVending Mobile**

O WinVending Mobile é uma aplicação móvel, para dispositivos Microsoft, com o sistema operativo Windows CE, que disponibiliza um conjunto de operações de pré-venda e auto venda.

No âmbito deste trabalho, explora-se as características concebidas, na ótica de comunicação, com a nova versão da solução. Neste campo, destaca-se o assistente de acompanhamento do operador, durante a visita à máquina. O operador pode consultar o estado da máquina e solicitar e visualizar as comunicações com a mesma.

O paradigma de comunicação entre a aplicação móvel e o *backoffice* foi alterado. Anteriormente mencionado, o modo de comunicação com o *backoffice* consistia numa comunicação direta com a aplicação *backoffice*, por GPRS, ou por meio de sincronização de ficheiros. Agora, faz-se uso da API para invocar funcionalidades do negócio.

### 3.2.8 WinGuard

A solução WinGuard surge para dar resposta à necessidade de monitorização de alarmes, fora do horário de funcionamento dos estabelecimentos. É constituída por uma aplicação *desktop*, por uma API e por um serviço de *callback*. O aplicativo *desktop* fornece uma interface com o operador, de modo a possibilitar a consulta, em tempo real, dos alarmes e a localização das máquinas no mapa. A API implementa a lógica do negócio e o acesso a dados da solução. O serviço de *callback* recebe os dados da API do WinVending e armazena-os numa estrutura de dados dedicada. Esta aplicação pode ser instalada no ambiente do cliente ou, num ambiente externo, por exemplo, numa empresa de segurança responsável pela monitorização e tratamento dos alarmes recebidos. No decorrer deste trabalho, pormenoriza-se detalhes de conceção do WinGuard.

## 3.3 Ferramentas

A decisão da construção de uma nova solução é uma oportunidade para se redefinir as ferramentas utilizadas na análise, prototipagem e implementação da mesma.

Este capítulo justifica as escolhas das ferramentas adotadas, ponderadas segundo critérios de robustez, fiabilidade, segurança, experiência dos intervenientes do projeto, facilidade de utilização, integração entre as ferramentas e reutilização de *software* adquirido.

### 3.3.1 IDE & Linguagem de Programação

Tomada a decisão de criar, de raiz, uma nova solução, torna-se necessário a escolha do IDE (Delphi ou Visual Studio) e da linguagem da programação (Object Pascal, C# ou VB.NET).

Desde a sua aquisição até à atualidade a solução desenvolvida, evoluiu para a versão 7 do Delphi, um IDE e compilador da linguagem de programação Object Pascal, num ambiente RAD<sup>51</sup>, detentor de uma série de características que lhe permite rivalizar com IDE's, como: Eclipse e Visual Studio. Destacam-se as seguintes características: não possuir dependências, consumir

---

<sup>51</sup> RAD do termo inglês Rapid Application Development

poucos recursos de memória, apresentar um alto desempenho e ser multiplataforma (Windows, Mac, Android e IOS).

O IDE Visual Studio disponibiliza um ambiente integrado de desenvolvimento para os programadores da plataforma Microsoft .NET<sup>52</sup>, como também, para outras linguagens. Distingue-se pela independência do sistema operativo e integração entre linguagens e capacidade de agregar várias linguagens, num único ambiente.

As características mencionadas não são suficientemente relevantes para se decidir o IDE a usar. Para esse efeito, recorre-se a outros critérios: experiência da equipa de programação, componentes de interface gráfica desenvolvidos e potencialmente reutilizados, custo, confiança no fabricante e a dimensão da comunidade. Em todos os pontos mencionadas a preferência recai sobre o Visual Studio e plataforma Microsoft .Net.

Como referido, o Visual Studio suporta várias linguagens, tendo a equipa de desenvolvimento igual experiência em duas delas: C# e VB.NET. Ambas as linguagens são totalmente orientadas a objetos e têm o mesmo desempenho, fruto da geração do mesmo código intermédio (MSIL<sup>53</sup>). A principal diferença reside na sintaxe da linguagem. Para o desenvolvimento do *backoffice*, o fator de reutilização de componentes e classes de interface gráfica é determinante para a seleção da linguagem VB.NET. A linguagem C# tem uma característica admirada pela equipa de programação: ser pouco verbosa. Tal característica facilita a leitura do código, tornando-o mais limpo. Adicionalmente, tem a capacidade de definir intrinsecamente a interface nos métodos, o que agiliza a tarefa de programar. Por estes motivos, nos restantes projetos adota-se o C#.

Desta forma, a solução Vodafone WinVending na sua componente *desktop* é uma aplicação *Windows Forms* desenvolvida sobre a *Framework* .NET 4.5.1, escrita em Visual Basic .NET 2015, acrescida de um conjunto de controlos de interface gráfica que integra o componente

---

<sup>52</sup> “A .NET Framework (pronuncia-se: dotNet) é uma iniciativa da empresa Microsoft, que visa uma plataforma única para desenvolvimento e execução de sistemas e aplicações. Todo e qualquer código gerado para .NET pode ser executado em qualquer dispositivo que possua a framework.” (Microsoft .NET 2017)

<sup>53</sup> MSIL do termo inglês Microsoft Intermediate Language

Infragistics NetAdvantage 2005. Os serviços completos (API da solução, *callbacks* e serviços de monitorização) são escritos na linguagem de programação C#.

### **3.3.2 Sistema Gestor de Base de Dados**

As bases de dados da atual solução estão assentes em Paradox, um SGBD baseado em ficheiros, segundo um sistema cliente. Esta abordagem implica a manipulação dos dados por cada cliente individual, ou seja, cada instância de uma aplicação concentra em si todo o processamento. Esta característica limita a sua utilização em ambientes multiutilizadores. No entanto, para aplicações com um único ou poucos utilizadores, apresenta desempenhos elevados (Leite, 2007).

O Paradox à semelhança do Microsoft Access está inserido numa categoria de “bases de dados de ambiente de trabalho” que apresentam limitações no acesso concorrente e na dimensão das bases de dados (Todd 2011). As limitações referidas, notadas ao longo do tempo pelos administradores de bases de dados, enunciam a frequência dos casos de corrupção de ficheiros e de desempenho menos satisfatório.

Por tudo isto, a empresa vê, neste ambiente de mudança, a oportunidade de evoluir para um SGBD empresarial, com suporte para multiutilizador, dimensões de bases de dados superiores e toda a panóplia de características suportadas por esses sistemas.

Enquadra-se, neste perfil, o SGBD SQL Server, selecionado pela familiaridade da equipa de desenvolvimento e administração de bases de dados com este sistema, a que se somam o desempenho, segurança e estabilidade demonstrados, nas implementações onde é adotado.

### **3.3.3 Prototipagem**

No desenvolvimento da interface das aplicações *backoffice* e WinGuard, seguem-se diferentes abordagens de prototipagem. Enquanto, no primeiro caso, é necessário apresentar um protótipo funcional, no segundo, o importante é o desenho e interações com o utilizador.

Devido ao cariz funcional do protótipo do *backoffice*, opta-se pela sua concretização com as ferramentas finais de desenvolvimento.

No WinGuard o cliente apresenta uma necessidade definida, embora não tenha noção dos detalhes. Por esse motivo, opta-se pela criação de um protótipo, numa ferramenta fácil de usar e que possibilite uma construção célere.

Analisa-se as ferramentas: Mockinbird, Indigno Studio Lite e Gliffy. A ferramenta Indigno Studio Lite sobressai perante a Mockinbird por ser gratuita e comparativamente à Gliffy apresenta características funcionais mais atrativas, destacando-se: desenho orientado a fluxos<sup>54</sup>, livre de codificação, simples, funcional, interativa e multiplataforma. O resultado final, sob a forma de HTML<sup>55</sup>, permite a sua execução em qualquer ambiente.

### **3.3.4 Ferramentas UML**

A linguagem de modelação, orientada a objetos UML, fornece uma gama de diagramas que podem ser usados para a fase de análise, tanto a nível de sistemas quanto de software.

As ferramentas de modelagem surgem para facilitar a criação desses diagramas. Algumas têm o potencial de geração de uma parte do código-fonte, a partir do modelo e, até mesmo, a geração da base de dados com as respetivas tabelas.

Pretende-se, para este trabalho, uma ferramenta que agilize a criação dos modelos. Deste modo, elegeram-se as ferramentas: Dia, Visio e Visual Paradigm. A ferramenta Dia tem a seu favor o cariz gratuito. No entanto, no que concerne ao aspeto gráfico e facilidade de trabalho, não corresponde às expectativas. O Visio está incluído no pacote de produtos Microsoft da empresa, por isso, não representa qualquer custo. Além disso, graficamente, é apelativo, mas peca pela pouca funcionalidade e integração entre diferentes diagramas. O Visual Paradigm, embora não gratuito, tem características de desenho, funcionalidade, aspeto gráfico que a tornam a ferramenta eleita.

---

<sup>54</sup> A prototipagem, orientada a fluxos, torna o protótipo mais próximo da realidade.

<sup>55</sup> HTML do termo inglês HyperText Markup Language.



## 4 Análise do Valor do Negócio

Inicia-se este capítulo com a referência ao processo de inovação, particularmente, a primeira fase, denominada modelo de desenvolvimento de novo conceito, cujas componentes são exemplificadas no contexto da solução em estudo. Refere-se a Implementação do modelo de negócios de Canvas proposto por Alexander Osterwalder (Osterwalder 2016). Através do modelo de negócio, define-se o modo como a empresa cria valor, entrega ao cliente e gera receita. Descreve-se a relação entre os benefícios e os sacrifícios, com ênfase para os segmentos de clientes detetados no modelo de negócio. Apresenta-se a proposta de valor da solução. Expõe-se e justificam-se cenários de negócio possíveis. Esquematiza-se a rede de valores da solução Vodafone WinVending. No final, identificam-se e justifica-se as formas de analisar/modelar e/ou quantificar a criação de valor, apoiado na adoção do método quantitativo AHP (Analytic Hierarchy Process), na resolução de um problema com que a ABS Informática se defronta, no decorrer da seleção da ideia: qual o método de notificação de alarmes a utilizar?

### 4.1 Modelo de Desenvolvimento de Novo Conceito

No atual cenário de globalização e competitividade, as organizações percebem a importância e pertinência da geração e incorporação de inovações nos seus produtos e serviços, como forma de manter a satisfação dos seus clientes e atrair novos.

Entende-se por inovação, o desenvolvimento de algo com valor acrescentado para os clientes (atuais ou novos). Destacam-se dois tipos de inovação: a inovação radical que adiciona valor, nunca antes explorado, caracterizado por uma rutura nas tecnologias e métodos existentes e a inovação incremental que adiciona valor ao atual produto ou serviço, por via do lançamento de

novas versões que melhoram, significativamente, o seu desempenho ou, por meio da reconfiguração de uma tecnologia já existente para novos propósitos (Williams & Staal 2016).

#### **4.1.1 Processo de inovação**

Tradicionalmente, o processo de inovação de produtos divide-se em três fases: a proposta teórica que reúne as atividades de pré-desenvolvimento, denominada de “Fuzzy Front End” (FFE)<sup>56</sup>, a conceção técnica, onde a anterior proposta é transformada num produto ou serviço, também conhecida por “New Product Development” (NPD) e a comercialização que concretiza todo o processo de inovação (Vigna 2015).

A primeira fase, compreende as etapas desde o surgir da ideia, ou identificação da oportunidade, até à aprovação do projeto para desenvolvimento. Esta fase, caracteriza-se pela incerteza e dinamismo das informações utilizadas e pela dificuldade na gestão do processo que apresenta um baixo nível de formalização. Por se tratar da fase inicial do processo, a tomada de decisões tem impacto no produto como um todo. Como consequência, decisões como alterar ou cancelar o projeto, devem ser tomadas o mais cedo possível, sob pena do custo aumentar exponencialmente, à medida que se avança no ciclo de desenvolvimento do novo produto (Ferreira 2010).

Diversos autores desenvolvem modelos de planeamento de inovação, que descrevem como realizar as atividades e sugerem o modo de diminuir as incertezas e guiar o fluxo de informação. De entre os propostos, destaca-se o modelo de relacionamento circular e não sequencial de Peter Koen “the New Concept Development” (NCD), que consiste em três partes principais: o motor, os fatores de influência e as cinco atividades internas da empresa.

---

<sup>56</sup> A primeira fase do processo de inovação é mencionada, no decorrer do projeto, pelos seguintes termos sinónimos: Fuzzy Front End (FFE) e planeamento de inovação.

### **4.1.2 Motor**

O motor corresponde aos fatores liderança, cultura e estratégia de negócio da empresa. A sua responsabilidade reside na condução dos cinco elementos internos da organização, por caminhos externos que fogem ao seu controlo.

A liderança da ABS Informática, composta pelo seu corpo gerente, envolve toda a organização em volta da compreensão da inovação, como forma de acrescentar valor aos seus cliente e fá-lo ao partilhar histórias<sup>57</sup> que reforçam os princípios e práticas da inovação. A aposta em produtos de cariz inovador demonstra também a importância que a inovação tem para a empresa e é complementada com a liberdade de atuação concedida aos seus funcionários, para a exploração de ideias, com potencial de adicionar valor às atuais soluções. A cultura da empresa fomenta uma relação de cumplicidade e proximidade com os seus clientes, o que permite o conhecimento das suas necessidades e desejos e cria um potencial de geração de ideias que ocasionam um nível de inovação. A postura de crescimento que a empresa adota nos mercados onde atua, motiva e incute na equipa um espírito de responsabilidade e inovação como fator de diferenciação e competitividade.

Acréscem-se as seguintes ferramentas, técnicas e métodos para o complemento de uma cultura colaborativa, encorajadora de criação de conhecimento: comunidades de prática<sup>58</sup>, ferramentas facilitadoras do contacto entre pessoas (ex. Skype, Microsoft Teams) e o ambiente de trabalho colaborativo (Koen et al. 2002).

### **4.1.3 Fatores de influência**

O planeamento da inovação encontra-se envolvido por um conjunto de fatores que influenciam a sua atuação. O ambiente externo, as competências organizacionais da empresa, os clientes, a

---

<sup>57</sup> Um dos gerentes da empresa partilhou uma lição de inovação da Netflix, ao destronar o seu direto concorrente Blockbuster, com uma visão inovadora do futuro do mercado de aluguer de filmes (Scott 2017).

<sup>58</sup> As comunidades de prática são compostas por um conjunto de pessoas que discutem temas de interesse comum

concorrência, as tendências socioeconómicas, os avanços tecnológicos e as políticas governativas afetam o modo como a empresa lida com a inovação.

A ABS Informática vive num ambiente repleto de oportunidades encorajadoras para inovar a sua solução de Vending e pretende abraçá-las e incluí-las no seu produto. A tecnologia de M2M encontra-se num estado de evolução maduro e fiável. A Vodafone age como um agente facilitador tecnológico ao proporcionar a comunicação remota com dispositivos a uma velocidade, estabilidade e custos que permitem a sua implementação. Os fabricantes de máquinas de tabaco estão dispostos a partilhar os seus protocolos de comunicação privados e os clientes confiam na mais-valia da comunicação remota no seu negócio. Contudo, algumas ameaças espreitam do exterior e necessitam de uma avaliação e conhecimento. Por um lado, a concorrência aposta na mesma área de inovação tecnológica, por outro lado, as políticas e leis governamentais podem colocar em causa a venda de tabaco em máquinas automáticas.

Os fatores de influência são, em grande parte, incontrolláveis pela organização, no entanto, a resposta dada pelo coração da empresa sobre os cinco elementos do NCD pode inverter situações aparentemente desfavoráveis (Koen et al. 2002). No caso prático do objeto de estudo, comportamentos sociais como o assalto a máquinas e carrinhas transportadoras de tabaco são encarados como uma ameaça ao setor de Vending. Contudo, a ABS Informática vê-as como uma oportunidade para melhorar o seu produto e melhor servir o seu cliente.

Consideram-se as seguintes abordagens eficazes para enfrentar os fatores de influência: a capacidade de executar a estratégia ou plano de ação, quando ocorrem mudanças e o estabelecimento de parcerias e alianças (Koen et al. 2002). Por exemplo, a ABS Informática estabelece parcerias com a Vodafone e a KTS para implementar, no menor espaço de tempo, uma solução de telemetria que satisfaça as necessidades e desejos dos seus clientes.

A parte interior do modelo é constituída por cinco elementos: identificação de oportunidades, análise de oportunidade, geração de ideias, seleção de ideias e a definição do conceito.

#### **4.1.4 Identificação das oportunidades**

Na identificação de oportunidades, a empresa reconhece as oportunidades a perseguir orientadas para os seus objetivos de negócio. Recorre a processos formais de identificação de oportunidades alinhados com os fatores de influência, de entre os quais se refere: a análise das

tendências tecnológicas, dos clientes e da concorrência, pesquisas de mercado, mapeamento de cenários futuros e métodos de resolução de problemas<sup>59</sup> (Koen et al. 2002). Por exemplo, a ABS Informática, através da análise de tendências, identifica a oportunidade de incorporar a tecnologia de M2M na sua solução de Vending, de modo a acelerar processos operacionais e reduzir custos de gestão de máquinas de tabaco. Um outro exemplo: através da observação e contacto com seus clientes e da análise do meio envolvente, a empresa nota que os atos de roubo e vandalismo sobre as máquinas causam transtornos económicos aos seus clientes que se sentem inseguros. Incluem-se, também, atividades informais de identificação de oportunidades, das quais se destacam: conversas informais e em fóruns e intuição individual.

#### **4.1.5 Análise de oportunidades**

A análise de oportunidades consiste na avaliação das oportunidades de modo a confirmar se vale a pena prosseguir com o envolvimento e investimento da organização. A tradução da identificação de uma oportunidade em algo específico para o negócio carece da avaliação das tendências tecnológicas, de mercado e da concorrência, bem como, do estudo das necessidades dos clientes. Nesta fase, são, igualmente, avaliadas as competências e capacidades da empresa para abraçar as oportunidades (Koen et al. 2002). Por exemplo, a ABS Informática determina que as necessidades de segurança e monitorização remota de máquinas de tabaco não são atualmente satisfeitas pelo seu produto WinVending e que, os seus concorrentes com tecnologia de telemetria, não respondem a alguns requisitos, como: a comunicação com as máquinas de todos os fabricantes, a ligação a diferentes ERP's e o acesso à localização física da máquina. Na mesma altura, a empresa valida os conhecimentos técnicos da sua equipa e a necessidade de expandir, futuramente, o número de elementos da equipa de suporte e desenvolvimento.

As mesmas ferramentas utilizadas na identificação das oportunidades são agora invocadas na sua análise, mas, desta feita, com maior detalhe e recursos envolvidos para a sua obtenção.

---

<sup>59</sup> Um dos métodos de resolução de problema é o diagrama de espinha de peixe que identifica as relações entre a causa e o efeito.

Apesar, do esforço alocado, algumas incertezas de mercado e tecnologia mantêm-se (Koen et al. 2002).

#### **4.1.6 Geração de ideia**

A geração de ideia consiste num processo iterativo e evolucionário, onde as oportunidades são transformadas em ideias que, posteriormente, são: combinadas, reformuladas, destruídas, modificadas e atualizadas, até se alcançar um ponto de satisfação do cliente e se verificar a sua viabilidade de produção e comercialização. Destacam-se as seguintes técnicas, métodos e ferramentas de geração de ideias: discussões de grupo, banco de ideias, observação e entrevista com os clientes, envolvimento dos clientes líderes de opinião, identificação de novas soluções tecnológicas, cultura da empresa ao encorajar os seus funcionários para validar as suas ideias, incentivos para estimular a criação de ideias, rotação frequente de funções dos funcionários na empresa, para fomentar a partilha de conhecimentos e a inclusão de pessoas com diferentes perfis no enriquecimento de ideias (Koen et al. 2002).

Nesta fase, a ABS Informática cria um conjunto de ideias para implementar a tecnologia de telemetria nas máquinas de tabaco e aproveitar o seu potencial, para conceder aos seus clientes a capacidade de identificar atos de roubos e vandalismos. Tais ideias passam por comunicar todos os alarmes suspeitos <sup>60</sup> através de: notificações SMS e / ou correio-electrónico direccionadas para utilizadores específicos, comunicação com uma solução onde os utilizadores podem visualizar o posicionamento e alarmes gerados pelas suas máquinas e comunicação com sistemas externos de empresas de segurança. No processo de geração de ideias, a ABS Informática envolve, desde o início do processo, os seus clientes líderes de opinião<sup>61</sup> e identifica as suas necessidades através de técnicas etnográficas de observação e entrevista (Rushikesh 2008).

---

<sup>60</sup> Na fase de geração de ideias o conceito de alarmes suspeitos ainda não está claramente definido. Será clarificado na fase de análise da solução.

<sup>61</sup> Os utilizadores pioneiros e líderes de opinião são bem conhecidos no mercado de venda automática de tabaco e, alguns deles, são clientes da solução WinVending.

#### **4.1.7 Seleção da ideia**

A seleção da ideia consiste em determinar, do conjunto de ideias geradas, aquelas com potencial de atingir maior valor para o cliente. Consideram-se os seguintes critérios de seleção: risco tecnológico, custos envolvidos, análise da concorrência, capacidade e competências da empresa e retorno financeiro (Vigna, 2015).

Recorre-se a metodologias de múltiplos fatores baseadas em escalas de ancora<sup>62</sup> para selecionar ideias, de entre as quais se refere: probabilidade de sucesso técnico e comercial, recompensa, ajuste estratégico e alavancagem estratégica (Koen 2002).

Por exemplo, na seleção da ideia de incorporação de uma componente de segurança na solução WinVending, o dono do projeto opta pela criação de uma nova aplicação de monitorização de alarmes, a instalar na infraestrutura da empresa de segurança, pela credibilidade e experiência no controlo alarmístico, oferecido por uma empresa especialista do ramo e pela redução de custos de manutenção da plataforma, que não são suportados pela organização.

#### **4.1.8 Definição do conceito**

A definição do conceito é o último elemento do NCD e fornece o único ponto de saída para o NPD. Nesta fase, o inovador deve apresentar um caso convincente para o investimento no negócio ou tecnologia propostos, o qual deve abordar os próximos pontos: objetivos a atingir, enquadramento do conceito na estratégia da empresa, impacto financeiro, necessidades e benefícios para o mercado ou cliente, fatores de risco comercial e técnico, meio envolvente e apresentação de um projeto que inclua tempos de desenvolvimento e recursos necessários (Koen 2002).

Utilizam-se os seguintes métodos e técnicas: definir os critérios para a organização que descrevem o quão atrativo o projeto é nas diversas dimensões (financeira, mercado),

---

<sup>62</sup> “Escalas ancoradas são medidas ordinais que utilizam indicadores numéricos, cada um dos quais associado a um conjunto de palavras que auxiliam a pessoa que responde a “ancorar” a sua avaliação.” (tradução nossa) (Koen 2002)

compreender e determinar as limitações da tecnologia, envolver o cliente, desde o início, em testes reais do produto, manter o foco e não dispersar com pormenores (Koen 2002).

No exemplo em estudo, estabelecem-se duas saídas do modelo NCD para o NPD: uma nova solução de telemetria, que complementa a atual aplicação de gestão de máquinas de tabaco e uma nova solução de monitorização de alarmes, a implementar nas empresas de segurança.

O NCD permite aos líderes empresariais e tecnológicos otimizarem as atividades do FFE. O resultado que daí advém deve apresentar um maior número de conceitos lucrativos que passam para a fase NPD e beneficiam a organização (Koen 2002).

## **4.2 Modelo de Negócio de Canvas**

O modelo de negócio de Canvas é um mapa de planeamento estratégico visual e pré-formatado composto por nove blocos. Nesta parte, pretende-se contextualizar cada bloco no ambiente da solução Vodafone WinVending.

Começa-se por identificar o segmento de clientes. Da análise do mercado, distinguem-se três: auto venda, pré-venda e pré-venda estatística. Esta divisão do segmento de proprietários de máquinas de Vending deve-se ao facto do valor, que cada segmento extrai da solução, não ser, exatamente, o mesmo.

Identifica-se, de seguida, o que a empresa oferece para o mercado que tem valor para os clientes: redução de custos, redução do risco, poupança de tempo, segurança, conforto, apoio à decisão, ambiente e pioneirismo.

A redução de custos consegue-se através da diminuição do número de quilómetros percorridos, da diminuição do tempo de inatividade das máquinas e da recuperação de máquinas roubadas. O primeiro argumento está diretamente relacionado com a criação do valor ambiental pela redução das emissões de dióxido de carbono.

A redução do risco é possível pela diminuição dos stocks em circulação nas viaturas, deste modo, em caso de assalto os valores perdidos são menores.

A poupança de tempo concretiza-se através de visitas mais rápidas e da eliminação de deslocações desnecessárias. Libertam-se, assim, recursos humanos para a execução de outras funções.

O valor segurança é conseguido pela monitorização dos alarmes da máquina e pelo conhecimento da sua posição geográfica. Este controlo condiciona atos de roubos e vandalismo e auxilia na recuperação das máquinas roubadas ou, pelo menos, dos seus componentes funcionais.

O conforto e o apoio à decisão concretizam-se pelo fato do proprietário da máquina não ter de se deslocar à máquina, para conhecer o seu estado e níveis de *stock*.

O pioneirismo atinge-se ao conceder, ao cliente, a posse de uma tecnologia de ponta, que é novidade no setor e permite uma diferenciação tecnológica, face aos seus concorrentes.

Pretende-se que a proposta de valor, anteriormente identificada, chegue aos clientes através dos seguintes canais: site, equipa de comerciais, equipa de assistência técnica (telefone, email), formadores e consultores.

Estabelece-se a relação com os clientes da solução, por meio de formações e no apoio de consultadoria prestado, pelo atendimento telefónico e/ou por respostas a questões colocadas, por correio eletrónico. O atendimento pessoal prefere-se na fase de arranque do sistema, enquanto que o personalizado pode justificar-se em clientes de elevada dimensão e atribuir um elemento da equipa para gerir a comunicação.

As receitas da solução advêm dos seguintes produtos e serviços: mensalidade por máquina que varia consoante o número de máquinas contratadas para telemetria, licença do *backoffice* por utilizador, serviço de consultadoria, cobrado à hora, e venda do módulo a instalar por máquina.

Consideram-se os seguintes recursos principais para a entrega da proposta de valor: humanos (especificamente, programadores e consultores), *software* de desenvolvimento e as marcas Vodafone e ABS Informática, respetivamente, na área de telecomunicações e no negócio de máquinas de Vending de tabaco.

Enuncia-se as principais atividades que asseguram a entrega da proposta de valor: desenvolvimento de *software*, apoio ao cliente em caso de dúvidas ou falhas, consultadoria (para o auxiliar na obtenção do máximo partido da solução), plataforma do cliente (que aloja a base de dados) e do middleware (que gere todas as comunicações entre os clientes e as máquinas) e, por último, a rede de comunicações, por onde circulam todos os dados comunicados.

Esta solução é construída, como anteriormente referido, numa parceria a três (ABS Informática, KTS e Vodafone) onde cada parceiro tem um papel bem definido, no entanto, com uma dependência e interligação, que exige a constante troca de ideias e tratamento do *feedback* do cliente relativamente a cada uma das partes envolvidas. Os fabricantes das máquinas de tabaco são parceiros chave, na medida em que partilham os protocolos específicos de comunicação, peças fulcrais para uma comunicação com as máquinas. A importância dos fabricantes dos componentes que equipam o módulo a instalar na máquina é notória na exposição do modo de funcionamento e restrições dos componentes, bem como nos prazos de entrega das quantidades encomendadas.

Os custos da solução dividem-se em dois grupos: fixos e variáveis. Os fixos correspondem aos gastos com recursos humanos dos quais se destacam: os programadores, a equipa de assistência técnica, os vendedores e os consultores. Os custos variáveis são aqueles que suportam a construção dos equipamentos a instalar nas máquinas e variam de acordo com as encomendas.

As ideias representadas nos nove blocos definem a forma como se pretende operar e gerar valor, para o mercado de Vending de máquinas de tabaco. Consegue-se este objetivo através da definição dos principais fluxos e processos, que permitem uma análise e visualização do modelo de atuação, no mercado específico.

Visualiza-se na Figura 8 a concretização do modelo Canvas.

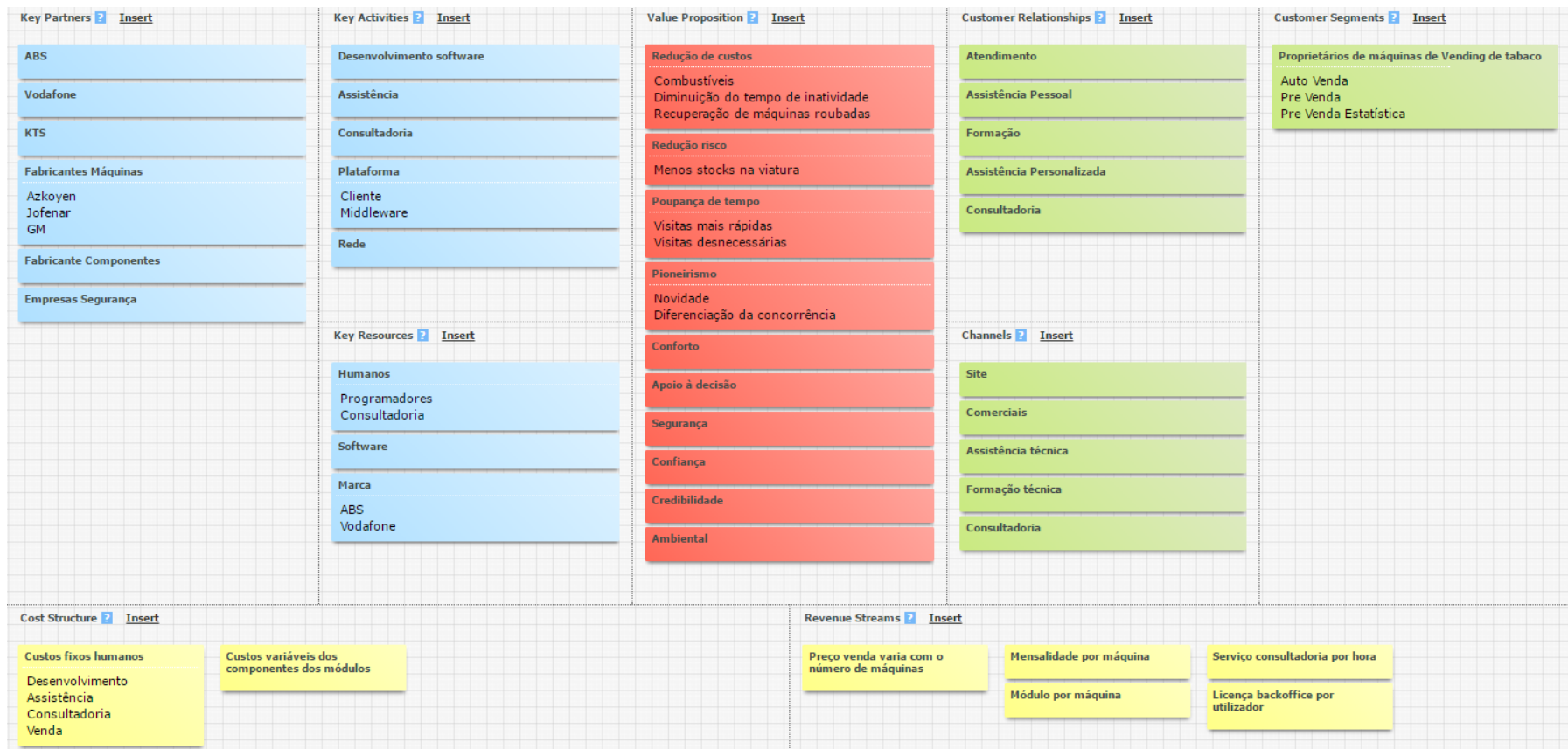


Figura 8 – Modelo de negócio CANVAS – Vodafone WinVending

### 4.3 Benefícios e Sacrifícios

O valor percebido pelo cliente é comumente apresentado pela diferença entre os benefícios e os sacrifícios. De uma outra perspectiva, os ganhos (aquisição do produto ou serviço) devem compensar as perdas da aquisição (custos e risco). De modo a aferir-se o valor percebido, realiza-se, nesta secção, um levantamento dos benefícios e sacrificios dos clientes-alvo.

Clarifica-se os benefícios específicos de cada segmento. Assim, no modelo de auto venda, reduz-se o *stock* da viatura pelo conhecimento antecipado, das necessidades da máquina, o que evita a posse de produtos excedentes. O tempo de visita é igualmente reduzido, pelo facto, do operador ter, na sua posse, os produtos já preparados pelo armazém. Desta forma, elimina-se a necessidade de regresso à viatura para colmatar alguma falta de produto.

Em pré-venda reduz-se, aproximadamente, 50% o número de visitas às máquinas, devido à subtração de uma visita à máquina para o conhecimento das suas necessidades de produtos e trocos. Com a nova solução, os valores usados como base para o cálculo dos abastecimentos são obtidos por leituras remotas. Consequentemente, os custos de combustíveis reduzem para metade e o ambiente ganha.

Na pré-venda estatística reduz-se o *stock* circulante e o tempo de visita. Os ganhos não são tão evidentes como em auto venda, contudo, são perceptíveis pela eliminação do fator de aleatoriedade do consumo de tabaco, que obriga a um aumento de *stock* na viatura, para fazer face às diferenças observadas. Em simultâneo, pode evitar uma deslocação desnecessária à viatura, para obter produtos consumidos imprevisivelmente.

Na Tabela 4 visualizam-se os benefícios particulares de cada segmento e os benefícios gerais comuns a todos os segmentos.

Destacam-se os seguintes benefícios gerais da telemetria: diminuição de custos, poupança de tempo, diminuição de risco, maior segurança, melhor ambiente e conforto, apoio à decisão e pioneirismo.

O benefício de poupança de tempo é perceptível, quer pela diminuição do número de visitas, quer por meio da redução do tempo de visita e, em alguns casos, por prevenir visitas desnecessárias. Tal benefício, liberta recursos humanos para a execução de outras funções.

O aumento do tempo de atividade da máquina reduz os custos da sua inatividade. Este aumento obtém-se pela observação de avarias detetadas, através de leituras remotas.

Tabela 4 – Benefícios Telemetria

Regime	Pré Venda Telemetria (Benefícios)	
Auto Venda	Redução de stock circulante (25%-30%)	Pioneirismo Segurança Conforto Apoio à decisão Libertação de recursos
	Redução no tempo de visita (25%-35%)	
	Redução de prejuízos em caso de assaltos	
Pré Venda	Redução do número de visitas (50%)	
	Redução de custos em combustíveis (50%)	
	Melhor ambiente	
Pré Venda Estatística	Redução de stock circulante (15%-20%)	
	Redução no tempo de visita (10%-25%)	
	Redução de prejuízos em caso de assaltos	

No que diz respeito aos sacrifícios, dividem-se em dois tipos: monetários e não monetários. Dos monetários realçam-se: mensalidade por máquina, licença *backoffice* por utilizador, licença móvel por dispositivo, módulo por máquina, custo de reparação e serviço de consultadoria. Os não monetários referem-se: ao tempo despendido na instalação / reparação do módulo e na formação aos utilizadores, para montagem do módulo.

O risco incorrido pelo cliente na aquisição desta solução deve-se à incerteza de fazer a escolha certa e do custo oneroso que advém de uma escolha errada.

Da diferença entre os benefícios e os sacrifícios e da ponderação dos eventuais riscos, o cliente percebe o valor da solução.

#### 4.4 Proposta de Valor

Depois de se conhecer em profundidade quem é o cliente e o seu problema, é possível avançar-se para o estudo da proposta de valor.

A seleção de uma proposta de valor é uma peça central para a estratégia de negócio de um produto, uma vez que, define o seu objetivo, preciso e específico (Lanning 2003).

Na definição da proposta é importante que o fornecedor se coloque na posição do cliente e que interprete as suas necessidades, expectativas e experiências. Com base nessa percepção deve desenhar e contruir uma proposta de valor que crie, na mente do cliente, o ambiente procurado e valorizado.

Numa proposta de valor, nem tudo é crítico e, o que importa salientar, na sua construção, são as experiências onde a empresa gera valor percebido pelo seu cliente. Segundo Kotler, o valor percebido é o valor atribuído pelos clientes ao produto ou serviço, baseado na relação entre os benefícios que trará, segundo a visão do consumidor, e os custos para a sua aquisição (Sigfried 2000).

A definição da proposta de valor tem implicações diretas na decisão do cliente levando-o a escolher uma solução em detrimento de todas as outras. A consciência da relevância da proposta de valor para o lançamento e apresentação da solução focada, nesta dissertação, leva à sua criação, cujo resultado se reflete no próximo parágrafo.

Vodafone WinVending é uma solução de gestão e monitorização de máquinas de venda automática de tabaco à distância e em tempo real. Permite aos depositários de tabaco: reduzir até 50% o número de quilómetros percorridos, diminuir o *stock* nas viaturas em 20% e detetar atos de roubo ou vandalismo. É a primeira solução do mercado nacional a aliar, a todas estas características, a capacidade de integração com o ERP do cliente.

Da análise, do anterior parágrafo, identifica-se sumariamente as dimensões da proposta: produto, clientes-alvo, valor proposto e carácter único. O produto é uma solução de gestão e monitorização de máquinas de venda automática de tabaco à distância e em tempo real. Os clientes-alvo são os proprietários das máquinas. Os valores propostos são a redução de custos através da diminuição dos quilómetros percorridos, a diminuição de risco pela redução dos *stocks* nas viaturas, a segurança com a deteção de roubos ou vandalismo e a integração com qualquer ERP. O que torna o produto único é a agregação de todas as características, isto é, cada uma das características, isoladamente, encontra-se presente na concorrência. No entanto, não existe nenhuma solução que concentre, em si, todas elas.

Para medir a eficácia e recetividade da proposta de valor, pode questionar-se, diretamente o cliente, acerca da sua opinião sobre a proposta de valor. “Não é fácil calçar os sapatos do cliente o melhor é ouvi-lo.” (Cruz 2011)

A proposta de valor deve ser partilhada e enraizada em todos os elementos da equipa e parceiros, bem como, estar presente em todos os processos de comunicação do produto, independentemente do canal e da forma.

## **4.5 Cenários de Negócio**

O contexto global de mudança onde a solução se insere encontra-se repleto de oportunidades e riscos. Visualizar e administrar os riscos pode garantir a continuidade do negócio (Análise de Cenários 2014).

Para enfrentar o ambiente de incerteza em que se insere, segue-se uma estratégia de articulação de possibilidades credíveis e internamente consistentes, designada por análise de cenários. Esta análise auxilia o entendimento dos riscos a médio / longo prazo.

A concretização da análise de cenários passa pela eleição de cenários de maior probabilidade e pelo desenho de diferentes estratégias. Assim, prepara-se um cenário pessimista, com quebras inesperadas no mercado e dois cenários otimistas de aumento da procura.

O cenário pessimista relaciona-se com a pressão política de impedimento da venda de tabaco em máquinas de Vending. Em Inglaterra, a 1 de outubro de 2011, o governo britânico proibiu a venda de tabaco em máquinas de Vending, com a justificação da inadequação dos mecanismos presentes nas máquinas, para travar o ímpeto de consumo dos menores de idade (ASH 2011). Perante este cenário, os fabricantes das máquinas criam mecanismos eletrónicos para o controlo da idade do comprador de tabaco. Em Portugal, as máquinas de venda automática de tabaco têm algumas especificidades, por força da lei n.º 37/2007, de 14 de Agosto, nomeadamente: a disponibilização de um dispositivo eletrónico, ou bloqueador similar, que impeça o acesso a menores de 18 anos (Máquinas de Vending 2016).

De modo a minimizar o impacto de tal lei, tenciona-se abranger outros segmentos de máquinas de Vending e pretende-se expandir o âmbito geográfico de atuação por meio da internacionalização.

Num cenário otimista, a procura da solução pode chegar a clientes de maior dimensão que trabalham com ERP's para grandes empresas. Para este cenário, a arquitetura da aplicação deve estar preparada para a incorporação de diferentes ERP's, de um modo transparente, para os restantes componentes da solução.

O cenário mais otimista equacionado é a possibilidade de algum fabricante de máquinas de Vending manifestar interesse em agregar a solução apresentada, como complemento à venda da máquina. Neste cenário, é necessário, eventualmente, a implementação centralizada da solução com uma camada de apresentação diferente, nomeadamente, baseada na Web.

Em todos cenários delineados, a força de trabalho (especificamente programadores) pode aumentar. Esta decisão deve ser refletida e planeada. Invoca-se a lei de Brooks para reforço do fundamento: “adicionar pessoas a um projeto atrasado, na verdade atrasa ainda mais o projeto” (tradução nossa) (Brooks 1995). Segundo o autor, o número máximo de pessoas no projeto determina-se de acordo com o número de tarefas independentes que o constituem (Brooks 1995).

Adota-se a previsão de cenários para preparar a empresa com um “plano B”, de modo a enfrentar potenciais riscos e maximizar o usufruto das oportunidades que surjam.

## **4.6 Rede de Valor**

“As pessoas naturalmente trabalham em rede, então porque não modelar o trabalho em si como uma rede?” (tradução nossa) (Allee 2014). A autora desta citação recupera um comportamento natural do ser humano, para gerir qualquer atividade de negócio. Assim, desenvolve uma metodologia, denominada “análise de rede de valor”, com o intuito de entender, usar, visualizar e otimizar, interna e externamente, a rede de valor de uma organização.

Uma rede de valor gera valor por meio de complexas e dinâmicas trocas, entre dois ou mais indivíduos, grupos ou organizações. Essas redes envolvem transações em torno de ativos tangíveis (bens, serviços e receitas), ou intangíveis (conhecimento e benefícios) (Value network 2016).

A ABS Informática crê que o sucesso da sua solução passa pela construção de uma rede de relacionamentos confiável e estável e, guiada por esse objetivo, procura estabelecer laços de confiança com parceiros que transmitam credibilidade e segurança, nos ramos onde atuam, como: em telecomunicações a Vodafone, no fabrico de máquinas de tabaco a Azkoyen, Jofemar e GM e no fabrico de módulos equipados com sensores a KTS. A organização concede e recebe um conjunto de valores, que agrega à solução, para satisfazer as necessidades dos seus clientes.



que o cliente pode experimentar. Relativamente aos ganhos, cabe surpreender e compreender os benefícios que o cliente espera e deseja (ganhos sociais, utilidades funcionais, emoções positivas e redução de custo). Assim, ao longo do processo, é possível visualizar e construir diferentes cenários e hipóteses de acordo com o cliente e segmento potencial. Recomenda-se que este modelo seja utilizado em conjunto com o Canvas do Modelo de Negócio (Canvas da Proposta de Valor 2015).

Na conceção do produto, é necessário tomar decisões estratégicas que podem passar pela escolha da adição de valor à solução, com determinado custo. No auxílio da ponderação entre os benefícios de determinado valor face aos custos que representa, usa-se o método quantitativo de apoio à decisão: a teoria dos jogos.

A teoria de jogos é um estudo de como matematicamente se determina a melhor estratégia, a partir de um conjunto de condições, com o objetivo de otimizar o retorno (Nicola<sup>63</sup> 2016). Na situação particular que se analisa, tem-se pela frente um cenário de decisão de adição de valor. Desse modo, é possível utilizar o tipo de jogo onde ambas as partes ganham (Win-Win). Visualizam-se as próximas vantagens neste jogo. Primeiro, obtém-se a possibilidade de existência de oportunidades de negócio não exploradas. Segundo, como o cliente não perde valor, oferece uma menor resistência a futuras abordagens. Terceiro, como este modelo não força os intervenientes a retaliarem, torna o negócio mais sustentável (Azevedo, Carvalho & Silva 2016). Pretende-se, com esta decisão, manter um relacionamento duradouro com o cliente, de modo a garantir a sua cooperação em futuros projetos (Nicola<sup>64</sup>2016).

Clarifica-se a existência de vários métodos quantitativos de apoio à decisão que se enumeram: teoria dos conjuntos aproximados<sup>65</sup>, teoria dos conjuntos nebulosos<sup>66</sup> e vários métodos de apoio multicritério, dos quais se destaca o AHP, cuja análise e exemplificação é aprofundada, neste subcapítulo.

---

<sup>63</sup> Apontamento Análise de Valor de Negócio / Negotiation

<sup>64</sup> Apontamento Análise de Valor de Negócio / Methods

<sup>65</sup> Teoria dos conjuntos aproximados é um modelo matemático para representação do conhecimento e tratamento de incerteza.

<sup>66</sup> O conceito formal de conjunto nebuloso é uma teoria capaz de expressar fenómenos vagos e incertos.

#### 4.7.1 Abordagem Prática do Método AHP

Define-se AHP como um método de decisão multicritério, que recorre a técnicas numéricas de auxílio aos decisores, na escolha de uma opção de um conjunto discreto de alternativas. É um processo efetuado com base no cruzamento das alternativas com os critérios existentes (Methods 2016). A metodologia é concretizada por etapas: definição do objetivo, distinção dos critérios, identificação das alternativas e o seu relacionamento com os critérios e a aplicação do processo AHP (Nicola<sup>67</sup> 2016).

Observa-se na Figura 10 um cenário real onde é evidenciada a decomposição de um problema de decisão da solução em estudo. O problema em causa relaciona-se com a decisão de qual o método de notificação / monitorização de alarmes a incluir na solução. Selecionam-se os critérios de decisão: (i) rapidez de desenvolvimento, (ii) custo de implementação e manutenção da solução incutidos à parceria, (iii) qualidade das características disponibilizadas e (iv) flexibilidade de manutenção, instalação e acesso aos alarmes despoletados. De seguida, identifica-se as alternativas para resolver o problema: (i) implementar um sistema de envio automático de SMS's, aquando da ocorrência de um alarme categorizado como prioritário, para utilizadores específicos, (ii) desenvolvimento de uma solução web centralizada no servidor dos fabricantes e disponibilizada aos clientes, (iii) implementação de uma aplicação *desktop* a instalar na plataforma do cliente, ou outra entidade responsável pela monitorização dos alarmes.

---

<sup>67</sup> Apontamento Análise de Valor de Negócio / Methods

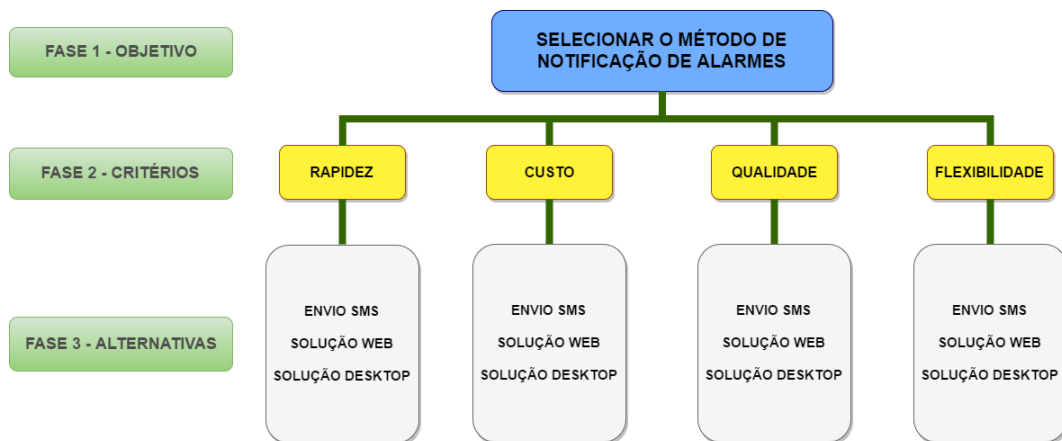


Figura 10 – Diagrama de Seleção do Método de Notificação de Alarmes

#### 4.7.2 Definição de Prioridades

Na definição de prioridades, solicita-se aos decisores do projeto que comparem os pares de critérios, com o objetivo de determinar a importância relativa<sup>68</sup> de um critério sob o outro. Apresenta-se o resultado dessa comparação na Tabela 5.

Tabela 5 – Níveis de importância de comparações binárias (<sup>69</sup>Nicola 2016)

Nível de importância	Definição	Explicação
1	Igual importância	As duas atividades contribuem igualmente para o objetivo
3	Fraca importância	A experiência e o julgamento favorecem levemente uma atividade em relação à outra
5	Forte importância	A experiência e o julgamento favorecem fortemente uma atividade em relação à outra
7	Muito forte importância	Uma atividade é muito fortemente favorecida em relação a outra

<sup>68</sup> Utilizam-se os níveis de importância de comparações binárias constantes Tabela 5, para determinar o grau de importância relativa de um valor face a um outro

<sup>69</sup> Apontamento Análise de Valor de Negócio / Methods

<b>9</b>	Importância absoluta	A evidência favorece uma atividade em relação a outra, com o mais alto grau de certeza
<b>2,4,6,8</b>	Valores intermediários	Quando se procura uma condição de compromisso entre duas definições

Tabela 6 – Comparação paritária entre critérios

	<b>Rapidez</b>	<b>Flexibilidade</b>	<b>Qualidade</b>	<b>Custo</b>
<b>Rapidez</b>	1	2	1/6	1/5
<b>Flexibilidade</b>	½	1	1/7	1/6
<b>Qualidade</b>	6	7	1	3
<b>Custo</b>	5	6	1/3	1

Define-se a classificação dos critérios segundo um vetor de prioridade (x), também conhecido por Eigen Vector X, cujos resultados são apresentados na Tabela 7. Destaca-se, da análise dos resultados, o critério de “qualidade” (0.55), que é considerado o mais importante e a “flexibilidade” (0.06), o menos importante.

Tabela 7 – Matriz Normalizada e Vetor de Prioridade Relativa dos Critérios

	<b>Rapidez</b>	<b>Flexibilidade</b>	<b>Qualidade</b>	<b>Custo</b>	<b>Vetor Prioridade</b>
<b>Rapidez</b>	0.08	0.13	0.1	0.05	0.09
<b>Flexibilidade</b>	0.04	0.06	0.09	0.04	0.06
<b>Qualidade</b>	0.48	0.44	0.61	0.69	<b>0.55</b>
<b>Custo</b>	0.4	0.38	0.2	0.23	0.30

Adotam-se, para o julgamento paritário das alternativas, os mesmos métodos de avaliação e intervenientes utilizados, para avaliar os critérios. Assim, para cada critério, compara-se, par a par, a importância relativa de cada alternativa. Apresentam-se, na Tabela 8, as comparações paritárias dos critérios (rapidez, flexibilidade, qualidade e custo), face às respetivas alternativas.

Tabela 8 – Comparação paritária dos critérios

<b>Rapidez</b>			
	SMS	Solução Web	Solução Desktop
<b>SMS</b>	1	7	5
<b>Solução Web</b>	1/7	1	1/4
<b>Solução Desktop</b>	1/5	3	1
<b>Flexibilidade</b>			
<b>SMS</b>	1	2	5
<b>Solução Web</b>	½	1	4
<b>Solução Desktop</b>	1/5	1/4	1
<b>Qualidade</b>			
<b>SMS</b>	1	1/7	1/6
<b>Solução Web</b>	7	1	2
<b>Solução Desktop</b>	6	1/2	1
<b>Custo</b>			
<b>SMS</b>	1	5	1/3
<b>Solução Web</b>	1/5	1	1/7
<b>Solução Desktop</b>	3	7	1

Verifica-se, na Tabela 9, que a alternativa “SMS” é a mais rápida a ser implementada, em contraponto, com a “Solução Web”, cujos decisores apontam a sua complexidade como fator causador da lentidão da conceção. Lê-se, também, que a “Solução Web” é a alternativa que os decisores consideram de manutenção, instalação e acesso mais flexível em oposição à “Solução Desktop” que, pelo facto de ser instalada em cada cliente, é considerada menos flexível. Observa-se, ainda, que a alternativa da “Solução Web” é a que tem melhor qualidade para os decisores, que justificam a decisão pela quantidade e diversificada oferta de funcionalidades aos seus clientes, a qual nenhuma das outras, no seu conjunto, oferece. Por outro lado, considera-se a solução “SMS” a mais limitada. Constata-se, por último, que a “Solução Desktop” é a que apresenta menor custo de manutenção, contrariamente, à “Solução Web”, que é considerada a mais cara. Os decisores clarificam a opção pelo facto dos encargos da “Solução Desktop” serem imputados aos clientes, enquanto que na “Solução Web” os custos de

administração e configuração dos servidores e base de dados é da responsabilidade da parceria da solução Vodafone WinVending.

Tabela 9 – Matriz Normalizada para os critérios e Prioridade Local

<b>Rapidez</b>				
	SMS	Solução Web	Solução Desktop	Prioridade Local
<b>SMS</b>	0.74	0.64	0.79	<b>0.72</b>
<b>Solução Web</b>	0.11	0.09	0.05	0.08
<b>Solução Desktop</b>	0.15	0.27	0.16	0.19
<b>Flexibilidade</b>				
<b>SMS</b>	0.59	0.62	0.50	<b>0.57</b>
<b>Solução Web</b>	0.29	0.31	0.40	0.33
<b>Solução Desktop</b>	0.12	0.08	0.10	0.10
<b>Qualidade</b>				
<b>SMS</b>	0.07	0.09	0.05	0.07
<b>Solução Web</b>	0.50	0.61	0.63	<b>0.58</b>
<b>Solução Desktop</b>	0.43	0.30	0.32	0.35
<b>Custo</b>				
<b>SMS</b>	0.24	0.38	0.23	0.28
<b>Solução Web</b>	0.05	0.08	0.10	0.07
<b>Solução Desktop</b>	0.71	0.54	0.68	<b>0.64</b>

Percebe-se, através da Tabela 9, que não existe uma alternativa que seja globalmente melhor às demais. Nestas situações e segundo a abordagem do método AHP, a solução mais satisfatória é obtida a partir do cálculo da prioridade global de cada alternativa, cujos resultados são apresentados na Tabela 13.

#### 4.7.3 Análise de Consistência dos Julgamentos

Verifica-se a validade dos julgamentos efetuados, por meio do cálculo do rácio de consistência (RC), onde se valida o quão consistentes são, relativamente a grandes escalas de dados puramente aleatórios. As avaliações do AHP baseiam-se no pressuposto de que os decisores são racionais, isto é, se A é preferível a B e B é preferível a C, então A é preferível a C. Se o rácio

de consistência é inferior a 0.1, os julgamentos são confiáveis, isto porque estão afastados da zona de aleatoriedade (<sup>70</sup>Nicola 2016).

O cálculo do RC envolve os seguintes procedimentos (Marins et al. 2009):

- começa-se por calcular “Ax”, cujos valores são resultado da multiplicação da matriz normalizada (A) com o vetor de prioridade (x);
- de seguida divide-se o vetor “Ax” pelo vetor “x”;
- obtém-se “λmax” pela média dos resultados de cada linha;
- calcula-se o índice de consistência (IC) através da fórmula “ $IC = (\lambda_{max} - n) / (n - 1)$ ” onde “n” é o número de critérios;
- atinge-se o valor do RC pela divisão do IC pelo Índice de Inconsistência Aleatória (IIA), uma constante, cujo valor depende da dimensão da matriz que, no caso concreto, varia entre 4 para os critérios e 3 para as alternativas. Observa-se que o valor IIA, obtido da Tabela 10, é “0.9” para os critérios e “0.58” para as alternativas.

Tabela 10 – Índice de Inconsistência Aleatória (Marins et al. 2009)

Dimensão da matriz	1	2	3	4	5	6	7	8
Inconsistência Aleatória	0.00	0.00	0.58	0.90	1.12	1.24	1.32	1,41

Para a determinação do rácio de consistência dos critérios, começa-se por calcular λmax através da fórmula “ $\lambda_{max} = \text{media}(0.35/0.09, 0.23/0.06, 2.39/0.55, 1.27/0.30) = 4.15$ ”. Posteriormente, calcula-se o índice de consistência através da fórmula “ $IC = (4.15 - 4) / (4 - 1)$ ” e chega-se ao resultado de “0,05”. Finalmente, pela aplicação da fórmula do RC “ $RC = 0.05 / 0.90$ ” obtém-se o valor de “0.06”, que é menor do que “0.1”, como tal, consideram-se consistentes as ponderações de critérios efetuadas pelos decisores.

---

<sup>70</sup> Apontamento Análise de Valor de Negócio / Methods

Tabela 11 – Vetor Ax dos Fatores

	<b>Vetor Ax</b>
<b>Rapidez</b>	0.35
<b>Flexibilidade</b>	0.23
<b>Qualidade</b>	2.39
<b>Custo</b>	1.27

De modo a estabelecer-se os rácios de consistência das opções dos decisores face aos critérios existentes, começa-se por determinar o vetor das prioridades das alternativas, apresentado na Tabela 7. Posteriormente, calculam-se o  $\lambda_{max}$  e o IC de cada critério para se atingir o valor do RC correspondente, observado na Tabela 12. Nessa matriz constam, igualmente, os rácios de consistência calculados. Conclui-se que todos os julgamentos dos critérios são consistentes, uma vez que apresentam valores do RC inferiores a “0.10”.

Tabela 12 – Índices de Consistência e Razões de Consistência

	<b><math>\lambda_{max}</math></b>	<b>IC</b>	<b>RC</b>
<b>Rapidez</b>	3.07	0.03	0.06
<b>Flexibilidade</b>	3.02	0.01	0.02
<b>Qualidade</b>	3.03	0.02	0.03
<b>Custo</b>	3.07	0.03	0.06

#### 4.7.4 Análise de Resultados

Procede-se à criação de uma matriz, visível na Tabela 13, de relação das alternativas disponíveis com vetor de importância de critérios. Cada elemento da nova matriz é multiplicado pelo valor da tabela de Vetor de Prioridade de Critérios (Tabela 7), equivalente ao respetivo critério. Por último, obtém-se um vetor de prioridades global, visível na última coluna da Tabela 13, onde são apresentadas, por ordem crescente, as intensidades das importâncias de cada alternativa. Conclui-se que a melhor alternativa para o problema em causa é o desenvolvimento de uma “Solução *Desktop*”, de seguida uma “Solução *Web*” e, por último, uma solução de envio de SMS’s. Justifica-se a conclusão obtida, pelo facto da “Solução *Desktop*” apresentar maior importância nos critérios com maior influência para a tomada da decisão e é mesmo considerada a mais relevante no critério de custo.

Tabela 13 – Matriz classificação das alternativas e prioridade global

	<b>Rapidez</b>	<b>Flexibilidade</b>	<b>Qualidade</b>	<b>Custo</b>	<b>Prioridade Global</b>
<b>SMS</b>	0.72	0.57	0.07	0.28	0.22
<b>Solução Web</b>	0.08	0.33	0.58	0.07	0.37
<b>Solução Desktop</b>	0.19	0.10	0.35	0.64	<b>0.41</b>

Importa considerar, na hora de tomar decisões estratégicas, o recurso a métodos de medição de criação de valor, quer sejam conceptuais ou quantitativos. Estas decisões são as que têm maior impacto no negócio do produto ou serviço, por isso devem ser tomadas com o máximo de consciência e ponderação.

## 5 Requisitos & Análise

Neste capítulo, apresentam-se os conceitos de negócio específicos da área de Vending e telemetria, enumeram-se os principais atores dos sistemas e referem-se os requisitos funcionais e não funcionais dos sistemas suportados pela classificação <sup>71</sup>FURPS+.

### 5.1 Conceitos de Negócio

Nesta seção, esclarecem-se e contextualizam-se os conceitos de negócio: comunicação, programação, ocorrência e rota.

#### 5.1.1 Comunicação

O conceito inerente a uma comunicação compreende os seguintes intervenientes: emissor, recetor, canal e mensagem. Assim que recebe a mensagem, o recetor envia um *feedback* ao emissor. Por vezes, o canal por onde circula a mensagem tem ruído, o que provoca falhas de comunicação.

---

<sup>71</sup> FURPS+ é uma classificação de requisitos (funcionais e não funcionais) desenvolvida por Robert Grady na empresa Hewlett Packard. “Usa-se esta classificação para descrever as principais categorias de requisitos: funcionais, usabilidade, confiabilidade, desempenho e suporte. O carácter “+” complementa o acrónimo de modo a incluir todos os outros requisitos como: restrições de desenho, requisitos de interface e requisitos físicos.” (Getronics 2003)

Ao transpor o conceito geral de comunicação para o sistema de telemetria, consideram-se dois processos de comunicação distintos: (i) o primeiro entre a máquina (emissor) e o *middleware* (recetor), através do canal TCP, onde a mensagem é constituída por um conjunto de dados da máquina (contabilísticos, incidências, moedeiros), (ii) o segundo processo compreende a comunicação do *middleware* (emissor) com o *backoffice* (recetor), por meio de uma ligação TCP segura, onde se envia a anterior mensagem, convertida para um formato acordado entre as partes. Ambos os processos podem incluir ruído na comunicação, embora de intensidades diferentes. Veja-se que, no primeiro cenário, o canal é suscetível de falhas da rede móvel, o segundo cenário pode, também, ser vítima de problemas de rede, embora de menor intensidade.

Neste sistema, a comunicação tem como objetivo enviar, por via remota, os dados desde a máquina até ao *backoffice*. Consideram-se os seguintes modos distintos de se despoletarem comunicações: (i) de modo automático, no ato de abertura / fecho de porta, (ii) via opção do sistema móvel, (iii) por meio do assistente de comunicações do *backoffice*, ou (iv) através do agendamento automático de comunicações.

### Estado

Quando o processo de comunicação é iniciado pelo *backoffice*, define-se um sistema de controlo do estado da comunicação. Parte-se desse pressuposto para se estabelecerem os próximos estados (c.f. Figura 11): (i) agendada, (ii) pendente, (iii) completa, (iv) suspensa, (v) anulada, (vi) falha.

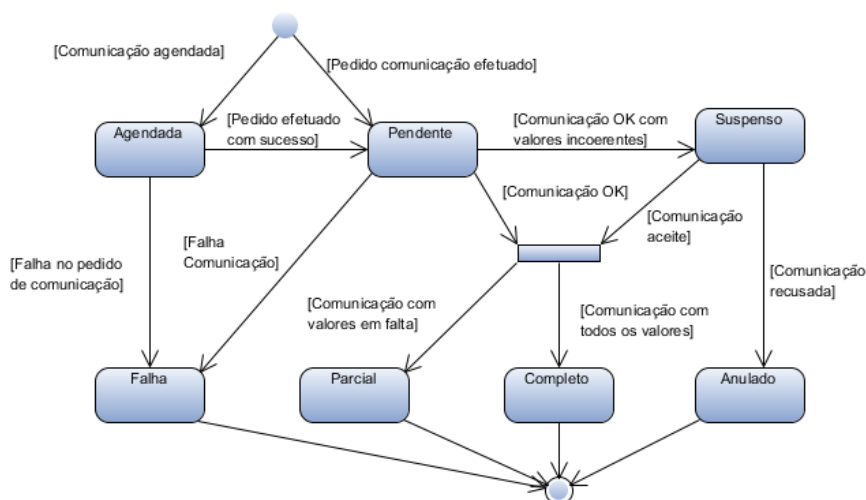


Figura 11 – Diagrama de estados de comunicação

Uma comunicação agendada fica no estado “Agendada” até que seja processado o pedido de comunicação. Nesse momento, ou em qualquer pedido efetuado, o estado fica assinalado como “Pendente”. Após a recepção, a mensagem pode passar para um dos seguintes estados: “Completo” se bem-sucedida, “Parcial” se faltarem apenas os dados de moedeiros e / ou incidências, “Falha” se ocorreu algum ruído no processo de comunicação que impossibilitou a entrega da mensagem ou, “Suspensa” quando a mensagem é entregue, no entanto, o seu conteúdo levanta suspeitas. O estado “Suspensa” ocorre, por exemplo, no caso de o código da máquina ser diferente do presente, na ficha da máquina. Em situações de suspensão, cabe ao utilizador, com as devidas permissões, a decisão de aceitar ou não. Caso aceite, o estado passa para “Completo”, caso contrário, fica como “Anulado”. Qualquer uma das decisões é irreversível.

#### *Tipo*

Consoante o momento em que a comunicação é recebida, pode assumir os seguintes tipos: (i) inicialização, (ii) estatística, (iii) contabilística e (iv) configuração. Visualizam-se (c.f. Figura 12), por ordem crescentemente temporal, os diferentes tipos de comunicação mencionados. A comunicação de inicialização é sempre a primeira comunicação de uma máquina e coloca os valores parciais a zero e os acumulados são guardados para servirem de base a futuras comunicações. Normalmente, estas comunicações só voltam a ocorrer quando o cliente associado à máquina é alterado. As comunicações estatísticas são solicitadas pelo *backoffice* com o objetivo de atualizar os seus valores. Solicita-se através de um assistente para esse efeito ou através de agendamento. As comunicações contabilísticas efetuam-se no momento de abertura da porta da máquina. Assim, ao abrir a porta da máquina, é automaticamente despoletada uma comunicação que, posteriormente ao seu tratamento, pode originar um documento de venda. As comunicações de configuração concretizam-se ao fechar a porta da máquina. Todos os contadores da máquina são atualizados, com a exceção dos valores colocados / retirados. O principal objetivo desta comunicação é definir uma base de valores acumulados de referência para o cálculo de futuras comunicações.

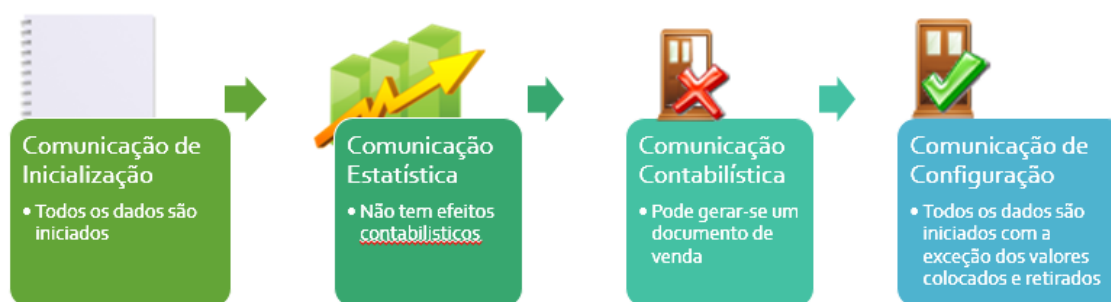


Figura 12 – Tipos de comunicação (ABS Informática 2014)

### 5.1.2 Programação

Compreende-se por programação todo o tipo de operação realizada na máquina, no sentido de alterar os seus dados. Distinguem-se os seguintes tipos de programação: (i) alteração do código de operador, (ii) alteração da mensagem publicitária e (iii) definição das seleções e canais da máquina e, em alguns modelos de máquina, inclui a definição da associação entre as seleções e os canais, também conhecida por união de canais.

O código de operador é uma chave de acesso a operações sensíveis da máquina (e.g. programação das seleções e canais). Mesmo que a programação se realize remotamente, é necessário enviar o código de operador para se prosseguir com a sua execução. O ato de alterar o código de operador implica o envio do anterior código.

A mensagem publicitária é um texto com um determinado número de caracteres, a apresentar no visor da máquina, direcionado para o consumidor final. De seguida, enumeram-se os diferentes objetivos destas mensagens: saudar, felicitar, agradecer e promover algum produto ou serviço junto do consumidor.

A programação de seleções e canais e respetiva união de canais é o processo de programação mais complexo e consiste na concretização das seguintes etapas principais: (i) configurar as seleções / canais na ficha da máquina, (ii) programar remotamente a máquina e (iii) manter o histórico de alterações. No decorrer destas etapas, a programação assume diferentes estados: (i) agendada, (ii) pendente, (iii) atual, (iv) cancelada e (v) histórico.

O estado inicial da programação assume três estados diferentes, consoante o método adotado para a sua execução: (i) "Agendada" ao proceder-se a um agendamento de programação, (ii) "Pendente" ao efetuar-se uma programação e (iii) "Atual" caso se opte por, apenas, configurar

a máquina (a ficha da máquina é alterada sem se alterarem os dados físicos da máquina). Uma comunicação agendada é apresentada ao vendedor que efetua a visita à máquina, para que aprove a sugestão dada e realize os devidos ajustes, antes de se efetuar uma programação por telemetria ou *Bluetooth*. A programação por *Bluetooth* passa diretamente para o estado “Atual”, já a programação por telemetria assume o estado “Pendente”. Quando a configuração ou programação em curso substitui uma anterior, procede-se à alteração do estado da anterior para “Histórico”. Uma configuração ou programação no estado “Agendada” ou “Pendente” pode passar para o estado “Cancelada”. Considera-se os estados “Cancelada” e “Histórico” como os estados finais do diagrama de estados apresentados na Figura 13.

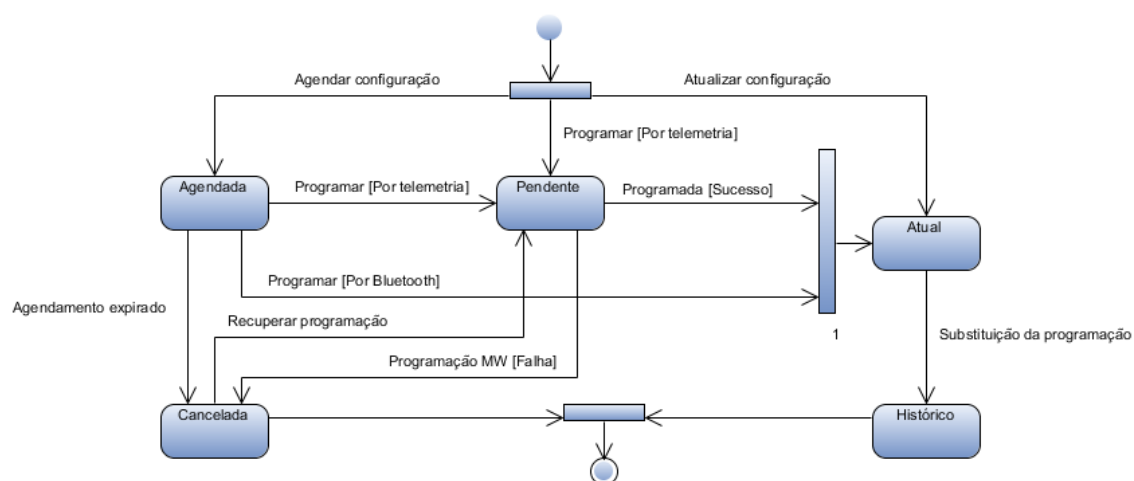


Figura 13 – Diagrama de estados de programação da máquina

### 5.1.3 Ocorrência

Entende-se por ocorrência qualquer evento capturado pelos sensores do módulo, ou provenientes da comunicação com máquina, ou ainda, através da dedução realizada a partir de uma combinação de fatores. Divide-se o mencionado conceito em dois sub conceitos: alarmes e anomalias.

#### *Alarmes*

Um alarme é qualquer evento capturado pelos sensores instalados na máquina, dos quais se destacam: movimento, localização, porta, ligada, desligada e cabo de dados.

### *Anomalias*

Entende-se por anomalia qualquer evento cujo estado se possa reverter, ou seja, todas as situações que possam ser corrigidas. Detetam-se dois modos diferentes de anomalias: diretas e deduzidas. As anomalias diretas, também denominadas de incidências, são obtidas juntamente com os dados da comunicação e descrevem os eventos capturados pelo *software* da própria máquina<sup>72</sup>. As anomalias deduzidas são geradas com base noutros eventos ocorridos, como por exemplo, a partir de um evento de canal esgotado, gera-se uma ocorrência de produto esgotado, se todos os canais estiverem nas mesmas condições do primeiro.

#### **5.1.4 Rota**

Entende-se por rota o conjunto de clientes e máquinas (visitas) que um vendedor deve visitar, no decorrer de um mês genérico. Cada dia da semana desse mês é composto por um conjunto específico de visitas.

Determina-se a rota de um vendedor de dois modos distintos: (i) manual ou (ii) automaticamente. No modo manual, o utilizador define para cada dia quais as visitas a realizar. Invoca-se o conceito periodicidade para auxiliar a compreensão da construção automática da rota. Entende-se por periodicidade o conjunto de dias da semana definidos para um mês genérico (e.g. todas as segundas-feiras). Este conceito materializa-se num campo da ficha do cliente onde se menciona a periodicidade com que é visitado, ou seja, ao aplicar-se o anterior exemplo, num determinado cliente, significa que deve ser visitado pelo vendedor definido na sua ficha, todas as segundas-feiras. No método automático, percorre-se a ficha de cada cliente cujo vendedor seja igual ao da rota e, para os dias da semana associados à respetiva periodicidade, adiciona-se uma nova visita.

Posteriormente à definição da rota em *backoffice*, o vendedor, através do sistema móvel, inicia todos os dias a sua rota diária. Nesse momento, cria-se automaticamente um plano de rota com os clientes e máquinas definidos na rota para o dia e atribui-se o número de quilómetros da

---

<sup>72</sup> Alguns exemplos de incidências são: avaria do módulo extrator, avaria do devolvedor, moedeiro esgotado e canal esgotado. As incidências comunicadas variam consoante o fabricante e nem todos as disponibilizam.

viatura e a correspondente hora de início. No decorrer do dia, efetuam-se visitas dentro do plano e outras fora do plano. Para as primeiras, atualizam-se os horários efetivos. Para as segundas, criam-se novas entradas no plano, com a indicação de visita realizada fora de rota e regista-se a hora de fim de visita efetiva.

Os utilizadores em *backoffice* acompanham, no painel geral do WinVending, as rotas de cada vendedor e o respetivo horário das visitas, bem como a percentagem de concretização da rota a nível individual e coletivo.

### **5.1.5 Categorização de Alarmes**

No sistema WinGuard, os alarmes podem entrar no sistema num ritmo e volume elevados, o que torna a sua análise complexa. De modo a auxiliar a tarefa de análise por parte dos operadores, opta-se por diferenciar os alarmes em duas categorias: “informativo” e “prioritário”.

Entende-se por alarme informativo todo aquele que ocorre fora do horário de funcionamento do estabelecimento não sendo, contudo, suficiente para se considerar que se está perante um caso de roubo ou vandalismo. Realça-se que a sua análise não deve ser descurada.

Um alarme prioritário informa o operador da “Central de Alarmes” que se reúnem condições suficientes para se entrar em contacto com o estabelecimento, ou com o agente de autoridade mais próximo, de forma a despistar ou evitar um possível ato de roubo ou vandalismo.

Elucida-se o conceito de intervalo consecutivo para a compreensão da categorização de alarmes. Assim, se a diferença de tempo entre o último alarme registado e o atual for menor ou igual a um valor predefinido (e.g. 15 minutos) significa que os alarmes estão relacionados.

Expõe-se, na Figura 14, o sistema de categorização dos alarmes a partir do momento em que entram no sistema e se executam uma série de condições de modo a categorizar o alarme entre “informativo” e “prioritário”. Começa-se por verificar se a localização da máquina está fora do raio de segurança e, em caso positivo, considera-se o alarme “prioritário”. No caso negativo, confirma-se se o último alarme dentro do intervalo consecutivo é “prioritário” e, se verdade, elege-se como “prioritário”. Caso contrário, obtém-se o tipo de alarme e (i) para os alarmes de porta aberta / fechada, considera-se “prioritário”, (ii) para os alarmes de máquina ligada / desligada, categoriza-se como “informativo”, (iii) para os alarmes de movimento, verifica-se a

existência de mais de X movimentos, dentro do intervalo consecutivo e, se verdade, marca-se como “prioritário”, no caso contrário, “informativo” e (iv) para os alarmes de GPS, analisa-se a coordenada atual e compara-se com a última coordenada conhecida, dentro do intervalo consecutivo e, se diferente, assinala-se como “prioritário”, se igual, atribui-se a categoria “informativo”.

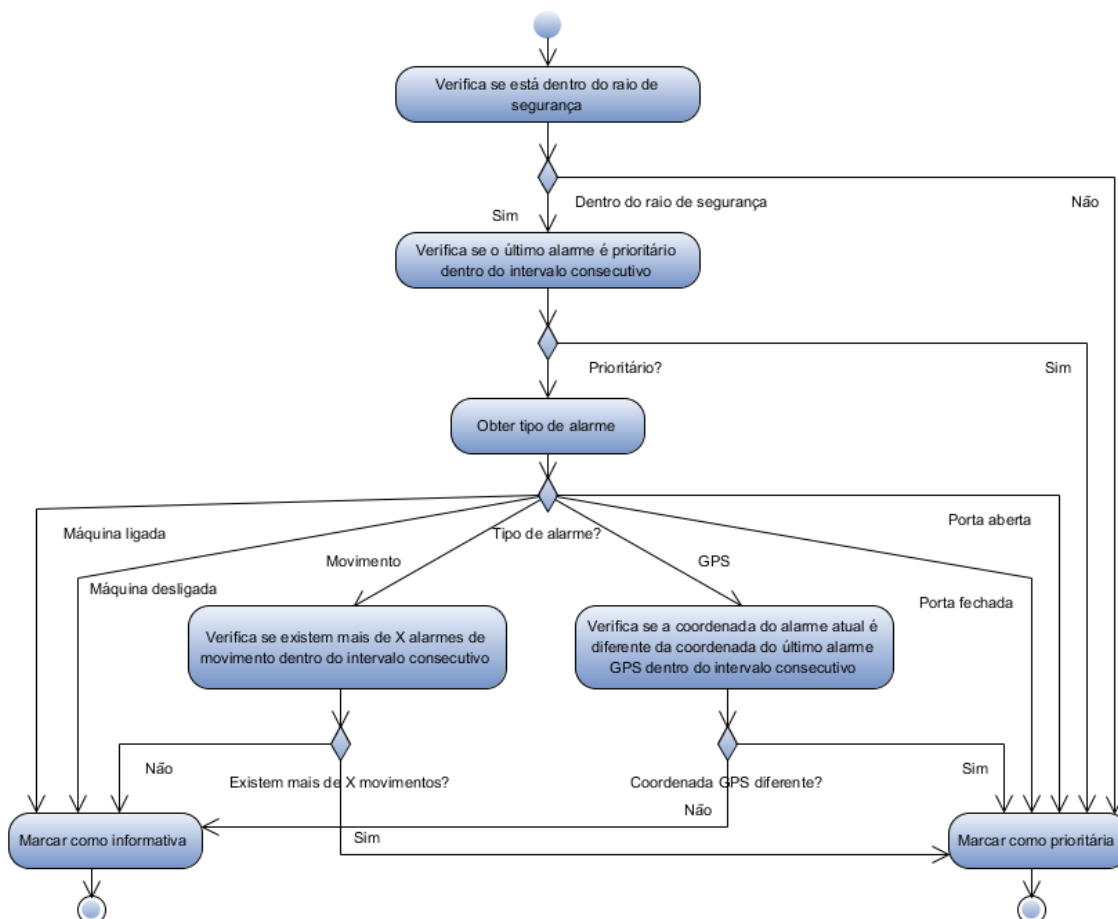


Figura 14 – Diagrama de atividade de prioridade do alarme

## 5.2 Atores

Nesta secção, identificam-se os atores dos sistemas: Vodafone WinVending e WinGuard.

### **5.2.1 Utilizador**

O utilizador é qualquer entidade que tenha autoridade para usar a aplicação WinVending. Distinguem-se três tipos de utilizadores, consoante as funcionalidades que estão disponíveis: administradores, administrativos e vendedores.

### **5.2.2 Administrador**

O administrador é um utilizador responsável por efetuar tarefas de manutenção do sistema, como por exemplo, gerir: empresas, utilizadores e exceções<sup>73</sup>.

### **5.2.3 Administrativo**

O administrativo é um utilizador responsável por gerir e monitorizar as máquinas, assim como, assegurar a manutenção dos dados necessários ao funcionamento do sistema (e.g. clientes, produtos, periodicidades, rotas, máquinas, entre outros).

### **5.2.4 WinVending Backoffice**

O WinVending *Backoffice* é um sistema composto por diversos componentes que tratam os dados provenientes das máquinas e apresentam aos utilizadores.

### **5.2.5 WinVending Mobile**

O WinVending *Mobile* é o sistema móvel que acompanha o vendedor e o técnico, na visita a uma máquina.

---

<sup>73</sup> “Uma exceção é um evento, que ocorre durante a execução de um programa, que interrompe o fluxo normal das instruções do programa.” (tradução nossa) (Oracle 2015)

### **5.2.6 Middleware**

É o sistema responsável por tratar os dados enviados pelos módulos instalados nas máquinas e comunicá-los ao sistema WinVending *Backoffice*.

### **5.2.7 ERP**

O ERP é uma aplicação responsável por gerir: clientes, produtos e documentos de venda.

### **5.2.1 Operador**

É a entidade responsável, no sistema WinGuard, pela monitorização e tratamento dos alarmes provenientes do sistema WinVending *Backoffice*.

### **5.2.2 WinGuard**

É o sistema que concede aos operadores a capacidade de monitorizar os alarmes provenientes das máquinas com os respetivos dados de contacto do estabelecimento, do proprietário e do agente de autoridade mais próximo.

## **5.3 Requisitos Funcionais**

Guiado pelo “F” do acrónimo FURPS+, nesta secção, descreve-se todo o aspeto funcional dos sistemas WinVending e WinGuard. Nas secções referentes a cada um dos sistemas, começa-se por contextualizar as partes interessadas e o seu contributo na definição dos requisitos. Posteriormente, concretiza-se a apresentação dos requisitos, através de diagramas UML (diagrama de casos de uso, diagramas de estados e diagramas de sequência) e remete-se, para anexos, uma análise mais detalhada de cada requisito (i.e. caso de uso) individual.

### **5.3.1 WinVending BackOffice**

Referem-se, nesta secção, os requisitos da nova solução de Vending que, lembra-se, tem dois objetivos: (i) implementar a nova componente de telemetria e (ii) migrar as funcionalidades essenciais da solução legada (e.g. manutenção da ficha da máquina).

Envolvem-se diferentes entidades, consoante o objetivo a analisar. Por um lado, na negociação dos requisitos da incorporação da componente de telemetria invocam-se: (i) o cliente (depositário de tabaco), (ii) a ABS Informática e (iii) os respetivos parceiros na área de M2M (Vodafone e KTS). Por outro lado, na migração das funcionalidades da solução legada, colaboram as entidades: (i) ABS Informática e (ii) o cliente.

Inicialmente, e no que diz respeito à inclusão da componente de telemetria, o cliente final apresentou o desejo e necessidade de manter e monitorizar remotamente o seu parque de máquinas. Das reuniões e entrevistas com o cliente, estabelece-se um conjunto de requisitos (e.g. comunicar e programar remotamente a máquina) que, devido às inerentes especificidades técnicas, necessita de uma análise e negociação com os parceiros. Sucede-se um conjunto de: (i) reuniões físicas e por videoconferência, (ii) conversas telefónicas e (iii) troca de *emails* entre os parceiros para discutir o levantamento dos requisitos atuais e definir novos (e.g. manter módulos).

Recorde-se, como marco no processo de definição dos requisitos, a primeira reunião formal entre os três parceiros, realizada em junho de 2012, nas instalações da KTS, onde cada um apresenta o produto que possui: (i) a ABS Informática, a versão do Vending 2012, (ii) a Vodafone, o cartão SIM de comunicação com as máquinas e (iii) a KTS, a componente física de comunicação com a máquina (o módulo) e a arquitetura de *software* que permite a ligação entre os vários componentes (máquina, cartão SIM, módulo e *Middleware*). Delimita-se, ainda, o âmbito de atuação de cada parceiro, as suas responsabilidades e os canais de comunicação, bem como respetivos intervenientes e responsáveis. Verifica-se, também, se o potencial oferecido pela infraestrutura dos parceiros se coaduna com os requisitos do cliente final. Por fim, registam-se os próximos passos: (i) envio, por parte da KTS, do documento do desenho de *software* e a API REST e (ii) a ABS Informática concretiza uma apresentação de um protótipo da solução que implemente a comunicação com a máquina através do recurso à API do parceiro. Posteriormente, a KTS envia o email que contém o documento, em formato rascunho, que considera: a visão geral e a arquitetura do sistema, o modo de autenticação e a API de comunicação com o *Middleware*. A partir desse momento, e daí em diante, a comunicação entre os parceiros não cessou.

Relativamente ao levantamento dos requisitos de migração do sistema legado, desenrolou-se por meio de: (i) reuniões com atuais clientes, (ii) *feedback* informal dos clientes e utilizadores

da aplicação (críticas, sugestões e opiniões), (iii) observação direta dos utilizadores em ambiente de trabalho e (iv) comunicação interna (dos consultores, equipa de sistemas e suporte da ABS Informática).

Da conjugação dos fatores (i) feedback do cliente final, (ii) características a migrar da aplicação WinVending legada e (iii) potencial oferecido pelas soluções da parceria (cartão SIM, módulo e API do *Middleware*) resultam as seguintes necessidades a incorporar na nova solução:

- credenciar utilizador (UC\_WVB\_0001);
- seleccionar empresa (UC\_WVB\_0002);
- consultar um painel de bordo (UC\_WVB\_0003);
- consultar a localização de todas as máquinas (UC\_WVB\_0004);
- manter consultas personalizadas (UC\_WVB\_0005);
- consultar o resultado das análises personalizáveis e exportá-las para XLS (UC\_WVB\_0006 e UC\_WVB\_0007);
- manter tabelas gerais da aplicação (UC\_WVB\_1000);
- manter rotas (UC\_WVB\_1120 e UC\_WVB\_1130);
- agendar a recolha remota de informações das máquinas (UC\_WVB\_1180, UC\_WVB\_1181, UC\_WVB\_1182 e UC\_WVB\_1183);
- manter a ficha técnica da máquina (UC\_WVB\_2000);
- consultar a visão geral da máquina (e.g. estado de sensores, stock, anomalias e últimos alarmes) (UC\_WVB\_2001);
- programar remotamente produtos e preços (UC\_WVB\_2040);
- consultar os dados não contabilísticos da máquina, nomeadamente, vendas perdidas, avarias e produto esgotado (UC\_WVB\_2050);
- consultar as ocorrências da máquina (UC\_WVB\_2060);
- consultar a localização da máquina (UC\_WVB\_2080);
- comunicar remotamente os dados contabilísticos das máquinas (e.g. vendas de artigos, vendas perdidas, dinheiro em cofre e dinheiro no moedeiro) (UC\_WVB\_0008, UC\_WVB\_0009 e UC\_WVB\_0010);
- notificar ocorrências (UC\_WVB\_0011).

### WinVending Backoffice (Visão Geral)

Começa-se pela apresentação da visão geral do sistema WinVending Backoffice (c.f. Figura 15) com a esquematização hierárquica do ator “Utilizador” (administrador, administrativo e vendedor) e a sua relação com os casos de uso acessíveis, diretamente, ao entrar na aplicação. Introduzem-se as entidades “Sistema Middleware”, “Sistema WinGuard” e “Sistema ERP”, bem como, os seus relacionamentos com casos de uso.

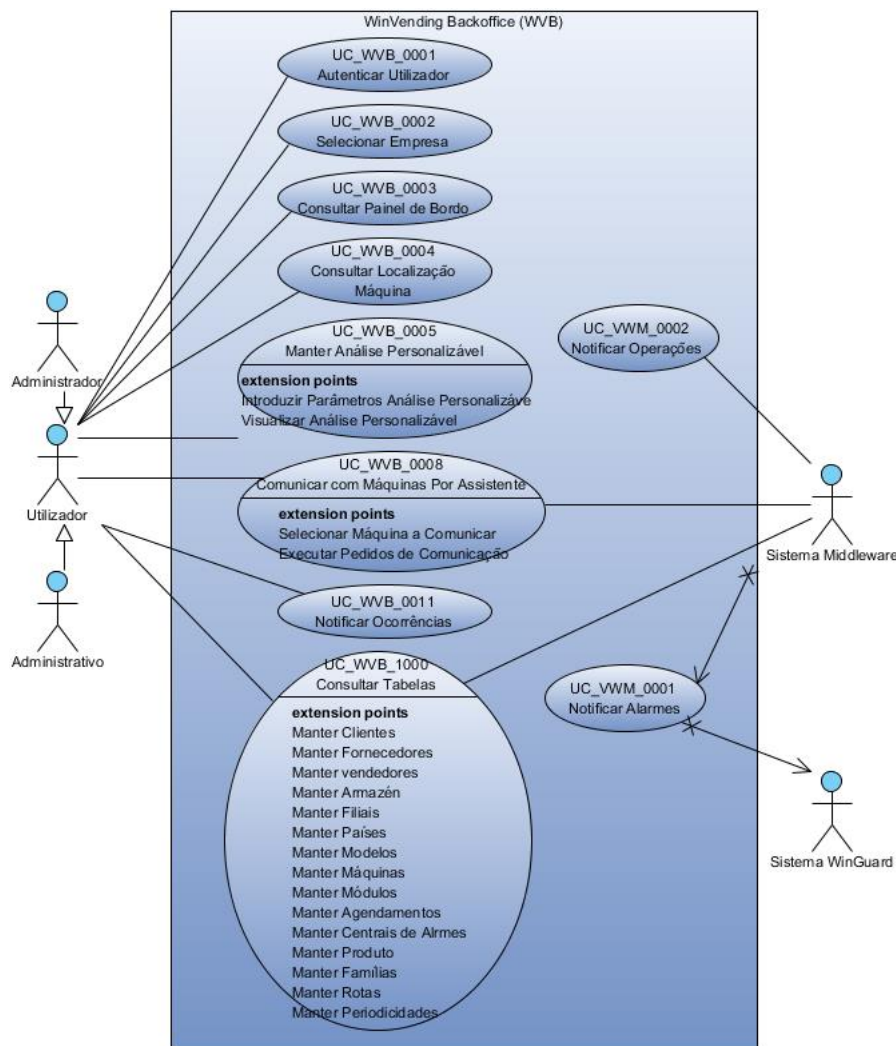


Figura 15 – Caso de uso do WinVending Backoffice

#### *Autenticar<sup>74</sup> (UC\_WVB\_0001)*

Ao entrar na aplicação, o utilizador introduz as credencias: nome de utilizador e palavra-passe. Se estiverem corretas, entra na sua área de utilizador, caso contrário, é-lhe apresentada uma mensagem de erro.

#### *Selecionar Empresa<sup>75</sup> (UC\_WVB\_0002)*

Ao abrir a janela da empresa, apresenta-se uma lista de empresas associadas à aplicação WinVending e seleciona-se aquela à qual se pretende aceder.

#### *Consultar Painel de Bordo<sup>76</sup> (UC\_WVB\_0003)*

Visão geral da informação operacional da empresa: progressão das rotas dos vendedores do dia, listagem das últimas comunicações por máquina e listagem das anomalias agrupadas por categoria.

#### *Consultar Localização de Máquinas<sup>77</sup> (UC\_WVB\_0004)*

Visualiza-se a posição geográfica das máquinas com a possibilidade de navegar diretamente para a ficha da máquina.

#### *Manutenção de Análises personalizáveis (UC\_WVB\_0005)*

Concede-se, com as análises personalizáveis, a capacidade de um utilizador administrador criar consultas em SQL com configurações específicas (c.f. Tabela 35).

Os utilizadores administrativos consultam as análises, anteriormente construídas, e em primeira instância, se definidos, solicita-se a introdução dos parâmetros da análise (c.f. Tabela 36) e, posteriormente, procede-se à sua visualização (c.f. Tabela 37) em formato tabular. O diagrama de caso de uso da Figura 16 ilustra as referidas entidades e seu relacionamento com os respetivos casos de uso.

---

<sup>74</sup> Para mais informações consultar o caso de uso (c.f. Tabela 31)

<sup>75</sup> Para mais informações consultar o caso de uso (c.f. Tabela 32)

<sup>76</sup> Para mais informações consultar o caso de uso (c.f. Tabela 33)

<sup>77</sup> Para mais informações consultar o caso de uso (c.f. Tabela 34)

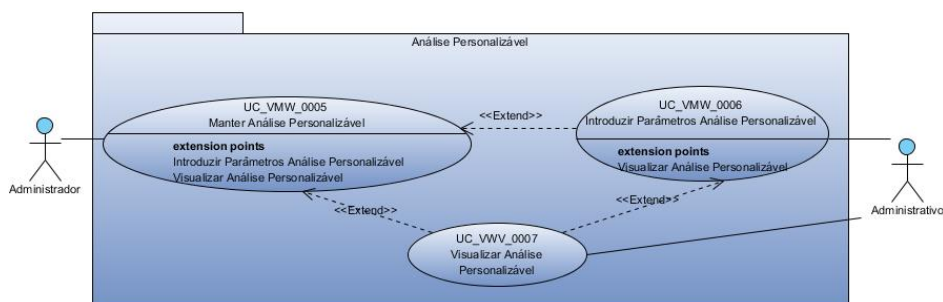


Figura 16 – Caso de uso de Análise Personalizável

### *Assistente de comunicação com as máquinas (UC\_WVB\_0008)*

No assistente de comunicação, o utilizador seleciona um conjunto de máquinas (individualmente ou por rota) para as quais pretende efetuar um pedido de comunicação. Na Figura 17, visualiza-se o caso de uso do assistente de comunicação com as máquinas (c.f. Tabela 38) e as duas etapas que o compõem: (i) a seleção das máquinas com as quais se pretende comunicar (c.f. Tabela 39) e a execução dos pedidos de comunicação (c.f. Tabela 40).

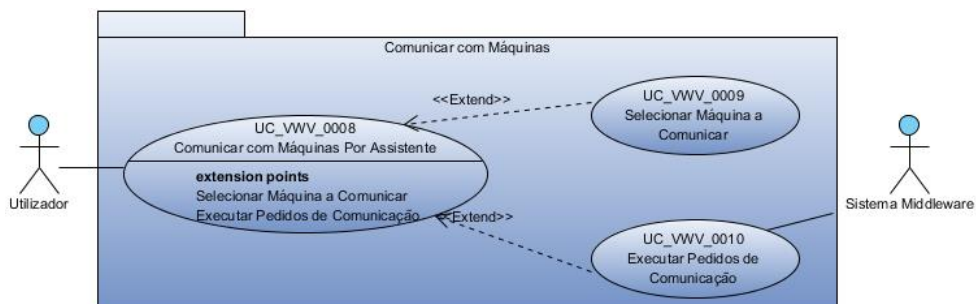


Figura 17 – Caso de uso de Comunicar com Máquinas

### *Notificar ocorrências<sup>78</sup> (UC\_WVB\_0011)*

O sistema verifica, de X em X tempo, a presença de um novo alarme e, caso detete um alarme de violação do raio de segurança de uma máquina, notifica o utilizador da sua ocorrência.

### *Manutenção de Tabelas Administrativas<sup>79</sup> (UC\_WVB\_1000)*

<sup>78</sup> Para mais informações consultar o caso de uso (c.f. Tabela 41)

<sup>79</sup> Para mais informações consultar o caso de uso (c.f. Tabela 42)

A manutenção das tabelas administrativas é a base para o funcionamento de todo o processo operacional, ao permitir a sua referência, na maioria dos registos introduzidos. A sua gestão fica a cargo dos administrativos, ou dos administradores da aplicação.

O utilizador seleciona a tabela que pretende administrar visualiza os registos da mesma e, mediante as suas permissões, disponibiliza-se um conjunto de operações de manutenção sobre os registos da tabela (criar, alterar, remover, atualizar).

No diagrama de caso de uso da Figura 18, constata-se a necessidade de manter as seguintes tabelas: clientes (c.f. Tabela 43), fornecedores (c.f.

Tabela 46), vendedores (UC\_WVB\_1020), armazéns (UC\_WVB\_1030), filiais (UC\_WVB\_1040), países (UC\_WVB\_1050), modelos (c.f. Tabela 49), máquinas (c.f. Tabela 60), módulos (c.f. Tabela 83), centrais de alarmes (UC\_WVB\_1090), produtos (c.f. Tabela 52), famílias (UC\_WVB\_1110), rotas (c.f. Tabela 55 – Caso de uso de Manter Rotas) e periodicidades (c.f. Tabela 54).

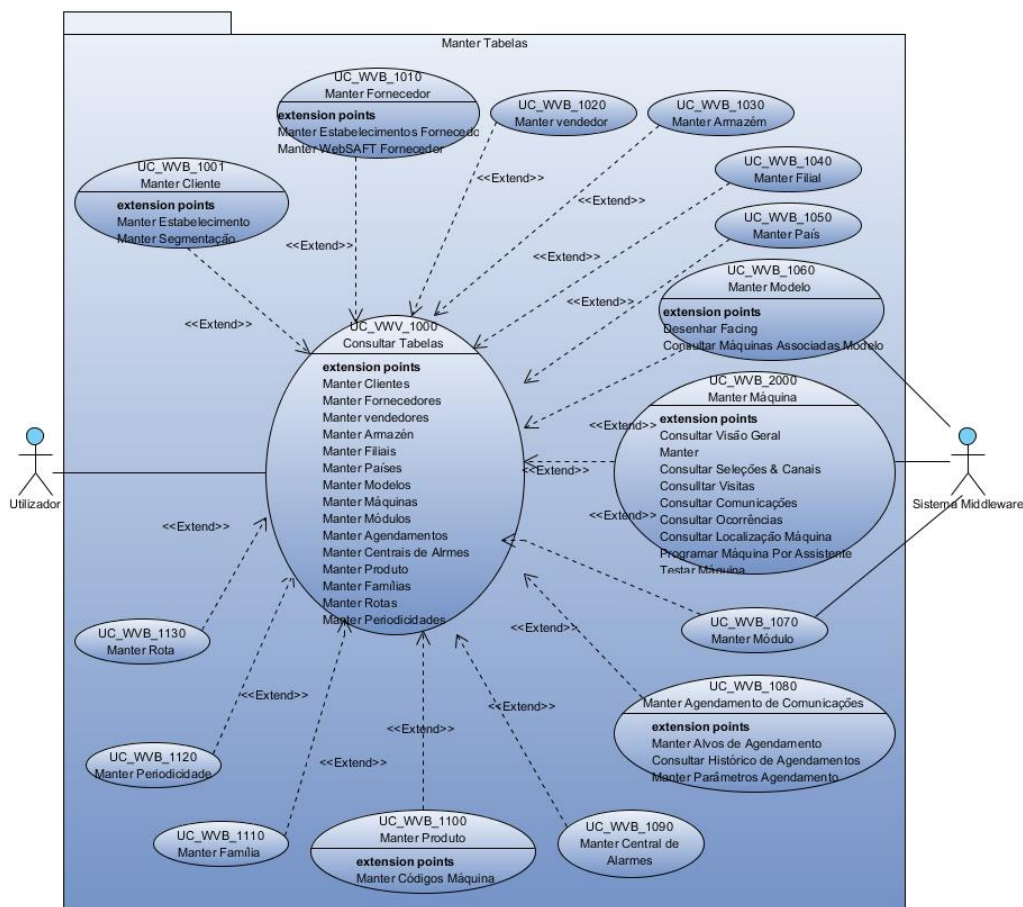


Figura 18 – Caso de uso de Manter Tabelas

### Manutenção de clientes (UC\_WVB\_1001)

Representa-se, na Figura 19, a manutenção: (i) da ficha do cliente (c.f. Tabela 43), (ii) dos seus estabelecimentos (c.f. Tabela 44) e (iii) dos dados estatísticos de <sup>80</sup>segmentação (c.f. Tabela 45).

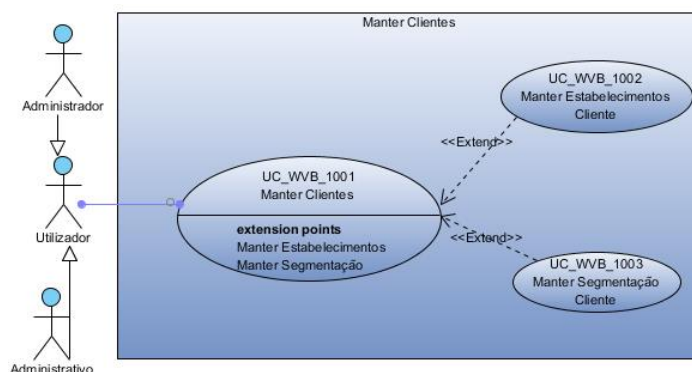


Figura 19 –Caso de uso de Manter Clientes

### Manutenção de Fornecedores (UC\_WVB\_1010)

Representa-se, na Figura 20, a manutenção: (i) da ficha do fornecedor (c.f. Tabela 46), (ii) dos seus estabelecimentos (c.f. Tabela 47) e (iii) os dados de sincronização com o WebSAFT<sup>81</sup> (c.f. Tabela 48).

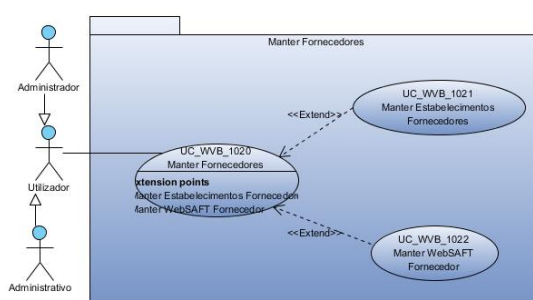


Figura 20 –Caso de uso de Manter Fornecedores

<sup>80</sup> A segmentação de clientes consiste na sua categorização segundo determinados critérios e métricas definidas por uma empresa especialista em estatística.

<sup>81</sup> WebSAFT é uma solução de alojamento de ficheiros SAF-T, desenvolvida pela ABS Informática (ABS Informática 2017). O site da solução é: <https://vending.absinformatica.pt:988/Account/LogOn> (WebSAFT 2017).

### Manutenção de Modelos (UC\_WVB\_1060)

Ilustra-se, na Figura 21, a manutenção do modelo da máquina (c.f. Tabela 49), a consulta do *facing* (c.f. Tabela 50) e o acesso à lista de máquinas com o modelo em causa (c.f. Tabela 51).

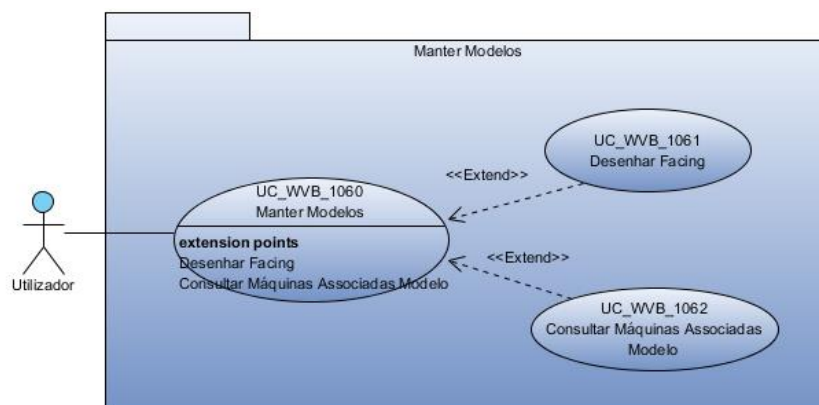


Figura 21 - Figura 20 –Caso de uso de Manter Fornecedores

### Manutenção de Agendamentos<sup>82</sup> (UC\_WVB\_1080)

Apresenta-se, na Figura 22, a manutenção do agendamento de comunicações, com destaque para a manutenção dos alvos e parâmetros, bem como a respetiva consulta do histórico dos agendamentos efetuados.

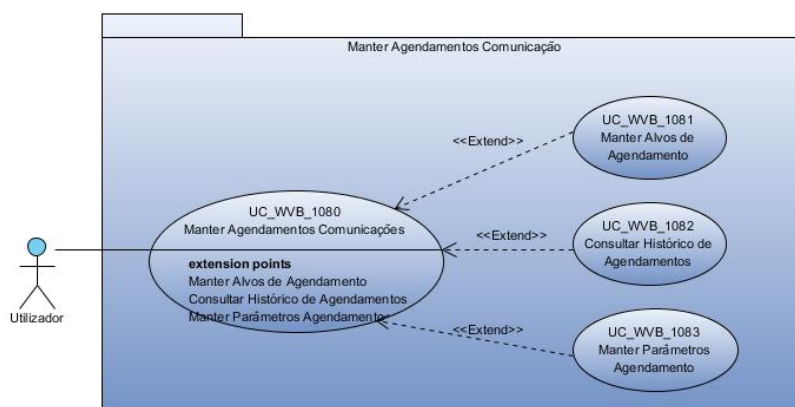


Figura 22 – Caso de uso de Manter Agendamentos de Comunicação

---

<sup>82</sup> Para mais informações, consultar o caso de uso (c.f. Tabela 56 – Caso de uso de Manter Agendamentos Comunicações)

### *Manter Produtos (UC\_WVB\_1110)*

Ilustra-se, na Figura 23, a manutenção: (i) da ficha do produto (c.f. Tabela 52) e (ii) dos códigos máquina (c.f. Tabela 53).

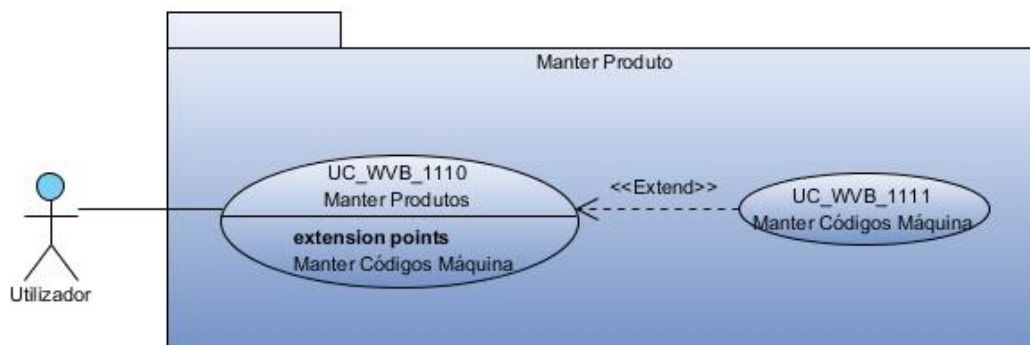


Figura 23 - Caso de uso de Manter Produtos

### *Manter Periodicidades<sup>83</sup> (UC\_WVB\_1120)*

Concede-se ao utilizador a possibilidade de manter, num calendário genérico com quatro semanas, os dias em que pretende efetuar uma visita (e.g. define-se uma periodicidade de visita às máquinas, todas as terças-feiras, de duas em duas semanas).

### *Manter Rotas<sup>84</sup> (UC\_WVB\_1130)*

O utilizador mantém para cada vendedor uma rota que corresponde aos clientes e máquinas que deve visitar num determinado dia.

### *Manutenção de Máquinas (UC\_WVB\_2000)*

Apresenta-se, na Figura 24, a manutenção da máquina e todas as operações que a envolvem: (i) consulta de visão geral (c.f. Tabela 60), (ii) manutenção dos dados principais (c.f. Tabela 63), (iii) consulta de visitas (c.f. Tabela 66), (iv) consulta de seleções & canais (c.f. Tabela 69), (v) consultar comunicações (c.f. Tabela 71), (vi) consultar ocorrências (c.f. Tabela 72), (vii) consultar localização da máquina (c.f. Tabela 73), (viii) programação da máquina através de assistente (c.f.

---

<sup>83</sup> Para mais informações consultar o caso de uso (c.f. Tabela 54)

<sup>84</sup> Para mais informações consultar o caso de uso (c.f. Tabela 55)

Tabela 74), (ix) testar a componente de telemetria da máquina (c.f. Tabela 61) e (x) manter estado da máquina (c.f. Tabela 62).

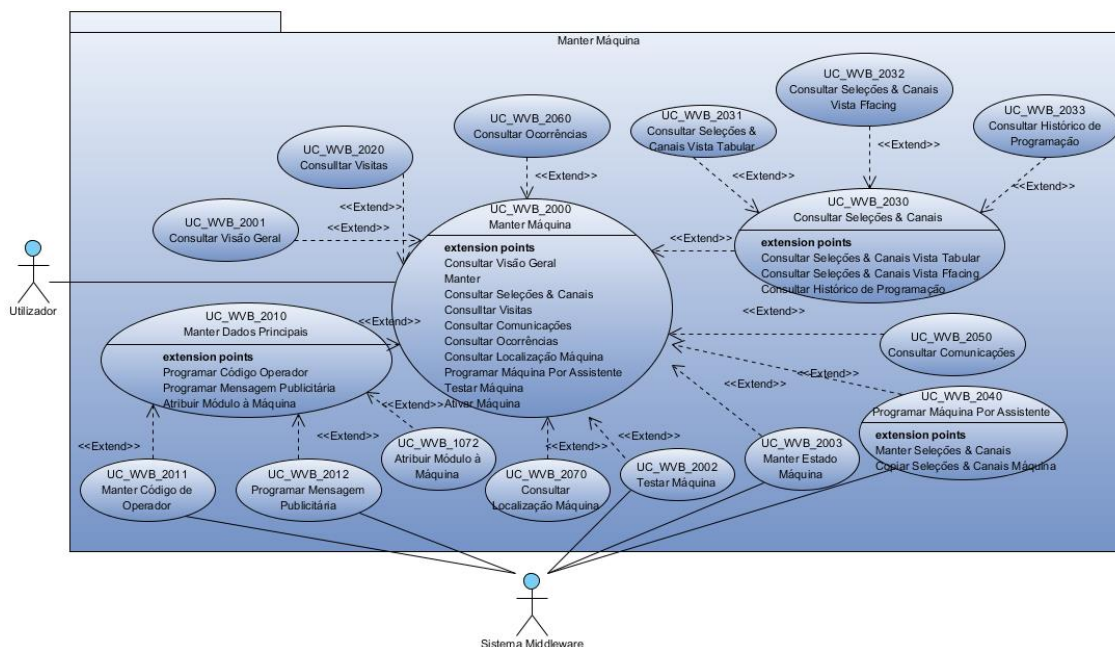


Figura 24 – Caso de uso de Manter Máquinas

#### Assistente de Programação da Máquina (UC\_WVB\_2040)

No pacote de “Programar a Máquina Por Assistente” (c.f. Figura 25)^, apresenta-se a interação entre o utilizador, o sistema *Backoffice* e o sistema *Middleware*. O sistema *Backoffice* disponibiliza um assistente que guia o utilizador na configuração / programação da máquina, através de um conjunto de etapas, algumas delas facultativas, das quais se destacam: (i) programar a máquina por assistente (c.f. Tabela 74), (ii) copiar seleções & canais de outra máquina (c.f. Tabela 75), (iii) manter seleções & canais (c.f. Tabela 76), (iv) seleccionar configuração (c.f. Tabela 80), (v) atualizar configuração da máquina (c.f. Tabela 81) e (vi) programar a máquina (c.f. Tabela 82).

A programação da máquina exige trocas de mensagens com o sistema *Middleware*, quer para programar, quer para comunicar remotamente com a máquina, de modo a validar a correta programação (c.f. Tabela 82).

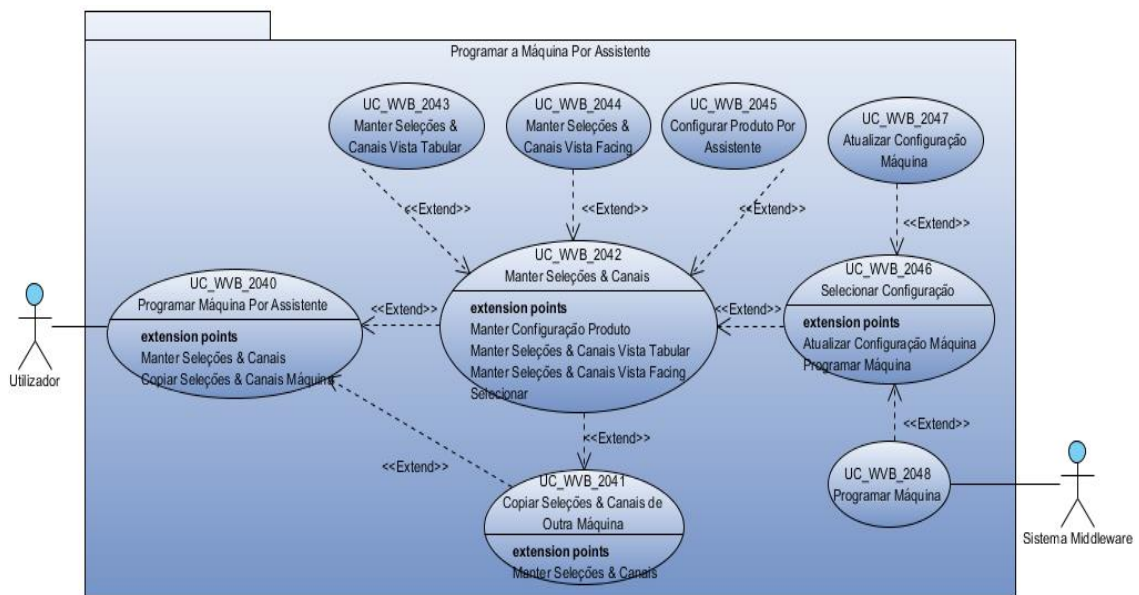


Figura 25 – Caso de Uso de Assistente de Programação da Máquina

#### *Manter Módulo*<sup>85</sup> (UC\_WVB\_1170)

No sistema *Backoffice* permite-se a associação de um módulo à máquina (UC\_WVB\_1072) e vice-versa (UC\_WVB\_1071), assim como, a gestão dos estados do módulo (c.f. Figura 27) e o respetivo histórico de alterações (UC\_WVB\_1073). As referidas alterações refletem-se, automaticamente, no sistema *Middleware*. Representa-se, na Figura 26, os principais casos de uso de manutenção do módulo e seus intervenientes (utilizador e sistema *Middleware*).

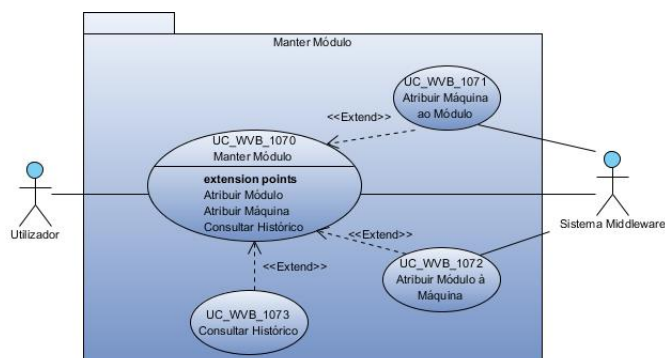


Figura 26 – Manter Módulo

<sup>85</sup> Para mais informações consultar o caso de uso (c.f. Tabela 83)

Visualiza-se, na Figura 27, as várias etapas de um módulo, desde a sua criação à sua remoção. Assim, inicia-se o ciclo de vida do módulo com a sua criação, no sistema *Backoffice*. Neste ponto, atribui-se o estado inicial predefinido, denominado “Livre”. Posteriormente, realiza-se a associação do módulo à máquina ou vice-versa (UC\_WVB\_1072 e UC\_WVB\_1071 respetivamente) e o seu estado torna-se “Desativo”. Neste estado, as comunicações estão desativadas (i.e. não se enviam alarmes e não se permitem operações). A ativação das comunicações da máquina (c.f. UC\_WVB\_2003 da Figura 24) coloca o módulo no estado “Ativo”. A partir desse momento, rececionam-se os alarmes e permite-se a execução de pedidos de comunicação e programação. Numa ação semelhante (UC\_WVB\_2003), o utilizador desativa as comunicações e o módulo passa, novamente, para o estado “Desativo”. Permite-se, nestas condições, a desassociação do módulo da máquina (UC\_WVB\_1072 e UC\_WVB\_1071). Como consequência, o estado regressa ao seu estado original “Livre” (pronto para ser associado a outra máquina). Nos estados “Livre” e “Desativo”, possibilita-se a indicação de avaria do módulo, com a atribuição do estado de “Danificado”. Numa situação extrema, onde o grau de dano não permite a sua recuperação, identifica-se o módulo, com o estado “Irrecuperável”. A ação de remover o módulo só é permitida caso não tenha sido realizada qualquer alteração ao módulo.

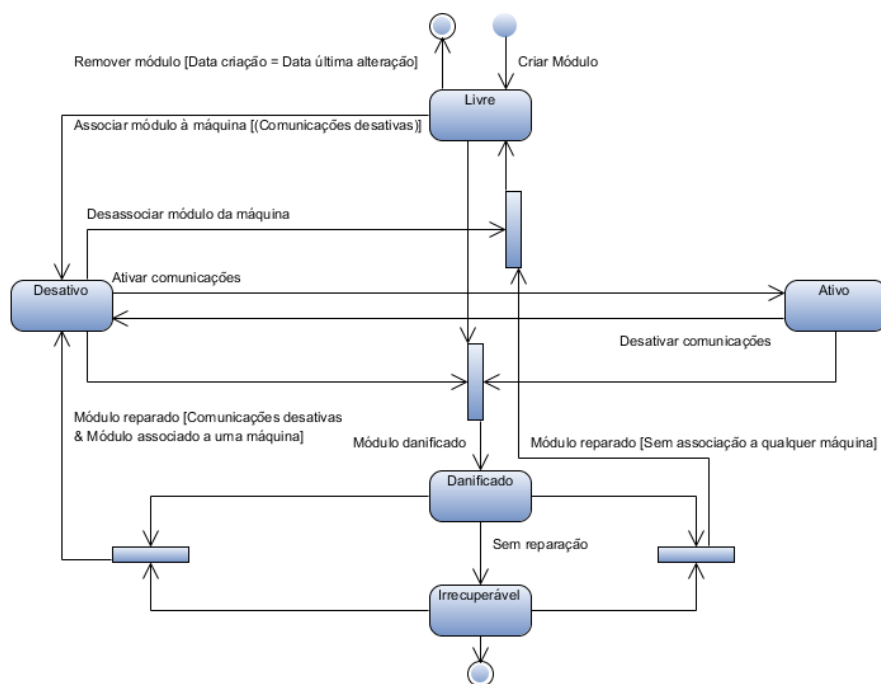


Figura 27 – Diagrama de estados – Estados Módulo

### Notificar Alarmes<sup>86</sup> (UC\_WVM\_0001)

A partir do momento, em que as comunicações da máquina se encontram ativas, o *Middleware* notifica o *Backoffice* dos alarmes ocorridos na máquina (e.g. porta aberta/fechada, máquina ligada/desligada, movimento e localização da máquina).

Aquando da notificação de um alarme de uma máquina, o sistema *Backoffice* verifica se o estabelecimento, onde a máquina está instalada, possui uma “Central de Alarmes” associada. No caso positivo, confirma se o alarme ocorre fora do horário de trabalho do estabelecimento e, ao verificar-se essa condição, reencaminha-se o alarme, automaticamente, para o sistema WinGuard (c.f. Figura 28).

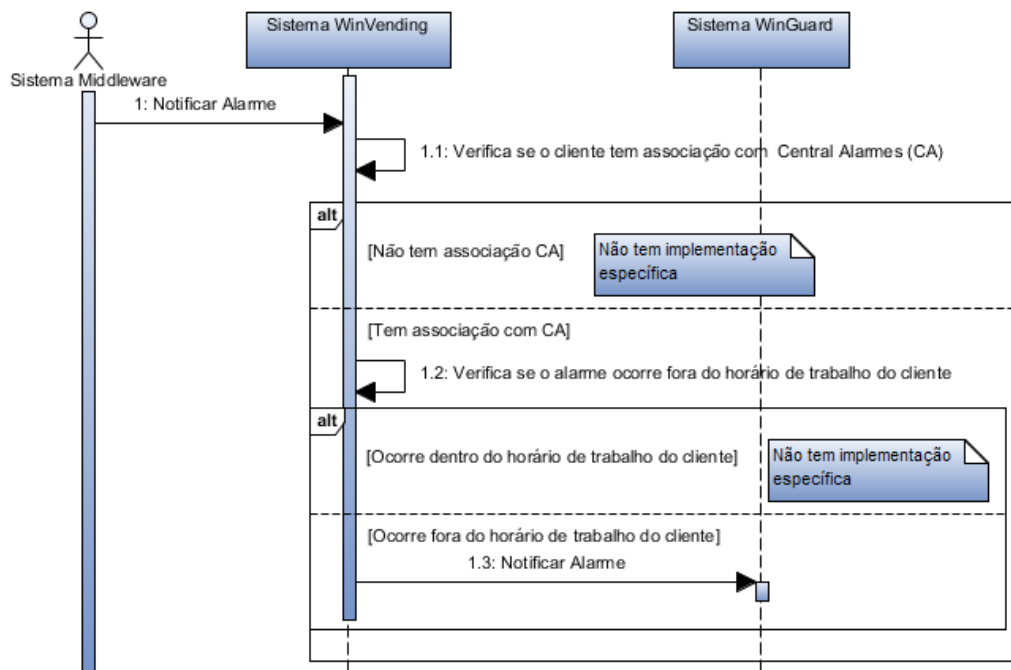


Figura 28 – Diagrama de Sequência de Notificação de Alarmes

### Notificar Operações<sup>87</sup> (UC\_WVM\_0002)

O *Backoffice* realiza pedidos de comunicação e programação (canais, seleções, preços, código de operador e mensagem publicitária) ao *Middleware*, o qual devolve um identificador do

<sup>86</sup> Para mais informações consultar o caso de uso (c.f. Tabela 87/Tabela 55)

<sup>87</sup> Para mais informações consultar o caso de uso (c.f. Tabela 88)

pedido. Após a execução do pedido, o *Middleware* envia ao *Backoffice* uma resposta positiva ou negativa (c.f. Figura 29) com os respetivos dados associados (e.g. valores de comunicação). Ao rececionar a operação, o *Backoffice*, associa a resposta recebida ao pedido solicitado e efetua o devido tratamento, de acordo com a operação em causa.

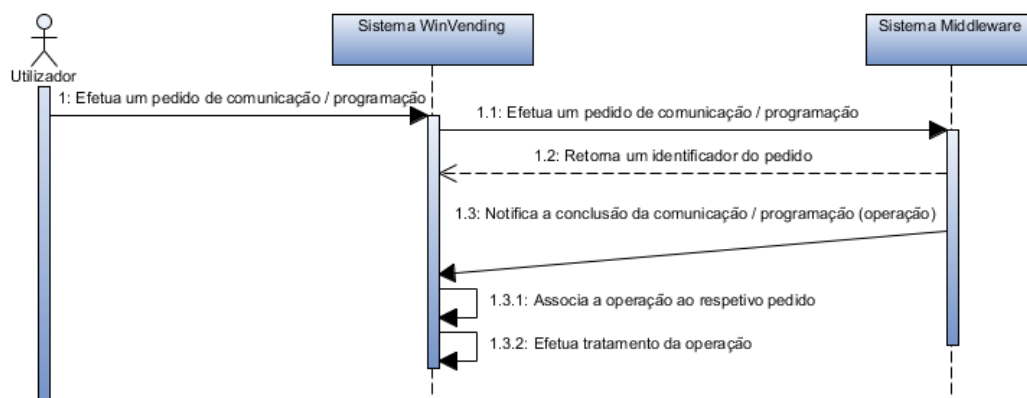


Figura 29 – Diagrama de Sequência de Notificação de Operações

### 5.3.2 WinGuard

Identificam-se, nesta secção, os requisitos do sistema WinGuard que, recorde-se, tem como objetivo filtrar os alarmes ocorridos nas máquinas, fora do horário de trabalho do estabelecimento.

Comprometem-se as seguintes entidades na definição e validação dos requisitos: (i) o cliente (depositário de tabaco), (ii) a ABS Informática, (iii) os parceiros na área de M2M (Vodafone e KTS) e (iv) uma empresa de segurança (<sup>88</sup>Global8Track).

Começa-se por testar a componente de telemetria e verificar o seu funcionamento, em ambiente real. Os testes realizados apresentam resultados positivos e segue-se a seleção do modo de disponibilizar os alarmes. Seleciona-se uma aplicação dedicada à monitorização dos alarmes (c.f. subcapítulo 4.7.4). Assim, a ABS Informática reúne-se com o seu cliente final, para

---

<sup>88</sup> “A Global8Track é uma empresa, filial do grupo Global8Link, que presta serviços na área da segurança (física e tecnológica), geolocalização e evacuações terrestres e aéreas.” (Global8Track 2015). Sítio da empresa: <http://www.global8track.com/>.

determinarem os requisitos iniciais do sistema. Dessa reunião, surgem os requisitos iniciais do sistema: (i) notificar os alarmes ocorridos em horário pós-laboral dos estabelecimentos (UC\_WVB\_001), (ii) consultar os alarmes (UC\_WCG\_001); (iii) identificar a máquina e a entidade a notificar (UC\_WCG\_010) e (iv) fechar os alarmes tratados (UC\_WCG\_020).

Implementam-se os requisitos essenciais ao funcionamento do WinGuard e o cliente contrata uma empresa de segurança, para monitorizar os alarmes recebidos. O sistema entra em produção e, após alguns meses de trabalho, os operadores da central de alarmes apresentam algumas observações: (i) apresentam-se demasiados alarmes e (ii) dificuldade em distinguir alarmes prioritários.

Para a resolução do primeiro ponto, a ABS Informática organiza uma reunião por vídeo conferência a três (ABS Informática, cliente final, empresa de segurança), com o objetivo de se ajustarem os requisitos definidos. Posteriormente, e devido a especificidades técnicas que originam a ocorrência de vários alarmes, a ABS Informática contacta os seus parceiros em M2M, de modo a validar a razão de ser para tal facto. Conclui-se que o motivo da ocorrência excessiva de alarmes se deve à sensibilidade do sensor de movimento. Para a resolução dessa situação, a parceria disponibiliza-se para a calibração da sensibilidade do sensor, mediante a concordância das partes envolvidas.

De modo a distinguir as diferentes prioridades de alarmes, sugere-se a categorização dos alarmes em: (i) prioritários (UC\_WVG\_003) e (ii) informativos (UC\_WVG\_004). Para facilitar a consulta de alarmes, cria-se um novo requisito relativo à consulta personalizada dos alarmes (UC\_WVG\_002).

A Figura 30 representa o caso de uso do sistema WinGuard e as suas interações com os atores: operador e sistema WinVending.

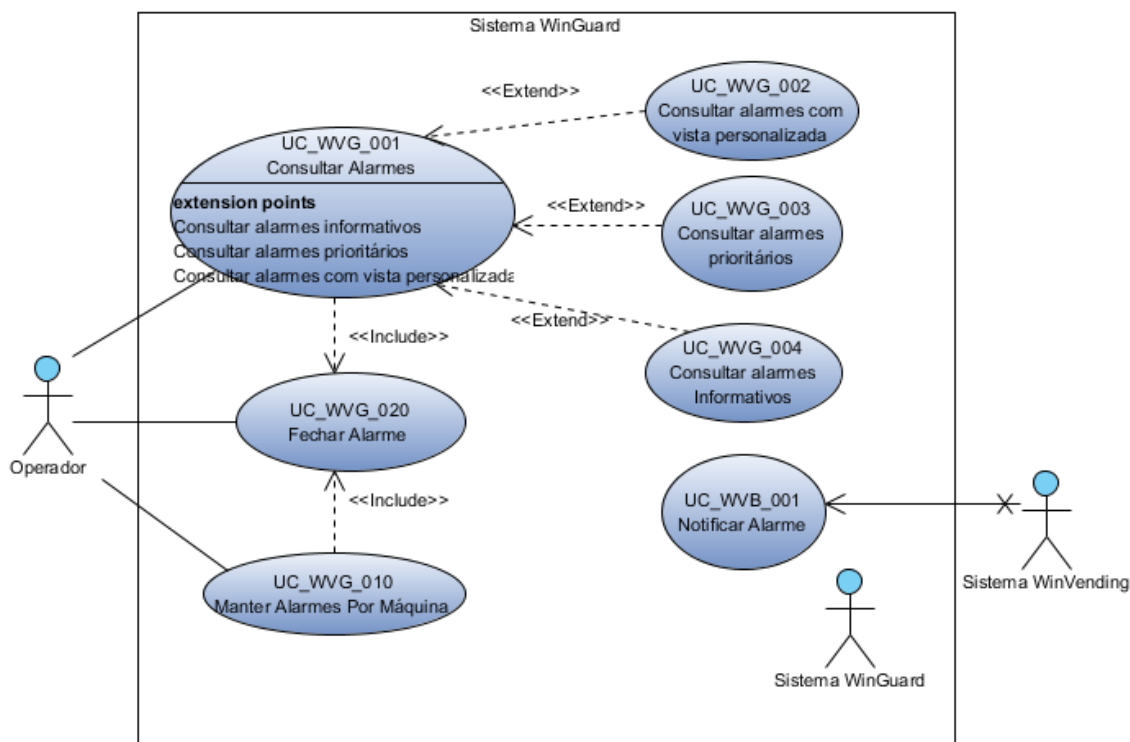


Figura 30 – Caso de uso “Central de Alarme”

*Notificar alarme<sup>89</sup> (UC\_WVB\_001)*

O sistema WinVending, mediante determinadas condições identificadas no diagrama de seqüência da Figura 28, notifica o WinGuard da ocorrência de certos alarmes. Ao rececionar o alarme, o sistema procede à sua categorização (c.f. Figura 14) e armazenamento.

*Consultar alarmes<sup>90</sup> (UC\_WVG\_001)*

Na abertura da aplicação WinGuard, observa-se, do lado direito, uma lista de alarmes e, do lado esquerdo, um mapa com a localização das respetivas máquinas (c.f. Figura 145).

A listagem das máquinas encontra-se agrupada em três categorias: (i) prioritários, (ii) informativos e (iii) visão personalizada. De modo predefinido, são automaticamente apresentados os alarmes prioritários. Cada linha da lista apresenta os seguintes dados da máquina: (i) o ícone figurativo do último alarme, (ii) o nome do estabelecimento, (iii) o número

<sup>89</sup> Para mais informações consultar o caso de uso (c.f. Tabela 89)

<sup>90</sup> Para mais informações consultar o caso de uso (c.f. Tabela 90)

da máquina, (iv) o contacto do estabelecimento, (v) a data e hora do alarme e (vi) uma ilustração dos diferentes alarmes ocorridos na máquina, com o respetivo número de ocorrências. Permite-se, após seleccionar-se uma máquina, o fecho de todos os alarmes associados.

O mapa apresenta a localização de todas as máquinas listadas, no separador selecionado. Ao efetuar-se uma aproximação do mapa, observa-se um ícone ilustrativo do último alarme rececionado da máquina em causa.

#### *Consultar alarmes com vista personalizada<sup>91</sup> (UC\_WVG\_002)*

Com a consulta personalizada (c.f. Figura 146), permite-se a seleção de um conjunto de filtros, de modo a visualizarem-se os alarmes pretendidos. Destacam-se os seguintes filtros: (i) intervalo de datas, (ii) intervalo de número de máquina, (iii) critério de abertura, (iv) critério de leitura e (v) listagem de tipos de ocorrência. Após a confirmação, apresenta-se uma lista de alarmes, segundo os filtros selecionados.

#### *Consultar alarmes prioritários<sup>92</sup> (UC\_WVG\_003)*

A consulta de alarmes prioritários é a consulta predefinida, ao abrir-se a aplicação e apresenta todos os alarmes prioritários, ainda não fechados.

#### *Consultar alarmes informativos<sup>93</sup> (UC\_WVG\_004)*

Permite-se nesta consulta a visualização de todos os alarmes informativos não fechados.

#### *Consultar alarmes por máquina<sup>94</sup> (UC\_WVG\_010)*

Ao seleccionar-se uma máquina, visualizam-se os dados completos, conhecidos da máquina (c.f. Figura 147): (i) nome do estabelecimento, (ii) número da máquina, (iii) morada, (iv) número de telefone, (v) número de telemóvel, (vi) nome do depositário, (vii) contacto da autoridade a contactar, em caso de emergência, (viii) horário de trabalho, (ix) mapa com a localização da máquina e (x) lista de alarmes da máquina.

---

<sup>91</sup> Para mais informações consultar o caso de uso (c.f. Tabela 91)

<sup>92</sup> Para mais informações consultar o caso de uso (c.f. Tabela 92)

<sup>93</sup> Para mais informações consultar o caso de uso (c.f. Tabela 93)

<sup>94</sup> Para mais informações consultar o caso de uso (c.f. Tabela 94)

Permite-se diferentes visões da lista de alarmes: prioritários, informativos e com uma vista personalizada. A listagem de máquinas é composta, independentemente da categoria selecionada, pelos seguintes campos: (i) a categoria do alarme, (ii) um ícone ilustrativo do tipo de alarme, (iii) a descrição do tipo de alarme e (iv) a data do alarme.

Possibilita-se, ainda, o fecho de um alarme específico ou de um conjunto de alarmes.

#### *Fechar alarme<sup>95</sup> (UC\_WVG\_020)*

Disponibiliza-se a opção de fecho na listagem de máquinas e na listagem de alarmes da máquina. Ao efetuar-se a opção de fecho, através da seleção de uma máquina, fecham-se todos os alarmes relacionados. Ao concretizar-se a operação de fecho, na lista de alarmes, permite-se a seleção dos alarmes que se pretendem fechar.

## **5.4 Requisitos Não Funcionais**

Nesta secção, realçam-se os requisitos não funcionais <sup>96</sup>gerais dos sistemas em estudo que, segundo a classificação FURPS+, engloba as categorias: (i) usabilidade (carácter <sup>97</sup>“U”), (ii) confiabilidade (carácter <sup>98</sup>“R”), (iii) desempenho (carácter <sup>99</sup>“P”), (iv) suporte (carácter <sup>100</sup>“S”) e (v) outras restrições específicas de cada sistema (carácter <sup>101</sup>“+”). Em cada categoria, descrevem-se os respetivos requisitos não funcionais, em duas tabelas distintas por sistema: (i) WinVending *Backoffice* (WVB) e WinGuard (WGD). Para cada requisito, atribui-se uma chave única composta por: (i) abreviatura do sistema (e.g. WVB), (ii) abreviatura da categoria do requisito (e.g. USB) e (iii) um número sequencial (e.g. 001).

---

<sup>95</sup> Para mais informações consultar o caso de uso (c.f. Tabela 95)

<sup>96</sup> Nesta secção, abordam-se os requisitos não funcionais comuns a mais do que um caso de uso. Os requisitos individuais de um caso de uso são mencionados, diretamente, na sua especificação.

<sup>97</sup> Carácter “U” do inglês “Usability”.

<sup>98</sup> Carácter “R” do inglês “Reliability”.

<sup>99</sup> Carácter “P” do inglês “Performance”.

<sup>100</sup> Carácter “S” do inglês “Support”.

<sup>101</sup> O carácter “+” permite a especificação de restrições que incluem o desenho e a implementação. Permite-se a inclusão de outros critérios que se ajustem ao sistema em análise.

### 5.4.1 Usabilidade (USB)

Avalia-se, na categoria usabilidade, a interface com o utilizador em vertentes como: (i) estética e (ii) consistência da interface. Na Tabela 14 e na Tabela 15, listam-se os requisitos não funcionais de “usabilidade” para os sistemas WinVending *Backoffice* e WinGuard.

Tabela 14 – Requisitos não funcionais de “usabilidade” do WinVending *Backoffice*

ID	Descrição
WVB_USB_001	Os ícones do sistema devem ser utilizados em contextos semelhantes.
WVB_USB_002	Na manipulação de tabelas, deve existir um menu fixo (superior, inferior ou lateral), com todas as funcionalidades e um menu contextual, com as funcionalidades principais.
WVB_USB_003	Os campos obrigatórios devem estar assinalados (e.g. negrito), de um modo intuitivo, para o utilizador final.
WVB_USB_004	Deve-se identificar, claramente, um campo em falta, ou incorretamente preenchido.
WVB_USB_005	As mensagens de erro devem ser claras e sugestivas de correção.
WVB_USB_006	O alinhamento e tamanho dos campos devem ser iguais para campos com o mesmo significado (e.g. data, hora, booleanos, numéricos, código, nome, moradas).
WVB_USB_007	O sistema deve assumir a temática e as cores (tonalidades verdes) do sistema WinVending <i>Backoffice</i> .
WVB_USB_008	Deve-se ter em consideração a resolução mínima de “1024*680” para a aplicação.

Tabela 15 – Requisitos não funcionais de “usabilidade” do *WinGuard*

ID	Descrição
WVG_USB_001	Os ícones do sistema devem ser utilizados em contextos semelhantes e com significado claro para o utilizador.
WVG_USB_002	Os campos mais importantes devem estar assinalados (e.g. negrito), de um modo intuitivo, para o utilizador final.
WVG_USB_003	As mensagens de erro devem ser claras e sugestivas de correção.
WVG_USB_004	O alinhamento e tamanho dos campo deve ser igual para campos com o mesmo significado (e.g. data, hora, booleanos, numéricos, código, nome, moradas).
WVB_USB_005	Deve-se ter em consideração a resolução mínima de “1024*680”, para a aplicação.

#### 5.4.2 Confiabilidade (CNF)

Inclui-se, na categoria confiabilidade, aspetos como: (i) disponibilidade, (ii) precisão e (iii) tolerância a falhas dos sistemas. Na Tabela 16 e na Tabela 17, listam-se os requisitos não funcionais de “confiabilidade” para os sistemas *WinVending Backoffice* e *WinGuard*.

Tabela 16 – Requisitos não funcionais de “confiabilidade” do *WinVending Backoffice*

ID	Descrição
WVB_CNF_001	A receção de comunicações e alarmes deve estar disponível 24 horas por dia, em todos os dias do ano.
WVB_CNF_002	Em caso de demora, ou não aviso da conclusão de uma comunicação, o sistema deve monitorizar, a partir de X minutos sem resposta, a conclusão da comunicação.
WVB_CNF_003	Em caso de falhas não expectáveis (e.g. falha de rede ou comunicações) o sistema deve apresentar mensagens de erro claras e possibilitar, caso o utilizador pretenda, a reprodução da operação.

Tabela 17 – Requisitos não funcionais de “usabilidade” do *WinGuard*

ID	Descrição
WVG_CNF_001	A receção de alarmes deve estar disponível 24 horas por dia, em todos os dias do ano.
WVG_CNF_002	Em caso de falhas não expectáveis (e.g. falha de rede ou comunicações), o sistema deve apresentar mensagens de erro claras e possibilitar, caso o utilizador pretenda, a reprodução da operação.

### 5.4.3 Desempenho (DSP)

Envolvem-se, na categoria desempenho, situações como: (i) tempos de resposta do sistema, (ii) tempos de recuperação e (iii) tempo de arranque. Na Tabela 33 e na Tabela 34, listam-se os requisitos não funcionais de “desempenho” para os sistemas *WinVending Backoffice* e *WinGuard*.

Tabela 18 – Requisitos não funcionais de “desempenho” do *WinVending Backoffice*

ID	Descrição
WVB_DSP_001	O carregamento de listas deve ser realizado em segundo plano, de modo a não afetar o desempenho de visualização de outros dados.
WVB_CNF_002	O sistema deve ser desenhado de modo a que o utilizador continue a trabalhar, enquanto se realizam tarefas, em segundo plano.
WVB_CNF_003	Os utilizadores acedem ao sistema em horário laboral, normalmente, entre as 07:00 e as 20:00, com a pressão de produzir resultados rapidamente. Por isso, o sistema deve apresentar uma resposta rápida às solicitações do utilizador, independentemente da execução de qualquer tarefa, em segundo plano.

Tabela 19 – Requisitos não funcionais de “desempenho” do *WinGuard*

ID	Descrição
WVG_DSP_001	O carregamento de listas deve ser realizado, em segundo plano, de modo a não afetar o desempenho de visualização de outros dados.
WVG_CNF_002	O sistema deve ser desenhado de modo a que o utilizador continue a trabalhar, enquanto se realizam tarefas em segundo plano.
WVG_CNF_003	Os operadores acedem ao sistema 24 horas por dia, 7 dias por semana, com a pressão de obter dados do sistema, rapidamente. Por isso, o sistema deve apresentar uma resposta célere às solicitações do utilizador, independentemente da execução de qualquer tarefa, em segundo plano.
WVG_CNF_004	De modo a que o operador tenha acesso imediato à aplicação, considera-se um minuto como o período máximo de arranque do sistema.

#### 5.4.4 Suporte (SPR)

Menciona-se, na categoria suporte, requisitos como: (i) testabilidade, (ii) adaptabilidade, (iii) compatibilidade, (iv) manutenção, (v) configuração, (vi) instalação, (vii) escalabilidade e (viii) internacionalização. Na Tabela 20 e na Tabela 21, listam-se os requisitos não funcionais de “suporte” para os sistemas *WinVending Backoffice* e *WinGuard*.

Tabela 20 – Requisitos não funcionais de “suporte” do WinVending *Backoffice*

<b>ID</b>	<b>Descrição</b>
WVB_SPR_001	O sistema deve estar preparado para suportar múltiplos ERP's.
WVB_SPR_002	O sistema deve estar preparado para ampliar as suas funcionalidades.
WVB_SPR_003	O sistema deve estar preparado para alterar a interface com o utilizador, sem comprometer a lógica de negócio e o acesso aos dados.
WVB_SPR_004	Alterações realizadas ao modelo de dados devem ser transparentes, para o técnico que efetua a instalação.
WVB_SPR_005	A aplicação deve ser compatível com os sistemas operativos da Microsoft Windows superiores ao Windows XP.
WVB_SPR_005	O sistema deve ser testável.
WVB_SPR_006	Após uma nova instalação, deve-se manter as anteriores configurações.
WVB_SPR_007	A atualização da aplicação deve ser sugerida ao utilizador, no momento em que entra.
WVB_SPR_008	O sistema deve estar preparado para um grande número de transações de alarmes e comunicações.
WVB_SPR_009	O sistema deve permitir a seleção de múltiplos idiomas. Inicialmente os idiomas: (i) português, (ii) castelhano e (iii) inglês.

Tabela 21 – Requisitos não funcionais de “suporte” do WinGuard

<b>ID</b>	<b>Descrição</b>
WVG_SPR_001	O sistema deve estar preparado para ampliar as suas funcionalidades.
WVG_SPR_002	O sistema deve estar preparado para alterar a interface com o utilizador, sem comprometer a lógica de negócio e o acesso aos dados.
WVG_SPR_003	Alterações realizadas ao modelo de dados deve ser transparente para o técnico que efetua a instalação.
WVG_SPR_004	A aplicação deve ser compatível com os sistemas operativos da Microsoft Windows, superiores ao Windows XP.
WVG_SPR_005	O sistema deve ser testável.
WVG_SPR_006	Após uma nova instalação, devem manter-se as anteriores configurações.

#### 5.4.5 Desenho (DSN)

Especifica-se, na categoria desenho, restrições de: (i) linguagem de programação, (ii) utilização de ferramentas de desenvolvimento e (iii) biblioteca de classe. Na Tabela 22 e na Tabela 23, listam-se os requisitos não funcionais de “desenho” para os sistemas WinVending *Backoffice* e WinGuard.

Tabela 22 – Requisitos não funcionais de “desenho” do WinVending *Backoffice*

ID	Descrição
WVB_DSN_001	A linguagem de programação para a interface com o utilizador deve ser Visual Basic .NET.
WVB_DSN_002	Deve-se utilizar a biblioteca de controlos de interface Infragistics <sup>102</sup> .
WVB_DSN_003	Para efeitos de autenticação, autorização e gestão de utilizadores e empresas, deve-se utilizar a solução proprietária da ABS Informática Administrador.
WVB_DSN_004	Deve existir um manual do utilizador que o guie nas tarefas da aplicação. Realça-se a gestão: (i) das máquinas, (ii) dos módulos e (iii) das comunicações.
WVB_DSN_005	Deve existir um manual de instalação da aplicação, que guia a equipa de suporte, na instalação e atualização do sistema.

Tabela 23 – Requisitos não funcionais de “desenho” do WinGuard

ID	Descrição
WVG_DSN_001	A linguagem de programação, para a interface com o utilizador, deve ser Visual Basic .NET.
WVG_DSN_002	Deve-se utilizar a biblioteca de controlos de interface Infragistics.

---

<sup>102</sup> Infragistics é uma empresa de soluções que aceleram o desenho, o desenvolvimento e a colaboração entre os membros da equipa. Sítio da empresa Infragistics: <https://www.infragistics.com>. (Infragistics 2017)

#### 5.4.6 Implementação (MPL)

Distinguem-se, na categoria implementação, requisitos que restringem o código ou a construção do sistema. Na Tabela 24 e na Tabela 25, listam-se os requisitos não funcionais de “implementação” para os sistemas WinVending *Backoffice* e WinGuard.

Tabela 24 – Requisitos não funcionais de “implementação” do WinVending *Backoffice*

ID	Descrição
WVB_MPL_001	A comunicação com serviços deve utilizar os padrões <i>Service Interface</i> <sup>103</sup> e <i>Service Gateway</i> <sup>104</sup> .
WVB_MPL_002	Deve-se adotar o padrão DTO <sup>105</sup> , como contrato entre os padrões <i>Service Interface</i> e <i>Service Gateway</i> .
WVB_MPL_003	A arquitetura interna das aplicações deve seguir o padrão três camadas <sup>106</sup> .
WVB_MPL_004	O nome das tabelas e campos das tabelas existentes, no sistema legado, devem manter-se inalteradas.
WVB_MPL_005	Todas as tabelas devem ter um campo de incremento automático, que pode, ou não, ser a chave primária da tabela.
WVB_MPL_006	As tabelas de manutenção devem incluir campos que armazenem os dados do utilizador e respetiva data de criação e alteração. Também, devem ter um campo automático que funcione como selo, para detetar situações de concorrência.

---

<sup>103</sup> *Service Interface* é um padrão arquitetural de serviços. (Trowbridge et. Al. 2003)

<sup>104</sup> *Service Gateway* é um padrão arquitetural de serviços. (Trowbridge et. Al. 2003)

<sup>105</sup> DTO do inglês Data Transfer Object é um padrão arquitetural de sistemas distribuídos. (Trowbridge et. Al. 2003)

<sup>106</sup> Três camadas é um padrão aplicacional. (Trowbridge et. Al. 2003)

Tabela 25 – Requisitos não funcionais de “implementação” do WinGuard

ID	Descrição
WVG_MPL_001	A comunicação com serviços deve utilizar os padrões <i>Service Interface</i> e <i>Service Gateway</i> .
WVG_MPL_002	Adotar o padrão DTO como contrato entre os padrões <i>Service Interface</i> e <i>Service Gateway</i> .
WVG_MPL_003	A arquitetura interna das aplicações deve seguir o padrão aplicativo de três camadas.
WVG_MPL_004	Todas as tabelas devem ter um campo de incremento automático que pode, ou não, ser a chave primária da tabela.
WVG_MPL_005	As tabelas de manutenção devem incluir campos que armazenem os dados do utilizador e respetiva data de criação e alteração. Também, devem ter um campo automático que funcione como selo, para detetar situações de concorrência.

#### 5.4.7 Interface (NTR)

Define-se, na categoria interface, requisitos para interagir com sistemas externos. (Eeles 2004)

Na Tabela 20 e na Tabela 21, listam-se os requisitos não funcionais de “interface” para os sistemas WinVending *Backoffice* e WinGuard.

Tabela 26 – Requisitos não funcionais de “interface” do WinVending *Backoffice*

<b>ID</b>	<b>Descrição</b>
WVB_NTR_001	Utilizar-se o estilo arquitetural REST para comunicar com as API's internas.
WVB_NTR_002	Usar-se o estilo arquitetural REST para comunicar com a API do Middleware.
WVB_NTR_003	Deter-se o documento guia da API do Middleware atualizado.
WVB_NTR_004	Deve-se utilizar o protocolo https, para a comunicação com a API do Middleware
WVB_NTR_005	Expor-se todos os DTO's das API's em inglês.
WVB_NTR_006	Representam-se os dados de data e hora no formato UTC <sup>107</sup> e ISO 8601 <sup>108</sup> .

Tabela 27 – Requisitos não funcionais de “interface” do WinGuard

<b>ID</b>	<b>Descrição</b>
WVG_NTR_001	Utilizar-se o estilo arquitetural REST para comunicar com as API's internas.
WVG_NTR_002	Expor-se todos os DTO's das API's em inglês.
WVG_NTR_003	Representam-se os dados de data e hora no formato UTC e ISO 8601.

---

<sup>107</sup> “O Tempo Universal Coordenado, abreviadamente UTC (do inglês Universal Time Coordinated), também conhecido como tempo civil, é o fuso horário de referência a partir do qual se calculam todas as outras zonas horárias do mundo.” (Tempo Universal Coordenado 2017)

<sup>108</sup> “A ISO 8601 é uma norma internacional para representação de data e hora emitida pela Organização Internacional para Padronização (International Organization for Standardization, ISO)” (ISO 8601 2017)



## 6 Desenho

Inicia-se, este capítulo, pela visão geral dos sistemas e o modo como comunicam. Posteriormente, especificam-se os componentes de cada um dos sistemas e as suas relações internas. Avança-se para a ilustração do modelo conceptual de classes e finaliza-se com a descrição do modelo de dados utilizado.

### 6.1 Visão Geral

Tal como referido na secção 3.2, o sistema Vodafone WinVending é constituído por diferentes elementos (e.g. máquina, módulo) que comunicam uns com os outros. Com a finalidade de dotar a máquina da capacidade de comunicação com o exterior, instala-se um módulo com um cartão SIM Global da Vodafone, habilitado a comunicar, por GSM ou 3G, com os sistemas: (i) GDSP e (ii) Middleware. Assim, o módulo comunica com os componentes internos da máquina e transmite os seus dados, bem como, os alarmes captados pelos seus sensores, por dois canais distintos: (i) UDP e (ii) TCP. Opta-se pelo canal UDP, para o envio de alarmes da máquina, que ocorrem com mais frequência e cujo controlo é menos sensível. Por sua vez, utiliza-se o canal TCP para a transmissão de dados mais complexos e cujo controlo de entrega é assegurado por meio de uma conexão estabelecida, entre o porto de origem e o porto de destino. Garante-se a gestão dos cartões SIM por meio de uma API concedida pelo sistema GDSP, que possibilita a ativação / desativação dos cartões, assim como, a verificação das condições de rede.

Quando o sistema Middleware recebe um alarme ou dados de comunicação da máquina, começa por armazená-los, temporariamente, numa base de dados local e, de seguida, transmiti-los para o respetivo sistema WinVending do depositário associado ao módulo instalado na máquina. A comunicação entre os sistemas Middleware e WinVending concretiza-

se nos dois sentidos. Por um lado, o sistema Middleware envia os alarmes e as comunicações para o sistema WinVending. Por outro lado, o sistema WinVending efetua pedidos de comunicação e consulta de dados (e.g. módulos, modelos base e máquinas) ao sistema Middleware. Realizam-se as comunicações entre os dois sistemas assentes no protocolo de comunicação segura HTTPS.

O sistema WinVending, ao receber os alarmes e as comunicações, armazena-os, numa base de dados local, associada à respetiva máquina. Encarrega-se, também, de gerir todo o modelo de dados da atividade de Vending (e.g. clientes, produtos, rotas, visitas, análises) e de conceder, por meio de uma API, um conjunto de serviços que mantém o sistema WinVending Mobile operacional e atualizado. Como meio de tolerância a falhas, o sistema móvel possibilita o trabalho, em modo desconectado e sincroniza, logo que possível, os dados com o sistema WinVending. Os utilizadores administrativos, habitualmente no escritório, acompanham o trabalho realizado pelos vendedores (e.g. visitas, comunicações, gestão dos módulos), através da aplicação WinVending Backoffice.

Caso o sistema do depositário faça uso do sistema WinGuard (internamente ou instalado numa empresa de segurança), no momento em que um alarme chega ao sistema WinVending, efetuam-se um conjunto de verificações (c.f. caso de uso UC\_WVM\_0001), de modo a confirmar-se a necessidade de notificação desse alarme ao WinGuard. No caso positivo, obtém-se o endereço da central de alarmes associada à máquina e envia-se o alarme para o devido sistema WinGuard. Realça-se que um depositário pode comunicar com mais do que um sistema WinGuard e que um sistema WinGuard pode gerir os alarmes de mais do que um depositário. Verifica-se esta realidade em depositários com múltiplas filiais, dispersas por todo o país e que realizam contratos com empresas de segurança, consoante a região, de modo a garantir uma cobertura total do seu parque de máquinas.

O sistema WinGuard, ao rececionar o alarme, armazena-o numa base de dados local e disponibiliza-o ao operador da central de alarme, segundo a categorização efetuada automaticamente pelo sistema (c.f. secção 5.1.5). Finalmente, o operador toma os devidos procedimentos (e.g. contactar a autoridade associada à ficha da máquina), face à análise efetuada por si e auxiliada pelo sistema.

Na Figura 31, observa-se um ponto de vista geral do sistema Vodafone WinVending, com enfoque nas entidades e sistemas envolvidos (máquina, sistema GDSP, sistema Middleware,

sistema *backoffice*, sistema móvel, sistema WinGuard) e do modo como comunicam e interagem.

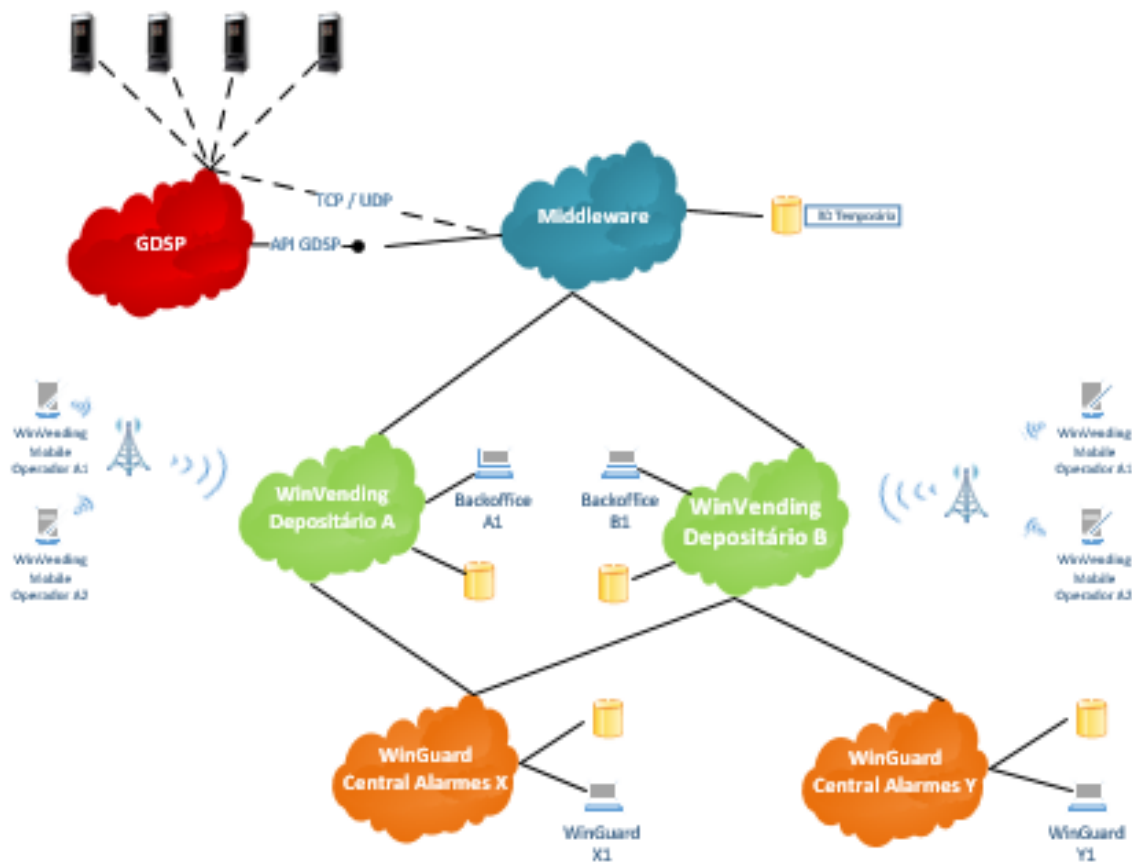


Figura 31 – Visão geral do sistema Vodafone WinVending

## 6.2 Visão WinVending

Pretende conhecer-se, nesta seção, o modo de funcionamento interno do sistema WinVending e as suas relações com os sistemas: Middleware, Mobile, Administrador e ERP's. Procura-se representar esse propósito, na Figura 32, através de um diagrama de componentes, onde se referem os principais componentes do sistema e as interfaces que disponibilizam e consomem.

Observa-se que o sistema Middleware disponibiliza uma interface (e.g. efetuar pedidos de comunicação e programação) e consome uma interface de um *callback* por si definido. O *callback* recebe alarmes e respostas de pedidos de comunicação e direciona-os para as respetivas filas de mensagens (e.g. alarmes, comunicações, programações). Por sua vez, existe

um serviço que consome as mensagens das respetivas filas e invoca os respetivos serviços da API do WinVending para os processar.

A API do WinVending, constituída por um conjunto de serviços com finalidades distintas, funciona como o motor do sistema. Abstrai a lógica de negócio, acesso a dados e acesso a serviços externos (e.g. Middleware, ERP's, Administrador, WinGuard) das aplicações que a invocam (*backoffice*, móvel, monitor e consumidor de mensagens). A aplicação *backoffice* começa por invocar a API WinVending, para efetuar a credenciação junto da API do Administrador. Em caso de sucesso, a API do Administrador devolve uma chave que permite a identificação do utilizador, na invocação dos pedidos seguintes. A aplicação móvel invoca a API WinVending para efetuar tarefas de: (i) manutenção do estado da rota / visita, (ii) comunicações via telemetria, ou atualização das comunicações efetuadas por *Bluetooth* e (iii) programação da máquina via telemetria, ou atualização de programações efetuadas por *Bluetooth*. O serviço de monitor executa: (i) operações agendadas (e.g. comunicações), (ii) comunicações pendentes e (iii) tarefas de sincronização (e.g. tabelas do ERP).

Comunica-se com os ERP's (e.g. Eticadata e PHC) através da invocação de pedidos junto da API Integrador ERP. Desses pedidos destacam-se: (i) a obtenção de dados das tabelas gerais (e.g. clientes, produtos, vendedores), (ii) a atualização de dados específicos das tabelas gerais e (iii) a geração de documentos (e.g. documentos de venda). A ABS Informática desenvolve a API Integrador ERP, de modo a tornar transparente o acesso aos dados dos diferentes ERP's e, ao mesmo tempo, reutiliza os serviços para todas as suas soluções. O modo de implementar o acesso aos dados e a lógica de negócio variam consoante o ERP. Por exemplo, o ERP Eticadata fornece um conjunto de bibliotecas que constituem a API do ERP e que implementam a lógica de negócio e o acesso aos dados. Já o ERP PHC não disponibiliza qualquer API e fica à responsabilidade do integrador ERP implementar a lógica de negócio e o acesso direto aos dados do ERP.

O sistema do depositário, ao conectar-se com o sistema WinGuard, permite o envio de alarmes para o mesmo, mediante as condições referidas no caso de uso UC\_WVM\_0001.

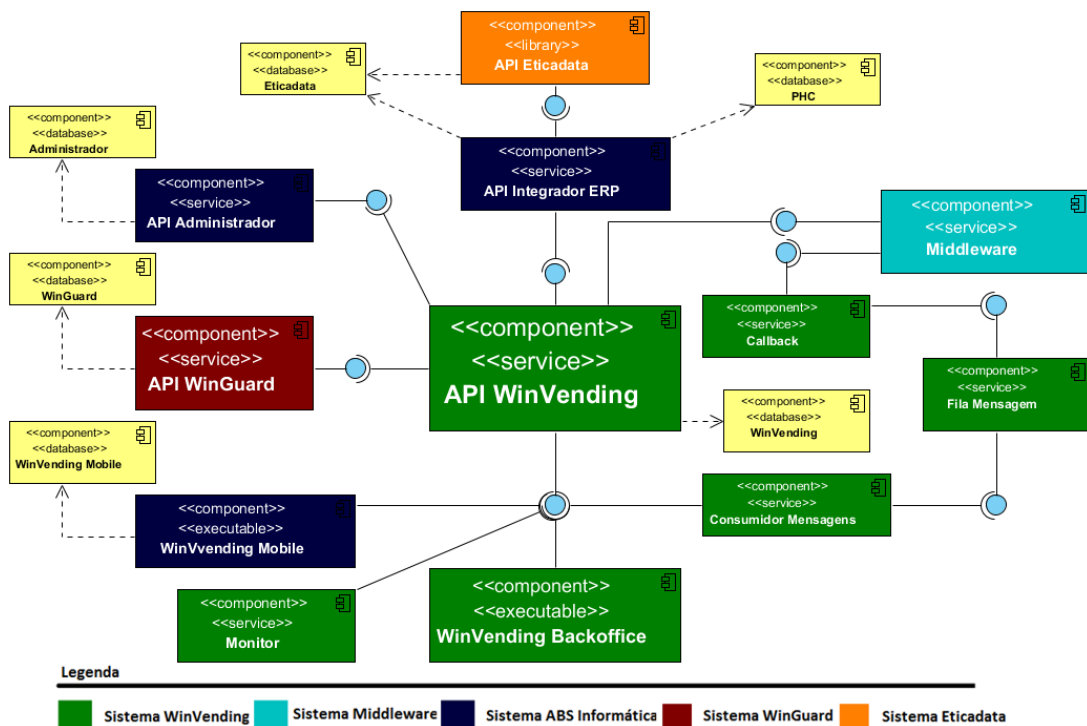


Figura 32 – Diagrama de componentes do sistema WinVending

### 6.3 Visão WinGuard

Visualiza-se, na Figura 33, o diagrama de componentes do sistema WinGuard onde se destacam: o serviço da API WinVending, o serviço da API WinGuard e a aplicação executável WinGuard. A API WinGuard concede uma interface (*callback* do sistema WinVending), invocada pela API do WinVending, para notificar a ocorrência de determinado alarme. Ao rececionar o alarme, a API WinGuard categoriza o alarme e armazena-o numa base de dados local. No final, o executável WinGuard apresenta ao operador da central de alarmes uma visão dos alarmes ocorridos.

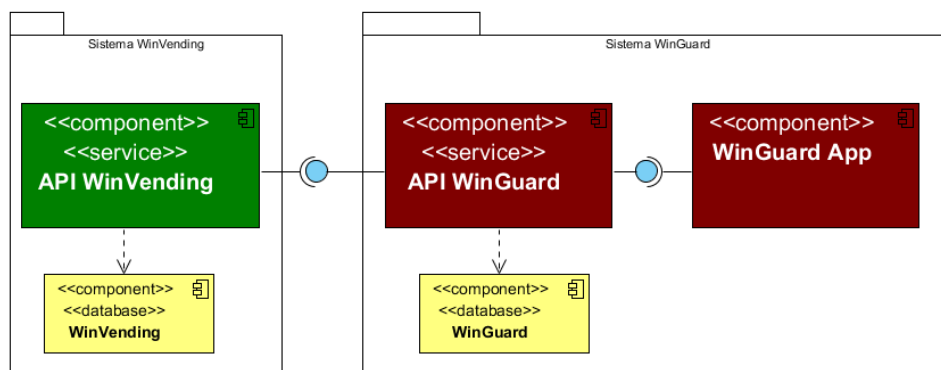


Figura 33 – Diagrama de componentes do sistema WinGuard



## Máquina

Na Figura 35, visualiza-se o diagrama conceptual de classes, sob o ponto de vista da máquina. Consta-se a constituição da máquina (e.g. estado, localização, propriedade e características técnicas) e a sua composição (moedeiros, seleções, canais, cabeçalho de comunicação, localizações, ocorrências e histórico de programação da máquina). Se os moedeiros possuírem um discriminador, capacita-se os mesmos de conterem, em simultâneo, dois tipos de moedas (unidade 1 e unidade 2). Liga-se sempre uma ocorrência à sua correspondente base, de modo a permitir a identificação do seu tipo (alarme ou anomalia).

Uma máquina tem um modelo, pode ou não ter um cliente (e.g. quando está em armazém, não tem qualquer cliente associado) e está sempre associada a uma filial. Caso se trate de uma máquina com telemetria, tem um módulo associado.

Registam-se, no sistema, todas as operações de programação da máquina: (i) seleções e canais, (ii) mensagem publicitária e (iii) código de operador.

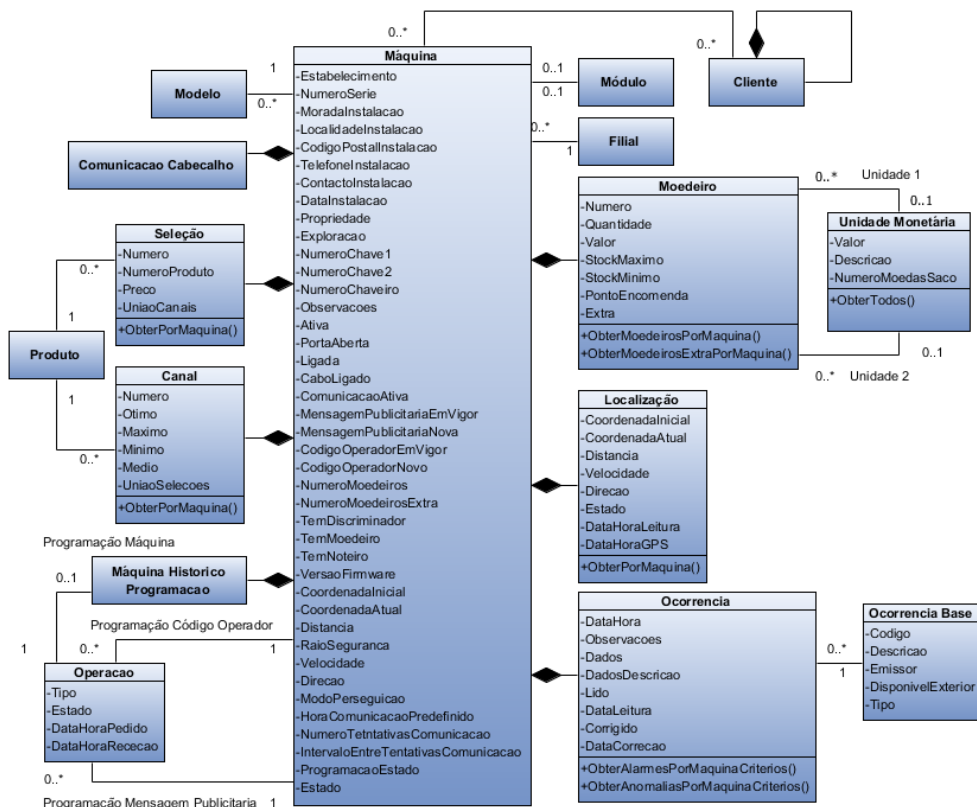


Figura 35 – Diagrama conceptual de classes da visão de máquina

## Comunicação

De acordo com o diagrama conceptual de classes da Figura 36, uma comunicação relaciona-se sempre com uma máquina e define-se a sua composição pelas seguintes entidades: (i) linhas de comunicação, (ii) moedeiros e (iii) incidências. Mantém-se, caso a comunicação seja resultado de um agendamento, a relação entre a comunicação e o agendamento que a originou.

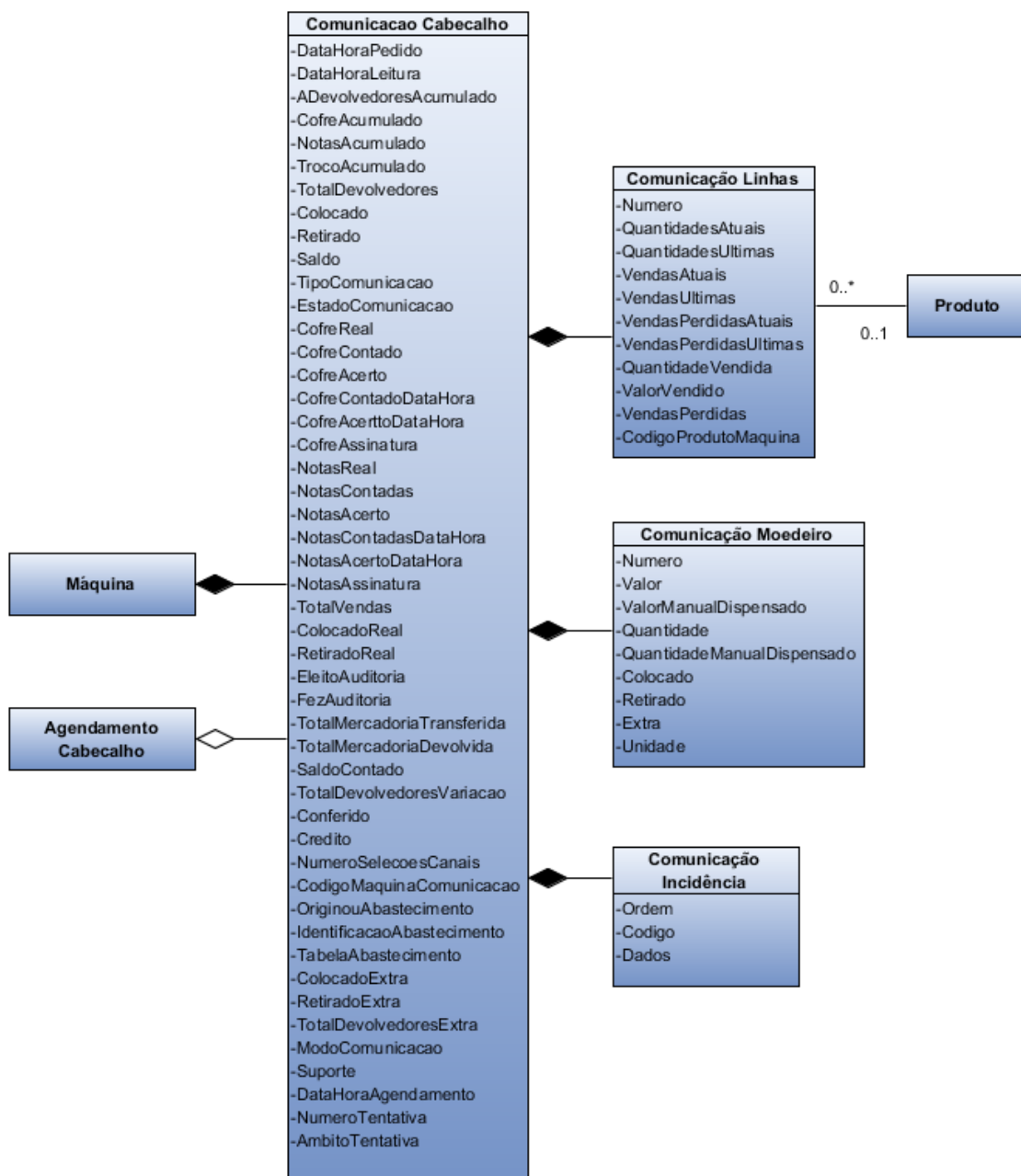


Figura 36 – Diagrama conceptual de classes da Visão de *Comunicação*

*Plano de Rota*

Verifica-se, na Figura 37, as entidades envolvidas na definição do plano de rota. Assim, na classe “Plano Rota Cabecalho”, registam-se os dados acerca da rota (e.g. início / fim, previsto / efetivo) e, na classe “Plano Rota Linha”, atualizam-se os dados da visita (e.g. início / fim, previsto / efetivo, fora de rota). Associa-se sempre uma rota a um vendedor e uma visita realiza-se a um cliente, ou a uma máquina.

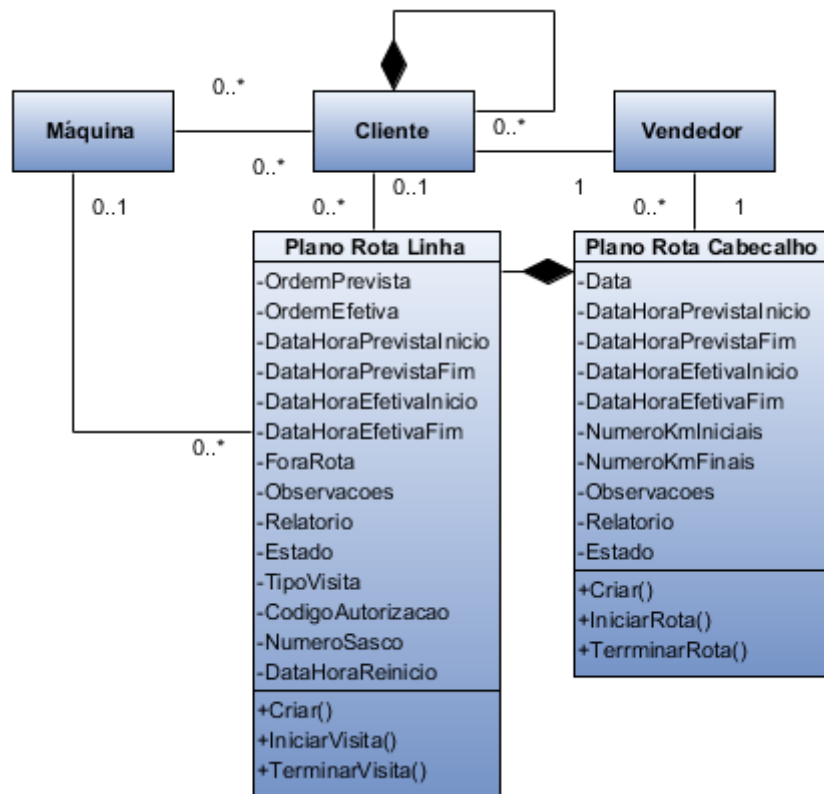


Figura 37 – Diagrama conceptual de classes da visão de plano de rota

### 6.4.1 WinGuard

Observa-se, na Figura 38, o relacionamento entre as entidades que constituem o sistema WinGuard. Destaca-se a presença de múltiplos proprietários (donos da máquina), cujas máquinas podem sofrer ocorrências de um determinado género (e.g. abertura/fecho porta, máquina ligada/desligado, movimento, GPS), as quais se categorizam e armazenam para posterior consulta.

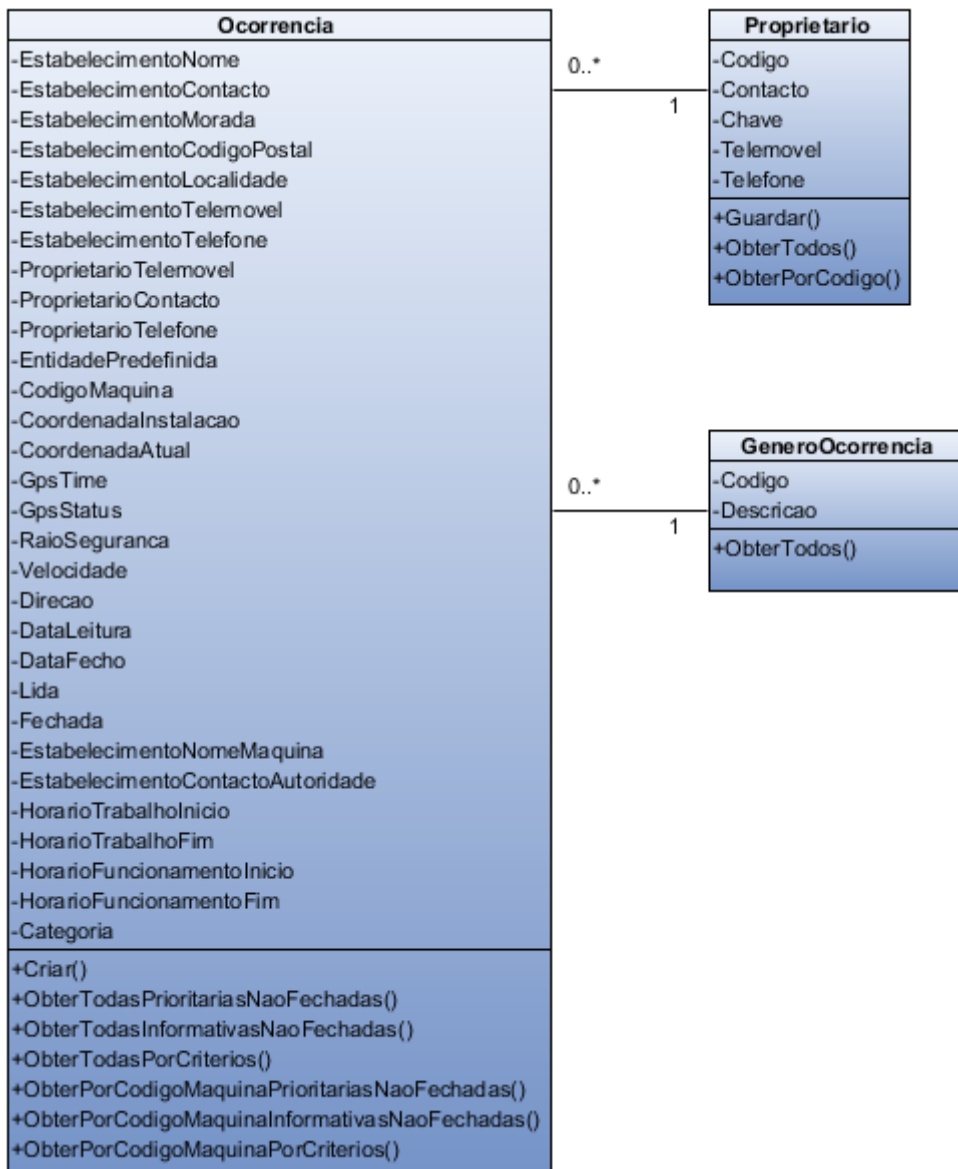


Figura 38 – Diagrama conceptual de classes do sistema WinGuard

## 6.5 Modelo de Dados

Começa-se por enunciar, nesta secção, as regras e práticas adotadas na criação: (i) das tabelas, (ii) dos campos, (iii) dos relacionamentos entre tabelas e (iv) dos índices. Apresentam-se como resultados finais os modelos de dados dos sistemas: WinVending e WinGuard.

### 6.5.1 Regras & Práticas

Consideram-se, no momento de criação das tabelas e campos dos sistemas WinVending e WinGuard, as respetivas restrições não funcionais de implementação: (i) limitação da atribuição de nomes existentes na solução legada e (ii) recomendação de criação de campos para gerir a data, hora e responsáveis pela criação e alteração dos registos, bem como, um campo para se detetarem alterações nos registos, quer se realizem através de aplicações, quer diretamente no SGBD. No que diz respeito, aos nomes das tabelas, o sistema legado abrevia, a três carateres, o nome das tabelas principais (e.g. “Maq” para representar a máquina e “Cli” para denominar clientes) e agrega mais três carateres, quando uma tabela complementa dados de uma outra tabela de nível superior (e.g. “MaqPrg”, ao conceder a componente de programação à máquina e “MaqPrgSel”, ao complementar a programação com os dados da seleção). Relativamente ao nome dos campos: (i) utilizam-se abreviaturas para campos específicos (c.f. Tabela 96) e (ii) define-se o nome das chaves estrangeiras, pela concatenação do nome da tabela estrangeira com o campo chave dessa tabela, separados pelo carácter “\_” (e.g. uma chave estrangeira, referente à tabela “Maq”, denomina-se “Maq\_Cod” e, para a tabela “Fil”, designa-se “Fil\_Id”). No que se refere à segunda restrição, salienta-se, para as tabelas principais expostas a operações de manutenção (e.g. criar, alterar), a necessidade de deterem, além dos campos específicos da tabela, quatro campos adicionais que permitem a monitorização (data / hora e responsáveis pela criação e alteração) e mais um campo de controlo de alterações (<sup>109</sup>*timestamp*).

A definição do modelo de dados, rege-se pelo princípio de normalizar primeiro e desnormalizar pelo desempenho, posteriormente. Segue-se a normalização até à terceira forma normal de

---

<sup>109</sup> “*Timestamp* é um tipo de dados que expõe números binários únicos gerados automaticamente pelo SGBD”. (tradução nossa) (Timestamp 2017)

modo a eliminarem-se dados redundantes e manter-se a integridade dos dados. Apesar dos benefícios mencionados, a normalização, por vezes, sacrifica o desempenho, e para colmatar essa lacuna efetuam-se ajustes, no sentido, de desnormalizar para melhorar o desempenho.

Recorre-se à desnormalização, no sistema WinVending, em situações onde se utilizam totalizadores que podem ser calculados a partir de campos de outras tabelas. Realça-se essa utilização na tabela de cabeçalho de comunicação (denominada de “CMAqC”) onde se armazenam os totalizadores da tabela de linhas de comunicação (designada de “CMAqL”) dos quais se referem, a título de exemplo, os seguintes: o total do valor de vendas (“TotVenda”) e o total das quantidades vendidas (“TotVendaQtd”).

Na construção das tabelas prioriza-se a definição: (i) das chaves primárias e estrangeiras, (ii) dos constrangimentos de unicidade (<sup>110</sup>*unique*), do tipo do campo (e.g. recorre-se ao tipo de campo <sup>111</sup>*geography* para se persistirem os dados de coordenadas de localizações) e dos índices das tabelas.

Colocam-se dois pontos em discussão para a atribuição da chave primária: (i) utilização de chave com ou sem significado e (ii) seleção de uma chave simples ou composta.

Considera-se uma chave com significado toda aquela que é do conhecimento prático do utilizador, por exemplo, no caso do sistema WinVending, o código da máquina. Designa-se por chave sem significado um número gerado automaticamente pelo SGBD, no momento da criação de um registo e que nunca pode ser alterado pelo utilizador, por exemplo, o campo “Id” do país. Adotam-se os seguintes critérios para a seleção da chave: (i) tem de ser única, (ii) tem de ser imutável e (iii) deve agilizar o desempenho da base de dados. Ambas as hipóteses em discussão contemplam o critério de unicidade da chave. No entanto, no que diz respeito ao segundo e terceiro critérios considera-se que variam consoante a tabela em análise. Por exemplo, no caso da máquina garante-se que o código da máquina não pode ser alterado após a sua criação, deste modo, o segundo critério também é cumprido por ambas. Nesta realidade, a solução que

---

<sup>110</sup> “A restrição *unique* garante que todos os valores em uma coluna são diferentes.” (tradução nossa) (w3schools 2017)

<sup>111</sup> “O tipo de dados espacial geográfico, *geography*, implementa-se como um tipo de dados de Common Language Runtime (CLR) .NET no SQL Server. Este tipo de dados armazena dados elipsoidais, como coordenadas de latitude e longitude do GPS.” (tradução nossa) (Geography 2017)

apresenta melhor desempenho é o código, uma vez que quando é utilizada em cenários de chave estrangeira evita a necessidade de se efetuar uma junção com a tabela de máquina para a obtenção do campo identificador automático. Analisa-se um outro exemplo, na definição do campo código como chave primária do país, depara-se com um problema de imutabilidade da chave, uma vez que, por erro humano ao digitar o código, existe a necessidade de alterá-lo e perde-se assim, o carácter imutável. Neste cenário, a chave sem significado torna-se a opção selecionada. Conclui-se que não existe uma solução ideal e a decisão deve-se tomar caso a caso.

A decisão entre chave simples ou composta fica, automaticamente, comprometida com a decisão tomada no ponto anterior, ou seja, caso se opte por uma chave sem significado esta é por natureza simples. No cenário, em que se decide por uma chave com significado esta pode assumir um âmbito simples ou composto consoante a situação.

Relativamente à utilização de índices efetua-se uma análise dos seus prós e contras. Por um lado, consideram-se os índices vitais para a eficiência no acesso aos dados, principalmente, em operações de leitura, onde a sua colaboração é indispensável para a localização de um registo. Por outro lado, a sua criação e manutenção, tem um custo associado que não deve ser negligenciado, nomeadamente, em operações de escrita que exigem a atualização dos índices associados.

Antes mesmo de se conhecer o histórico de utilização de uma tabela definem-se índices através da adoção dos seguintes critérios: (i) um índice por chave estrangeira (invocados frequentemente em operações de junção), (ii) índices em campos acedidos por range (e.g. "BETWEEN", "<" e ">") ou utilizados em cláusulas específicas (e.g. "WHERE", "ORDER BY", "GROUP BY" e "DISTINCT") e (iii) índices altamente seletivos, isto é, que apresentem muitos valores distintos e pretere-se a criação de índices em campos com poucos valores distintos (e.g. campos do tipo *bit*). Excluem-se da indexação as colunas: (i) raramente referenciadas em *queries*, (ii) com alta cardinalidade ou (iii) de certos tipos (e.g. "Ntext", "Text", "Image").

Após o conhecimento da utilização da base de dados pondera-se: (i) a criação, (ii) atualização ou (iii) remoção de índices. Recomenda-se, no auxílio à tomada de decisão, o recurso às



Tal como na apresentação do diagrama conceptual de classes, também na apresentação do modelo de dados opta-se pela divisão do modelo nas mesmas temáticas (máquina, comunicação e plano de rota) e transporta-se, igualmente, para os casos de uso os respetivos modelos de dados auxiliares à sua compreensão.

### Máquina

Destaca-se, na Figura 40, a tabela máquina possuidora de uma chave primária simples com significado e utilizada como chave estrangeira nas relações de um para vários (e.g. moedeiros, seleções, canais, localizações da máquinas, históricos de programação). Verifica-se que a máquina se associa: (i) ao cliente através de uma chave composta com significado (código e número de estabelecimento), (ii) à filial, à operação, ao modelo e ao módulo por via de uma chave simples sem significado (“id”) e (iii) ao armazém por uma chave simples com significado.

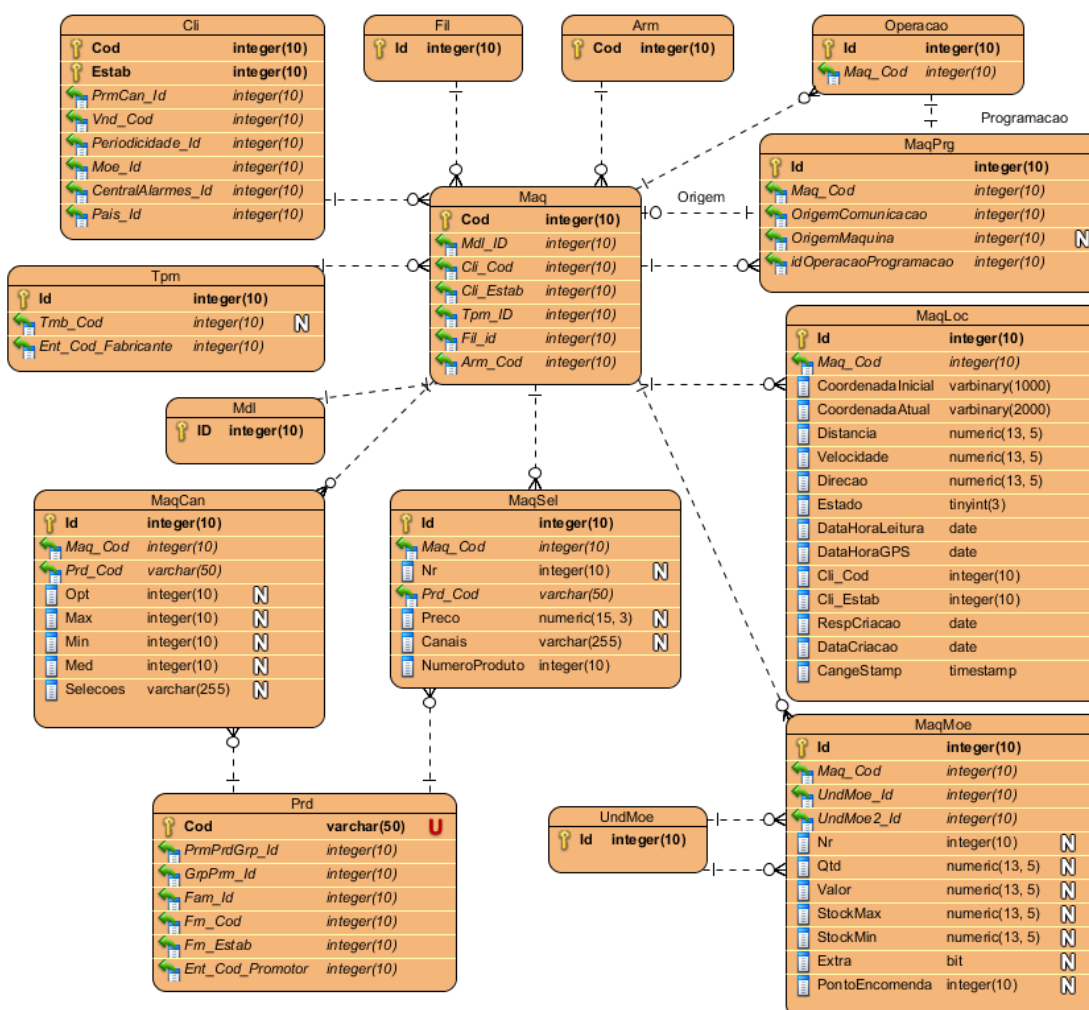


Figura 40 – Modelo de dados da visão da máquina

## Comunicação

Observa-se, na Figura 41, a tabela de cabeçalho de comunicação associada a um conjunto de tabelas satélites (e.g. linhas de comunicação, moedeiros, incidências) através de uma chave primária simples sem significado. Embora exista a associação entre a máquina e a comunicação, verifica-se uma relação aparentemente redundante entre o cliente e a comunicação, mas, tal existe, para garantir a rastreabilidade das comunicações por meio do conhecimento das comunicações realizadas em clientes anteriores.

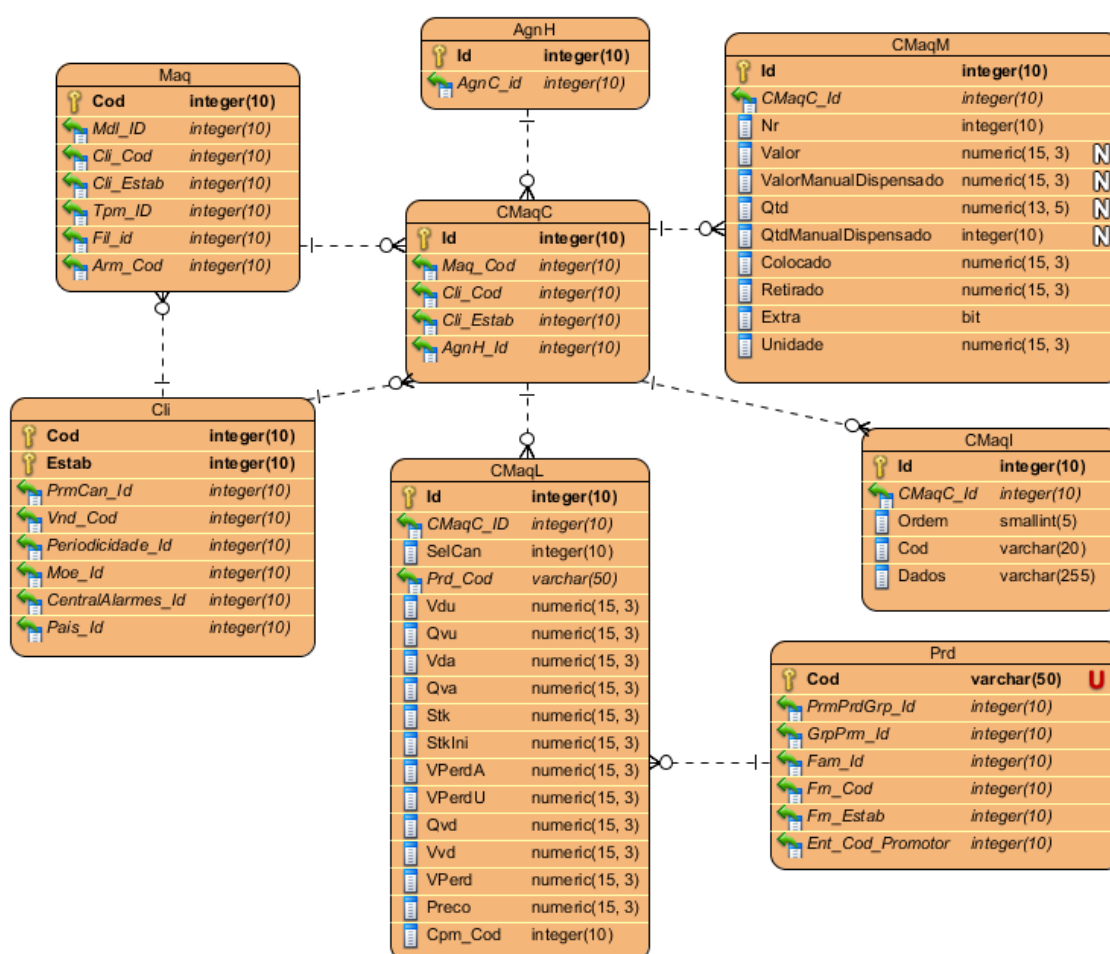


Figura 41 – Modelo de dados da visão da comunicação

### Plano de rota

Representa-se, na Figura 42, a tabela do cabeçalho do plano de rota (conhecida por “PRotC”) que especifica a rota de um vendedor num dado momento. Associa-se a esta tabela as respetivas linhas do plano de rota (denominada por “PRotL”) que concretizam as visitas efetuadas aos clientes e às máquinas.

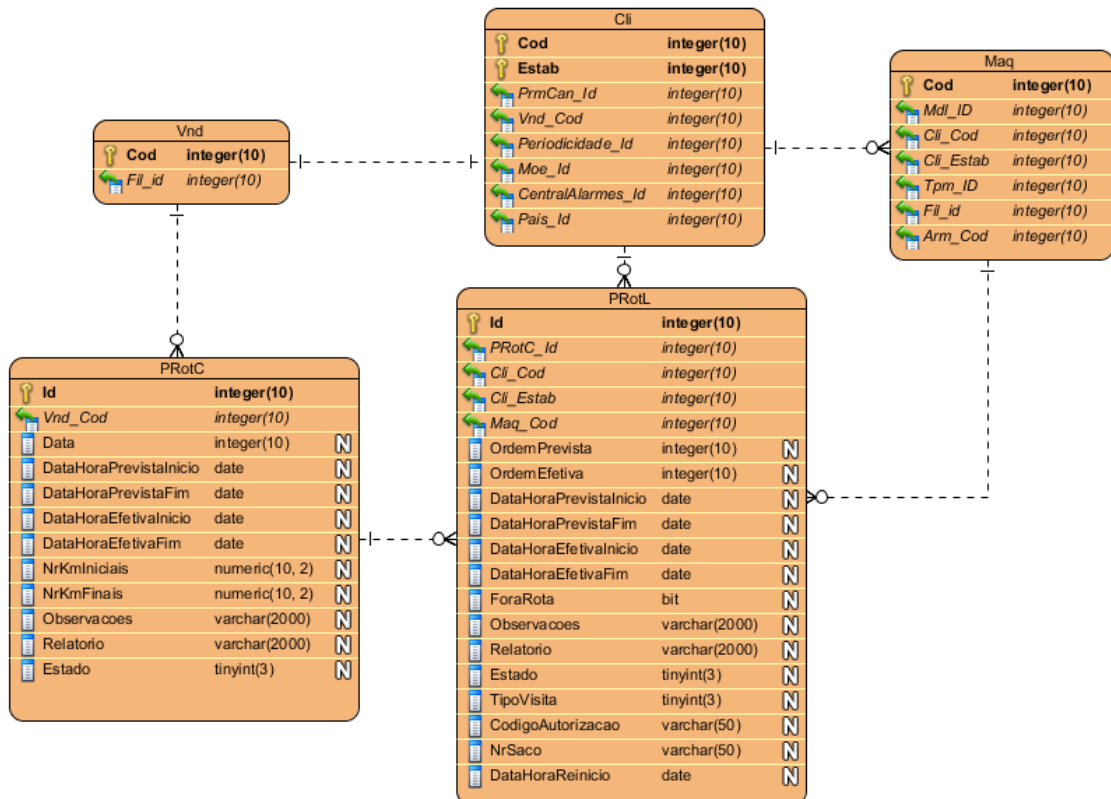


Figura 42 – Modelo de dados da visão do plano de rota

### 6.5.3 WinGuard

Destacam-se, na Figura 43, as tabelas do modelo de dados do WinGuard: (i) “Proprietario”, (ii) “Ocorrencia” e (iii) “TipoOcorrencia”. Confirma-se que um proprietário pode ter várias ocorrências e que uma ocorrência tem um determinado tipo.

Uma vez que as tabelas “Proprietario” e “Ocorrencia” permitem a criação e alteração de registos, adicionam-se campos para o controlo das alterações: (i) data de criação, (ii) data de alteração e (iii) <sup>114</sup>changestamp.

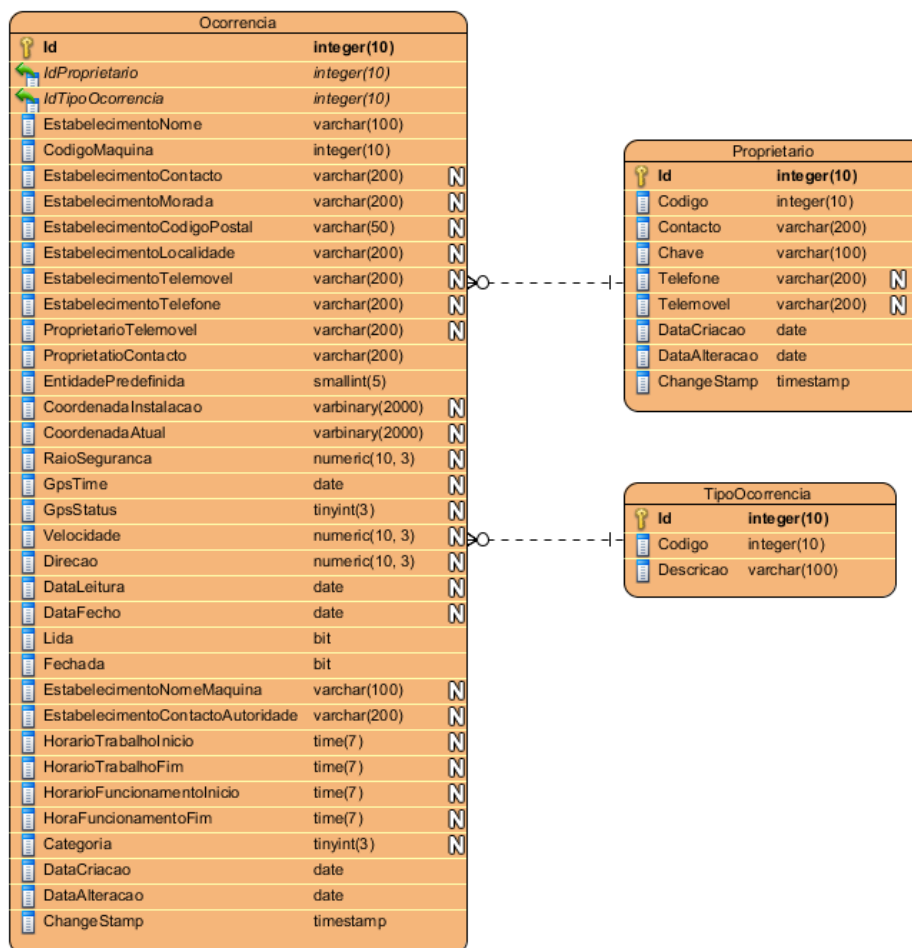


Figura 43 – Modelo de dados WinGuard

<sup>114</sup> O campo “changestamp” permite a confirmação da alteração de um registo, numa determinada tabela, por via da geração automática de um selo, mesmo que seja efetuada, diretamente, na base de dados.

# 7 Implementação

Neste capítulo, começa-se por descrever e comparar as tecnologias utilizadas para a concretização do desenho da solução. Segue-se a exposição do modelo arquitetural adotado no desenvolvimento das API's e das aplicações de interação com o utilizador. Pelo carácter distribuído do sistema, destacam-se os métodos de integração entre os diferentes componentes que o compõem: Middleware, WinVending Mobile e ERP.

## 7.1 Tecnologias

A escolha da tecnologia certa potencia a poupança de tempo de desenvolvimento, que é um recurso valioso quando se aborda esta temática. Pondera-se a seleção de cada tecnologia, com base em determinados critérios específicos para a tecnologia em análise.

### 7.1.1 Serviços de Internet

A comunicação entre sistemas é uma necessidade presente, nesta solução, e opta-se pela exposição de serviços, através da Internet, para atingir esse fim. Entre as abordagens de implementação de serviços de Internet, o protocolo SOAP e o modelo arquitetural REST destacam-se das demais pela sua maturidade e popularidade.

O protocolo SOAP é um protocolo para troca de informações estruturadas entre sistemas, baseado na linguagem XML como formato de mensagem. A construção de uma mensagem SOAP divide-se em três seções: (i) o envelope que envolve, descreve e informa como processar a mensagem, (ii) o cabeçalho que tem como função estender as funcionalidades das mensagens

(autenticação, transações, encriptação) e (iii) o corpo onde se encontra a informação propriamente dita. De modo a não comprometer a independência em relação às plataformas de desenvolvimento, o SOAP permite a utilização de uma variedade de protocolos, dos quais se destacam: HTTP, FTP, SMTP e BEEP. Para a comunicação entre o cliente e o servidor, um serviço de Internet SOAP utiliza um documento que descreve como aceder e utilizar os métodos que o constituem, designado WSDL (Web Services Description Language).

Consideram-se as seguintes vantagens do protocolo SOAP: (i) independência da linguagem, plataforma e camada de transporte; (ii) protocolo padronizado; (iii) especificações nativas para a gestão de estados, segurança e transações. Como desvantagens identificam-se: (i) a natureza prolixa e maior tempo de análise dos resultados, percebida pela utilização de XML e (ii) curva de aprendizagem elevada, de modo a apreender o padrão, a sua utilização e especificações.

REST (Representational State Transfer) é um estilo arquitetural, projetado para sistemas distribuídos, criado no ano de 2000 por Roy Thomas Fielding, que especifica um conjunto de princípios, tais como: não manter o estado de sessão, permitir cache, apresentar uma comunicação cliente-servidor e manter uma interface uniforme. Um serviço de Internet baseado em REST e respeitando os seus princípios arquiteturais é denominado por RESTful.

Reconhecem-se as seguintes vantagens do protocolo REST: (i) facilidade de aprendizagem; (ii) rapidez de execução; (iii) formatos curtos das mensagens e (iii) infraestrutura de cache assente no método GET do protocolo http. Verificam-se as seguintes desvantagens: (i) falta de padrões e (ii) limitação da utilização ao protocolo http.

Confrontam-se as vantagens e desvantagens de cada um dos protocolos, para se decidir qual deles se adequa melhor ao ambiente e aos requisitos dos sistemas a desenvolver. Desse confronto, seleciona-se a abordagem REST pelo seu carácter simples, rápido e com menor sobrecarga de parâmetros (e.g. cabeçalhos dos pedidos efetuados), adequado a plataformas leves, consideradas um dos alvos da API do WinVending (e.g. WinVending Mobile). Por outro lado, as limitações do protocolo escolhido (e.g. não mantém sessão, não gere transações e é limitado ao protocolo http) não interferem com a concretização dos requisitos do sistema.

### 7.1.2 Mapeamento relacional de objetos

“ORM (Object-Relational Mapping) é uma técnica de programação para converter dados entre sistemas de tipos incompatíveis (e.g. bases de dados relacionais, repositórios XML e outras fontes de dados) e as linguagens de programação orientadas a objetos.” (tradução nossa) (Object-relational mapping 2017). Esta técnica visa a criação de uma camada de mapeamento entre o modelo de objetos da aplicação e o modelo de dados específico, de modo a abstrair o acesso ao seu conteúdo.

No universo .NET, conhece-se o padrão <sup>115</sup>ADO.NET de acesso a dados relacionais, XML's e documentos Office (e.g. Excel, Word). Ao adotar-se uma solução baseada puramente nesta tecnologia, notam-se as seguintes limitações: (i) a necessidade de escrita manual de todo o código SQL, (ii) preocupação na utilização de código SQL comum a todos os SGBD's e (iii) restrições associadas a um modelo não orientado a objetos (e.g. não possui herança e polimorfismo).

Observam-se as seguintes vantagens na utilização de um ORM: (i) aumento de produtividade por via da redução de escrita de código, (ii) qualidade e elegância do código produzido, (iii) fácil manutenção e (iv) padronização no acesso aos dados da aplicação.

Para facilitar a aplicação desta técnica, ao longo do tempo, surgem diversas *frameworks* ORM, das quais se destacam para análise nesta seção as seguintes: NHibernate 4, Linq to SQL e Entity Framework 6. Selecionam-se estas três *frameworks* devido ao cumprimento dos seguintes requisitos, considerados mínimos: suporte para o SGBD SQL Server, disponível para as linguagens de programação .NET e inexistência de custos de aquisição e manutenção.

NHibernate é um ORM para a plataforma de desenvolvimento Microsoft .NET, de cariz livre e de código aberto, baseado no ORM para Java, denominado Hibernate. Para equipas com experiência de utilização em Hibernate, a transposição para o NHibernate realiza-se pacificamente com o auxílio da documentação existente. No entanto, o esforço de

---

<sup>115</sup> “ADO.NET é uma tecnologia de acesso a dados do Microsoft .NET Framework que fornece comunicação entre sistemas relacionais e não relacionais através de um conjunto comum de componentes.” (tradução nossa) (ADO.NET 2017)

aprendizagem para equipes sem experiência é considerável. Consulta-se, em novembro de 2013, a atividade do projeto NHibernate (<https://www.openhub.net/p/nhibernate>) com os resultados: (i) 12 submissões com tendência decrescente, (ii) 4 contribuidores com tendência crescente, (iii) 525961 linhas de código pouco comentadas. Suporta os próximos SGBD's: (i) MS SQL Server (inclui versão compacta), (ii) Oracle, (iii) MS Access, (iv) Firebird, (v) DB2 UDB, (vi) MySQL, (vii) SQLite. Implementa: (i) a gestão de transações, (ii) o mapeamento de código em primeiro e (iii) a configuração baseada em código. Contempla o conceito de migração apenas na geração do esquema inicial.

“Linq to SQL é um componente da .NET Framework da versão 3.5 que fornece uma estrutura, em tempo de execução, de gestão de dados relacionais como objetos.” (tradução nossa) (Hubbard et al. 2017). Para equipes com experiência em LINQ, a adaptação para SQL é intuitiva e rápida. Este componente, contempla: (i) a gestão de transações, (ii) a configuração baseada em código e (iii) como técnica de mapeamento, unicamente, do modelo de dados relacional (exclusivamente o SGBD MS SQL Server e a sua versão compacta) para um modelo de objetos. O conceito de migrações não tem qualquer implementação.

Entity Framework é um ORM para a plataforma de desenvolvimento Microsoft .NET desenvolvido por uma equipe composta pela Microsoft Open Tech Hub<sup>116</sup>, em colaboração com a comunidade de programadores de código aberto. Desenhado, de raiz, para integrar a Framework .NET, apresenta uma estrutura e sintaxe que facilitam a sua aprendizagem. Observa-se, em novembro de 2013, a atividade do projeto Entity Framework (<https://www.openhub.net/p/entityframework>) com próximos os valores: (i) 87 submissões com tendência crescente, (ii) 12 contribuidores com tendência crescente, (iii) 800448 linhas de código moderadamente comentadas. Suporta os SGBD's: MS SQL Server (inclui a versão compacta) e Firebird. Implementa: (i) a gestão de transações, (ii) o mapeamento de código em primeiro, (iii) a configuração baseada em código e (iv) o conceito de migração na sua plenitude.

Na Tabela 28, comparam-se os três ORM's em análise, segundo um conjunto de características a seguir enumeradas: (i) curva de aprendizagem, (ii) atividade do projeto (número de novas

---

<sup>116</sup> “Subsidiária integral da Microsoft conhecida como Microsoft Open Technologies, Inc. O objetivo declarado da nova empresa é promover o investimento da empresa na abertura (inclui interoperabilidade, padrões e código abertos).” (Walker 2012)

versões e periodicidade), (iii) SGBD's suportados, (iv) capacidade de executar transações, (v) mapeamento de código em primeiro, (vi) configuração baseada em código e (vii) migrações.

Tabela 28 – Comparação entre ORM's (dados de novembro de 2013)

Característica	NHibernate 4	Linq to SQL	Entity Framework 6
<b>Curva de aprendizagem</b>	Alta	Baixa	Média
<b>Atividade do projeto (novas versões e periodicidade)</b>	Média (tendência a diminuir)	Baixa (depende do fabricante)	Média (tendência a aumentar)
<b>SGBD's suportados</b>	MS SQL Server (inclui versão compacta), Oracle, MS Access, Firebird, PostgreSQL, DB2 UDB, MySQL, SQLite	MS SQL Server	MS SQL Server (inclui a versão compacta), Firebird
<b>Capacidade de executar transações</b>	Sim	Sim	Sim
<b>Mapeamento de código em primeiro</b>	Sim	Não	Sim
<b>Configuração baseada em código</b>	Sim	Sim	Sim
<b>Migrações</b>	Parcial	Não	Total

Dos três ORM's em análise, começa-se por excluir o Linq to SQL por não implementar os conceitos de: (i) mapeamento de código em primeiro e (ii) migrações. De seguida, exclui-se o NHibernate por apresentar uma curva de aprendizagem alta e implementar, apenas, de um modo parcial, a característica de migrações. Elege-se, por exclusão de partes, o ORM Entity Framework, que mostra uma curva de aprendizagem moderada e a implementação integral das características valorizadas nesta análise. Contudo, mantém-se a expectativa do ORM eleito suportar um leque mais alargado de SGBD's, num futuro próximo, o que, devido ao crescimento previsto do projeto, é uma possibilidade.

### 7.1.3 Ferramentas de registo<sup>117</sup>

O registo (por vezes, também referido como rastreamento <sup>118</sup>) é utilizado para manter informações sobre a execução de uma aplicação para fins de depuração e teste. Os programadores, os responsáveis por testes e as equipas de suporte recorrem frequentemente a técnicas de registo e rastreamento para: (i) identificar falhas de *software*, (ii) depuração após a instalação do sistema, (iii) monitorização e (iv) auditoria dos sistemas em ambiente de produção (Gurock Software GmbH 2017).

Normalmente, o registo envolve a escrita de mensagens de texto, em diferentes fontes de dados ou, o envio de dados para aplicações de monitorização. A manutenção das mensagens de registos pode ser efetuada com os recursos concedidos pela plataforma .NET (e.g. manipulação de ficheiros), ou por meio de ferramentas dedicadas a essa finalidade. Pela complexidade envolvida nas tarefas de manutenção dos registos (e.g. estrutura de dados complexas, pilha de chamadas, execução assíncrona, diferentes níveis de registos, rotação de ficheiros, diferentes fontes de dados), opta-se pela utilização de uma ferramenta de registo.

Define-se os próximos requisitos como mínimos para uma ferramenta de registo: (i) manipulação de diferentes fontes de dados (ficheiros, base de dados e eventos do sistema operativo), (ii) execução assíncrona de registos, (iii) capacidade de extensão e customização, (iv) flexibilidade de configuração por código e ficheiro, (v) sem custos de aquisição ou manutenção e (vi) suporte na plataforma .NET NuGet. Enquadram-se, nesse perfil, as seguintes ferramentas de registo: NLog, Apache Log4net e Enterprise Library.

NLog é uma ferramenta gratuita de registo da plataforma .NET, distribuída sob a licença BSD. Avalia-se, em novembro de 2013, a atividade do projeto (<https://www.openhub.net/p/nlog>) com os seguintes valores: cinco submissões com tendência a diminuir, um contribuidor com tendência a diminuir e código bem documentado. Observa-se, na Figura 44, um excelente desempenho no registo de eventos no sistema.

---

<sup>117</sup> Registo mais conhecida pelo estrangeirismo *log*.

<sup>118</sup> Rastreamento mais conhecida pelo estrangeirismo *tracing*.

A biblioteca Apache Log4net é um projeto livre de código aberto baseado no projeto log4j<sup>119</sup> para Java, iniciado por uma equipa da Neoworks Limited<sup>120</sup>, em junho de 2001 e acrescido de contribuições realizadas pela comunidade (Apache Software Foundation 2017). Observa-se, em novembro de 2013, a atividade do projeto (<https://www.openhub.net/p/log4net>) com os seguintes valores: 43 submissões com tendência a aumentar, dois contribuidores com tendência a manter e código bem documentado. Note-se, na Figura 44, um desempenho no registo de eventos no sistema que, comparativamente com as outras ferramentas em análise, é notoriamente inferior.

A Microsoft Enterprise Library é uma coleção de componentes de software reutilizados, também conhecidos por blocos de aplicação, projetados para auxiliar os programadores de *software* com desafios comuns de desenvolvimentos empresarial (e.g. registo e rastreamento) (Enterprise Library 2013). O facto de ser desenvolvido, na sua íntegra, pela Microsoft, evita a observação da atividade do projeto e levanta uma incógnita relativamente às futuras intervenções a implementar no projeto. Consta-se, pela Figura 44, um desempenho no registo de eventos, muito próximo do resultado excelente da ferramenta NLog.

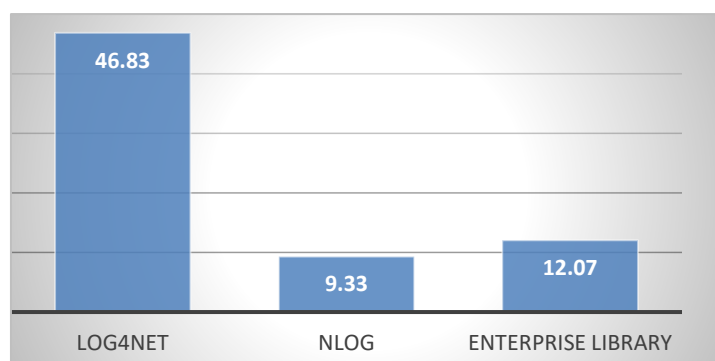


Figura 44 – Comparação do desempenho das ferramentas de *log* (segundos / 100.000 eventos de registo) (Marsh 2016)

Na Tabela 29, comparam-se as ferramentas de registo, segundo as características a avaliar para a seleção de uma delas: (i) filtro de mensagens disponíveis, (ii) rotação máxima de partes de

<sup>119</sup> “O Log4j é um software de registo livre de código aberto desenvolvido pela Apache Software Foundation para aplicações escritas na linguagem de programação Java.” (Log4j 2017)

<sup>120</sup> Neoworks, uma empresa Salmon, integra e suporta sistema de comércio eletrónico (Salmon 2017).

ficheiros (e.g. geração de ficheiros de registo por cada dia), (iii) atividade de projeto, (iv) desempenho e (v) facilidade de configuração.

Tabela 29 – Comparação entre ferramentas de registo (Gurock Software GmbH 2017)

Característica	NLog	Apache Log4net	Enterprise library
Licença	BSD	Apache	MS-PL
Fabricante	Não se aplica	Apache	Microsoft
Filtro de mensagens	Debug / Trace / Info / Warn / Error Fatal	Debug / Info / Warn / Error / Fatal	Verbose / Information / Warning / Error / Critical
Rotação máxima de partes de ficheiros	Sim	Sim	Não
Atividade do projeto	Média (tendência a diminuir)	Média (tendência a aumentar)	Desconhecida (depende do fabricante)
Desempenho	Excelente	Médio	Bom
Facilidade de configuração	Médio	Fácil	Difícil

Da análise das três ferramentas de registo, exclui-se a Enterprise Library pela dificuldade de configuração, por não implementar o conceito de rotação máxima de partes de ficheiros e pelo desconhecimento da atividade do projeto. Embora o excelente desempenho da ferramenta NLog, considera-se a aposta na ferramenta Apache Log4Net mais segura, pelo facto de outros projetos de código aberto também a dotarem, a que se acresce as já referidas características de: (i) facilidade de configuração e (ii) atividade do projeto com tendência a aumentar.

#### 7.1.4 Filas de Mensagens

“Na construção de aplicações de maior escala, geralmente, opta-se pela adoção de filas de mensagens, tanto para a distribuição, como para a agregação de carga de trabalho. Considera-se a distribuição de carga de trabalho uma abordagem intrínseca às filas de mensagens, frequentemente alcançada através de vários leitores à mesma fila de mensagens. A agregação, geralmente, implementa um leitor e vários escritores e utiliza-se para agrupar muitas pequenas

atualizações, num grande bloco. Tal facilita o recurso a técnicas de comunicação com as bases de dados, como por exemplo, a inserção em massa, ao invés de múltiplas inserções individuais” (tradução nossa) (Allen 2011).

Observam-se, no ecossistema .NET, uma panóplia de ofertas de filas de mensagens, das quais se destacam, pelo seu cariz popular e gratuito, as seguintes tecnologias: MSMQ, ZeroMq e Apache ActiveMQ.

Considera-se a opção MSMQ, como uma tecnologia madura para sistemas que apenas utilizam produtos Microsoft. Notam-se como principais vantagens: (i) a simplicidade da API, (ii) a qualidade da documentação concedida pelo MSDN, (iii) a não necessidade de um servidor ou bibliotecas adicionais (suficiente o Windows com a .NET Framework) e (iv) a disponibilização de características avançadas (e.g. gestão de transações distribuídas). Registam-se como desvantagens: (i) a disponibilidade apenas para sistemas Microsoft, (ii) limite de 4MB para o tamanho da mensagem, (iii) limite de 2GB para o tamanho da mensagem de uma fila e (iv) o reduzido suporte a padrões de integração empresarial.

O ZeroMQ é um sistema de mensagens muito leve, especialmente concebido para cenários de alta rendimento e baixa latência (e.g. universo financeiro). O “zero” no nome desta tecnologia refere-se à cultura minimalista presente no projeto: (i) zero intermediários, (ii) zero latência, (iii) zero custos e (iv) zero administração. Distinguem-se as seguintes vantagens: (i) o número de linguagens suportadas (mais de 30 das quais C, C++, Java, .NET e Python), (ii) o número de sistemas operativos compatíveis (e.g. Linux, Windows, OS X), (iii) a leveza, (iv) o alto desempenho, (v) a implementação de vários padrões de mensagens (e.g. publicador / subscritor e cliente / servidor) e (vi) a larga e ativa comunidade. Observa-se, em novembro de 2013, a atividade do projeto (<https://www.openhub.net/p/zeromq>) com os seguintes valores: 86 submissões com tendência a diminuir, 13 contribuidores com tendência a aumentar e código moderadamente documentado. Como desvantagens referem-se: (i) não garantia de entrega das mensagens, (ii) não implementação de algumas funcionalidades avançadas (e.g. transações distribuídas) e (iii) não permite a descoberta de componentes pela inexistência de mediador.

O Apache ActiveMQ é uma plataforma de gestão de mensagens de código aberto que implementa na íntegra o standard Java Message Service<sup>121</sup>. Em .NET, concede-se o acesso ao ActiveMQ pela Apache .NET Messaging API (NMS). Destacam-se, neste sistema, o fornecimento de recursos empresariais como: (i) tolerância a falha, (ii) alta disponibilidade, (iii) segurança (e.g. canais de comunicação seguros, autenticação por certificados, autorização por níveis de mensagem), (iv) monitorização (e.g. ActiveMQ Web Console), (v) transações, (vi) *clustering*<sup>122</sup>, (vii) múltiplos protocolos (e.g. ActiveMQ, MSMQ, STOMP) e (viii) múltiplas linguagens de suporte (e.g. Java, C++, .NET) (Dejan 2010). Observa-se, em novembro de 2013, a atividade do projeto (<https://www.openhub.net/p/activemq>) com os seguintes valores: 87 submissões com tendência a diminuir, sete contribuidores com tendência a diminuir e código moderadamente documentado. Conclui-se as próximas vantagens desta plataforma: (i) escalável, (ii) confiável, (iii) vocacionada para o universo empresarial e (iv) com uma comunidade dinâmica. Como desvantagens verificam-se: (i) arquitetura complexa para o desenho e implementação e (ii) necessidade elevada de configuração.

Da comparação dos três sistemas, a escolha recai no MSMQ devida à facilidade de instalação e configuração, à baixa curva de aprendizagem e ao suporte de transações distribuídas. Embora o excelente desempenho, o ZeroMQ não implementa o conceito de transações distribuídas e a garantia de entrega de mensagens, motivo pelo qual se exclui este sistema. Considera-se equilibrado o sistema ActiveMQ, no que diz respeito às características avançadas necessárias e ao desempenho. No entanto, a sua complexidade e necessidade de configuração elevada motiva a sua exclusão.

---

<sup>121</sup> “Java Message Service, ou JMS, é uma API da linguagem Java para middleware orientado para mensagens. Através da API JMS duas ou mais aplicações podem se comunicar por mensagens.” (JMS 2014)

<sup>122</sup> Pode definir-se *clustering* como: “um processo de organizar objetos em grupos cujos membros são semelhantes de alguma forma.” (tradução nossa) (Saxena & Amutha 2017)

### 7.1.5 Ferramentas de Integração Empresarial (EAI<sup>123</sup>)

A complexa e diversificada troca de mensagens entre diferentes sistemas (e.g. Middleware, WinVending, WinGuard, ERP), através de serviços, motiva a ponderação da adoção de uma ferramenta de integração, como meio de automatização dos serviços envolvidos.

Com a consciência que as ferramentas EAI aceleram e simplificam a integração entre sistemas, analisam-se, nesta secção, as ferramentas disponíveis para o ambiente de desenvolvimento .NET: (i) Microsoft BizTalk Server e (ii) MuleSoft.

Criado pela Microsoft, o Microsoft BizTalk Server funciona como “um sistema *middleware* que permite às empresas automatizar processos de negócios, através do uso de adaptadores que comunicam com diferentes sistemas de *software*.” (tradução nossa) (Microsoft BizTalk Server 2017)

Conhece-se o MuleSoft como: (i) um produto de código aberto desenvolvido em Java, (ii) um ESB leve e (iii) uma plataforma de integração. Esta plataforma possibilita aos programadores: (i) a conexão a aplicações de um modo rápido e fácil e (ii) a integração de sistemas existentes, independentemente das tecnologias utilizadas (e.g. JMS, serviços da Web, http).

Na Tabela 30, compara-se um conjunto de características entre as plataformas em análise e, para cada par (característica - plataforma), atribui-se um indicador que assume um dos seguintes três estados: (i) equiparadas (“=”), (ii) vantajosa (“↑”) e (iii) desvantajosa (“↓”).

---

<sup>123</sup> EAI do termo inglês Enterprise Application Integration considera-se “o uso de princípios arquitetónicos de *software* e sistemas informáticos para integrar um conjunto de aplicações empresarias” (tradução nossa) (Enterprise application integration 2017).

Tabela 30 – Comparação de características entre plataformas de integração (Aqeel 2015)

Característica	BizTalk Server	MuleSoft
Extensibilidade	=	=
SaaS e desenvolvimento híbrido	↓	↑
REST, JSON, RAML & gestão da API	↓	↑
Interoperabilidade entre .NET e componentes Java	=	=
Motor de regras	↑	↓
Transações de execução longa	↑	↓
EDI <sup>124</sup> e EDIFACT <sup>125</sup>	↑	↓
ESB <sup>126</sup>	↑	↓
Ferramentas de desenvolvimento e administração	↑	↓
Preço	=	=
Suporte	=	=
Inovação	↓	↑

As plataformas apresentam resultados equiparados para as características: (i) extensibilidade onde ambas mostram capacidades de extensão e personalização, (ii) interoperabilidade entre .NET e componentes Java presente em ambos, (iii) preço, cuja avaliação depende do uso e do tempo de utilização da plataforma e (iv) suporte disponível em múltiplos níveis em ambos os fabricantes. Considera-se o MuleSoft vantajoso nas seguintes características: (i) SaaS e desenvolvimento híbrido ao oferecer um desenvolvimento mais maduro para soluções na nuvem, (ii) gestão da API e (iii) inovação, ao mostrar-se ativo no trabalho de novas tecnologias

<sup>124</sup> EDI do termo inglês Electronic Data Interchange “significa troca estruturada de dados através de uma rede de dados qualquer” (EDI 2017).

<sup>125</sup> EDIFACT do termo inglês Electronic Data Interchange for Administration, Commerce and Transport considerado “um padrão internacional de EDI, para descrição textual de documentos com vista ao armazenamento e envio por meios eletrônicos.” (EDIFACT 2017)

<sup>126</sup> “ESB do termo inglês *Enterprise Service Bus* refere-se à arquitetura de construção de software implementado em tecnologias encontradas na categoria de produtos de infraestrutura de middleware.” (Enterprise Service Bus 2017)

(e.g. gestão da API e RAML<sup>127</sup>), em contraponto com o processo mais demorado do BizTalk na adição de funcionalidades e tecnologias. Verificam-se as próximas vantagens do BizTalk: (i) motor de regras próprio e compreensivo, (ii) implementação simples de transações de execução longa, face às necessidades de personalização do MuleSoft, (iii) melhor suporte para os formatos de EDI e EDIFACT, (iv) ESB com oferta de um melhor controlo e características (e.g. criação de itinerários) e (v) desenvolvimento e ferramentas de administração que, embora equiparadas, o BizTalk vence pela familiaridade das suas ferramentas (e.g. Visual Studio). (Aqeel 2015)

Da ponderação das vantagens e desvantagens de cada uma das plataformas, conclui-se que o BizTalk Server se adequa melhor ao cenário em estudo, caracterizado por: (i) implementar complexas regras de negócio em código existente .NET, (ii) conectar-se a sistemas empresariais (e.g. ERP), (iii) ter potencial de alto tráfego entre os sistemas e (iv) existência de uma maior familiaridade da equipa de desenvolvimento com as ferramentas Microsoft. (Aqeel 2015)

Embora com o conhecimento da importância da adoção de uma ferramenta de EAI, opta-se por não adotar qualquer uma, pelos seguintes motivos: (i) custo da plataforma e (ii) necessidade de aprendizagem de uma nova ferramenta. Tais fatores influenciam diretamente duas variáveis importantes na decisão: (i) custo e (ii) tempo.

## 7.2 Modelo Arquitetural

Apresenta-se, nesta secção, a arquitetura da solução Vodafone WinVending (c.f. Figura 45) baseada em duas API's REST (API WinVending e API WinGuard) e o consumo das mesmas por dois clientes *Windows Forms* (Backoffice WinVending e WinGuard).

As aplicações desenvolvidas, no âmbito deste trabalho, baseiam-se no modelo de três camadas que se considera uma arquitetura, mas, mais concretamente, denomina-se padrão. Isto porque não concede uma solução em concreto para um caso específico, mas sim um modelo que pode ser aplicado a vários casos (Sousa 2006). Enumeram-se as três camadas deste modelo: (i)

---

<sup>127</sup> RAML do termo inglês RESTful API Modeling Language conhece-se como uma linguagem de dados serializados, passível de leitura humana para descrever API's RESTful. (RAML 2017)

camada de apresentação, (ii) camada de regra de negócio e (iii) camada de serviço de dados. A camada de apresentação responsabiliza-se pela interação entre o utilizador, ou sistema externo e a aplicação. A camada de regras de negócio é o coração da aplicação, em termos de processamento e concentra as regras de todo o negócio. A camada de serviço de dados fornece serviços de persistência de dados.

Observa-se, na Figura 45, os componentes da arquitetura (API's REST e aplicações *Windows Forms*) e os fluxos de comunicação entre eles. Os utilizadores acedem às aplicações, através da camada de apresentação que interage com a camada de regras de negócio, onde se efetuam validações preliminares, antes de se estabelecer a comunicação com a respetiva API. Construídas sobre a tecnologia ASP.NET Web API<sup>128</sup>, as API's contêm os *controllers*, que invocam regras de negócio para um posterior acesso a dados de uma determinada base de dados. No caso particular da API WinVending, possibilita-se o envio de dados para a API WinGuard, através de um *gateway*.

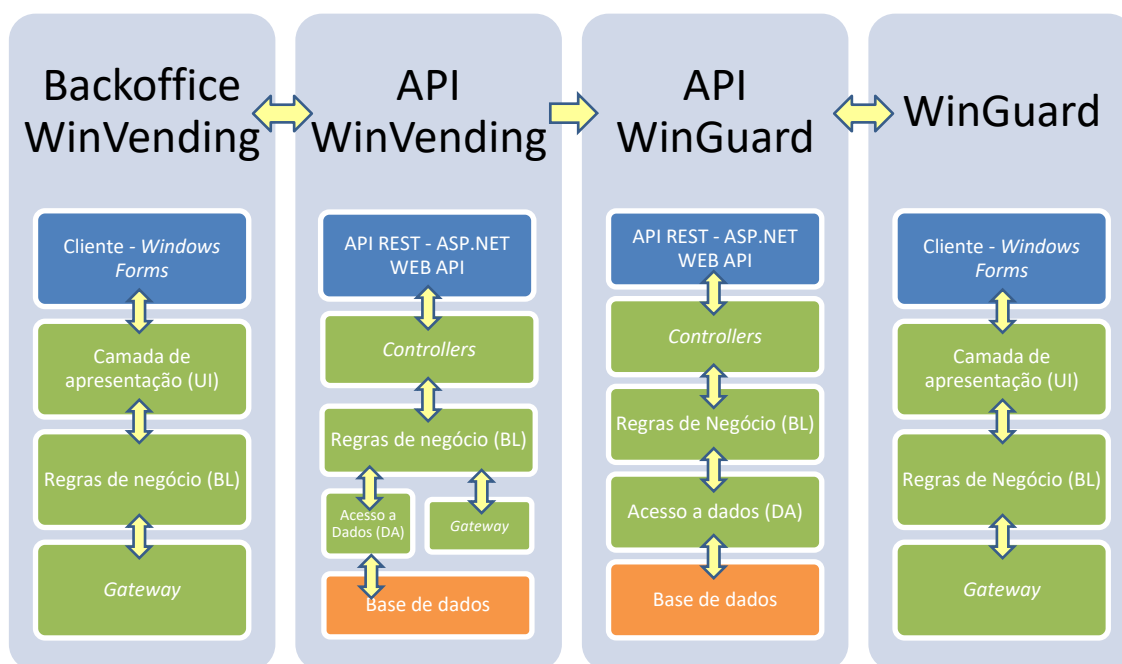


Figura 45 – Arquitetura da Solução Vodafone WinVending

<sup>128</sup> A ASP.NET Web API é uma estrutura para a construção de API's sobre a .NET Framework (tradução nossa) (Wasson 2015).

Nas próximas secções, detalha-se cada uma das camadas e os padrões e técnicas de comunicação utilizadas.

### 7.2.1 Camada de Apresentação

A camada de apresentação contém os componentes necessários de interação entre as aplicações e os utilizadores. No caso em estudo das aplicações WinVending e WinGuard, utiliza-se a tecnologia *Windows Forms*<sup>129</sup> sobre a *Framework .NET*, complementada por controlos de interface que incorporam o componente *Infragistics NetAdvantage 2005*, para expor os dados ao utilizador e obter e validar as entradas por ele introduzidas.

#### *Organização de pacotes*

Constitui-se a camada de apresentação por um único pacote (c.f. Figura 46), conhecido pelo acrónimo UI, que contém os componentes gráficos da aplicação e comunica com a camada de regras de negócio (BL), por via de objetos de negócio (BO), com o intuito de: (i) obter, (ii) validar e (iii) persistir os dados, desconhecendo todos os pormenores de implementação.

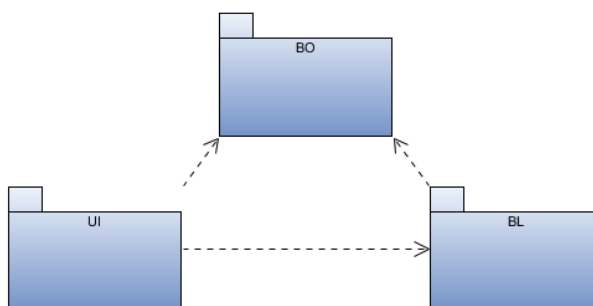


Figura 46 – Diagrama de pacotes UI

#### *Organização de classes*

Na Figura 47, observa-se a organização do pacote UI, em termos de classes. Constatase que todas as janelas da aplicação herdam direta ou indiretamente da classe "FrmComum". A

---

<sup>129</sup> "*Windows Forms* é uma biblioteca de classes gráfica (GUI) incluída como parte integrante da Microsoft .NET Framework" (tradução nossa) (*Windows Forms 2017*).

utilização da técnica de herança, nesta situação, permite que qualquer alteração comum a todas as janelas da aplicação, se centre num único ponto (e.g. alteração do ícone da aplicação, situado no canto superior esquerdo, de todas as janelas). Verifica-se, também, a inclusão de uma classe comum dedicada, exclusivamente, às janelas de manutenção<sup>130</sup> de registos (“FrmComumManutencao”). Deteta-se a presença da janela “FrmPrincipal”, considerada o contendor de janelas, ou seja, a partir dela invoca-se a generalidade das janelas da aplicação, onde ficam referenciadas. Tal referenciação permite a navegação e disponibilização de vistas simultâneas das janelas nela contidas (em cascata ou lado a lado). A classe “VarSessao”, acessível por todas as classes do pacote, contém as variáveis de sessão da aplicação e as propriedades necessárias para a respetiva atribuição e obtenção (e.g. utilizador, empresa, clientes, produtos). Determinadas variáveis de sessão, carregam-se para memória, segundo o padrão *lazy load*, ou seja, obtém-se os dados para memória, somente quando necessários (Patterns Of Enterprise Application Architecture, 2005).

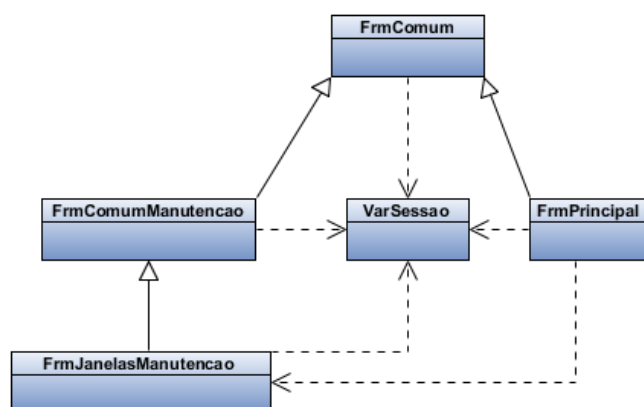


Figura 47 – Diagrama de classes UI

A janela “FrmComumManutencao” permite a criação de janelas de manutenção que implementam: (i) uma imagem ajustada ao registo a manter (c.f. ponto 1 da Figura 48), (ii) um conjunto de funcionalidades usuais na gestão de registos (c.f. ponto 2 da Figura 48) e (iii) a possibilidade da adição de novas funcionalidades (c.f. ponto 3 da Figura 48).

---

<sup>130</sup> Janelas de manutenção dedicam-se a tarefas de criação, alteração, remoção e visualização de um determinado registo (e.g. máquina).

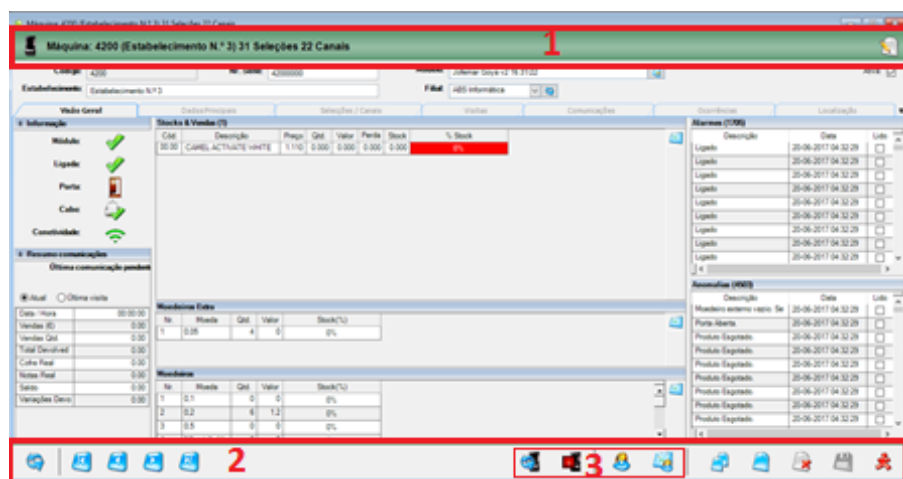


Figura 48 – Janela de manutenção de máquina

Consoante as necessidades específicas de cada aplicação, pondera-se, para cada caso particular, a criação de uma classe comum para concentrar em si todas as partes iguais (interface e funcionalidades). Por exemplo, na aplicação WinVending, procedeu-se à criação de uma classe “frmComumAssistente” dedicada à gestão de diferentes janelas de assistente (e.g. comunicação e programação), cujas partes comuns distinguem-se pela organização de diferentes etapas navegáveis que culminam na execução de uma tarefa final.

### *Databinding*

De modo a facilitar e agilizar a troca de dados entre as camadas de apresentação e regras de negócio, por via dos objetos de negócio, adota-se a técnica de vinculação de dados, mais conhecida pelo termo inglês *databinding*, que consiste na ligação da propriedade de um controlo (e.g. “text” numa caixa de texto) a uma propriedade de um objeto (e.g. “Codigo” do objeto “Maquina”). Desta forma, uma alteração na caixa de texto repercute-se na variável do objeto. No próximo trecho de código, apresenta-se uma implementação desta técnica em três controlos distintos: (i) uma caixa de texto (“txtCodigo”), (ii) uma opção de seleção (“optExploracao”) e (iii) uma caixa de verificação (“chkTemDiscriminador”).

```
Private Sub AtribuirBinds(ByVal Maquina As Estrutura.Maquina)
    ...
    Me.txtCodigo.DataBindings.Clear()
    Me.txtCodigo.DataBindings.Add("text", Maquina, "Codigo", True,
    DataSourceUpdateMode.OnPropertyChanged)
    Me.optExploracao.DataBindings.Clear()
    Me.optExploracao.DataBindings.Add("Value", Maquina, "Exploracao", True,
    DataSourceUpdateMode.OnPropertyChanged)
    Me.chkTemDiscriminador.DataBindings.Clear()

```

```

Me.chkTemDiscriminador.DataBindings.Add("Checked",           Máquina,
"TemDiscriminador", True, DataSourceUpdateMode.OnPropertyChanged)
...
End Sub

```

A utilização da técnica de *databinding* torna o código mais legível, simples e limpo, ao mesmo tempo, agiliza o desenvolvimento e minimiza tarefas de manutenção de código, fruto de potenciais falhas no mapeamento dos campos.

## 7.2.2 Camada de Regras de Negócio

Considera-se a camada de regras de negócio o motor da aplicação, em termos de processamento. Estabelece as regras de negócio relacionadas ao problema e às entidades de negócio e concede os seus serviços à camada de apresentação ou ao *controller*.

### Organização de pacotes

Nas aplicações *Windows Forms*, a camada de regras de negócio é constituída por um único pacote, conhecido pelo acrónimo BL, referenciado pelo pacote da camada da apresentação (UI) e dependente dos pacotes: (i) *gateway* e (ii) pacote de objetos de negócio (c.f. Figura 49).

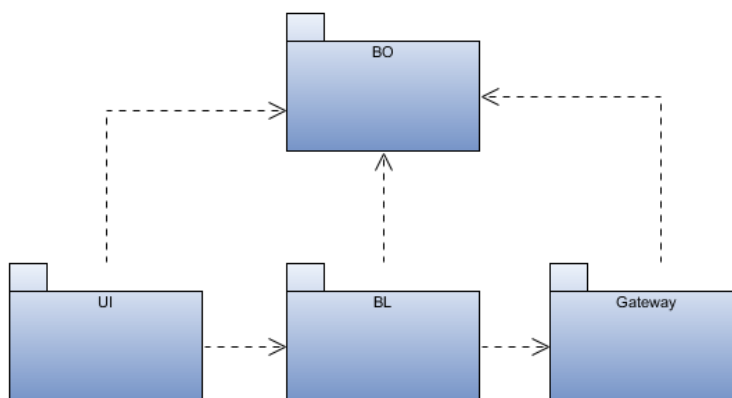


Figura 49 – Diagrama de pacotes BL - Aplicação

Nas API's, a camada de regra de negócios representa-se pelo pacote BL, referenciado pelo pacote *controller* e dependente de, pelo menos, um dos seguintes pacotes: (i) acesso a dados (DA) e (ii) *gateway* (c.f. Figura 50). Entre os diferentes pacotes trocam-se objetos de negócio.

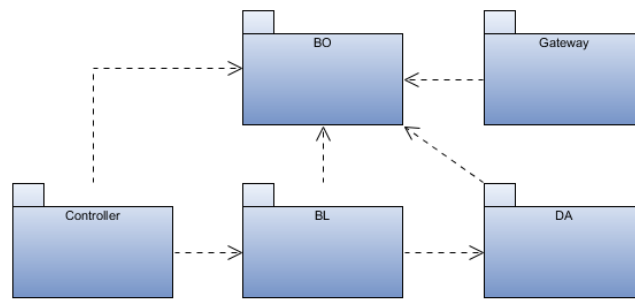


Figura 50 – Diagrama de pacotes BL - API

### Organização de classes

De acordo com a Figura 51, as classes sujeitas a operações de CRUD<sup>131</sup> “ClasseManutencaoBL” herdam da classe “BaseManutencao” que mantêm a informação da empresa e do utilizador, de modo a registar o responsável pela ação, numa determinada empresa. As classes desta camada relacionam-se com uma única classe de acesso a dados ou *gateway* e podem invocar várias classes da mesma camada que complementem a sua função.

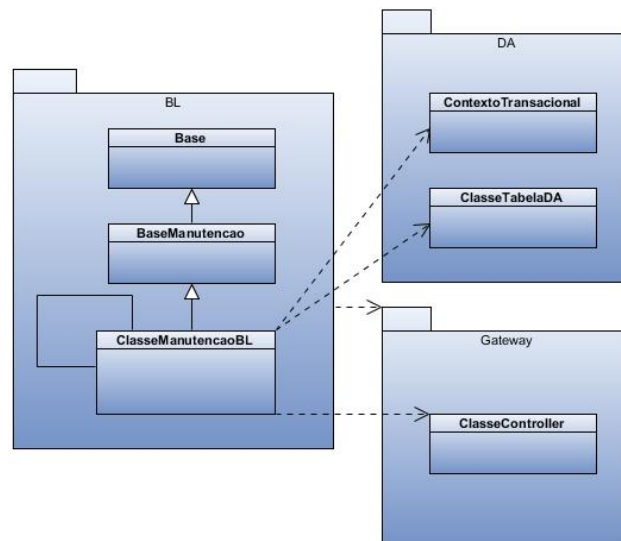


Figura 51 – Diagrama de classes BL

---

<sup>131</sup> CRUD do acrónimo inglês *Create, Read, Update e Delete* para as quatro operações básicas utilizadas em bases de dados relacionais ou em interface para utilizadores para criação, consulta, atualização e remoção de dados (CRUD 2017).

## Transações

A gestão de transações realiza-se na camada de regras de negócio e mantém-se a classe de “ContextoTransacional”, na camada de acesso a dados. Nesta abordagem, existe uma classe raiz da transação, situada na camada de regras de negócio, que se responsabiliza pela: (i) criação do contexto transacional, (ii) início da transação, (iii) conclusão e (iv) libertação dos recursos.

Segue-se o princípio de que cada classe da camada de regra de negócios acede, unicamente, a uma única classe da camada de acesso a dados (alto nível de coesão entre camadas). A classe raiz apenas invoca métodos da classe de acesso a dados, referente à mesma entidade de negócio e, quando necessita de invocar métodos de outras classes de acesso a dados, cria um objeto da classe de regras de negócio que comunica com a classe pretendida e envia-lhe, como parâmetro do seu construtor, o objeto do contexto transacional. Os métodos invocados retornam um valor indicativo do sucesso, ou falha de determinada operação. Mediante esse valor, decide-se se as próximas operações são executadas e, no final, decide-se qual a ação a efetuar: (i) *commit* ou (ii) *rollback* da transação, consoante o sucesso ou insucesso das operações.

Dado que, o recurso “transação” tem um elevado impacto no desempenho da base de dados, realça-se a importância de se iniciar a transação o mais tarde e terminar o mais cedo possível. Por esse motivo, qualquer regra de negócio, cuja execução se limite a operações em memória independentes da transação, deve ser executada antes do início da transação.

Apresenta-se, de seguida, um trecho de código exemplificativo da gestão de uma transação entre diferentes classes de regras de negócio.

```
try
{
    Object          contexto          =          contextoTransacional.Criar
(SuporteDA.Conexao.enumBD.BDApp, MaquinaDA.CodigoEmpresa);
    contextoTransacional.BeginTransaction();

    //Criar nova ocorrencia de alteracao do estado da porta
    BL.WinVending.IOcorrenciaBL  ocorrenciaBL  =  new  OcorrenciaBL(DA.
DaFabrica.NovaInstancia(contexto, MaquinaDA.CodigoEmpresa).Ocorrencia());

    ocorrenciaBL.CriarOcorrenciaAlterarEstadoPorta(this.Maquina, estadoPorta);

    //Alterar estado da porta
    DA.IMaquinaDa  maquinaContextoDA  =  DA.DaFabrica.NovaInstancia(contexto,
MaquinaDA.CodigoEmpresa).Maquina();
    maquinaContextoDA.AlterarEstadoPorta(this.Codigo, estadoPorta);
}
```

```

        contextoTransacional.Commit();
    }
    catch (Exception ex)
    {
        contextoTransacional.Rollback();
        throw new WinVending_API.Suporte.ExcecaoWinVending(ex);
    }
    finally
    {
        if (contextoTransacional != null)
        {
            contextoTransacional.Dispose();
        }
    }
}

```

Conclui-se que a abordagem adotada tem um baixo grau de acoplamento, na camada de acesso a dados e um grau, hipoteticamente, superior, na camada de regras de negócio, fruto da necessidade de gestão de múltiplas classes, justificada pela complexidade do problema. Relativamente à coesão, observa-se um alto grau entre as camadas ao limitar o acesso das classes de regras de negócio às classes da camada de acesso a dados, referentes à mesma entidade de negócio.

### **7.2.3 Camada de Serviço de Dados**

A camada de serviço de dados é a terceira camada do modelo de três camadas e responsabiliza-se pela persistência e consulta de dados armazenadas numa base de dados.

#### *Organização de pacotes*

Verifica-se, na Figura 52, que as classes do pacote de acesso a dados (DA) invocam-se a partir do pacote de regras de negócio (BL), cuja comunicação se concretiza por meio de objetos de negócios, referenciados por ambos os pacotes. O pacote de acesso a dados mapeia os objetos de negócio em objetos de transferência de dados (DTO.DB), para os persistir numa base de dados e vice-versa, para os consultar.

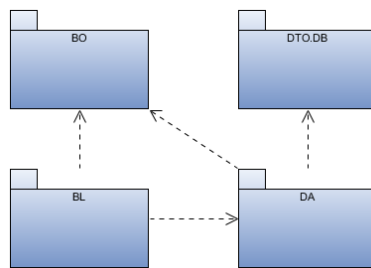


Figura 52 – Diagramas de pacotes DA

### Organização de classes

Visualiza-se, na Figura 53, que qualquer classe responsável pela comunicação, com uma tabela da base de dados (“TabelaDA”), herda da classe repositório (“Repositorio”) que gere o contexto de um determinado modelo (“Contexto”). Representa-se, em *Entity Framework*, o contexto como a abstração das leituras e escritas, numa base de dados. Para esse efeito, mapeiam-se em objetos de transferência de dados (“ClasseDTO.DB”) as tabelas de uma base de dados. A classe “Contexto” herda da classe “ContextoBase” responsável pela: (i) definição de valores predefinidos de configuração das bases de dados (e.g. definições de tempo de espera) e (ii) atribuição do texto de conexão à base de dados. Por sua vez, esta classe herda da classe “DBContext” parte da *Entity Framework* e intermediária entre as classes de entidade e as tabelas e vistas de uma base de dados.

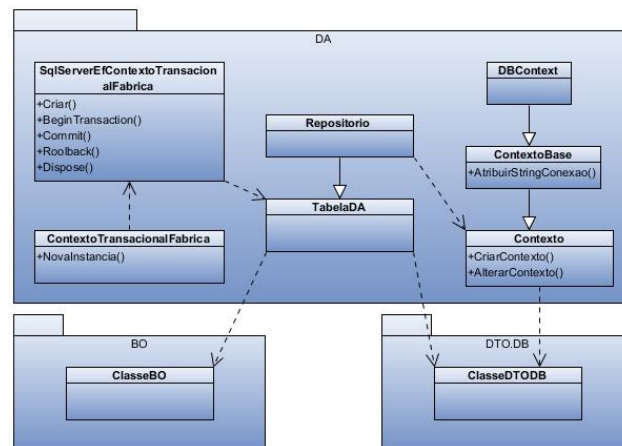


Figura 53 – Diagrama de classes DA

De modo a tornar a camada acesso a dados independente da tecnologia (e.g. *Entity Framework*), opta-se pela implementação do padrão *Factory*<sup>132</sup>, através de uma classe fábrica (“ContextoTransacionalFabrica”) que cria uma nova classe, mediante a tecnologia de acesso a dados utilizada (e.g. “SqlServerEFContextoTransacionalFabrica”), implementadora da interface “IContextoTransacional”.

### *Entity Framework*

A tecnologia *Entity Framework* simplifica o acesso a dados, através da escrita de código para a execução de operações de CRUD, sem necessidade de interagir diretamente com a base de dados (Kanjilal 2015). Enumeram-se três abordagens disponíveis para a modelação das entidades: modelo primeiro, base de dados primeiro e código primeiro. Na abordagem de modelo primeiro, disponibiliza-se um assistente visual para a criação do modelo, posteriormente, gere-se a base de dados. Considera-se a abordagem base de dados primeiro quando dela se dispõe e se pretende o ficheiro de modelo para, de seguida, se proceder à geração de classes, via modelos T4<sup>133</sup>. A abordagem código primeiro concede flexibilidade e controlo na criação das classes de domínio, de modo a proceder-se à posterior geração do modelo de dados.

Considera-se que abordagem a adotar deve contemplar todo o potencial de manutenção da base de dados, sem prejuízo para a estrutura, simplicidade e flexibilidade do código aplicacional gerado. Seleciona-se a abordagem código primeiro em conjunto com a técnica de engenharia reversa<sup>134</sup> da base de dados, também denominada de código segundo, pois, na verdade, em primeiro lugar gere-se a base de dados e, posteriormente, escreve-se o código. Assim, procede-se à criação do modelo de dados através de *scripts*<sup>135</sup> que permitem todo o potencial do SGBD

---

<sup>132</sup> Conhece-se *Factory* como um padrão de desenho criacional, que permite a criação de um objeto sem expor a lógica de criação ao cliente e refere-se ao objeto recém-criado, por meio de uma interface comum (Factory 2017).

<sup>133</sup> “Entende-se modelo T4 como uma mistura de blocos de texto e lógica de controlo que pode gerar um ficheiro de texto” (Code Generation and T4 Text Templates 2017).

<sup>134</sup> Entende-se engenharia reversa como “um processo de descobrir os princípios tecnológicos e o funcionamento de um dispositivo, objeto ou sistema, através da análise de sua estrutura, função e operação.” (Engenharia reversa 2017)

<sup>135</sup> Conhece-se script como um texto com uma série de instruções escritas para serem executadas por um programa de computador.

e, de seguida, recorre-se à ferramenta *EF Power Tools*<sup>136</sup> para a geração de código correspondente à aplicação do modelo código primeiro. Assim geram-se as seguintes classes: (i) “DBContext”, (ii) POCO<sup>137</sup> e (iii) classes responsáveis pelo mapeamento entre as classes e a base de dados.

A aborgadem de código primeiro com o suporte da técnica de engenharia reversa une as vantagens da abordagem de base de dados primeiro com a de código primeiro, das quais se salientam: (i) os *scripts* podem ser desenhados e mantidos por uma equipa especializada em base de dados, (ii) utiliza-se tanto em cenários de base de dados existentes como novas, (iii) suporta o conceito de migração de base de dados<sup>138</sup> e (iv) controla-se mais o que se cria e como as coisas funcionam.

### *Repositório*

O padrão repositório implementa a separação de responsabilidades através da abstração da lógica de negócio da persistência dos dados (Kanjilal 2016). Neste padrão, executam-se duas ações: (i) questiona-se a fonte de dados para a obtenção dos dados para de seguida os mapearem em entidades de negócio e (ii) persistem-se as alterações das entidades de negócio na base de dados. Recorre-se ao ORM *Entity Framework* para abstrair a comunicação entre a aplicação e a base de dados.

Utiliza-se “o padrão *Data Mapper* como uma camada de *software* que separa os objetos em memória da base de dados. A sua responsabilidade reside na transferência de dados entre os dois e, ao mesmo tempo, isolar cada um.” (tradução nossa) (Fowler 2003) O próximo treço de código gera-se, automaticamente, pela ferramenta *EF Power Tools* e expõe o mapeamento do objeto na respetiva tabela da base de dados.

```
using System.ComponentModel.DataAnnotations.Schema;
using System.Data.Entity.ModelConfiguration;
namespace ABSInformatica.WinVending_API.DTO.DB.WinVending.Models.Mapping
{
```

---

<sup>136</sup> *EF Power Tools* é uma ferramenta, desenvolvida pela Microsoft, de engenharia reversa do modelo de dados na abordagem de código primeiro.

<sup>137</sup> “Em engenharia de *software*, POCO (Plain old CLR object) é um objeto simples criado na CLR do .NET Framework livre de herança ou atributos.” (Plain old CLR object 2017)

<sup>138</sup> “Em engenharia de *software*, a migração de base de dados refere-se à gestão de mudanças incrementais e reversíveis nos modelos de dados.” (tradução nossa) (Schema migration 2017)

```

public class MaqMap : EntityTypeConfiguration<Maq>
{
    public MaqMap()
    {
        // Primary Key
        this.HasKey(t => t.Cod);
        // Properties
        this.Property(t => t.ID)
            .HasDatabaseGeneratedOption(DatabaseGeneratedOption.Identity);
        this.Property(t => t.NrSerie)
            .HasMaxLength(50);
        this.Property(t => t.Obs)
            .HasMaxLength(1000);
        this.Property(t => t.IDDispositivo)
            .HasMaxLength(100);
        this.Property(t => t.ChangeStamp)
            .IsRequired()
            .IsFixedLength()
            .HasMaxLength(8)
            .IsRowVersion();
        ...
        // Table & Column Mappings
        this.ToTable("Maq");
        this.Property(t => t.ID).HasColumnName("ID");
        this.Property(t => t.Cod).HasColumnName("Cod");
        this.Property(t => t.NrSerie).HasColumnName("NrSerie");
        this.Property(t => t.Obs).HasColumnName("Obs");
        this.Property(t
            t.IDDispositivo).HasColumnName("IDDispositivo");
        this.Property(t => t.ChangeStamp).HasColumnName("ChangeStamp");
        ...
        // Relationships
        this.HasRequired(t => t.Fil)
            .WithMany(t => t.Maqs)
            .HasForeignKey(d => d.Fil_Id);
        this.HasOptional(t => t.Mdl)
            .WithMany(t => t.Maqs)
            .HasForeignKey(d => d.Mdl_Id);
        this.HasOptional(t => t.Pai)
            .WithMany(t => t.Maqs)
            .HasForeignKey(d => d.Pais_Inst_ID);
    }
}

```

A implementação deste padrão possibilita que a camada de lógica de negócios seja agnóstica dos dados da camada de serviço de dados.

### Transação

A tecnologia *Entity Framework* intrinsecamente implementa um mecanismo de gestão de transação baseado no padrão *Unit Of Work*, isto é, mantém uma lista de objetos afetados por uma transação e coordena as operações de escrita e resolução de problemas de concorrência (Fowler 2003). Na prática todas as operações realizadas no âmbito de um contexto adicionam-

se a uma lista de objetos em memória e confirmam-se pela execução do método “SaveChanges()” que gere internamente o conceito de transação.

Embora, o conceito se encontre concretizado no ORM a arquitetura do sistema pretende assegurar a implementação deste requisito de integridade de dados independentemente da tecnologia adotada (e.g. ADO.NET). Persegue-se esse objetivo através da concretização da gestão da transação num ponto central materializado na classe “ContextoTransacionalFabrica”. Essa classe cria uma instância de um objeto que implementa a interface “IContextoTransacional” que, no caso prático deste trabalho, denomina-se “SqlServerEFContextoTransacional” (c.f. Figura 53).

O método de início de transação cria, explicitamente, uma transação na base de dados e pelos métodos de *commit* e *rollback* garante-se, respetivamente, a sua execução ou cancelamento. Deste modo, quando um determinado contexto está num ambiente transacional o método “SaveChanges()” não tem o efeito de persistir, imediatamente, os dados na base de dados e ficam pendentes da execução de um dos métodos de finalização de transação mencionados.

Conclui-se que a camada de regra de negócio responsabiliza-se pela invocação dos métodos de gestão da transação e a camada de acesso a dados assegura o seu correto funcionamento consoante a tecnologia adotada.

#### **7.2.4 Objetos de Negócio**

Um objeto de negócio representa um tipo de entidade do mundo real, tal como: cliente, fornecedor, máquina, produto, entre outros. Expõe-se essas entidades de diferentes modos nas diferentes plataformas de desenvolvimento, como por exemplo: *custom classes*, XML, *dataset* na plataforma .NET.

Representam-se, neste trabalho, os objetos de negócio, unicamente, sob a forma de *custom classes*. Estas classes estão contidas no pacote com o nome “BO” e funcionam como elo de ligação entre as diferentes camadas (apresentação, regras de negócio, acesso a dados).

As classes de objeto de negócio, constituem-se, de um modo geral, por um conjunto de variáveis e propriedades de negócio e outras de controlo (e.g. responsável e data de alteração). Excecionalmente, as aplicações do tipo *Windows Forms* incluem, nestas classes, a implementação de uma interface de auxílio à técnica de *databinding* denominada:

“INotifyPropertyChanged”. Ao implementarem essa interface permite ao mecanismo de *databinding* conhecer uma alteração realizada na propriedade de um objeto dessa classe. O próximo trecho de código exemplifica a implementação da referida interface na classe “Maquina” e a invocação do método “OnPropertyChanged”, na propriedade “Codigo”.

```
Public Class Maquina
    Implements INotifyPropertyChanged
...
Public Property Codigo() As Int32
Get
    Return _codigo
End Get
Set(ByVal Value As Int32)
    If (Me.estadoAlteracao =
ConstEnum.enumEstadoAlteracao.SEM_ALTERACAO) AndAlso (_codigo.Equals(Value)
= False) Then
        Me.estadoAlteracao =
ConstEnum.enumEstadoAlteracao.ALTERADO
    End If
    _codigo = Value
    OnPropertyChanged("Codigo")
End Set
End Property
...
End Class
```

## 7.2.5 Objetos de Transferência de Dados

Conhece-se o padrão de software “objetos de transferência de dados” (DTO) como facilitador de transferência de dados entre diferentes sistemas. No caso prático dos sistemas em análise, conhecem-se os próximos DTO’s: *middleware*, API WinVending e API WinGuard.

O próximo trecho de código refere-se ao DTO de um vendedor da aplicação WinVending (“Seller”).

```
namespace ABSInformatica.WinVending_API.DTO.WinVending
{
    public class Seller
    {
        public int ID { get; set; }
        public int Code { get; set; }
        public string Name { get; set; }
        public string Abbreviation { get; set; }
        public int StorageCode { get; set; }
        public bool ComercialType { get; set; }
        public bool MachineType { get; set; }
        public int? SectionID { get; set; }
        public bool Inactive { get; set; }
        public bool AuditSupervisor { get; set; }
        public int? SecondaryStorage { get; set; }
    }
}
```

```

    public string CashierTreasury { get; set; }
    public string CashierVaults { get; set; }
    public string CashierChange { get; set; }
    public string CashierCommissions { get; set; }
    public bool ActiveOnWinsight { get; set; }
    public bool WinVendingProfile { get; set; }
    public int? Colour { get; set; }
    public int? UserID { get; set; }
    public byte[] ChangeStamp { get; set; }
    public ConstantsEnumerations.Status Status { get; set; }
}
}

```

Note-se que o trecho de código do DTO do vendedor se encontra de acordo com os requisitos não funcionais de interface definidos (“WVB\_NTR\_005” e “WVG\_NTR\_002”), ou seja, encontra-se disponível em inglês.

Os DTO’s, caracterizam-se pela sua simplicidade e pela ausência de comportamento, exceto o de armazenagem e obtenção dos próprios dados.

### 7.2.6 Comunicação Entre API’s

Em conformidade com os requisitos não funcionais de implementação (“WVB\_MPL\_001” e “WVG\_MPL\_001”) a comunicação entre API’s baseia-se nos dois padrões da categoria de serviços: *Service Interface* e *Service Gateway*.

O primeiro padrão (*Service Interface*) surge como resposta a uma necessidade das aplicações empresariais em disponibilizar as suas funcionalidades para sistemas externos. Assim, desenha-se o sistema como uma coleção de serviços em que cada um tem uma interface disponível para que os consumidores da aplicação possam interagir com o serviço. A interface define e implementa um DTO que deve ser reproduzido pelo consumidor. (Trowbridge et. Al. 2003)

O segundo padrão (*Service Gateway*) advém da necessidade de consumir um serviço fornecido por outro sistema. Para atingir essa finalidade, procede-se ao encapsulamento: (i) do código de comunicação com os serviços externos e (ii) das traduções dos BO’s em DTO’s.

A Figura 54, representa o diagrama de pacotes da comunicação entre dois sistemas distintos onde o pacote “Interface” disponibiliza um conjunto de serviços que podem ser consumidos pelo pacote “Gateway” através de trocas de DTO’s. Referencia-se o pacote do “DTO.API” tanto pelo “Interface” como pelo “Gateway”.

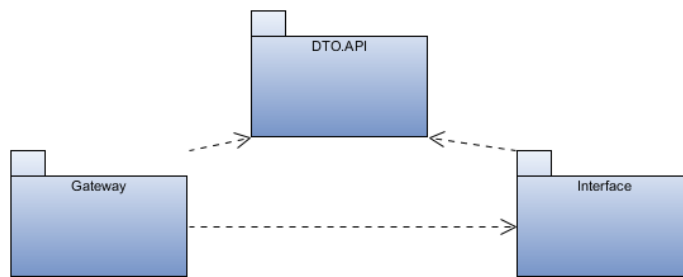


Figura 54 – Diagrama de pacotes comunicação API

### 7.2.7 Controllers / Interface

Em Web API, entende-se *controller* como um objeto que lida com pedidos *http* e cria respostas no mesmo formato.

#### Organização de pacotes

Nas API's, agrupam-se os *controllers* num pacote com o mesmo nome que depende dos seguintes pacotes: (i) objetos de transferência de dados da respetiva API (DTO.API), (ii) regras de negócio (BL) e (iii) objetos de negócio (BO) (c.f. Figura 55). Os objetos de transferência de dados concentram em si todos os dados necessários para invocações e respostas remotas. Mapeia-se o DTO, após a sua receção, num objeto de negócio BO para posterior invocação de determinado método de uma classe de regras de negócio. Finalizada a execução da regra de negócio devolve-se um objeto de negócio para o *controller* que o mapeia de volta num novo objeto de transferência de dados.

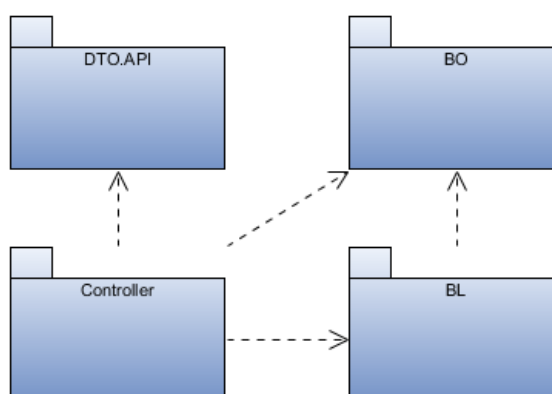


Figura 55 – Diagrama de pacotes Controller

#### Organização de classes

De acordo com a Figura 56, todos os *controllers* herdam de uma classe personalizada “BaseController”, derivada de “ApiController”, a qual possibilita a adição de novas funcionalidades comuns a todas as classes num único ponto (e.g. implementação de uma anotação de validação da chave da aplicação). Um *controller* pode depender de uma classe partilhada de mapeamento cuja responsabilidade reside na transformação de uma classe DTO numa classe BO e vice-versa.

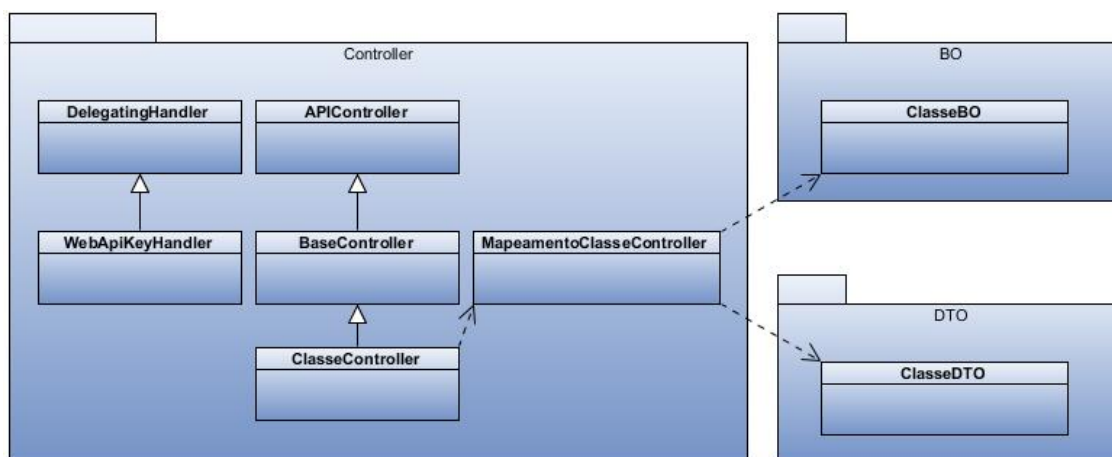


Figura 56 – Diagrama de classes *Controller*

### Autenticação

Uma vez exposta ao exterior, a API necessita de proteger alguns ou todos os serviços disponíveis por meio de mecanismos de autenticação<sup>139</sup>. Ao utilizar o mecanismo de autenticação assegura-se que todos os pedidos recebidos pelos serviços protegidos se realizam por clientes com credenciais apropriadas.

Em Web API, pode assegurar-se a autenticação em dois níveis distintos: (i) no *controller* ou (ii) num nível anterior conhecido pelo estrangeirismo *message handler*. Neste trabalho, adota-se o segundo sistema que consiste numa classe “WebApiKeyHandler” derivada da classe “DelegatingHandler” que interceta e valida os pedidos http antes que estes atinjam o *controller* (Poudel 2016).

---

<sup>139</sup> Autenticação é o processo de determinar que alguém ou alguma coisa é de facto quem diz ser.

## 7.2.8 Gateway

No contexto do desenvolvimento de aplicações empresarias que consomem serviços fornecidos por outra aplicação realça-se a importância da utilização do padrão *Gateway*<sup>140</sup>, mais concretamente, do tipo específico *Gateway de Serviço*, adaptado para o uso em arquiteturas orientadas a serviços, cuja principal preocupação reside na abstração do acesso a serviços externos (Trowbridge et. Al. 2003). A sua função consiste no encapsulamento dos detalhes de conexão com os serviços externos e na concretização de traduções entre os objetos de negócio e os objetos de transferência de dados de uma API concreta.

### *Organização de pacotes*

O *Gateway* encontra-se presente num pacote com o mesmo nome invocado pelas regras de negócio (BL) que lhe enviam objetos de negócio (BO) e esperam receber objetos do mesmo tipo. Assim, o BL referencia o *Gateway* e ambos dependem do BO. Espera-se que o *Gateway* traduza os objetos de negócio (BO) em objetos de transferência de dados (DTO.API) para que, de seguida, se enviem para API's externas. Representa-se, na Figura 57, sob a forma diagrama de pacotes, as dependências do pacote *Gateway*.

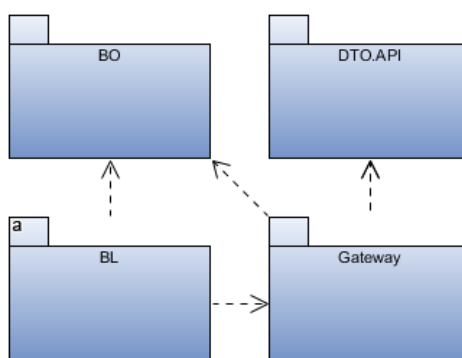


Figura 57 – Diagrama de pacotes *gateway*

### *Organização de classes*

---

<sup>140</sup> *Gateway* é um padrão que define um objeto que encapsula o acesso a sistemas ou recursos externos (Fowler 2003).

Conforme Figura 58, todas as classes “ClasseGateway” presentes no pacote “Gateway” herdam da classe “GatewayBase” cujo construtor recebe um conjunto de parâmetros (uri do serviço e credenciais do utilizador) e que fornece um método de conexão aos serviços externos “ObterURIPorParametros” que recebe argumentos (recurso e os respetivos parâmetros) e devolve o uri completo para a invocação de um serviço. Posteriormente, traduz-se, numa classe partilhada de mapeamento, os objetos de negócio em objetos de transferência de dados da API (DTO.API). Por último, invoca-se o serviço REST através da “FrameworkABS”<sup>141</sup> com os parâmetros, anteriormente, preparados (uri e DTO.API).

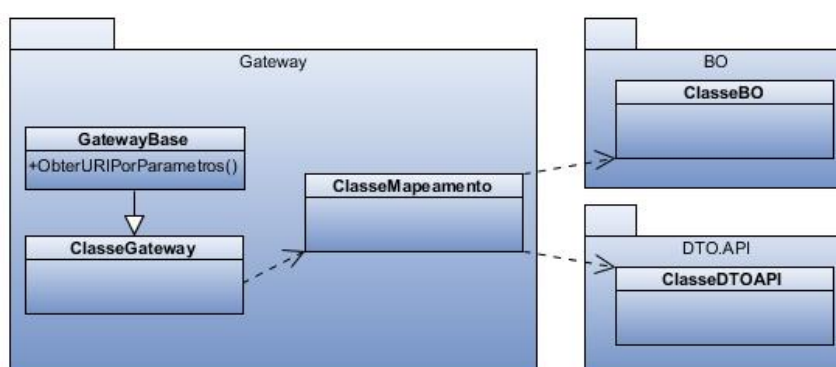


Figura 58 – Diagrama de classes *gateway*

### 7.3 Integração de Aplicações Empresariais

Nos dias de hoje, raramente as aplicações vivem isoladas e considera-se que qualquer uma pode fazer melhor através da integração com outras aplicações (Hohpe & Woolf 2004). Defini-se “integração empresarial como a tarefa de fazer com que aplicações dispares trabalhem juntas para produção de um conjunto de funcionalidades unificadas” (tradução nossa) (Hohpe & Woolf 2004). Por paralelismo com as anteriores afirmações, verifica-se que as funcionalidades disponíveis, neste trabalho, apenas se concretizam pela conjugação das funcionalidades disponíveis em cada um dos sistemas intervenientes: (i) *Middleware* na recolha e tratamento das comunicações e alarmes, (ii) *WinVending BackOffice* na apresentação dos

---

<sup>141</sup> FrameworkABS é uma biblioteca de utilidades proprietária da ABS Informática que reúne um conjunto de funcionalidade utilizadas frequentemente pela equipa de desenvolvimento da empresa (e.g. invocação de serviços REST).

dados provenientes da máquina e manutenção de rotas, (iii) WinVending *Mobile* na manutenção dos dados da máquina e gestão da rota diária, (iv) WinGuard na visualização de alarmes prioritários e (v) ERP na gestão das entidades (e.g. clientes, fornecedores, produtos) e dos documentos do sistema (documentos de venda).

Identificadas as aplicações e funcionalidades a partilhar equacionam-se as próximas abordagens de integração: (i) transferência de ficheiros em que cada aplicação produz ficheiros de partilha de dados para serem consumidos por outra aplicação, (ii) partilha de base de dados onde as aplicações armazenam os dados que desejam compartilhar numa base de dados comum, (iii) invocação de procedimentos remotos onde uma aplicação expõe procedimentos de modo a que possam ser invocados remotamente por outras aplicações e (iv) mensagens onde cada aplicação se conecta a um sistema de mensagens comum (e.g. MSMQ) para trocarem dados por meio de mensagens (Hohpe & Woolf 2004).

As abordagens de transferência de ficheiros e partilha de base de dados permitem, somente, a partilha dos dados das aplicações, no entanto, não é suficiente para alguns requisitos definidos em todas as aplicações os quais necessitam de socorrer-se das funcionalidades dos sistemas. Por esse motivo, descartam-se estas abordagens para os cenários de integração definidos na nova solução. Das restantes duas abordagens de integração justificam-se e fundamentam-se, nas próximas secções, a abordagem adotada para cada cenário particular dos sistemas: (i) *Middleware*, (ii) *backoffice*, (iii) móvel, (iv) WinGuard e (v) ERP.

### **7.3.1 Middleware**

O *Middleware*, desenvolvido pelo parceiro KTS, seleciona a abordagem de invocação de procedimentos remotos como modo de estabelecer a integração entre o WinVending e as máquinas de venda automática. Segue os princípios da abordagem mencionada ao encapsular e abstrair todos os detalhes específicos de cada fabricante de máquinas e das comunicações com o parceiro Vodafone através da implementação de uma API.

A API encontra-se dividida em duas partes: (i) *webhooks*<sup>142</sup> e (ii) API HTTP regular. Por via da técnica de *webhooks* concede-se ao WinVending a oportunidade de receber os eventos diretamente do Middleware e, por essa via, melhora-se o tempo de comunicação do sistema. A API HTTP utiliza o formato JSON para serializar e desserializar os dados e adota o estilo arquitetural REST. Os dados de data e hora representam-se no formato ISO 8601 e em UTC. Como método de autenticação adota o envio em cada pedido do nome de utilizador e palavra-passe, ambos sem qualquer tipo de encriptação. Parte-se do pressuposto que todas as comunicações entre os sistemas se encontram protegidas pelo protocolo de comunicação HTTPS.

### 7.3.2 Sistema BackOffice

O sistema *backoffice*, desenvolvido pela ABS Informática, elege o padrão de invocação de procedimentos remotos como modo de estabelecer a integração entre os sistemas: (i) Middleware, (ii) WinVending *BackOffice*, (iii) WinVending *Mobile*, (iv) WinGuard e (v) ERP. A integração bilateral com o Middleware consiste, por um lado, na disponibilização dos *webhooks* definidos e, por outro lado, no consumo dos serviços por ele implementados. Integra-se com o WinVending *BackOffice* unilateralmente por via da abstração e encapsulamento dos dados relativos às: (i) máquinas, (ii) comunicações, (iii) alarmes, (iv) rotas e (v) entidades do ERP (e.g. clientes e produtos). Realiza-se a integração unilateral com o WinVending *Mobile* através da disponibilização de um conjunto de serviços relativos às: (i) comunicações e (ii) rotas. Adota-se a técnica de *callback* para integrar os alarmes prioritários no sistema WinGuard. A integração com o ERP estabelece-se, num único sentido, e consiste na abstração e encapsolamento da: (i) sincronização com o ERP e (ii) geração de documentos.

Realça-se um caso particular de implementação: os *webhook's* definidos pelo Middleware. A particularidade, destes serviços, reside na necessidade de alta disponibilidade derivado da alta frequência de invocação das suas operações (alarmes, comunicações e programações). Pretende-se atingir, tal objetivo, através da rápida execução dos pedidos efetuados e, para isso,

---

<sup>142</sup> Defini-se “*webhooks* como *callbacks* HTTP definidos pelo utilizador” (tradução nossa) (Webhooks 2017).

pondera-se a adoção de duas estratégias de execução de operações: (i) lançamento da operação numa *thread* ou (ii) colocação da operação numa fila de mensagens.

A técnica de criação de uma *thread* para a execução do pedido implementa-se mais rapidamente, no entanto, apresentam-se as seguintes desvantagens: (i) aumento do consumo de memória pelo facto de ocorrerem várias instâncias de execução de operações em paralelo e (ii) menor tolerância a falha (e.g. base de dados indisponível ou aplicação termina inesperadamente).

Para colmatar essas desvantagens, utilizam-se mensagens para transferir as operações mais frequentes de uma forma imediata, confiável e assíncrona. Esta técnica apresenta como vantagens: (i) baixo acoplamento, (ii) isolamento de falhas, (iii) nivelamento de carga e (iv) operações desconectadas. O baixo acoplamento permite independência: (i) do estado das aplicações, ou seja, as aplicações enviam mensagens para as filas sem necessitarem de conhecer se a aplicação que recebe se encontra ou não disponível e (ii) de processamento, pelo facto da aplicação poder enviar mensagens com uma frequência que não depende da rapidez de processamento da aplicação que recebe. Com o isolamento de falhas uma das aplicações pode falhar sem afetar a outra. O nivelamento de carga possibilita a parametrização da frequência de produção e consumos de mensagens de modo a evitar a sobrecarga do recetor. A característica de operações desconectadas permite, em ambientes de alta latência ou disponibilidade limitada, o envio, receção e processamento de mensagens. Posteriormente, e assim que se estabelece a conexão, a fila encaminha as mensagens para a aplicação de receção (Reitan et al. 2017).

Salientam-se os seguintes aspetos técnicos da implementação da API WinVending: (i) utiliza o formato de dados JSON, (ii) respeita o requisito não funcional de interface “WVB\_NTR\_006” ao representarem-se as variáveis de tempo no formato ISO 8601 e em UTC e (iii) como método de autenticação envia-se a chave de utilizador encriptada. Normalmente, a chave de utilizador obtem-se após a invocação bem-sucedida do serviço de autenticação onde se envia o nome de utilizador e a palavra-passe e recebe-se como retorno a respetiva chave.

Recorre-se à tecnologia de gestão de filas de mensagens MSMQ para a gestão das mensagens relativas aos alarmes. Posteriormente, pretende-se estender a aplicação deste estilo de integração às comunicações e programações.

### 7.3.3 Sistema Móvel

Atualmente, o sistema móvel, desenvolvido pela ABS Informática, adota o padrão de integração de transferência de ficheiro para a sincronização dos dados com a solução legada (c.f. subcapítulo 2.3).

Verifica-se uma integração bilateral entre o sistema móvel e a solução legada. Por um lado, a solução legada exporta um ficheiro (com os dados das entidades e configurações do sistema), no formato CSV, para uma localização a partir da qual a aplicação móvel importa e transforma os dados para o seu sistema. Por outro lado, a aplicação móvel gera um ficheiro (com os documentos criados), também no formato CSV, que coloca numa localização onde a solução legada o lê e o integra internamente.

Confirma-se como vantagem, desta abordagem, o baixo acoplamento entre as aplicações, ou seja, verifica-se uma baixa dependência entre ambas, pelo facto da comunicação estabelecida se restringir à troca de ficheiros. Noutro ponto de vista, detetam-se as próximas desvantagens: (i) não garante a atualização dos dados, (ii) nem a validade da estrutura dos ficheiros. De modo a colmatar as desvantagens detetadas necessita-se de implementar controlos adicionais que aumentam o custo da solução. Pelos motivos apontados, assume-se que esta abordagem não se adequa para a integração com novas funcionalidades do sistema.

De acordo com a decisão tomada e acrescida pelo facto da existência de uma API do WinVending delega-se para a última a responsabilidade de fornecer as novas funcionalidades requeridas pelo sistema móvel, na ótica da telemetria, das quais se destacam: (i) conhecimento do estado da máquina, (ii) gestão da rota e (iii) obtenção das comunicações.

Note-se a necessidade de implementações específicas de determinados serviços, já existentes na API, de modo a satisfazerem particularidades técnicas que o sistema móvel o exige (e.g. tratamento de campos booleanos como inteiros). Devido às limitações da plataforma móvel a que se acresce as características conhecidas das aplicações móveis (e.g. comunicações lentas) assume-se, como uma realidade, a necessidade de implementações particulares cuja implementação pode seguir uma das seguintes estratégias: (i) criação de uma camada de abstração da API WinVending, ou (ii) criação de novos serviços para satisfazerem novos requisitos. Pondera-se a criação de uma camada de abstração entre o sistema móvel e a API do WinVending materializada, por exemplo, na construção de uma API WinVending Móvel que estabelece a comunicação entre a aplicação móvel e a API do WinVending. Devido à perda de

desempenho que esta solução acarreta a que se somam os esforços de implementação, manutenção e testes exigidos, decide-se pela criação de serviços específicos da API WinVending para os requisitos transmitidos pela aplicação WinVending *Mobile*.

#### **7.3.4 Sistema WinGuard**

Igualmente, desenvolvido pela ABS Informática, o sistema WinGuard adere ao padrão de integração de procedimentos remotos para: (i) estabelecer a comunicação entre o sistema WinVending e o sistema WinGuard e (ii) abstrair o acesso aos dados da aplicação WinGuard. Implementa-se, o referido padrão, através da construção de uma API REST constituída por: (i) serviços de *callback* de receção de alarmes e (ii) um conjunto de serviços de acesso aos dados do WinGuard.

Descrevem-se os próximos aspetos técnicos da implementação da API WinGuard: (i) adota o formato de dados JSON, (ii) respeita o requisito não funcional de interface “WVG\_NTR\_003” ao representarem-se as variáveis de tempo no formato ISO 8601 e em UTC e (iii) como método de autenticação utiliza o envio de uma chave encriptada e partilhada entre os sistemas envolvidos.

#### **7.3.5 Sistema ERP**

A solução legada, em funcionamento paralelo com a nova solução, integra com o ERP PHC suportado pelo padrão de partilha de base de dados. A utilização de uma base de dados partilhada garante uma estrutura comum, padronizada e atualizada. No entanto, apresenta algumas dificuldades que consistem: (i) na elaboração de um desenho genérico num ambiente de constantes alterações e (ii) custo de partilha de dados síncronos que envolvem controlo transacional (e.g. degradação do desempenho do sistema). O foco principal, desta abordagem, é a partilha de dados, embora, se possam executar ações na base de dados (Rocha 2013). No caso prático, da integração da aplicação legada partilham-se os dados das entidades (e.g. clientes e produtos), em modo de leitura, e executam-se transações para efetivar documentos (e.g. documentos de venda) na base de dados.

No desenho da nova solução, e na ótica da seleção da abordagem de integração com o ERP, invoca-se o quinto princípio de David Hooker (“O Quinto Princípio – Estar aberto para o futuro”) que refere que uma solução de *software* deve: (i) adaptar-se às constantes alterações das

necessidades e preferências dos utilizadores e (ii) assegurar-se da flexibilidade e extensibilidade da arquitetura da solução de modo a capacitar-se da envolvimento em cenários futuros (Hooker 1996).

De modo a cumprir o princípio mencionado, efetuam-se as seguintes tarefas: (i) analisam-se as limitações da solução legada e (ii) observa-se as necessidades atuais e futuras dos clientes e do mercado. Desse trabalho resulta a deteção dos seguintes pontos: (i) potenciais clientes utilizam diferentes ERP's, (ii) aquando da indisponibilidade do ERP a solução de Vending não funciona e (iii) na solução legada, a criação de novos campos realiza-se na base de dados partilhada, mesmo que só sejam relativos à solução de Vending.

Deste modo, consideram-se os próximos requisitos na integração com o ERP: (i) múltiplos ERP's com múltiplas abordagens de integração (e.g. transferência de ficheiros, partilha de base de dados e invocação de procedimentos remotos), (ii) acesso aos dados do ERP a qualquer momento independentemente da disponibilidade do ERP e (iii) criação de novos campos referentes às entidades do ERP sem necessidade de os persistir no modelo de dados do ERP.

Reflete-se acerca do primeiro requisito e da sua aplicabilidade em outras soluções da empresa e, a propósito, invoca-se o sexto princípio de David Hooker ("Sexto Princípio – Planear com antecedência para reutilizar") assente nas seguintes premissas: (i) a reutilização economiza tempo e trabalho, (ii) premeditar para potenciar a reutilização, (iii) existem várias técnicas de reutilização em todos os níveis do processo de desenvolvimento e (iv) comunicar as oportunidades de reutilização a outros elementos da organização (Hooker 1996).

Da anterior reflexão, resulta a ideia de criar uma API REST de integração com o ERP denominada de: "API Integrador ERP". Esta API tem como finalidade principal integrar qualquer aplicação da empresa com qualquer ERP e, ao mesmo tempo, manter a porta aberta para a sua comercialização como uma solução de integração com ERP's. No caso concreto da solução Vodafone WinVending, a API do WinVending integra com o ERP através da API Integrador ERP que abstrai toda a lógica de negócio e acesso a dados do ERP. Realça-se que o desenvolvimento desta API não se enquadra no âmbito deste trabalho.

Avalia-se o segundo e terceiro requisitos e decide-se replicar um subconjunto de dados principais das entidades (e.g. clientes e produtos) que residem no sistema ERP e aumentá-los, no contexto do sistema do WinVending, isto é, criam-se tabelas de entidades no novo sistema e, todos os novos campos específicos, persistem-se aí. "Associa-se, por vezes, esta técnica com

a integração da cadeira giratória, no entanto, não é a mesma. Na verdadeira integração com cadeira giratória, a mesma informação insere-se em múltiplas aplicações que não se integram diretamente. Com contextos limitados, os dados mestre são replicados, e são contextualmente aumentados e transformados em aplicativos dependentes.” (tradução nossa) (Byars 2013)

A implementação, desta técnica, exige esforços de desenvolvimento de procedimentos de sincronização entre o sistema ERP e o sistema interno., também eles, não se enquadram no âmbito desta dissertação.

Concluída a sincronização entre os sistemas, disponibiliza-se o acesso desconectados aos dados do ERP, em modo leitura, e permite-se a alteração dos campos específicos da nova solução. A sincronização entre os sistemas, despoleta-se de dois modos distintos: (i) automaticamente, através de uma programação de de X em X tempo e (ii) manualmente, pela execução de uma ação de atualização do utilizador.

A API do WinVending abstrai e encapsula o acesso aos dados das entidades do ERP, ou seja, quem invoca o serviço desconhece se se obtém as entidades da base de dados da solução ou, da base de dados do ERP



## 8 Implantação

Neste capítulo, detalham-se dois casos particulares de processos e rotinas de trabalho disponibilizadas aos utilizadores finais: (i) representação da localização da máquina e (ii) definição da estratégia de distribuição das aplicações (por *setup* ou publicação). Refere-se, também, como o sistema gere a mudança de: (i) arquitetura da aplicação, (ii) atualização do modelo de dados e (iii) atualização da aplicação.

### 8.1 Representação da Localização da Máquina

Atualmente, as máquinas de venda automática de tabaco enfrentam sistemáticos atos de roubo e vandalismo, causadores de prejuízos e transtornos no negócio dos donos das máquinas. Convictos em enfrentar esses crimes, os depositários de tabaco procuram investir em soluções que permitam a sua deteção e atuação no sentido de dissuadir e deter os responsáveis.


Conhecedores dessa realidade, a ABS Informática cria, na solução *WinVending Backoffice*, o requisito “UC\_WVB\_2080” dedicado à deteção das alterações da localização da máquina e a respetiva apresentação da localização atual e, caso exista histórico de localizações, do percurso efetuado. Como complemento ao requisito mencionado, e numa perspetiva mais ativa, a ABS Informática propõe, aos seus clientes, uma solução dedicada, exclusivamente, à deteção e visualização de alarmes com características suspeitas, para que se analisem, por entidades especialistas, no ramo da segurança. Traçam-se para a solução *WinGuard* os requisitos “UC\_WVG\_001” e “UC\_WVG\_010” dedicados ao conhecimento da localização da máquina.

Os requisitos referentes à localização da máquina, pressupõem uma apresentação visual da posição da máquina, num mapa. Com a motivação de cumprir esses requisitos, de um modo funcional e atrativo, socorre-se do componente GMap .NET, para a sua implementação.

Conhece-se o GMap.NET como um controlo gratuito para .NET, que disponibiliza funcionalidades de mapeamento para aplicações do tipo: (i) *Windows Forms*, (ii) *Windows Presentation Foundation*<sup>143</sup> e (iii) *Windows Mobile*. Oferece suporte para uma panóplia de fornecedores de mapas, dos quais se destacam: (i) *OpenStreetMap*, (ii) *Yahoo Maps*, (iii) *Bing Maps*, (iv) *ArcGis* e (v) *Google Maps*.



Distinguem-se os seguintes componentes para o entendimento da utilização do referido controlo: (i) “GMapControl” considera-se a primeira camada e a responsável pela renderização do mapa, (ii) “GMapOverlay” entende-se como a camada acima do controlo do mapa e representa um conjunto de marcadores (e.g. rasto da máquina) e (iii) “GMapMarket” conhece-se como o marcador específico colocado numa determinada camada e representa uma coordenada específica (e.g. marcador de localização atual da máquina) (Khan 2011).

### 8.1.1 Mapa

Adota-se, nas soluções desenvolvidas, a renderização do mapa da Google e disponibilizam-se as próximas capacidades de interação com o mapa: (i) mover o mapa, (ii) efetuar *zoom in / out* e (iii) centrar o mapa entre os pontos conhecidos (c.f. Figura 59). Para mover o mapa, pressiona-se o botão do lado esquerdo do rato, seguido da movimentação do rato para a direção que se pretende. O *zoom in* realiza-se de três modos distintos: (i) duplo *click* com o botão esquerdo, no mapa (*zoom* para o centro do mapa), (ii) roldanas do rato (*zoom* para o ponto onde se encontra o ponteiro do rato) e (iii) pressionar o botão  (*zoom* para o centro do mapa). O *zoom out* concretiza-se de três formas: (i) duplo *click* com o botão direito no mapa (*zoom* para o centro do mapa), (ii) roldanas do rato (*zoom* para o ponto onde se encontra o ponteiro do

---

<sup>143</sup> Conhece-se o *Windows Presentation Foundation*, também conhecido pelo acrónimo WPF, como “um subsistema gráfico na Framework .NET 3.0 que utiliza uma linguagem de marcação, conhecida como XAML para desenvolvimento de GUI’s” (*Windows Presentation Foundation* 2017).

rato) e (iii) pressionar-se o botão  (zoom para o centro do mapa). O botão  permite centrar o mapa mediante os marcadores presentes (e.g. coordenadas de instalação e atual).

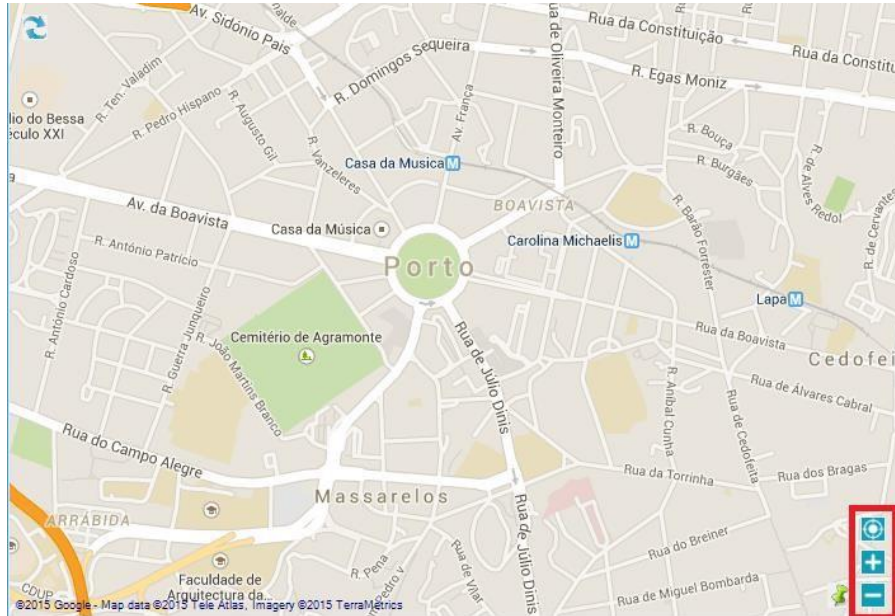


Figura 59 – Mapa Google da aplicação WinVending *BackOffice*

O próximo trecho de código apresenta as seguintes funcionalidades: (i) criação do mapa, (ii) *zoom in*, (iii) *zoom out* e (iv) centrar o mapa.

```

Private Sub IniciarMapa()
    ...
    If _marcadoresMaquinaIn Is Nothing Then
        Me.gmpLocalizacao.MapProvider =
GMap.NET.MapProviders.GoogleMapProvider.Instance
GMap.NET.GMaps.Instance.Mode =
GMap.NET.AccessMode.ServerAndCache
Me.gmpLocalizacao.Zoom = 17
Me.gmpLocalizacao.Visible = True
Me.gmpLocalizacao.ShowCenter = False
Me.gmpLocalizacao.CanDragMap = True
Me.gmpLocalizacao.DragButton = Windows.Forms.MouseButtons.Left
Me.gmpLocalizacao.MouseWheelZoomType =
GMap.NET.MouseWheelZoomType.MousePositionWithoutCenter
    ...
End Sub
Private Sub MapaZoomIn()
    If Me.gmpLocalizacao.Zoom < maximoZoom Then
        Me.gmpLocalizacao.Zoom += 1
    End If
End Sub
Private Sub MapaZoomOut()
    If Me.gmpLocalizacao.Zoom > 0 Then
        Me.gmpLocalizacao.Zoom -= 1
    End If
End Sub

```

```

End Sub

Private Sub MapaCentrar()
    If Me.gmpLocalizacao.Zoom < 17 Then
Me.gmpLocalizacao.ZoomAndCenterMarkers("marcadoresMaquinaZoomOut")
    Else
        Me.gmpLocalizacao.ZoomAndCenterMarkers("marcadoresMaquina")
    End If
    Me.gmpLocalizacao.Refresh()
End Sub

```

### 8.1.2 Marcadores

Disponibilizam-se, nos mapas da aplicação, os seguintes tipos de marcadores<sup>144</sup>: (i) local de instalação da máquina, (ii) localização atual da máquina e (iii) rasto da máquina. Recorre-se ao espaço de nome<sup>145</sup> “System.Drawing” (fornecedor do acesso à funcionalidade de gráficos da GDI<sup>146</sup>) para desenhar, em tempo de execução, os marcadores. O próximo trecho de código demonstra o desenho do marcador da máquina.

```

Shared Function DesenharMaquinaZoomIn(ByVal cor As Estrutura.Cor.enumCor _
, ByVal maquina As Estrutura.Maquina)
As Bitmap
    Dim brush As Brush = New SolidBrush(Color.White)
    Dim fonteMarca As New Font("Arial", 6, FontStyle.Bold)
    Dim fonteCodigoMaquina As New Font("Courier New", 7, FontStyle.Bold)
    ...
    Dim tamanhoImagemMaquina As New Size(imagemMaquinaFacing.Width,
IIf(maquina.RaioSegurancaViolado, imagemAlarme.Height / 2, 0))
    tamanhoImagemMaquina.Height += imagemMaquinaMoedeiroNoteiro.Height
    tamanhoImagemMaquina.Height += imagemMaquinaFacing.Height

    'Imagem maquina
    Dim posicaoYCamadaMaquina As Integer = 0
    Dim imagemMaquina As Image = New Bitmap(tamanhoImagemMaquina.Width,
tamanhoImagemMaquina.Height)
    Dim graficoMaquina = Graphics.FromImage(imagemMaquina)
    If maquina.RaioSegurancaViolado Then
        graficoMaquina.DrawImage(imagemAlarme,
CType(tamanhoImagemMaquina.Width / 2 - imagemAlarme.Width / 4, Integer), 0,

```

<sup>144</sup> Entende-se por marcador a imagem que é colocada no mapa, para representar uma localização.

<sup>145</sup> Conhece-se “espaço de nomes pelo termo inglês *namespace* e funciona como um delimitador abstrato que fornece um contexto para os itens que ele armazena (e.g. nomes, termos técnicos, conceitos) o que permite uma desambiguação para itens que possuem o mesmo nome, mas que residem em espaços de nomes diferentes.” (Espaço de nomes 2017)

<sup>146</sup> “GDI, ou *Graphics Device Interface*, corresponde a um dos três subsistemas principais do Microsoft Windows para representar objectos gráficos e transmiti-los para dispositivos de saída, como monitores e impressoras.” (GDI 2017)

```

CType(imagemAlarme.Width / 2, Integer), CType(imagemAlarme.Height / 2,
Integer))
    posicaoYCamadaMaquina = imagemAlarme.Height / 2
End If
...
End Function

```

Disponibilizam-se os marcadores de localização em diferentes cores, onde cada uma corresponde à cor da rota a que pertence. A máquina, num zoom mais afastado, representa-se por uma pequena seta (e.g. cinzento ▼). Num zoom mais próximo, distinguem-se as seguintes informações: (1) fora do raio de segurança (🚫), (2) tem noteiro (📄), (3) iniciais do fabricante da máquina (AZ – Azkoyen / JO – Jofemar / GM), (4) código da máquina (e.g. 1010), (5) informação de telemetria (📶) (não visível se não tiver telemetria), (6) porta (aberta 🟢 / fechada 🔴 / desconhecida 🟡), (7) conectividade (com 🟢 / sem 🔴 / limitada 🟡 / desconhecida 🟡), (8) cabo (ligado 🟢 / desligado 🔴 / desconhecido 🟡), (9) máquina (ligada 🟢 / desligada 🔴 / desconhecida 🟡) e (10) localização da máquina ▼ (c.f. Figura 60).

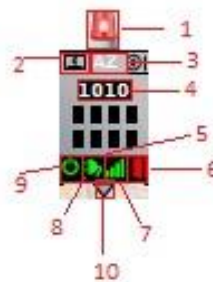


Figura 60 – Marcador máquina

O rasto da máquina representa as localizações anteriores à atual posição da máquina. Num zoom mais afastado, o rasto contém as seguintes informações: (1) ordem cronológica (número mais pequeno corresponde a coordenadas mais antigas), (2) máquina sem movimento (círculo perfeito) e (3) em movimento (círculo com uma serrilha no lado esquerdo) (c.f. Figura 61).

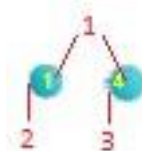


Figura 61 – Marcador rasto da máquina (zoom out)

Num zoom mais próximo, distingue-se os seguintes dados: (1) ordem cronológica, (2) máquina sem movimento (círculo perfeito), (3) em movimento (círculo com uma serrilha no lado esquerdo) e (4) direção (seta direcional) (c.f. Figura 62).

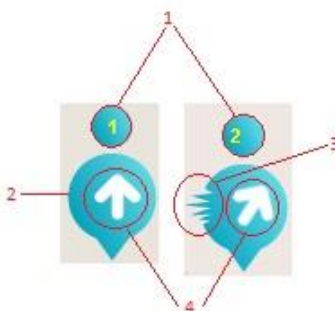


Figura 62 – Marcador rasto da máquina (zoom in)

### 8.1.3 Camadas

Uma camada corresponde a um conjunto de marcadores desenhados sobre o mapa. Na aplicação WinVending BackOffice, distinguem-se as seguintes camadas: (i) máquinas (zoom in), (ii) máquinas (zoom out), (iii) rasto (zoom in) e (iv) rasto (zoom out) (c.f. Figura 63 e Figura 64).

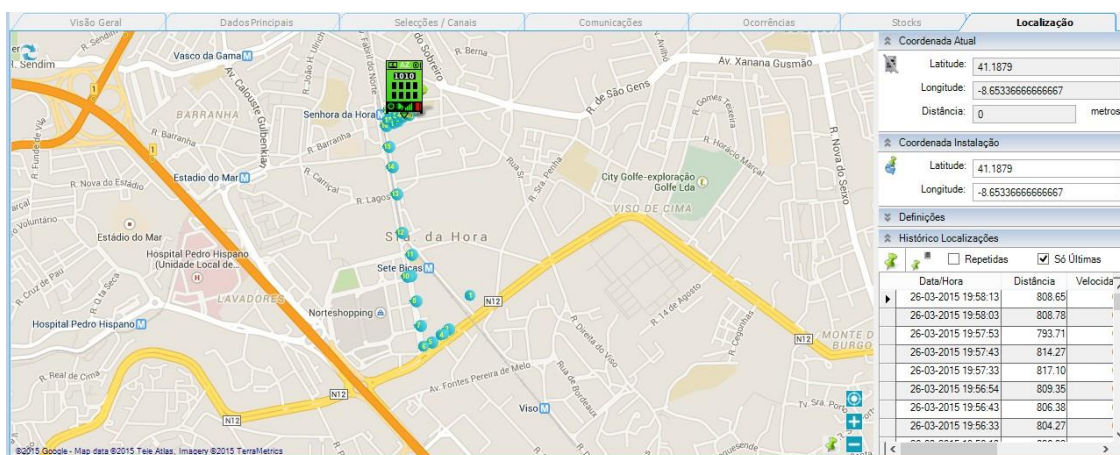


Figura 63 – Histórico de localização da máquina (zoom out)

Decide-se separar por camadas o zoom, devido ao facto de se utilizarem imagens diferentes para representarem determinados níveis de zoom. Isto é, a partir de um nível de zoom mais próximo, apresentam-se imagens de maior dimensão e com mais detalhe e, num determinado nível de zoom mais afastado, utilizam-se imagens mais pequenas e com menos detalhe, devido à possibilidade da ocorrência de múltiplos marcadores no mesmo ponto. Opta-se, igualmente,

por separar a máquina do respetivo rasto, devido à funcionalidade de ocultarem o rasto da máquina. Tecnicamente, para a concretização desta funcionalidade, é suficiente esconder a camada relativa ao rasto em determinado *zoom*.

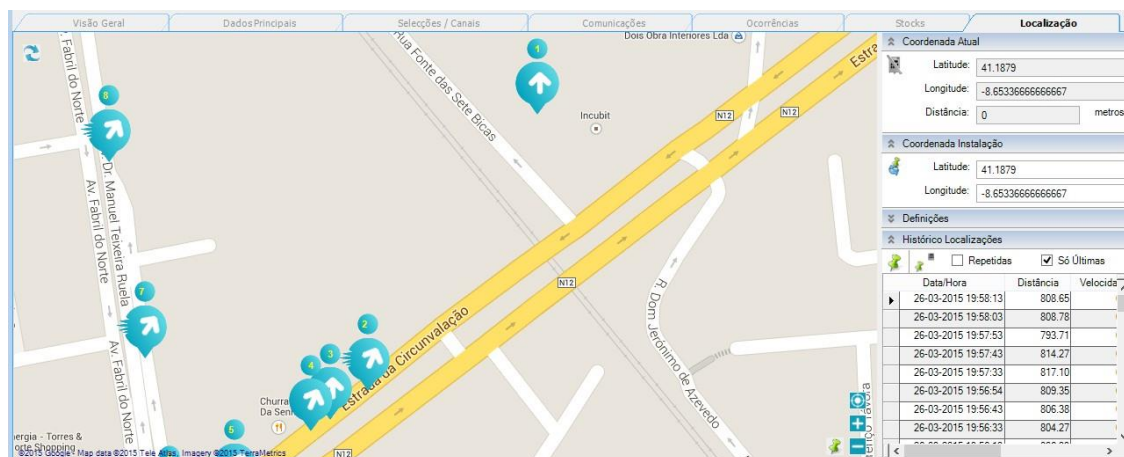


Figura 64 - Figura 63 – Histórico de localização da máquina (*zoom in*)

## 8.2 Instalação

Concluído o desenvolvimento dos componentes do sistema, torna-se necessário disponibilizá-los aos clientes. Com esse propósito, apresentam-se duas estratégias diferentes de instalação: (i) por *setup* (*InstallShield*) e (ii) por publicação (*ClickOnce*). Cada uma delas aplica-se em cenários específicos, condicionados pelas características pretendidas na instalação, que se expõem nas próximas secções.

### 8.2.1 Setup (InstallShield)

A utilização do *InstallShield* possibilita a criação de um pacote de instalação e respetiva distribuição aos utilizadores, para que possam instalar uma aplicação, serviço ou componente, sem necessidade de conexão a uma rede.

Considera-se a adoção do *InstallShield*, caso se pretenda alguma das próximas características: (i) instalação de componentes COM, (ii) instalação de serviços, (iii) instalação em local específico

ou na *Global Assembly Cache*<sup>147</sup> (GAC), (iv) interação do utilizador durante a instalação, (v) configuração dos serviços a nível de sistema (e.g. *Active Directory*<sup>148</sup>), (vi) instalação da aplicação em todos os utilizadores do sistema e (vii) instalação de múltiplos projetos num único pacote (Sanford 2005).

O *InstallShield* substitui a tecnologia de instalação *Windows Installer* (não mais suportada). As versões da ferramenta *Visual Studio Professional* e *Enterprise* incluem o *InstallShield*, na sua versão mais limitada (*Limited Edition*).

Enumera-se os requisitos de instalação dos vários projetos da solução Vodafone WinVending: (i) instalação de serviços, (ii) instalação de múltiplos projetos, num único pacote de instalação, (iii) capacidade de manter os ficheiros de configuração, após a remoção da aplicação e (iv) possibilidade de inibir a sobreposição dos ficheiros de configuração, caso já existam.

Analisa-se os requisitos de instalação e confirma-se a viabilidade de implementação, na versão limitada do *InstallShield* e verifica-se o não suporte dos requisitos dois e três. De modo a colmatar essa lacuna, avaliam-se as várias versões disponíveis do *InstallShield* (*Express*, *Professional*, *Premier*) e mantém-se em mente o princípio de David Hooker (“O Quinto Princípio – Estar aberto para o futuro”), ao verificar funcionalidades úteis para outros projetos da empresa, ou para projetos futuros.

Da ponderação de vários critérios (e.g. preço, requisitos presentes e requisitos futuros), elege-se a versão *Express* por cumprir todos os requisitos presentes a um baixo custo. Destaca-se um requisito futuro, não suportado por esta versão, que se relaciona com a possibilidade de incluir, no mesmo pacote de instalação, projetos de outras soluções.

Enumeram-se os cenários de adoção do estilo *setup*: (i) projetos WinVending e WinGuard BackOffice, em situações de utilização de múltiplos utilizadores (e.g. utilização de máquinas virtuais) e (ii) projetos de serviços (e.g. monitor e API's).

---

<sup>147</sup> Entende-se por *Global Assembly Cache* (GAC) como o diretório especial onde se armazenam todas as *assemblies* partilhadas.

<sup>148</sup> “Entende-se *Active Directory* como uma implementação de serviço de diretório que armazena informações sobre objetos em rede de computadores e disponibiliza essas informações a utilizadores e administradores dessa rede.” (*Active Directory* 2017)

A título demonstrativo, descrevem-se, de seguida, os passos necessários para a instalação do projeto de serviços do WinVending, com a tecnologia *InstallShield Express*, sem o auxílio do assistente de manutenção do projeto.

Começa-se pela criação de um novo projeto do tipo “*InstallShield Express*” e atribui-se um nome ao projeto (e.g. *InstaladorWinVending*) (c.f. Figura 65).

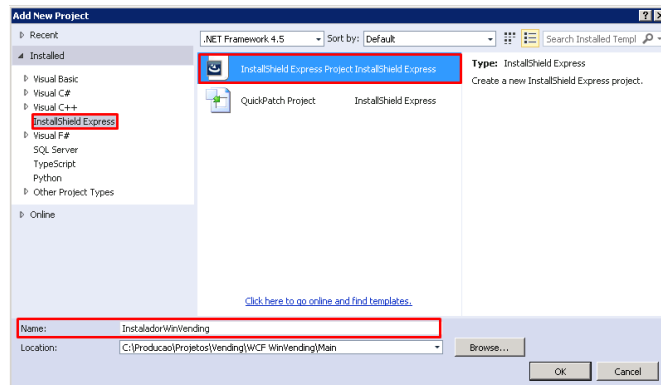


Figura 65 – Criação de projeto *InstallShield Express*

Prossegue-se com a atribuição dos dados gerais do projeto: (i) nome, (ii) versão, (iii) diretório de instalação e (iv) ícone na entrada de adicionar / remover programas do *Windows*.

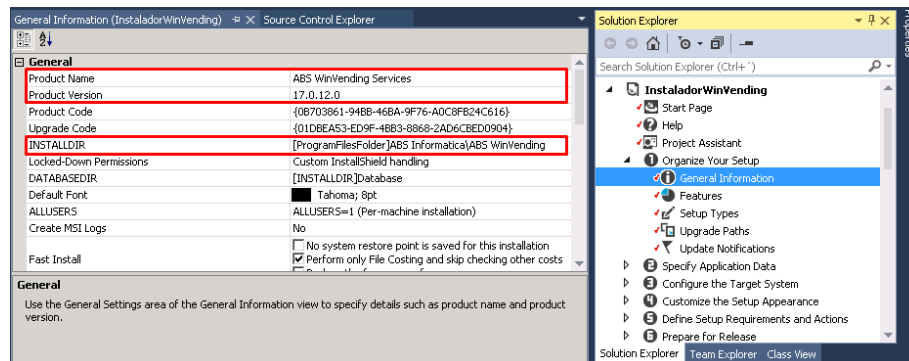


Figura 66 – Dados gerais do projeto de instalação

Definem-se as características a instalar, equivalentes a cada um dos projetos da aplicação (e.g. Monitor, API) e as sub-características correspondentes ao tipo de projeto (serviço ou consola) (c.f. Figura 67).

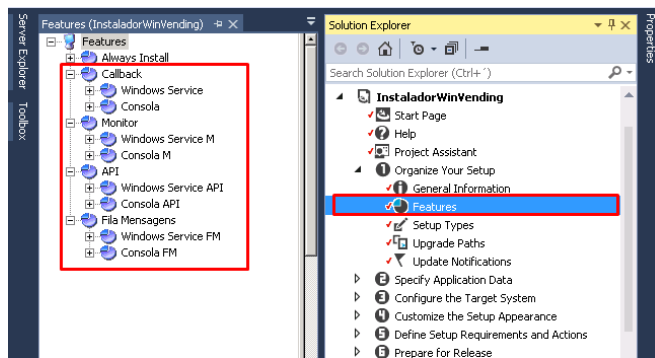


Figura 67 – Características e sub-características do projeto de instalação

Atribuem-se os tipos de instalação disponíveis para o utilizador (“Typical” e “Custom”) e as respetivas características associadas (c.f. Figura 68).

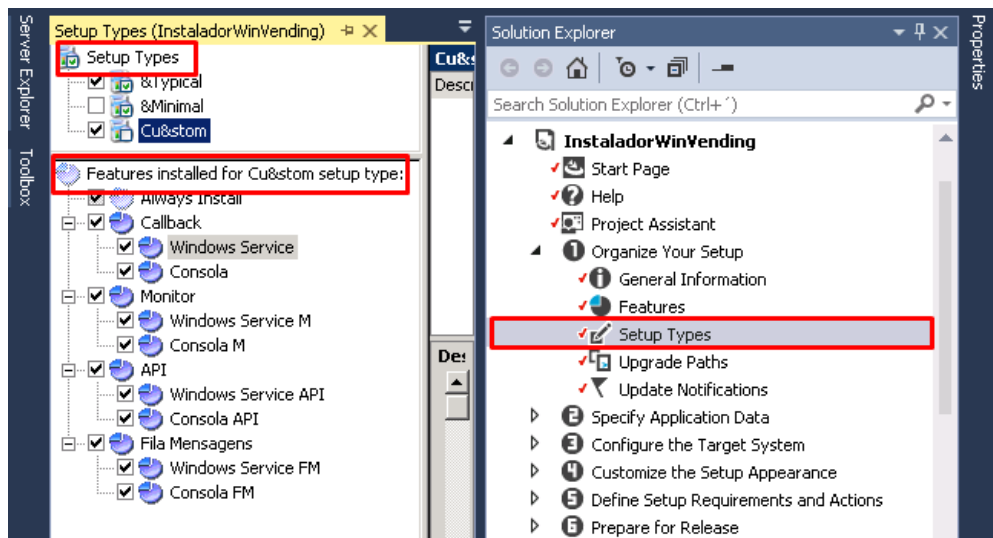


Figura 68 – Tipos de instalação

Selecionam-se os ficheiros a incluir na instalação e a sua localização (c.f. Figura 69). Para cada projeto, escolhem-se os recursos relativos aos ficheiros de saída primária (e.g. ficheiros executáveis e respetivas dependências) e os ficheiros de conteúdo (e.g. ficheiros de configuração personalizados).

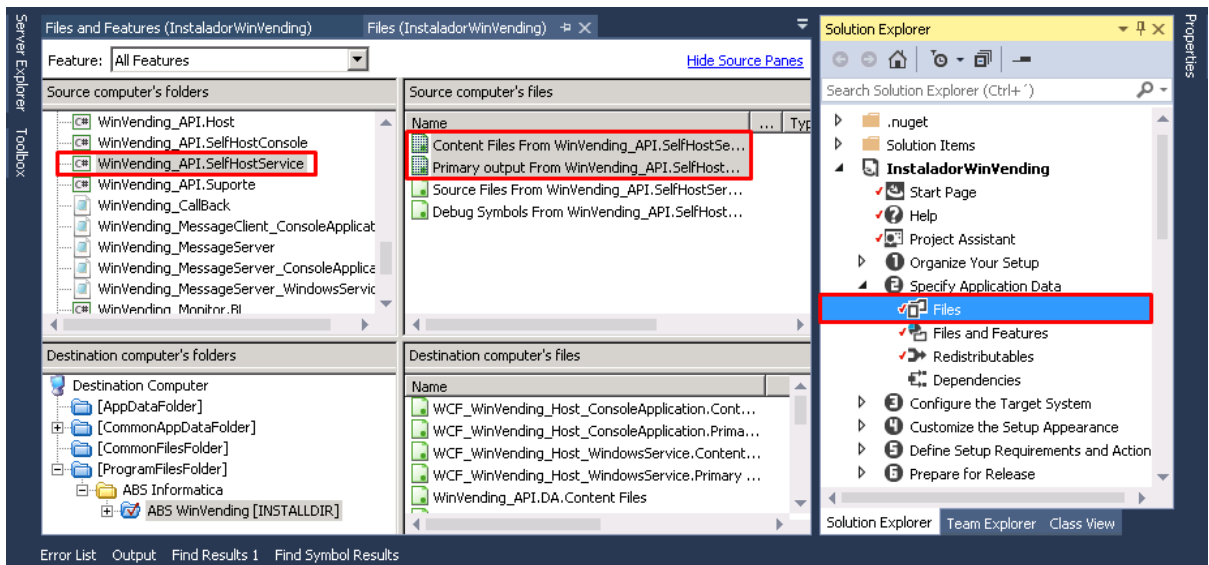


Figura 69 – Ficheiros de instalação

A instalação dos serviços exige a ativação da propriedade “*Installer Class*”, no recurso do ficheiro de saída primária (c.f. Figura 70).

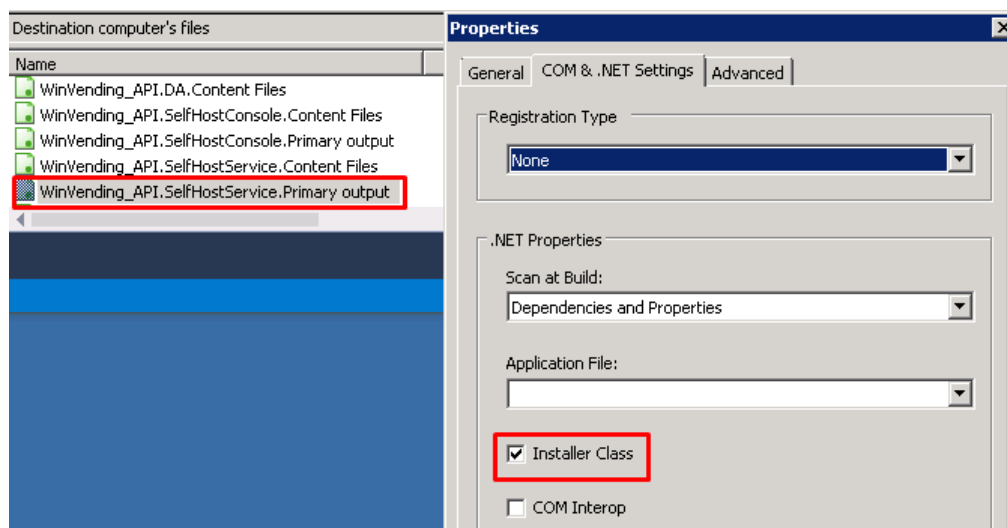


Figura 70 – Propriedades do ficheiro de saída primária de um serviço

Alteram-se as seguintes propriedades dos ficheiros de recursos de conteúdo: ativa-se a propriedade de permanência dos ficheiros, após remoção (“*Permanent*”) e seleciona-se a opção de nunca sobrepôr (“*Never Override*”) os ficheiros de configuração, aquando de uma nova instalação, de modo a salvaguardar as alterações personalizadas realizadas aos ficheiros (c.f. Figura 71).

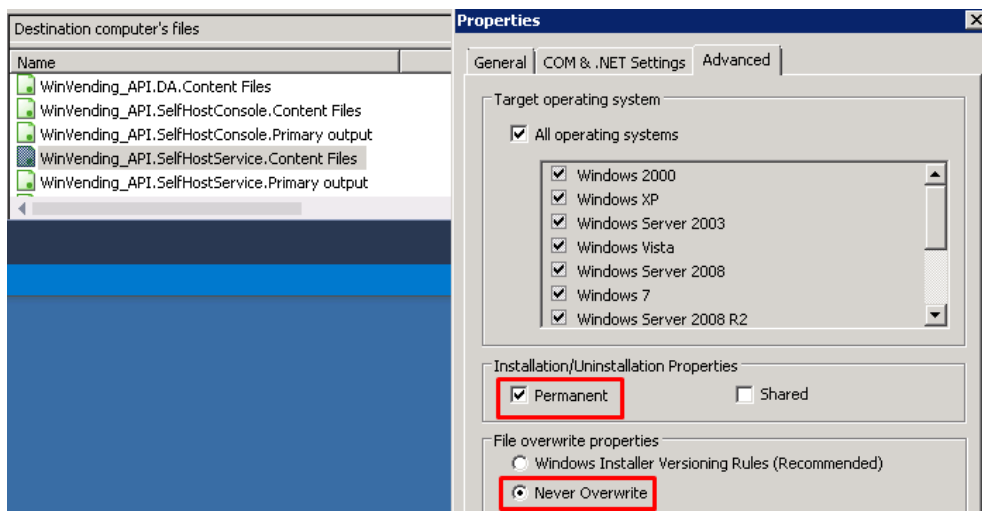


Figura 71 – Propriedades do ficheiro de conteúdo

Visualiza-se, na Figura 72, a associação dos ficheiros de recursos a cada uma das características do projeto de instalação.

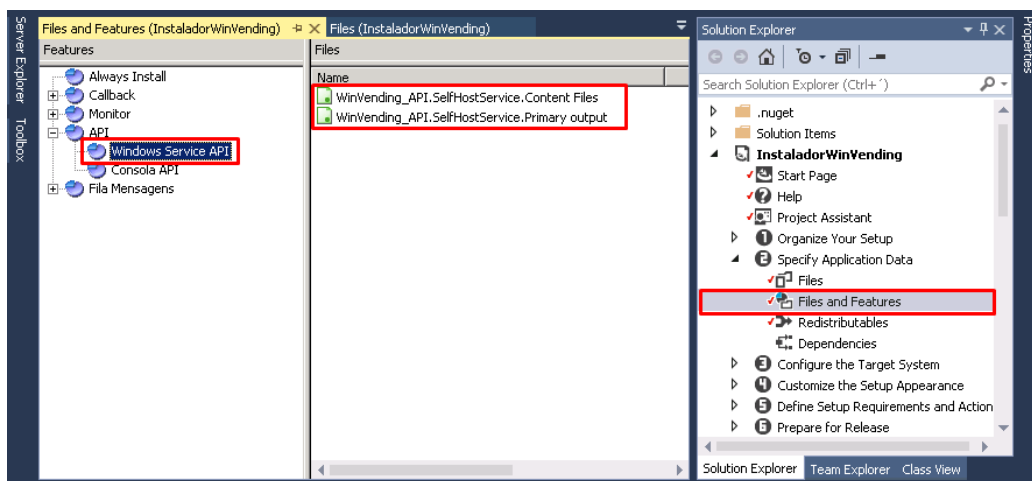


Figura 72 – Ficheiros e características de instalação

Na preparação do lançamento da instalação, define-se o formato e a localização do pacote de instalação gerado, o qual, no caso prático em análise, consiste num único ficheiro ("SingleImage").

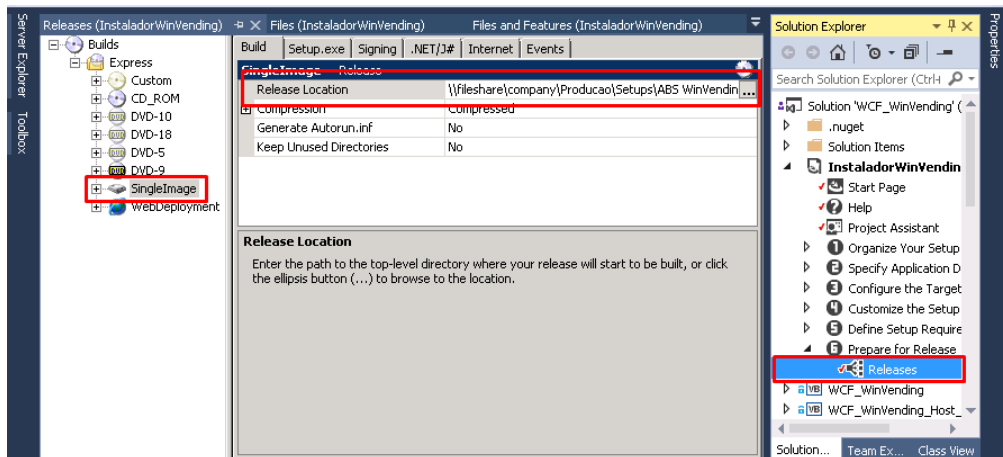


Figura 73 – Preparação do lançamento da instalação

Para terminar, procede-se à compilação do projeto, no modo parametrizado na preparação do lançamento da instalação (e.g. “SingleImage”), do qual resulta um ficheiro no formato executável (e.g. “setup.exe”), pronto para ser distribuído pelos vários clientes.

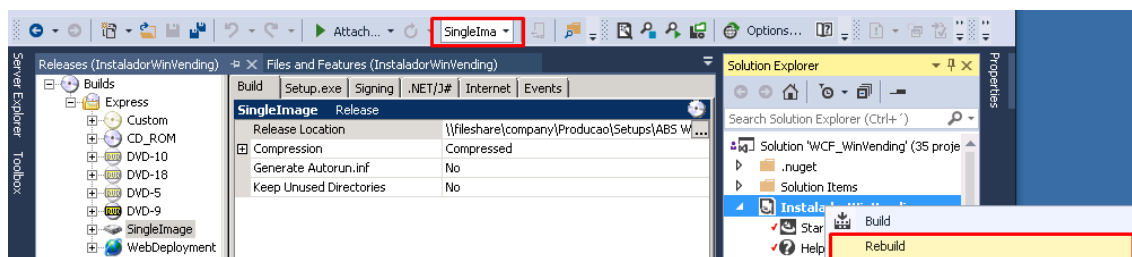


Figura 74 – Compilação do projeto

No cliente, executa-se o referido ficheiro (e.g. “setup.exe”) e, auxiliado por um assistente, prossegue-se à instalação de um ou vários componentes disponíveis, no pacote de instalação.

## 8.2.2 Publicação (ClickOnce)

Considera-se o *ClickOnce* uma tecnologia de distribuição de aplicações, inserida na *.NET Framework 2.0*. Esta tecnologia foca-se, essencialmente, em dois aspectos: (i) simplicidade e (ii) segurança na instalação e atualização das aplicações. Para distribuir uma aplicação *ClickOnce*, colocam-se os ficheiros de publicação num diretório partilhado e fornece-se aos utilizadores uma ligação para o manifesto das aplicações (Sanford 2005).

Utiliza-se a tecnologia *ClickOnce*, na publicação dos próximos tipos de aplicações: (i) *Windows Forms* (.exe), (ii) *Console Application* (.exe), (iii) *Windows Presentation Foundation* (.xbap) e (iv) solução *Office* (.dll). Efetua-se a publicação das aplicações de três modos distintos: (i) página Web, (ii) partilha de ficheiros na rede e (iii) através de um dispositivo multimédia.

Consideram-se as próximas vantagens da tecnologia em análise: (i) atualizações automáticas, (ii) reversão de uma atualização e (iii) execução segura e isolada da aplicação. Uma das mais-valias da tecnologia *ClickOnce* caracteriza-se pela capacidade de ser auto-atualizada, isto é, quando o utilizador entra na aplicação, verifica-se, automaticamente, a existência de versões mais recentes e, caso se detete uma nova versão, atualiza-se a aplicação. A atualização pode ser condicionada, mediante parametrização da publicação, por uma caixa de mensagem a questionar o utilizador acerca da possibilidade de atualizar a aplicação. As atualizações podem ser revertidas para uma versão anterior, caso o utilizador assim o pretenda. As aplicações instalam-se e executam-se, a partir de um diretório seguro, por utilizador. O facto da sua execução ser isolada, evita que interfira com outras aplicações instaladas.

No universo das soluções implementadas, no âmbito desta dissertação, candidatam-se à utilização desta tecnologia as aplicações: *WinForms* (*WinVending* e *WinGuard*) e *Console Application* (utilizadas como réplicas dos serviços). Pelo facto das aplicações do tipo consola serem instaladas, apenas, numa única máquina, não se justifica a adoção desta tecnologia. Na aplicação *WinGuard*, devido ao reduzido número de utilizadores que usufruem da aplicação, também se opta pela não adoção desta tecnologia. No que diz respeito à aplicação *WinVending*, e, pelo seu cariz multi-utilizador, decide-se pela implementação do estilo de publicação aqui abordado. Descrevem-se, de seguida, os passos necessários para a publicação da aplicação: *WinVending*.

Inicia-se o processo pela criação de um certificado digital a utilizar na assinatura do manifesto<sup>149</sup> do *ClickOnce*. Atribui-se o certificado a partir das propriedades do projeto *Windows Forms*, no separador “*Signing*” (c.f. Figura 75).

---

<sup>149</sup> Conhece-se manifesto do *ClickOnce* como um ficheiro XML que descreve a aplicação implementada, de acordo com os seguintes atributos: (i) identificação da assembly principal da aplicação, (ii) descrição

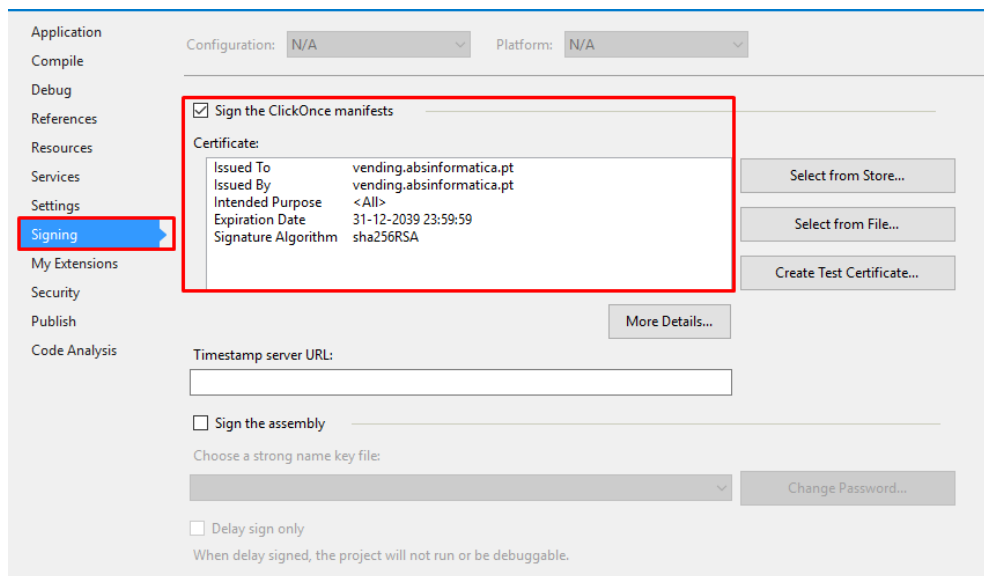


Figura 75 – Assinar o manifesto *ClickOnce*

Prosegue-se com a atribuição dos dados de configuração da publicação: (i) caminho do diretório, (ii) modo de instalação e (iii) número de versão (c.f. Figura 76).

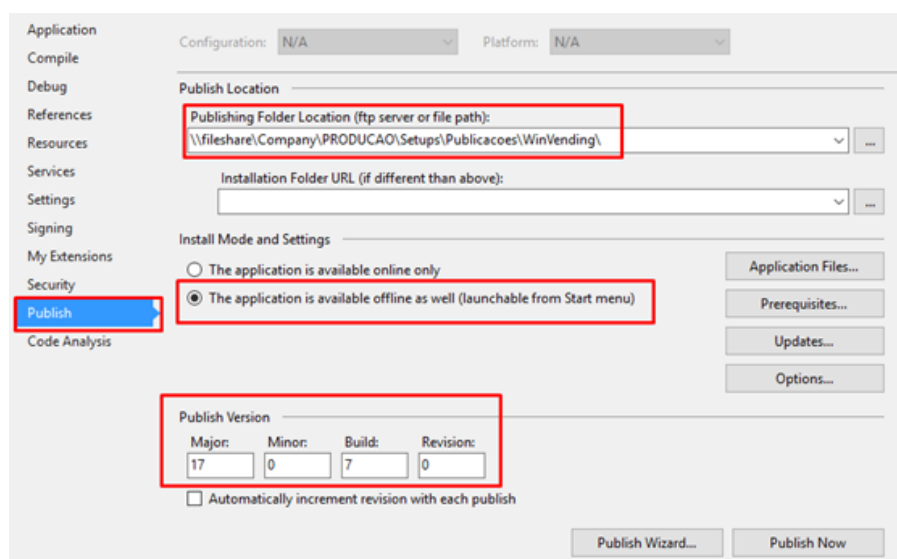


Figura 76 – Propriedades da publicação *ClickOnce*

dos requisitos de segurança, (iii) identificação das dependências e (iv) dados do publicador e da assinatura.

Defini-se, de seguida, a política de atualizações da aplicação. Visualiza-se, na Figura 77, a escolha do método de verificação da existência de atualizações, antes da sua execução.

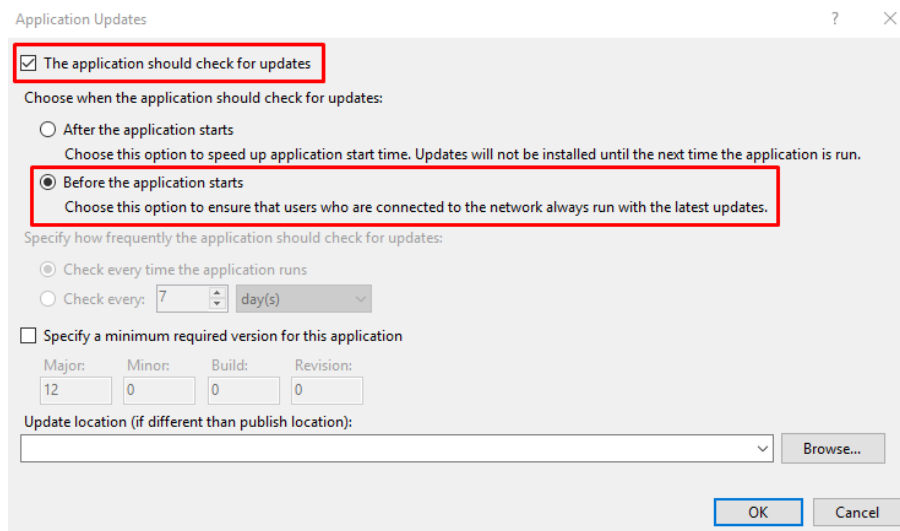


Figura 77 – Política de verificação de atualização do *ClickOnce*

Outras opções encontram-se disponíveis para configuração da publicação, como por exemplo: (i) os dados do publicador e do produto, (ii) o endereço de suporte e de erro e (iii) ativação da opção de criação de atalho, no ambiente de trabalho.

Efetua-se, então, a geração da publicação. Concluída a publicação, disponibilizam-se, automaticamente, os respetivos ficheiros, no diretório parametrizado. Neste ponto, a aplicação ainda não se encontra preparada para ser distribuída pelos vários clientes, isto porque, no manifesto, encontra-se atribuído o diretório de publicação, utilizado no acto da geração da publicação. Necessita-se, neste momento, de se proceder à redefinição dos diretórios de publicação, de acordo com a estrutura de diretórios de cada cliente. O próximo trecho de código mostra as instruções definidas num ficheiro *batch*<sup>150</sup> (e.g. “Assinar\_Cliente\_WV17\_0\_5\_0.bat”), necessárias para se redefinir a localização do diretório de publicações, de um dado cliente, e se proceder à posterior assinatura do manifesto.

```
Mage.exe -Update WinVending.application -AppManifest "Application
Files\WinVending_17_0_5_0\WinVending.exe.manifest" -ProviderURL
```

<sup>150</sup> Entende-se *batch* como um “ficheiro do sistema operativo *Windows* utilizado para a otimização das tarefas dos técnicos e programadores” (Batch 2017).

```
file://ts01/WinVending/Publicacoes/WinVending.application -CertFile  
vendingAbsInformatica.pfx -Password XPT0 -a sha256RSA
```

Finalmente, disponibiliza-se, no diretório de publicações do cliente, a nova versão da aplicação com o respetivo manifesto atualizado. A partir deste ponto, o utilizador, ao entrar na aplicação, recebe uma notificação de atualização que, após aceitar, origina a atualização automática da aplicação, na sua máquina e no seu utilizador em particular.

## 8.3 Gestão da Mudança

Considera-se a mudança um processo evolutivo necessário e positivo, que necessita de uma gestão apropriada, para que não impacte negativamente o sistema. Neste contexto, a gestão de mudança não implica prevenir ou evitar a mudança, mas sim identificar e gerir potenciais alterações que ocorram no sistema (Engholm 2013).

Nas próximas secções, descreve-se como a solução Vodafone WinVending se prepara para enfrentar o futuro e lidar com as mudanças. Esse processo guia-se pelo quinto e sexto princípios de David Hooker (“Estar aberto para o futuro” e “Planear com antecedência para reutilizar”, respetivamente).

### 8.3.1 Arquitetura da aplicação

A nível arquitetural, desenha-se a solução com o objetivo de agilizar o processo de mudança, com o mínimo esforço e prejuízo, na duração do projeto e no desempenho do sistema.

Uma característica arquitetural que demonstra essa preocupação, observa-se na construção de API's, como forma de abstrair a lógica de negócio e o acesso a dados da camada física de apresentação (e.g. aplicação *Windows Forms*). Tal implementação, permite, por exemplo, que se altere a abordagem de apresentação dos dados ao utilizador de uma aplicação *desktop*, para um ambiente Web.

Da observação dos clientes atuais, deteta-se uma maior abertura para alterarem de ERP e, alguns potenciais clientes, apresentam como requisito manter o seu ERP, que se verifica diferente do adotado nos atuais clientes. A pensar nesse cenário, a arquitetura da solução idealiza uma API que integra com múltiplos ERP's. Essa API implementa o padrão criacional *factory*, de modo que a adição de um novo ERP não tenha qualquer impacto para o negócio em causa.

A arquitetura das aplicações, nomeadamente a nível da camada de acesso a dados, implementa um mecanismo de prevenção de alteração de SGBD, ou de tecnologia de acesso a dados, por via, mais uma vez, do padrão *factory*. Ou seja, para uma situação em que o cliente tenha um SGBD diferente e não pretenda utilizar o SQL Server, constata-se suficiente a criação de um novo projeto dedicado a esse SGBD que, no limite, pode originar a alteração da tecnologia de acesso a dados.

De um ponto de vista geral, a arquitetura das aplicações encontra-se preparada para enfrentar alterações das funcionalidades existentes, ou para a criação de novas. Por um lado, na programação das aplicações, teve-se em consideração o segundo (“Segundo Princípio – Manter as coisas simples”) e o quarto (“Quarto Princípio – O que é produzido alguém consumirá”) princípios de David Hooker ao se efetuarem contínuos atos de *refactory*<sup>151</sup>, para tornar o código mais simples e, ao mesmo tempo, comentar o código e prever que alguém, ou o próprio programador, necessite de ler o código produzido. Por outro lado, a clara separação entre as camadas facilita a implementação de alteração e a criação de novas funcionalidades, através da reutilização das funcionalidades, anteriormente implementadas.

### **8.3.2 Atualização do modelo de dados**

Prevê-se a necessidade de atualização do modelo de dados decorrente de tentativas de melhoria no desempenho da base de dados (e.g. criação de índices), ou do levantamento de novos requisitos junto de clientes que, porventura, ocasionam a necessidade de manter o modelo de dados: (i) novas tabelas, (ii) novos campos, (iii) novas funções, entre outras operações a realizar junto do SGBD.

No processo de atualização do modelo de dados, correm-se riscos, pois, se durante o processo algo falhar, pode danificar-se ou corromper-se os dados da base de dados. Pela existência dessa possibilidade, recomenda-se que qualquer atualização ao modelo de dados seja antecedida de uma cópia de segurança da base de dados a atualizar.

---

<sup>151</sup> Considera-se *refactoring* como o processo de modificar um sistema de software para melhorar a estrutura interna do código, sem alterar seu comportamento externo (Refatoração 2017).

De modo a organizar, otimizar e gerir as alterações sobre o modelo de dados e com o auxílio da tecnologia de *Entity Framework Code First*, implementa-se o conceito de migração.

O conceito de migração adotado permite a definição de um ou vários ficheiros portadores de instruções SQL a executar, junto do SGBD do cliente, com o objetivo de atualizar o modelo de dados. Sempre que se atualiza o modelo de dados, numa determinada versão, cria-se um ou vários ficheiros (extensão “.sql”), que se distribuem juntamente com a aplicação e que se executam automaticamente, por ordem crescente de versão, desde a última versão instalada, exclusive, até à versão atual. Logo que detetado o sucesso da execução da migração, atualiza-se a tabela que gere as migrações com a indicação da última versão. Assim, evita-se que a migração seja, novamente, executada.

A escolha da estratégia de migração em que as suas instruções se colocam em ficheiros separados, de acesso aberto aos utilizadores, prende-se com o facto de, por vezes, poderem ocorrer falhas na execução da migração, devido a particularidades do sistema do cliente e ao aceder-se, diretamente, ao ficheiro da migração, torna-se mais ágil o diagnóstico da falha. Realça-se que a migração se efetua, apenas, no contexto da API que concentra, em si, a gestão do acesso aos dados.

Com a implementação da tecnologia de migrações, a base de dados evolui e as alterações estão devidamente controladas.

### **8.3.3 Atualização da aplicação**

Considera-se a atualização das aplicações uma tarefa habitual, no ciclo de vida *software* e assume diferentes propósitos: (i) correção de erros, (ii) melhorias e (iii) novas funcionalidades.

As atualizações visam a alteração de ficheiros e implementam-se por meio de uma nova distribuição da aplicação, que assume dois modos distintos: (i) *setup* ou (ii) publicação. O primeiro método utiliza-se em serviços ou em aplicações instaladas, tipicamente, num servidor ou em poucas máquinas; a segunda adota-se, preferencialmente, em aplicações com múltiplos utilizadores, onde se observa o ganho notório da sua utilização, pelo modo automático e ágil de funcionamento.

Posteriormente ao lançamento de uma nova atualização, verifica-se a importância de: (i) informar-se os utilizadores da disponibilização de uma nova versão e (ii) indicar-se quais as

alterações nela contidas. Com esse propósito e guiado pelo quarto princípio de David Hooker (“Quarto Princípio – O que é produzido alguém consumirá”), cria-se um ficheiro de histórico de alterações, onde se descrevem as alterações efetuadas e se disponibilizam ao cliente final. Nesse ficheiro, acumulam-se, por ordem decrescente de data, as diferentes versões publicadas e, para cada uma delas, enumeram-se os próximos elementos: (i) o número da versão, (ii) a data de publicação e (iii) as alterações realizadas por categoria (c.f. 11.3).

## 9 Avaliação do Sistema

Neste capítulo, identificam-se as grandezas utilizadas para a avaliação dos resultados de negócio da solução e das aplicações (Vodafone WinVending e WinGuard). Identificadas as métricas, passa-se à definição de uma metodologia para avaliar o seu valor. Com esse objetivo, utilizam-se: (i) testes de hipóteses e (ii) inquéritos.

### 9.1 Grandezas

Começa-se pela descrição das grandezas de avaliação do sistema, segundo diferentes visões: solução, utilização e suporte da aplicação. A primeira visão abrange a solução como um todo e direciona-se para a área de gestão e administração do cliente; a segunda é específica de cada uma das aplicações e mede-se junto dos utilizadores finais; a terceira, e última, refere-se ao suporte concedido a todas as aplicações e analisa-se junto da equipa responsável pela instalação e suporte das aplicações. Paralelamente, elucidam-se os métodos, técnicas e ferramentas utilizadas para maximizar os valores de cada métrica.

A quantificação de cada métrica é referida nos subtítulos: 9.3 através da realização de testes de hipóteses e, por meio de inquéritos, onde cada pergunta representa um valor para a métrica correspondente.

#### 9.1.1 Métricas da solução

Avalia-se o benefício da solução para o negócio da empresa, através do incremento dos ganhos face aos sacrifícios inculcados ao cliente. Demonstra-se a mais valia apresentada na proposta de valor da empresa, através de duas métricas: tempo de visita e nível de stock circulante nas

viaturas. Analisam-se estas métricas, com maior pormenor e detalhe, no capítulo “Testes de Hipóteses”.

### **9.1.2 Métricas de Utilização da Aplicação**

Procede-se à avaliação da utilização das aplicações, através das métricas: compreensão, estética, maturidade, segurança, tolerância a falhas, estabilidade, desempenho e escalabilidade.

A compreensão representa a facilidade com que o utilizador apreende as funcionalidades da aplicação e avalia se satisfazem as suas necessidades específicas (ISO/IEC 9126 2016). Os responsáveis pela análise e desenho das aplicações procuram disponibilizar todas as novas funcionalidades, em coerência com a organização geral da solução e, se necessário e pertinente, auxiliados por assistentes que os guiam durante a execução das tarefas.

A estética da interface com o utilizador envolve todas as características que atraíam os utilizadores do sistema, desde a adequação das informações prestadas, até aos pormenores visuais utilizados na interface gráfica (ISO/IEC 9126 2016). Os responsáveis pelo desenho preocupam-se com a organização das informações prestadas, das cores utilizadas, da expressividade dos ícones e imagens e dos textos e expressões apresentadas.

A segurança evidencia a capacidade da aplicação em evitar o acesso não autorizado, acidental ou deliberado, a programas e dados. Nesta área, as aplicações disponibilizam métodos de autenticação e autorização que garantem ao utilizador, ao entrar na aplicação, a veracidade da sua identidade (autenticação) e a permissão para aceder a um conjunto de recursos (autorização).

A maturidade é entendida como a capacidade de a aplicação evitar falhas decorrentes de defeitos nela detetados (ISO/IEC 9126 2016). A construção de testes de unidade e de integração garantem uma maior perceção e resolução de falhas.

A tolerância a falhas implica que a aplicação mantenha o funcionamento adequado mesmo quando ocorrem defeitos nela ou, nas interfaces externas associadas. Os programadores adotam a técnica de captura de exceções, de modo a evitar que falhas não controláveis afetem as suas funcionalidades. Por outro lado, os analistas idealizam estratégias alternativas para as situações de indisponibilidade do sistema. Por exemplo, no requisito de agendamentos

automáticos as com comunicações indisponíveis, aguardam um alarme de máquina ativa (máquina ou cabo ligado) para tentar novamente a comunicação.

A estabilidade é a característica da aplicação que permite evitar efeitos inesperados ocasionados por modificações. A construção de testes de integração proporciona a minimização desses efeitos.

O desempenho relaciona-se com o tempo de resposta das aplicações aos pedidos efetuados pelos utilizadores. Os analistas preveem a demora em determinadas opções e definem o carregamento parcelar dos dados; noutra perspetiva, os programadores aplicam o conceito de *threads*<sup>152</sup> para possibilitar o usufruto de outras funcionalidades, enquanto o sistema aguarda que todos os elementos sejam preenchidos.

A escalabilidade permite conhecer o comportamento da solução face à necessidade de crescimento do parque de máquinas do cliente, com consequências diretas no aumento de transações de comunicações entre o sistema e a máquina. A adoção da abordagem RESTfull, mais precisamente a sua característica de não manter o estado de sessão, na arquitetura dos serviços e a utilização da tecnologia de filas de mensagens garante a escalabilidade do sistema.

### **9.1.3 Métricas de Suporte da Aplicação**

A equipa de suporte é responsável pela instalação e manutenção das aplicações e é solicitada pelo cliente, em situações de instalação ou, ocorrência de falhas. Daí advém a importância de ouvir estes intervenientes e definir métricas que avaliem a capacidade de as aplicações responderem às suas solicitações. Para esse efeito, definem-se as seguintes métricas: análise de falhas, capacidade de instalação e coexistência com outros recursos.

A análise de falhas consiste na disponibilização de meios para detetar erros ocorridos no sistema com a finalidade de os compreender e resolver, num curto espaço de tempo. Nas

---

<sup>152</sup> *Thread* é uma forma de um processo do sistema operativo se dividir em duas ou mais tarefas que são executadas concorrentemente (*Threads* 2017).

aplicações desenvolvidas, utilizam-se técnicas de registo desses dados, nos seguintes repositórios: base de dados, ficheiros de texto e no gestor de eventos do sistema operativo.

A capacidade de instalação identifica a facilidade de instalar uma aplicação no sistema atual ou, num novo sistema. Os métodos detalhados no capítulo 8.2 agilizam a instalação das aplicações principais e dos respetivos componentes (API, monitor, entre outros).

Entende-se coexistência com outros recursos como a capacidade de instalar a aplicação num ambiente onde existem outras aplicações sem que, por esse motivo, falhem. Para melhorar esta métrica, os responsáveis pelo desenho da aplicação procuram minimizar as dependências utilizadas nas aplicações e, se estritamente necessárias, tentam movê-las para soluções mais centrais com menor número de instâncias ativas, como por exemplo, as API's.

## 9.2 Testes

Procura-se garantir a qualidade das soluções, através de um conjunto de testes, dos quais se destacam: (i) testes de unidade, (ii) testes de integração, (iii) testes de sistema e (iv) testes de aceitação.

Adota-se uma estratégia incremental de teste, começa-se com os testes unidades e avança-se para os testes de integração dos sistemas e culmina-se com testes que exercitam o sistema construído. Detalham-se, nas próximas secções, cada uma das classes de teste.

### 9.2.1 Testes Unidade

Os testes de unidade auxiliam no garante dos requisitos definidos e asseguram a maturidade da aplicação. O teste de unidade foca-se na lógica interna de processamento e na estrutura de dados, dentro dos limites de um componente. Testa-se a interface do módulo para garantir que a informação flui adequadamente para dentro e para fora do componente a ser testado (Pressman 2005).

Na solução, iniciam-se os testes nas API's para, posteriormente, se expandir para as aplicações de intereção com os utilizadores. De modo a testar o padrão de comunicação entre sistemas *Service Interface* concretizam-se testes com ênfase nos aspetos técnicos de implementação. Um caso de teste foca-se no retorno da informação da base de dados e outro dedica-se ao teste

da tradução dos objetos de negócio, em objetos de transferência de dados. O padrão *Service Interface* encapsula todos os detalhes de fornecimento do serviço e desacopla-os da lógica da aplicação. Tal separação permite a substituição da lógica de uma aplicação por uma implementação *mock*<sup>153</sup>. Este estilo de implementação substitui o código real por uma implementação fictícia que emula o código real. Utiliza-se a implementação de *mock* de modo a permitir a escrita de testes que verificam o funcionamento do código, sem depender do código atual da aplicação. Pode-se estender a implementação de *mock* para situações de erro que dificilmente se simula com o código real (Trowbridge et. Al. 2003).

Após a execução desses testes, confia-se que a informação obtida da base de dados e todo o processo de tradução até aos objetos de transferências de dados, funciona corretamente.

### 9.2.2 Testes de Integração

Os testes de integração incidem nas interligações dos diferentes componentes do sistema e mantêm a estabilidade da solução. Estes sucedem os testes de unidade. Embora individualmente, as unidades funcionem corretamente, quando juntas podem originar erros devido a problemas de compatibilidade ou algum outro motivo. Assim, os testes de integração, visam detetar se as unidades funcionam, adequadamente, em conjunto.

Consideram-se duas abordagens a aplicar nos testes de integração: (i) incremental e (ii) não incremental. Na abordagem incremental, agrupa-se o sistema em etapas, de modo a facilitar a descoberta de erros, uma vez que se encontra mais isolado. Esta abordagem pode concretizar-se de duas formas: ascendente e descendente. Na ascendente, testam-se primeiro os módulos de baixo nível durante a sua integração. Na descendente, integra-se em primeiro lugar os módulos de maior nível, enquanto se testam. Na abordagem não incremental agrupa-se o sistema de modo completo (Pressman 2005).

A realização de testes de integração deteta falhas que, talvez, passassem despercebidas durante os testes anteriores, uma vez que as unidades funcionavam corretamente.

---

<sup>153</sup> Entende-se por “*mock* um objeto que simula o comportamento de objetos reais de um modo controlado” (Object Mock 2017).

### 9.2.3 Testes de Sistema

Divide-se os testes de sistema em três grupos: segurança, carga e desempenho.

Teste de segurança simulam situações de ataque às aplicações e em caso de sucesso procede-se à devida análise e correção.

Os testes de carga e desempenho avaliam o desempenho da solução quando esta é submetida a um elevado número de pedidos e pretende-se, também, testar a capacidade de escalabilidade do sistema.

### 9.2.4 Testes de Aceitação

O comportamento do utilizador é imprevisível e a sua perceção dos dados pode não coincidir com a perspetiva de quem os idealizou. As instruções de utilização podem não ser corretamente interpretadas, combinações inesperadas de dados podem ser aplicadas e a apresentação da informação que parece clara para quem testa, pode ser ininteligível pelo utilizador final. Compensa-se essa característica imprevisível com testes de aceitação onde se procura envolver os utilizadores finais no processo de testes (Pressman 2005).

Uma vez que ambas as aplicações, estudadas no âmbito desta dissertação, são desenvolvidas para um leque diversificado de clientes, não é prático realizar testes de aceitação com cada cliente. Seleciona-se, então, os processos conhecidos por testes *alfa* e *beta* para descobrir erros apenas desvendáveis pelo utilizador que trabalha diariamente com a aplicação.

Os responsáveis do projeto selecionam um cliente alvo para a concretização dos testes. Realiza-se, em primeiro lugar, o teste *alfa* num ambiente natural e controlado onde os utilizadores finais operam as aplicações como se do seu quotidiano se tratasse, enquanto os desenvolvedores supervisionam o comportamento da aplicação e registam erros e problemas de utilização (Pressman 2005).

Quando o comportamento da solução se considera estável avança-se para o teste beta conduzido no ambiente real do cliente. Nestes testes, os desenvolvedores não se encontram presentes e os utilizadores finais não controlam o ambiente. O cliente regista os problemas detetados e comunica-os ao suporte para serem analisados pela equipa de desenvolvimento.

Como consequência dos problemas e falhas detetadas, nesta fase, são lançadas atualizações que incluem as modificações necessárias para o correto funcionamento do sistema.

### 9.3 Testes de Hipóteses

Pretende-se comparar, através de um caso de estudo, o modelo tradicional de visita acompanhado por Bluetooth, com a adoção da solução de telemetria. Por essa via, testam-se duas hipóteses: nível de stock circulante inferior a 15% e duração da visita por telemetria inferior a 10%, ambas, face a uma visita realizada com a tecnologia Bluetooth. Considera-se para a concretização dos testes uma população de 40500 máquinas (Informa D&B 2014).

Utiliza-se a ferramenta *Real Statistics*<sup>154</sup> para a execução de testes estatísticos não fornecidos nativamente pela ferramenta *Microsoft Excel*<sup>155</sup>.

#### 9.3.1 Nível de Stock Circulante

Compreende-se por nível de stock circulante de uma viatura, a quantidade de tabaco que uma viatura tem à saída da empresa, com a finalidade de abastecer as máquinas.

Recolhem-se os dados das amostras diretamente das bases de dados operacionais dos clientes. De modo a reduzir fatores exteriores ao âmbito da comparação, selecionam-se todas as máquinas eleitas para a incorporação da tecnologia de telemetria e que se encontrem no mesmo estabelecimento, cujo proprietário não tenha sido modificado, no período de tempo em análise. Os dados recolhidos representam dois períodos diferentes: antes (de julho a setembro de 2014) e depois (de julho a setembro de 2016) da adoção da componente de telemetria (c.f. 11.4). Note-se que os valores com a componente de telemetria incluem um stock de reserva para colmatar necessidades inesperadas. Realça-se que este valor tende a

---

<sup>154</sup> Considera-se *Real Statistics* como um suplemento estatístico do *Microsoft Excel*. Sítio do suplemento: <http://www.real-statistics.com>

<sup>155</sup> Entende-se o *Microsoft Excel* como uma folha de cálculo desenvolvida pela Microsoft com as seguintes características: execução de cálculos, ferramentas gráficas, tabelas dinâmicas e uma linguagem de programação macro.

diminuir com o aumento da confiança do cliente nos valores sugeridos pela solução e pela otimização do algoritmo de pré-cargas ao modelo de negócio do cliente.

Com a nova solução de telemetria, tenciona-se reduzir em pelo menos 15% o nível de stock circulante. Assim, define-se como hipótese nula a fórmula SCT (stock circulante telemetria) = SCB (stock circulante Bluetooth) \* 0.85 e como hipótese alternativa  $SCT > SCB * 0.85$ .

Considera-se, para efeitos de seleção do teste estatístico, que se está perante duas amostras emparelhadas, uma vez que o universo de máquinas a comparar é o mesmo, antes e depois da utilização da tecnologia da telemetria. Verifica-se a normalidade das amostras através da aplicação do teste de normalidade: *Shapiro-Wilk*. Confirma-se que a amostra, relativa aos valores antes da adoção da telemetria, não segue a distribuição normal (c.f. Figura 78).

1	Shapiro-Wilk Test		
2			
3		Group 1	Group 2
4	W	0.954166	0.971574
5	p-value	0.040751	0.235842
6	alpha	0.05	0.05
7	normal	no	yes

Figura 78 – Teste de normalidade das amostras de stock circulante (*Shapiro-Wilk*)

Como não se verificou a normalidade, numa das amostras, seleciona-se o teste *Wilcoxon* para amostras emparelhadas de modo a verificarem-se as hipóteses definidas. Após a execução deste teste confirma-se o valor da hipótese nula (c.f. Figura 79).

1	Wilcoxon Signed-Rank Test for Paired Samples		
2			
3		Antes Tel Com Telemetria	
4	median	2924	3095
5			
6	count	53	
7	# unequal	53	
8	T+	462	
9	T-	969	
10	T	462	
11			
12		one tail	two tail
13	alpha	0.05	
14	mean	715.5	
15	std dev	112.9591	
16	z-score	2.244176	
17	effect r	0.217974	
18	T-crit	529.6989	494.1043
19	p-value	0.012411	0.024821
20	sig (norm)	yes	yes

Figura 79 – Teste “*Wilcoxon*” para avaliar as hipóteses de stock circulante

### 9.3.2 Duração de Visita

Entende-se por duração de visita, o intervalo de tempo compreendido entre a chegada de um vendedor a um cliente até à sua saída. Durante esse intervalo, o vendedor realiza um conjunto de tarefas na máquina, relacionadas com o abastecimento e recolha de valores monetários.

De modo a estabelecer-se um comparativo entre o antes e o depois da implementação da nova solução, seleciona-se um novo cliente, como caso de teste, e solicita-se a recolha de dados antes da sua implementação. Esse levantamento implica as seguintes fases:

1. identificação das máquinas onde se pretende implementar a telemetria;
2. eleição das máquinas alvo da medição;
3. acompanhamento dos vendedores às máquinas, alvo do estudo, para efeitos de medição da duração das visitas.

Com a implementação do novo sistema, pretende-se provar a redução de, pelo menos, 10% no tempo de visita. Com essa finalidade, procura-se medir os novos tempos de visita. No entanto, e de modo a reduzir o impacto do fator de familiaridade com o novo sistema, opta-se por efetuar tais medições três meses após a implementação do novo sistema. Assim, nessa data, efetuam-se os mesmos passos realizados aquando da medição do anterior sistema.

O esforço realizado no levantamento da informação condiciona o número de elementos da amostragem e, conseqüentemente, a robustez dos resultados apresentados.

Define-se como hipótese nula a fórmula  $DVT$  (duração da visita com telemetria) =  $DVB$  (duração da visita com Bluetooth) \* 0.9 e como hipótese alternativa  $DVT > DVB * 0.9$ .

Considera-se, para efeitos de seleção do teste estatístico, que se está perante duas amostras emparelhadas, uma vez que o universo de máquinas a comparar é o mesmo, antes e depois da utilização da tecnologia da telemetria. Verifica-se a normalidade das amostras por meio de um teste de normalidade em estatísticas de frequência: *Shapiro-Wilk*. Caso se confirme a normalidade, adotam-se os testes *t* para amostras emparelhadas, no caso contrário, seleciona-se o teste *Wilcoxon* para amostras emparelhadas (Gomes 2016).

## **9.4 Inquéritos**

Considera-se as especificidades funcionais dos diversos intervenientes para a elaboração de questionários que medem o grau de satisfação dos mesmos, relativamente à nova solução.

Tenciona-se elaborar um conjunto de questões, a colocar no inquérito de cada solução, com o objetivo de conhecer as opções dos utilizadores.

Cada questão do inquérito está associada a uma métrica: compreensão, estética da interface com o utilizador, segurança, maturidade, tolerância a falhas, estabilidade, desempenho, análise de falhas, capacidade de instalação e coexistência com outros recursos.

## 10 Conclusões

Neste capítulo, responde-se às questões colocadas na introdução, referem-se as limitações da solução e os respetivos métodos para as superar, enumeram-se novas funcionalidades a implementar e termina-se com uma apreciação geral do trabalho realizado.

### 10.1 Objetivos Realizados

Neste trabalho, discute-se a realidade da empresa ABS Informática que se confronta com o desafio de incluir uma componente de telemetria na sua solução: WinVending. Percebe-se, desde logo, a importância e pertinência de inovar o produto, como forma de manter a satisfação dos seus clientes e atrair novos.

Envolta num ambiente externo repleto de oportunidades, a ABS Informática, mostra-se motivada e encorajada a abraçá-las e incluí-las no seu produto. Por um lado, a tecnologia, M2M encontra-se num estado de evolução maduro e fiável. Por outro, a Vodafone age como um agente facilitador tecnológico ao proporcionar a comunicação remota com dispositivos a uma velocidade, estabilidade e custos que permitem a sua implementação.

Da análise e observação do mercado detetam-se algumas ameaças que necessitam de uma avaliação e conhecimento, das quais se destacam: (i) a concorrência aposta na mesma área de inovação tecnológica, (ii) as políticas e leis governamentais colocam em causa a venda de tabaco em máquinas de venda automática e (iii) atos de roubo e vandalismo a máquinas e carrinhas transportadoras de tabaco que causam prejuízos e transtornos na gestão do negócio. A ABS Informática encara a última ameaça como uma oportunidade para melhorar o seu produto e servir melhor o seu cliente.

De modo a aferir-se o valor percebido pelo cliente, realiza-se um levantamento dos benefícios e prejuízos dos clientes-alvo. De um ponto de vista geral, consideram-se os seguintes benefícios da incorporação da componente de telemetria: diminuição de custos, poupança de tempo, diminuição de risco, maior segurança, melhor ambiente, mais conforto, apoio à decisão e pioneirismo. Relativamente aos sacrifícios, dividem-se em dois tipos: (i) monetários e (ii) não monetários. Dos monetários distinguem-se: mensalidade da máquina, licença da aplicação por utilizador, licença móvel por dispositivo, módulo da máquina, custo da reparação e serviço de consultadoria. Nos não monetários realçam-se: o tempo despendido na instalação / reparação do módulo e na formação aos utilizadores, para a montagem do módulo.

Note-se a existência de um fator de risco inerente à aquisição desta solução, que se deve à incerteza de se efetuar a escolha certa e do custo oneroso que advém da escolha errada.

Do confronto entre os benéficos e os sacrificios, em conjunto com a ponderação dos eventuais riscos, o cliente percebe o valor da solução.

O conhecimento das necessidades e interesses do cliente permite o avanço para o estudo da proposta de valor que culmina na sua criação, agora mencionada. Vodafone WinVending é uma solução de gestão e monitorização de máquinas de venda automática de tabaco, à distância e em tempo real. Permite aos depositários de tabaco: reduzir até 50% o número de quilómetros percorridos, diminuir o stock nas viaturas em 20% e detetar atos de roubo ou vandalismo. É a primeira solução do mercado nacional a aliar, a todas estas características, a capacidade de integração com o ERP do cliente.

Convicta do valor da telemetria para a gestão inovadora de uma empresa gestora de máquinas de tabaco, a ABS Informática estabelece com a Vodafone, reconhecida operadora de telecomunicações e a KTS, produtora de dispositivos eletrónicos, um acordo de colaboração com o objetivo de reunir o conhecimento especializado de cada parceiro, na respetiva área de atuação.

A incorporação das funcionalidades de telemetria, na atual solução de Vending, coloca em aberto um conjunto de possíveis cenários: (i) implementar as novas características no ambiente atual, ou numa nova solução; (ii) alojar os dados localmente, ou na nuvem; (iii) desenvolver a nova solução na web, ou em desktop; (iv) comunicar os alarmes das máquinas por SMS, ou através de uma aplicação web centralizada, ou por meio de uma aplicação *desktop*.

Apresenta-se uma solução para cada cenário, após se ponderar os prós e contras das suas alternativas. Decide-se enveredar pela construção de uma nova solução e aproveita-se a oportunidade para: (i) melhorar a interface gráfica e interação com o utilizador, (ii) remodelar o modelo de dados e (iii) criar uma arquitetura para a solução, de modo a facilitar e enfrentar a mudança. Opta-se por alojar os dados por cliente, em servidores controlados pelos seus administradores de sistemas. Tal decisão, deve-se à posição defendida pelos principais clientes da empresa que se manifestam, perentória e notoriamente, contra a possibilidade dos seus dados estarem na nuvem. Prefere-se o ambiente *desktop*, após ponderação dos fatores: (i) tempo, (ii) experiência da equipa de programação e (iii) reutilização de componentes já existentes. Elege-se o desenvolvimento de uma solução específica e disponibilizada a uma ou mais centrais de alarmes, para ser possível consultar, analisar e atuar face à ocorrência de alarmes que indiquem situações de furtos, ou vandalismos.

Da ponderação dos cenários equacionados e das respetivas decisões, nascem as soluções: (i) WinVending e (ii) WinGuard, objeto de estudo nesta dissertação.

Ao decidir-se pela construção de uma nova solução, aproveita-se a oportunidade para se redefinir as ferramentas utilizadas na análise, prototipagem e implementação da mesma. Assim, opta-se: (i) pelo IDE Visual Studio e as respetivas linguagens de programação .NET (C# e VB.NET), (ii) pelo SGBD SQL Server, (iii) pela ferramenta de prototipagem Indigno Studio Lite e (iv) pela linguagem de modelação, orientada a objetos UML, Visual Paradigm.

Segue-se a descrição de todo o aspeto funcional dos sistemas WinVending e WinGuard, guiado pelo “F” do acrónimo FURPS+. Contextualizam-se as partes interessadas e o seu contributo na definição dos requisitos e concretiza-se a apresentação dos requisitos, através de diagramas UML (diagrama de casos de uso, diagramas de estados e diagramas de sequência).

Recorde-se que o levantamento de requisitos da nova solução tem dois objetivos: (i) implementar a nova componente de telemetria e (ii) migrar as funcionalidades essenciais da solução legada. Da conjugação dos fatores (i) feedback do cliente final, (ii) características a migrar da aplicação WinVending legada e (iii) potencial oferecido pelas soluções da parceria (cartão SIM, módulo e API do Middleware) resulta um conjunto de necessidades a incorporar na nova solução. Posteriormente, aos testes realizados à componente de telemetria e ao confirmar-se o seu correto funcionamento e potencial, procede-se ao levantamento de requisitos da solução de notificação e registo de alarmes, denominada: WinGuard.

Como complemento aos requisitos funcionais, agrega-se a verificação dos requisitos não funcionais dos sistemas que, segundo a classificação FURPS+, engloba as categorias: (i) usabilidade (carácter “U”), (ii) confiabilidade (carácter “R”), (iii) desempenho (carácter “P”), (iv) suporte (carácter “S”) e (v) outras restrições específicas de cada sistema (carácter “+”).

Conhecidos os requisitos das aplicações a desenvolver, apresenta-se uma visão geral do sistema e o modo como se estabelece a comunicação entre os seus sub-sistemas (Middleware, WinVending, WinVending Mobile e WinGuard). Detalha-se o funcionamento interno dos sistemas WinVending e WinGuard, bem como, as suas relações com os sistemas: Middleware, Mobile, Administrador e ERP’s. Representa-se esse propósito com o auxílio de diagramas de componentes, onde se apresentam os principais componentes do sistema e as interfaces que disponibilizam e consomem. Avança-se para a ilustração do modelo conceptual de classes e finaliza-se com a descrição do modelo de dados utilizado.

A decisão da arquitetura da aplicação passa pela adoção do estilo arquitetural em camadas, mais concretamente o padrão três camadas (apresentação, lógica de negócio e acesso a dados), onde as camadas comunicam entre si, através do padrão: objeto de negócio.

A comunicação com sistemas externos estabelece-se pela implementação dos padrões: (i) *Service Interface* e (ii) *Service Gateway*. O primeiro padrão define e implementa o padrão de objeto de transferência de dados, que deve ser reproduzido pelo consumidor. O segundo padrão encapsula o código de comunicação com os serviços externos e traduz os objetos de negócio em objetos de transferência de dados.

Define-se um procedimento de gestão de transações contido na camada de regras de negócio.

Na camada de acesso a dados, implementam-se os seguintes padrões: (i) *Repository* e (ii) *Data Mapper*. O primeiro executa duas ações: (i) questiona a fonte de dados para a obtenção dos dados para, de seguida, os mapearem em entidades de negócio e (ii) persiste as alterações das entidades de negócio, na base de dados. Socorre-se de um ORM para abstrair a comunicação entre a aplicação e a base de dados. O padrão *Data Mapper* utiliza-se como uma camada de mapeamento entre os objetos em memória e a base de dados.

Definida a arquitetura interna das aplicações e o modo como interagem com o exterior, prossegue-se para a escolha das tecnologias que a suportam. Assim: (i) escolhe-se o estilo arquitetural REST como tecnologia de serviços de Internet, (ii) como ferramenta relacional de

objetos (ORM) prefere-se a Entity Framework na abordagem *Code First*, (iii) como ferramenta de registo elege-se o Log4Net, (iv) nomeia-se a tecnologia MSMQ para gerir filas de mensagens e (v) opta-se por não utilizar qualquer ferramenta de integração empresarial (EAI).

O sistema estudado compõe-se por um conjunto de aplicações com funcionalidades específicas que complementam as funcionalidades de outras aplicações, de modo a formar um conjunto comum e coeso. Constata-se que os sub-sistemas comunicam, maioritariamente, por via de serviços REST.

Por forma a conceder uma melhor experiência de utilização, recorre-se ao controlo GMap .NET para representar a localização da máquina, no mapa da Google, imbutido nas aplicações.

Concluído o desenvolvimento dos componentes do sistema, torna-se necessário disponibilizá-los aos clientes. Com esse propósito, adotam-se duas estratégias diferentes de instalação: (i) por setup (InstallShield) e (ii) por publicação (ClickOnce). Escolhe-se, a primeira estratégia, nos próximos cenários: (i) projetos WinVending e WinGuard, em situações de utilização de múltiplos utilizadores e (ii) projetos de serviços. Utiliza-se, a segunda estratégia, na aplicação WinVending, pelo seu cariz multi-utilizador.

Realça-se o modo como o sistema se prepara para lidar com a mudança, em diferentes níveis: (i) arquitetura, (ii) modelo de dados e (iii) atualização automática da aplicação.

Procura-se garantir a qualidade das aplicações, através de um conjunto de testes, dos quais se destacam: (i) testes de unidade, (ii) testes de integração, (iii) testes de sistema e (iv) testes de aceitação. Embora idealizada, a sua implementação não se encontra implementada.

Finalmente e de modo a avaliar o valor da solução, utilizam-se: (i) testes de hipóteses e (iii) inquéritos. Os primeiros carecem de maiores amostras e os segundos necessitam de desenvolvimento e submissão aos respetivos interessados.

## **10.2 Limitações e Trabalho Futuro**

Concluído o trabalho e observado numa perspectiva autocrítica construtiva, constatam-se as seguintes limitações e potenciais necessidades de desenvolvimento futuro.

Em relação ao desenho da aplicação e, no que diz respeito, à comunicação entre a API do WinVending e o Middleware, nota-se a necessidade de implementar o padrão *factory*, de modo

a possibilitar a invocação de outros sistemas *middleware* (e.g. comunicação com o *middleware* do fabricante GM), possivelmente, em conjunto com o padrão *adapter*<sup>156</sup>, caso se verifique a necessidade de simplificar as *interfaces* entre os sistemas.

No âmbito da seleção dos padrões de integração, deve-se ponderar alargar a implementação do padrão de mensagens para as comunicações entre: (i) os sistemas *Middleware* e *WinVending* e (ii) entre os sistemas *WinVending* e *WinGuard*. Tal implementação tem como benefícios: (i) aumentar a disponibilidade do sistema e (ii) aumentar a rapidez das comunicações.

A complexidade, as nuances técnicas (e.g. limitação da utilização de campos booleanos) e o crescimento dos serviços dedicados ao *WinVending Mobile* implicam a reconsideração do desenvolvimento de uma API dedicada ao sistema móvel, de modo a abstrair as especificidades da plataforma móvel da API do *WinVending*.

Relacionado com a implementação das aplicações, mais concretamente, com a efetivação de comunicações e programações, sugere-se uma abordagem orientada a mensagens para a sua implementação. No caso particular das comunicações, realça-se a importância de avaliar a utilização de filas com prioridades ou múltiplas filas, consoante a prioridade das comunicações.

Sugere-se a adoção do padrão *factory* na utilização do controlo *GMap .NET*, de modo a que cada cliente possa escolher qual o fornecedor de mapas que prefere utilizar (e.g. mapas Bing).

Atualmente, o método adotado de notificação de alarmes, utiliza a técnica de *thread* para efetuar pedidos periódicos à API, com o objetivo de obter novos alarmes. Considera-se esse método passível de melhoramento ao nível de: (i) recursos consumidos e (ii) desempenho. Para colmatar esses défices, aconselha-se a utilização da tecnologia *SignalR*<sup>157</sup> por forma a: (i) diminuir o número de pedidos à base de dados e (ii) aumentar a rapidez de disponibilização de novos alarmes.

---

<sup>156</sup> “O padrão *Adapter* converte uma interface de uma classe para outra interface que o cliente espera encontrar. O Adaptador permite que classes com interfaces incompatíveis trabalhem juntas.” (*Adapter* 2017)

<sup>157</sup> Conhece-se *SignalR* como “uma tecnologia Microsoft de construção de aplicações em tempo real, onde os clientes podem colocar dados no servidor e o servidor pode enviar dados para todos os clientes. Funciona com a nova API HTML5 WebSockets que permite a comunicação bidirecional entre clientes (e.g. aplicações *desktop* e Web) e servidor” (tradução nossa) (Deysel 2014).

Relativamente à avaliação do sistema, considera-se fulcral a implementação da abordagem sugerida no capítulo de testes, com auxílio de ferramentas de testes (e.g. xUnit.net<sup>158</sup> com NSubstitute<sup>159</sup>). Julgam-se as hipóteses levantadas como pertinentes para a análise e prova do valor da solução, no entanto, as amostras temporais recolhidas verificam-se reduzidas, bem como, o número de clientes submetidos ao teste. A idealização dos questionários deve materializar-se e submeter-se aos respetivos interessados, de modo a aferir o seu grau de satisfação.

Como novas funcionalidades / requisitos de negócio da solução consideram-se:

- migração total da solução legada;
- implementação de um módulo de gestão de avarias;
- gestão do plano de rota com diferentes visões, no formato de calendário;
- sugestão de planos de rota;
- apresentação de gráficos e análises, no painel de bordo;
- integrar com outros sistemas middleware (e.g. fabricante de máquinas GM);
- integração com aplicações de gestão de frotas.

Além das especificidades do negócio de Vending, tenciona-se implementar as seguintes funcionalidades gerais:

- conceder-se a opção de multi-idioma para potenciar a internacionalização;
- controlar-se licenças de acesso com base no número de utilizadores e máquinas;
- enviar-se, automaticamente, relatórios aos respetivos interessados (e.g. lista de máquinas com falhas de comunicação ou avarias);
- criar-se o manual da aplicação e sugestões / ligações de ajudas contextualizadas à navegação efetuada (e.g. janela ou campo selecionado);
- consultar-se diretamente, na aplicação, o histórico de versões.

---

<sup>158</sup> Denomina-se “xUnit.net como a ferramenta de teste de unidade livre, de código aberto, focada na comunidade para o .NET Framework. Escrito pelo inventor original da NUnit v2, o xUnit.net considera-se uma tecnologia para testar unidades C #, F #, VB.NET” (tradução nossa) (xUnit.net 2017).

<sup>159</sup> Considera-se NSubstitute como uma biblioteca *mocking* para a plataforma .NET.

### **10.3Apreciação Final**

Consideram-se cumpridos os objetivos idealizados para este trabalho. Não obstante, a solução continua a ser aperfeiçoada e acrescida de novas funcionalidades.

No decorrer deste trabalho e de acordo com o terceiro princípio de David Hooker (“Terceiro Princípio – Manter a visão”), assegura-se o foco no objetivo de concretizar a implementação da componente de telemetria, na solução WinVending, de um modo claro, sem ambiguidades e com integridade conceptual. Atribuíram-se responsabilidades, ao autor deste escrito, que lhe permitem manter uma visão e garantir o seu cumprimento.

A concretização desta solução só se torna possível com o trabalho e cooperação das três entidades envolvidas: ABS Informática, Vodafone e KTS. No resultado deste trabalho subjaz a máxima de Aristóteles: “o todo é maior do que a simples soma das suas partes”.

A solução construída afigura-se recheada de futuras hipóteses de crescimento, algumas, enunciadas na secção 10.2 e que se acredita motivo para novas reflexões.

Crê-se ter proposto um sentido com o presente escrito, pelo menos aquele que sustenta o percurso desta tarefa: comunicar, colaborar, cooperar, ouvir, observar, pesquisar, refletir, planejar, construir e pensar (sétimo e último princípio de David Hooker).

# Referências

- [ABS Informática 2017] ABS Informática 2017, Empresa. Disponível em <<http://absinformatica.pt/pt/go/empresa2>>. [11 de fevereiro de 2016]
- [ABS Informática 2014] ABS Informática 2014, WinVending Telemetria. [8 de fevereiro de 2016]
- [Active Directory 2017] 'Art Wikipedia', Active Directory 2017, (artigo wiki), 18 de outubro de 2017. Disponível em: <[https://pt.wikipedia.org/wiki/Active\\_Directory](https://pt.wikipedia.org/wiki/Active_Directory)>. [18 de outubro de 2017]
- [Adapter 2017] 'Art Wikipedia', Adapter 2017, (artigo wiki), 22 de junho de 2017. Disponível em: <<https://pt.wikipedia.org/wiki/Adapter>>. [21 de outubro de 2017]
- [ADO.NET 2017] 'Art Wikipedia', ADO.NET 2017, (artigo wiki), 27 de agosto de 2017. Disponível em: <<https://en.wikipedia.org/wiki/ADO.NET>>. [28 de março de 2017]
- [Allee 2000] Allee, Verna 2000, 'Reconfiguring the Value Network', International Small Business Journal, vol. 21, N 4. Disponível em: <<http://www.sveiby.com/articles/Allee-ValueNets.htm>>. [20 de fevereiro 2017]
- [Allen 2011] Allen, Jonathan 2011, Message Queuing Options for .NET. Disponível em: <<https://www.infoq.com/news/2011/04/Message-Queues>>. [10 de setembro de 2017]
- [Análise de Cenários 2014] Análise de Cenários 2014 (ficheiro de vídeo). Disponível em: <[https://www.youtube.com/watch?v=WFwr\\_sq2jY](https://www.youtube.com/watch?v=WFwr_sq2jY)>. [19 de fevereiro de 2016]
- [Apache Software Foundation 2017] Apache Software Foundation 2017, What is Apache log4net. Disponível em: <<https://logging.apache.org/log4net/index.html>>. [10 de setembro de 2017]
- [Aqeel 2015] Aqeel, Kashif, "Battle of the Integration Plataforms (Biztalk Vs. MuleSoft)". [5 de agosto de 2015]
- [ASH 2011] ASH 2011, ASH Briefing: Tobacco Vending Machines. [28 de setembro de 2015]
- [Azevedo, Carvalho & Silva 2016] Azevedo, Guilherme, Carvalho, Henrique Fernandes & Silva, Jorge Ferreira da, A Teoria dos Jogos na Estratégia de Negócios: uma Contribuição Relevante?. [18 de fevereiro de 2016]
- [Batch 2017] 'Art Wikipedia', Batch 2017, (artigo wiki), 7 de outubro de 2017. Disponível em: <<https://pt.wikipedia.org/wiki/Batch>>. [19 de outubro de 2017]
- [Bluetooth 2017] 'Art Wikipedia', Bluetooth 2017, (artigo wiki), 25 de maio de 2017. Disponível em: <<https://pt.wikipedia.org/wiki/Bluetooth>>. [3 de junho de 2017]
- [Brooks 1995] Brooks, Frederick P. 1995, The Mythical Man-Month, Anniversary Edition: Essays On Software Engineering, Edition 2, Addison-Wesley, United States.
- [Byars 2013] Byars, Brandon, "Enterprise Integration Using REST". [18 de novembro de 2013]
- [Canvas da Proposta de Valor 2015] Canvas da Proposta de Valor 2015, Você conhece o Canvas da Proposta de Valor?, 11 de Agosto de 2015. Disponível em: <<http://www.inovativabrasil.com.br/canvas-proposta-de-valor/>>. [20-02-16]
- [Code Generation and T4 Text Templates 2017] 'Art Microsoft Developer Network', Code Generation and T4 Text Templates. Disponível em: <<https://msdn.microsoft.com/en-us/library/bb126445.aspx>>. [14 de outubro de 2017]
- [Comma-separated values 2017] 'Art Wikipedia', Comma-separated values 2017, (artigo wiki), 22 de maio de 2017. Disponível em: <[https://en.wikipedia.org/wiki/Comma-separated\\_values](https://en.wikipedia.org/wiki/Comma-separated_values)>. [19 de junho de 2017]
- [Conectividade M2M 2016] Conectividade M2M 2016. Disponível em: <<http://negocios.vodafone.pt/aplicacoes-empresariais/solucoes-m2m/conectividade.html>>. [16 de fevereiro de 2016]

- [Costa 2010] Costa, João Matriz da 2010, 'Como Analisar a Concorrência', Portal Gestão 2 de abril. Disponível em: <<https://www.portal-gestao.com/artigos/2630-como-analisar-a-concorr%C3%83%C2%AAncia.html>>. [10 de fevereiro de 2016]
- [CRUD 2017] 'Art Wikipedia', CRUD 2017, (artigo wiki), 28 de setembro de 2017. Disponível em: <<https://pt.wikipedia.org/wiki/CRUD>>. [12 de outubro de 2017]
- [Cruz 2011] Cruz, Carlos Pereira da 2011, 'Qual é a proposta de valor?', Redsigma – Sistemas de Gestão. Disponível em: <<http://pt.slideshare.net/ccruz1234/qual-a-proposta-de-valor>>. [2 de fevereiro de 2016]
- [Dejan 2010] Dejan, Bosanac, Apache "ActiveMQ – Enterprise messaging in action". [5 de julho de 2010]
- [Delphi 2017] Delphi 2017, Delphi Overview. Disponível em: <<https://www.embarcadero.com/products/delphi>>. [1 de junho de 2017]
- [Delphi 2017] 'Art Wikipedia', Delphi 2017, (artigo wiki), 7 de junho de 2017. Disponível em: <[https://en.wikipedia.org/wiki/Delphi\\_\(programming\\_language\)#Embarcadero\\_Delphi\\_XE7](https://en.wikipedia.org/wiki/Delphi_(programming_language)#Embarcadero_Delphi_XE7)>. [11 de junho de 2017]
- [Deysel 2014] Deysel, Dieter, "Implementing SignalR in Desktop Applications". [6 de agosto de 2014]
- [Dynamics NAV 2017] Dynamics NAV, Nav-Overview. Disponível em: <<https://www.microsoft.com/pt-pt/dynamics365/nav-overview>>. [4 de junho de 2017]
- [EDI 2017] 'Art Wikipedia', EDI 2017, (artigo wiki), 13 de março de 2017. Disponível em: <<https://pt.wikipedia.org/wiki/EDI>>. [9 de outubro de 2017]
- [EDIFACT 2017] 'Art Wikipedia', EDIFACT 2017, (artigo wiki), 9 de outubro de 2014. Disponível em: <<https://pt.wikipedia.org/wiki/EDIFACT>>. [9 de outubro de 2017]
- [Eeles 2004] Eeles, Peter 2004, What, no supplementary specification, IBM DeveloperWorks. Disponível em: <<https://www.ibm.com/developerworks/rational/library/3975-pdf.pdf>>. [17 de julho de 2017]
- [Embarcadero 2015] Embarcadero 2015, BDE Overview. Disponível em: <[http://docwiki.embarcadero.com/RADStudio/Tokyo/en/BDE\\_Overview](http://docwiki.embarcadero.com/RADStudio/Tokyo/en/BDE_Overview)>. [11 de junho de 2016]
- [Embarcadero 2016] 'Art Wikipedia', Embarcadero 2016, (artigo wiki), 26 de janeiro de 2015. Disponível em: <[https://pt.wikipedia.org/wiki/Embarcadero\\_Technologies](https://pt.wikipedia.org/wiki/Embarcadero_Technologies)>. [11 de junho de 2017]
- [Engenharia reversa 2017] 'Art Wikipedia', Engenharia reversa 2017, (artigo wiki), 12 de abril de 2017. Disponível em: <[https://pt.wikipedia.org/wiki/Engenharia\\_reversa](https://pt.wikipedia.org/wiki/Engenharia_reversa)>. [14 de outubro de 2017]
- [Engholm 2013] Engholm, Helio, Gerenciamento e Controle de Mudanças. [17 de outubro de 2013]
- [Enterprise application integration 2017] 'Art Wikipedia', Enterprise application integration 2017, (artigo wiki), 13 de julho de 2017. Disponível em: <[https://en.wikipedia.org/wiki/Enterprise\\_application\\_integration](https://en.wikipedia.org/wiki/Enterprise_application_integration)>. [9 de outubro de 2017]
- [Enterprise Library 2013] 'Art Microsoft', Enterprise Library, abril de 2013. Disponível em: <<https://msdn.microsoft.com/library/cc467894.aspx>>. [10 de setembro de 2017]
- [Enterprise Service Bus 2017] 'Art Wikipedia', Enterprise Service Bus 2017, (artigo wiki), 13 de janeiro de 2016. Disponível em: <[https://pt.wikipedia.org/wiki/Enterprise\\_Service\\_Bus](https://pt.wikipedia.org/wiki/Enterprise_Service_Bus)>. [8 de outubro de 2017]
- [Espaço de nomes 2017] 'Art Wikipedia', Espaço de nomes 2017, (artigo wiki), 9 de janeiro de 2017. Disponível em: <[https://pt.wikipedia.org/wiki/Espa%C3%A7o\\_de\\_nomes](https://pt.wikipedia.org/wiki/Espa%C3%A7o_de_nomes)>. [19 de outubro de 2017]
- [Eticadata 2017] Eticadata 2017, Início. Disponível em: <<https://pt.eticadata.com/>>. [3 de junho de 2017]
- [EVA DTS 2015] EVA DTS 2015. Disponível em: <[http://www.vending-europe.eu/en/standards\\_protocols/eva-dts.html](http://www.vending-europe.eu/en/standards_protocols/eva-dts.html)>. [18 de fevereiro de 2016]
- [Factory 2017] 'Art Wikipedia', Factory (object-oriented programming) 2017, (artigo wiki), 19 de julho de 2017. Disponível em: <[https://en.wikipedia.org/wiki/Factory\\_\(object-oriented\\_programming\)](https://en.wikipedia.org/wiki/Factory_(object-oriented_programming))>. [13 de outubro de 2017]

- [Ferreira 2010] Ferreira, Henrique Santos, Um estudo sobre métricas para o fuzzy front end e as suas contribuições para o processo de inovação, Universidade Federal de Pernambuco.
- [FireDAC 2015] FireDAC 2015, Overview (FireDAC), 13 de maio de 2015, Embarcadero. Disponível em: <[http://docwiki.embarcadero.com/RADStudio/Seattle/en/Overview\\_\(FireDAC\)](http://docwiki.embarcadero.com/RADStudio/Seattle/en/Overview_(FireDAC))>. [11 de junho de 2017]
- [Firmware 2015] 'Art Wikipedia', Firmware, (artigo wiki), 30 de setembro de 2015. Disponível em: <<https://pt.wikipedia.org/wiki/Firmware>>. [4 de fevereiro de 2016]
- [Fowler 2003] Fowler, Martin, "Patterns Of Enterprise Application Architecture". [2003]
- [Geography 2017] 'Art Microsoft', Spatial Types - geography. Disponível em: <<https://docs.microsoft.com/en-us/sql/t-sql/spatial-geography/spatial-types-geography>>. [22 de julho de 2017]
- [Getronics 2003] Getronics 2013, Software Requirements Specification. [21 de fevereiro de 2003]
- [Global8Track 2015] Global8Track 2015, Sobre Nós. Disponível em: <[http://www.global8track.com/?page\\_id=4415](http://www.global8track.com/?page_id=4415)>. [2 de julho de 2017]
- [GMBOS 2017] GMBOS 2017, GM BOS - LA TECNOLOGÍA AL SERVICIO DE LOS PROFESIONALES. Disponível em: <<http://www.gmvending.com/gm-vending-/productos/maquinas-de-tabaco/sistema-telemetria-gmbos.html>>. [5 de junho de 2017]
- [Gomes 2016] Gomes, Elsa Ferreira 2016, Experimentação e Avaliação, Instituto Superior de Engenharia do Porto. [12 de fevereiro de 2017]
- [Gurock Software GmbH 2017] Gurock Software GmbH 2017, Comparison of .NET Logging Frameworks and Libraries. Disponível em: <<http://www.dotnetlogging.com/comparison/>>. [28 de agosto de 2017]
- [Heinkele & Pousttchi 2004] Heinkele, Christian & Pousttchi, Key 2004, Mobile Business in the Vending Industry, University of Augsburg.
- [Hohpe & Woolf 2004] Hohpe, Gregor, Woolf, Bobby, "Enterprise Integration Patterns". [2004]
- [Hooker 1996] Hooker, David, Seven Principles of Software Development, 1996. Disponível em: <<http://c2.com/cgi/wiki?SevenPrinciplesOfSoftwareDevelopment>>. [7 de fevereiro de 2016]
- [Hubbard et al. 2017] Hubbard, Jennifer, Latham, Luke, Blome, Mike, Wenzel, Maira 2017, Linq to SQL. Disponível em: <<https://docs.microsoft.com/en-us/dotnet/framework/data/adonet/sql/linq/>>. [3 de março de 2017]
- [Infopédia 2017] Infopédia 2017, Horeca. Disponível em: <<https://www.infopedia.pt/dicionarios/lingua-portuguesa/HORECA>>. [18 de junho de 2017]
- [Informa D&B 2014] Informa D&B 2014, Setores de estudo da DBK Portugal – Vending. Disponível em: <<https://www.informadb.pt/idbweb/resourcesRepository/setoresportugal2014/jun-vending.pdf>>. [3 de fevereiro de 2016]
- [Informa D&B 2015] Informa D&B 2015, Setores de estudo da DBK – Vending. Disponível em: <<https://www.informadb.pt/idbweb/resourcesRepository/setores-espanha2015/jun-vending.pdf>>. [4 de fevereiro de 2016]
- [Infragistics 2017] Infragistics 2017, About Us. Disponível em: <<https://www.infragistics.com/about-us>>. [17 de julho de 2017]
- [Intelligent Vending Machines 2015] Intelligent Vending Machines 2015, 'Rising Popularity of Wireless Vending Machines Drives the Global Intelligent Machines Market, According to a New Report by Global Industry Analysts, Inc.', Global Industry Analysts 10 de junho. Disponível em: <<http://www.strategyr.com/pressMCP-7036.asp>>. [22 de janeiro de 2016]
- [ISO 8601 2017] 'Art Wikipedia', ISO 8601 2017, (artigo wiki), 7 de agosto de 2017. Disponível em: <[https://pt.wikipedia.org/wiki/ISO\\_8601](https://pt.wikipedia.org/wiki/ISO_8601)>. [16 de outubro de 2017]
- [ISO/IEC 9126 2016] 'Art Wikipedia', ISO/IEC 9126, (artigo wiki), 27 de outubro de 2016. Disponível em: <[https://pt.wikipedia.org/wiki/ISO/IEC\\_9126](https://pt.wikipedia.org/wiki/ISO/IEC_9126)>. [25 de fevereiro de 2017]
- [James 2015] Pearce, James 2015, 'Vodafone world' top M2M provider, according to Analysys Mason', Mobile News Online 30 de junho. Disponível em:

- <<http://www.mobilenewscwp.co.uk/2015/06/30/vodafone-worlds-top-m2m-provider-according-to-analysis-mason/>>. [14 de fevereiro de 2016]
- [JMS 2014] 'Art Wikipedia', JMS 2014, (artigo wiki), 21 de outubro de 2014. Disponível em: <<https://pt.wikipedia.org/wiki/JMS>>. [23 de setembro de 2017]
- [Kanjilal 2015] Kanjilal, Joydip, "Explore the different approaches to model entities in Entity Framework". [3 de setembro de 2015]
- [Kanjilal 2016] Kanjilal, Joydip, "How to implement the Repository design pattern in C#". [12 de agosto de 2016]
- [Khan 2011] Khan, Saif, "Using GMap.NET – Great Maps for Windows Forms & Presentation". [3 de agosto de 2011]
- [Koen et al. 2002] Koen, Peter A., Ajamian, Greg M., Boyce, Scott, Clamen, Allen, Fisher, Eden, Fountoulakis, Stavros, Johnson, Abert, Puri, Pushpinder & Seibertt, Rebecca Fuzzy Front End: Effective Methods, Tools, and Techniques.
- [KTS 2017] KTS 2017, Sobre a KTS. Disponível em: <[http://www.kts.pt/sobre\\_sp.html](http://www.kts.pt/sobre_sp.html)>. [11 de fevereiro de 2016]
- [Kulkarni 2008] Kulkarni, Rushikesh 2008, Lead Users. Disponível em: <<http://pt.slideshare.net/idrush/lead-user>>. [11 de fevereiro de 2017]
- [Lanning 2003] Lanning, Michael J. 2003, An introduction to the market-focused philosophy, framework and methodology called Delivering Profitable Value. Disponível em: <[http://kentblumberg.typepad.com/kent\\_blumberg/files/dpv\\_introduction.pdf](http://kentblumberg.typepad.com/kent_blumberg/files/dpv_introduction.pdf)>. [2 de fevereiro de 2016]
- [Leite 2007] Leite, Mário 2007, Acessando a Banco de Dados com ferramentas RAD Aplicações em Visual Basic, Brasport, Brasil.
- [Log4j 2017] 'Art Wikipedia', Log4j 2017, (artigo wiki), 15 de abril de 2017. Disponível em: <<https://pt.wikipedia.org/wiki/Log4j>>. [10 de setembro de 2017]
- [Máquinas de Vending 2016] Máquinas de Vending 2016, Máquinas de Vending. Disponível em: <<http://www.portaldolicenciamento.com/maquinas-vending.html>>. [19 de fevereiro de 2016]
- [Marins et al. 2009] Martins, Cristiano Souza, Souza, Daniela de Oliveira, Barros, Magno da Silva, O uso do método de análise hierárquica (AHP) na tomada de decisões gerenciais – um estudo de caso.
- [Market Publishers 2014] Market Publishers 2014, 'Vending Machines: Market Research Report', Market Publishers de 9 de janeiro. Disponível em: <[https://pdf.marketpublishers.com/gia/vending\\_machines\\_market\\_research\\_report.pdf](https://pdf.marketpublishers.com/gia/vending_machines_market_research_report.pdf)>. [22-01-16]
- [Marsh 2016] Marsh, Jennifer 2016, Benchmarking 5 popular .NET logging libraries. Disponível em: <<https://www.loggly.com/blog/benchmarking-5-popular-net-logging-libraries/>>. [12 de janeiro de 2016]
- [Mendes 2012] Mendes, Edson, "Vantagens e Desvantagens de SOA". [8 de outubro de 2017]
- [Microsoft .NET 2017] 'Art Wikipedia', Microsoft .NET 2017, (artigo wiki), 27 de agosto de 2017. Disponível em: <[https://pt.wikipedia.org/wiki/Microsoft\\_.NET](https://pt.wikipedia.org/wiki/Microsoft_.NET)>. [4 de agosto de 2017]
- [Microsoft BizTalk Server 2017] 'Art Wikipedia', Microsoft BizTalk Server 2017, (artigo wiki), 8 de setembro de 2017. Disponível em: <[https://en.wikipedia.org/wiki/Microsoft\\_BizTalk\\_Server](https://en.wikipedia.org/wiki/Microsoft_BizTalk_Server)>. [8 de outubro de 2017]
- [MS Access 2017] MS Access 2017, Início. Disponível em: <<https://products.office.com/pt-pt/access?legRedir=true&CorrelationId=8538c744-abfd-4699-bb78-f132d2822945>>. [3 de junho de 2017]
- [Nicola 2016] Nicola, Susana 2016, Análise de valor de negócio – Methods, Instituto Superior de Engenharia do Porto. [2 de fevereiro de 2016]
- [Nicola 2016] Nicola, Susana 2016, Análise de valor de negócio – Negotiation, Instituto Superior de Engenharia do Porto. [2 de fevereiro de 2016]
- [Nielsen 2017] Nielsen 2017, SOBRE A NIELSEN. Disponível em: <<https://www.clubnielsen.pt/About>>. [18 de junho de 2017]

- [Objecto Mock 2017] 'Art Wikipedia', Objecto Mock 2017, (artigo wiki), 29 de dezembro de 2014. Disponível em: <[https://pt.wikipedia.org/wiki/Objeto\\_Mock](https://pt.wikipedia.org/wiki/Objeto_Mock)>. [20 de outubro de 2017]
- [Object-relational mapping] 'Art Wikipedia', Object-relational mapping 2017, (artigo wiki), 27 de agosto de 2017. Disponível em: <[https://en.wikipedia.org/wiki/Object-relational\\_mapping](https://en.wikipedia.org/wiki/Object-relational_mapping)>. [7 de julho de 2017]
- [Olso Stirling 2003] Olson Stirling 2003, Fields nchar & nvarchar in SQL Server. Disponível em: <<http://www.delphigroups.info/2/44/233785.html>>. [11 de junho de 2017]
- [Oracle 2015] Oracle 2015, What Is an Exception?. Disponível em: <<https://docs.oracle.com/javase/tutorial/essential/exceptions/definition.html>>. [15 de junho de 2017]
- [Osterwalder 2016] Osterwalder, Alexander, Hi! I'm Alexander Osterwalder. Disponível em: <<http://alexosterwalder.com/>>. [19 de fevereiro de 2016]
- [Paradox 2017] Paradox 2017, About Paradox. Disponível em: <<https://www.spectralcore.com/databases/paradox>>. [1 de junho de 2017]
- [PHC 2017] PHC 2017, Quem Somos. Disponível em: <<http://www.phcsoftware.com/sobrenos/>>. [1 de junho de 2017]
- [Plain old CLR object 2017] 'Art Wikipedia', Plain old CLR object 2017, (artigo wiki), 27 de julho de 2017. Disponível em: <[https://en.wikipedia.org/wiki/Plain\\_old\\_CLR\\_object](https://en.wikipedia.org/wiki/Plain_old_CLR_object)>. [14 de outubro de 2017]
- [Poudel 2016] Poudel, Sovit, "Securing ASP.NET Web API". [7 de julho de 2016]
- [Pressman 2005] Pressman, Roger S. 2005, Engenharia de Software, McGraw-Hill, São Paulo.
- [Primavera 2017] Primavera 2017, Início. Disponível em: <<http://pt.primaverabss.com/pt/>>. [4 de junho de 2017]
- [RAML 2017] 'Art Wikipedia', RAML 2017, (artigo wiki), 4 de outubro de 2017. Disponível em: <[https://en.wikipedia.org/wiki/RAML\\_\(software\)](https://en.wikipedia.org/wiki/RAML_(software))>. [9 de outubro de 2017]
- [Refatoração 2017] 'Art Wikipedia', Refatoração 2017, (artigo wiki), 13 de fevereiro de 2017. Disponível em: <<https://pt.wikipedia.org/wiki/Refatora%C3%A7%C3%A3o>>. [20 de outubro de 2017]
- [Reitan et al. 2017] Reitan, Erik, Latham, Luke, Wenzel, Maira, 'Queues in Windows Communication Foundation', 30 de março de 2017. Disponível em: <<https://docs.microsoft.com/en-us/dotnet/framework/wcf/feature-details/queues-in-wcf>>. [17 de outubro de 2017]
- [Rocha 2013] Rocha, Helder, "Padrões de integração de sistemas com aplicações java". [21 de agosto de 2015]
- [Rodrigues 2007] Rodrigues, Telmo 2007, Arquitectura de Aplicações Windows Forms Estudo de caso – Desenvolvimento de uma aplicação de transitários, Instituto Superior de Engenharia do Porto.
- [Rodrigues 2013] Rodrigues, Fátima 2013, Algumas Orientações para a Elaboração de uma Tese de Mestrado, Instituto Superior de Engenharia do Porto.
- [SAGE 2017] SAGE 2017, Software ERP. Disponível em: <<http://www.sage.pt/software-e-servicos/erp>>. [4 de junho de 2017]
- [Salmon 2017] Salmon 2017, Início. Disponível em: <<https://www.salmon.com/en/>>. [10 de setembro de 2017]
- [Sanford 2005] Sanford, Michael, "Choosing Between ClickOnce and Windows Installer". [janeiro de 2005]
- [Saxena & Amutha 2017] Saxena, Swasti & Amutha, B., "Clustering and Categorization of Html Parsed Data Using Histogram Technique Deploying the TF-IDF Weight in the Field of Text Mining". [2017]
- [Schema migration 2017] 'Art Wikipedia', Schema migration 2017, (artigo wiki), 6 de setembro de 2017. Disponível em: <[https://en.wikipedia.org/wiki/Schema\\_migration](https://en.wikipedia.org/wiki/Schema_migration)>. [14 de outubro de 2017]

- [Scott 2017] Scott, Daniel, Visitei a Netflix e descobri o seu segredo para o sucesso. Disponível em: < <https://www.linkedin.com/pulse/visitei-netflix-e-descobri-o-segredo-do-seu-sucesso-daniel-scott>>. [24-01-17]
- [Service-oriented architecture 2017] 'Art Wikipedia', Service-oriented architecture 2017, (artigo wiki), 8 de abril de 2017. Disponível em: <[https://pt.wikipedia.org/wiki/Service-oriented\\_architecture](https://pt.wikipedia.org/wiki/Service-oriented_architecture)>. [8 de outubro de 2017]
- [Sigfried 2000] Dominguez, Sigfried 2000, O valor percebido como elemento estratégico para obter a lealdade dos clientes, Caderno de Pesquisas em Administração, São Paulo.
- [Sousa 2005] Sousa, Paulo, "Apontamentos de Engenharia de Informação". [2005-06]
- [SQL Server 2017] SQL Server 2017, Início. Disponível em: <<https://www.microsoft.com/pt-pt/sql-server>>. [3 de junho de 2017]
- [SQL Server Profiler, 2013] 'Art Microsoft', SQL Server Profiler. Disponível em: <<https://docs.microsoft.com/en-us/sql/tools/sql-server-profiler/sql-server-profiler>>. [22 de julho de 2017]
- [SQL UNIQUE Constraint] 'Art W3Schools', SQL UNIQUE Constraint. Disponível em: <[https://www.w3schools.com/sql/sql\\_unique.asp](https://www.w3schools.com/sql/sql_unique.asp)>. [22 de Julho de 2017]
- [SQLite 2017] SQLite 2017, About SQLite. Disponível em: <<https://www.sqlite.org/about.html>>. [4 de junho de 2017]
- [Tempo Universal Coordenado 2017] 'Art Wikipedia', Tempo Universal Coordenado 2017, (artigo wiki), 5 de junho de 2017. Disponível em: <[https://pt.wikipedia.org/wiki/Tempo\\_Universal\\_Coordenado](https://pt.wikipedia.org/wiki/Tempo_Universal_Coordenado)>. [16 de outubro de 2017]
- [Thread 2017] 'Art Wikipedia', Thread, (artigo wiki), 17 de fevereiro de 2017. Disponível em: <[https://pt.wikipedia.org/wiki/Thread\\_\(ci%C3%A4ncia\\_da\\_computa%C3%A7%C3%A3o\)](https://pt.wikipedia.org/wiki/Thread_(ci%C3%A4ncia_da_computa%C3%A7%C3%A3o))>. [27 de fevereiro de 2017]
- [Timestamp 2017] 'Art Microsoft', Timestamp (Transact-SQL). Disponível em: <[https://msdn.microsoft.com/pt-pt/library/ms182776\(v=sql.90\).aspx](https://msdn.microsoft.com/pt-pt/library/ms182776(v=sql.90).aspx)>. [22 de julho de 2017]
- [Todd 2011] Todd, Bill 2011, Advantage Database Server or Microsoft SQL Server – which one is right for you?. Disponível em: <<http://www.abox.com/documentos/ADSvsSQLServerWhitePaper.pdf>>. [9 de fevereiro de 2016]
- [Transmission Control Protocol 2015] 'Art Wikipedia', Transmission Control Protocol, (artigo wiki), 8 de novembro de 2015. Disponível em: <[https://pt.wikipedia.org/wiki/Transmission\\_Control\\_Protocol](https://pt.wikipedia.org/wiki/Transmission_Control_Protocol)>. [17 de fevereiro de 2016]
- [Trowbridge et. Al. 2003] Trowbridge, David, Mancini, Dave, Quick, Dave, Hohpe, Gregor, Newkirk, James, Lavigue, David 2003, Enterprise Solutions Patterns Using Microsoft .NET, Version 2.0.
- [TSUNAMI 2017] TSUNAMI 2017, Início. Disponível em: <<http://www.tsunami.pt/>>. [3 de junho de 2017]
- [Ultimate Delphi 2016] Ultimate Delphi 2016, Database connectivity frameworks in Delphi. Disponível em: <<https://ultimatedelphi.wordpress.com/2016/04/14/database-connectivity-frameworks-in-delphi/>>. [14 de abril de 2016]
- [Unicode 2017] 'Art Wikipedia', Unicode 2017, (artigo wiki), 8 de abril de 2017. Disponível em: <<https://pt.wikipedia.org/wiki/Unicode>>. [11 de junho de 2017]
- [UNISYS 2017] UNISYS 2017, Início. Disponível em: <<http://www.tsunami.pt/>>. [3 de junho de 2017]
- [User Datagram Protocol 2016] 'Art Wikipedia', User Datagram Protocol, (artigo wiki), 28 de janeiro de 2016. Disponível em: <[https://en.wikipedia.org/wiki/User\\_Datagram\\_Protocol](https://en.wikipedia.org/wiki/User_Datagram_Protocol)>. [17 de fevereiro de 2016]
- [Value network 2016] 'Art Wikipedia', Value network, (artigo wiki), 24 de agosto de 2016. Disponível em: <[https://en.wikipedia.org/wiki/Value\\_network](https://en.wikipedia.org/wiki/Value_network)>. [19 de fevereiro de 2017]
- [VB.NET 2017] VB.NET 2017, Visual Basic .NET: A Grande Virada - Parte I. Disponível em: <<https://msdn.microsoft.com/pt-br/library/cc580665.aspx>>. [3 de junho de 2017]

- [Vending machine 2016] 'Art Wikipedia', Vending Machine, (artigo wiki), 16 de fevereiro de 2016. Disponível em: <[https://en.wikipedia.org/wiki/Vending\\_machine](https://en.wikipedia.org/wiki/Vending_machine)>. [17 de fevereiro de 2016]
- [Vendon 2017] Vendon 2017, Soluções de telemetria para gerenciamento de venda automática. Disponível em: <<http://www.vendon.net/>>. [5 de junho de 2017]
- [Vigna 2015] Vigna, Claudio Marcos, Fuzzy Front End: Linha de Frente da Inovação, Disponível em: <<https://www.ibm.com/developerworks/community/blogs/tlcb/entry/mp238?lang=en>>. [12 de fevereiro de 2017]
- [Vodafone 2017] Vodafone 2017, Quem Somos. Disponível em: <<http://www.vodafone.pt/main/A+Vodafone/PT/Quem+Somos/default>>. [3 de junho de 2017]
- [Walker 2012] Walker, Shaun 2012, Microsoft Announces Open Technologies Inc. Subsidiary. Disponível em: <<http://www.dnnsoftware.com/community-blog/cid/134821/microsoft-announces-open-technologies-inc-subsidiary>>. [23 de abril de 2012]
- [Wasson 2015] Wasson, Mike, "Getting Started with ASP.NET Web API 2 (C#)". [28 de maio de 2015]
- [Webhooks 2017] 'Art Wikipedia', Webhook 2017, (artigo wiki), 19 de setembro de 2017. Disponível em: <<https://en.wikipedia.org/wiki/Webhook>>. [16 de outubro de 2017]
- [WebSAFT 2017] WebSAFT 2017, WebSAFT. Disponível em: <<https://vending.absinformatica.pt:988/Account/LogOn>>. [18 de junho de 2017]
- [Williams & Staal 2016] Williams, David & Staal Gert, What is innovation?. [Janeiro de 2016]
- [Windows CE 3 2017] Windows CE 3 2017, Product Documentation. Disponível em: <[https://msdn.microsoft.com/en-us/library/gg162630\(v=winembedded.0\).aspx](https://msdn.microsoft.com/en-us/library/gg162630(v=winembedded.0).aspx)>. [3 de junho de 2017]
- [Windows CE 6 2017] Windows CE 6, Windows Embedded CE 6. Disponível em: <[https://msdn.microsoft.com/en-us/library/dn600283\(v=winembedded.60\).aspx](https://msdn.microsoft.com/en-us/library/dn600283(v=winembedded.60).aspx)>. [3 de junho de 2017]
- [Windows Forms 2017] 'Art Wikipedia', Windows Forms 2017, (artigo wiki), 15 de maio de 2017. Disponível em: <[https://en.wikipedia.org/wiki/Windows\\_Forms](https://en.wikipedia.org/wiki/Windows_Forms)>. [9 de outubro de 2017]
- [Windows Mobile Center 2017] Windows Mobile Center 2017, Descrição do centro do dispositivo do Windows Mobile. Disponível em: <<https://support.microsoft.com/pt-pt/help/931937/description-of-windows-mobile-device-center>>. [27 de abril de 2017]
- [Windows Presentation Foundation 2017] 'Art Wikipedia', Windows Presentation Foundation 2017, (artigo wiki), 16 de março de 2017. Disponível em: <[https://pt.wikipedia.org/wiki/Windows\\_Presentation\\_Foundation](https://pt.wikipedia.org/wiki/Windows_Presentation_Foundation)>. [19 de outubro de 2017]
- [Wintrans 2017] Wintrans 2017, Início. Disponível em: <<http://wintrans.pt/#inicio>>. [3 de junho de 2017]



# 11 Anexos

Neste capítulo, apresentam-se os anexos do trabalho.

## 11.1 Casos de uso

Nesta secção, detalham-se os casos de uso dos sistemas: WinVending Backoffice e WinGuard.

### 11.1.1 WinVending Backoffice

Nesta secção, consultam-se os casos de uso da solução Vodafone WinVending e, caso se justifique, visualizam-se imagens ilustrativas das janelas da aplicação WinVending.

Tabela 31 – Caso de uso de Autenticar

<i>Nome</i>	Autenticar
<i>ID</i>	UC_WVB_0001
<i>Descrição</i>	Autenticar um utilizador na aplicação.
<i>Atores</i>	Utilizador
<i>Pré-condições</i>	Todas as configurações de sistema devem estar parametrizadas.
<i>Cenário Principal</i>	Ao entrar na aplicação, o utilizador introduz as credencias: nome de utilizador e palavra-passe (c.f. Figura 80). Se estiverem corretas, entra na sua área de utilizador, caso contrário, é-lhe apresentada uma mensagem de erro. Pode adicionalmente e, antes de entrar, selecionar um idioma à sua escolha.
<i>Exceções</i>	EX 1 O nome de utilizador ou palavra-passe estão incorretas.


	<ol style="list-style-type: none"> <li>1. O sistema notifica o utilizador para o facto de o nome de utilizador, ou palavra-passe, estarem incorretas;</li> <li>2. Retorna à janela inicial de introdução de credenciais.</li> </ol>
<i>Janela</i>	 <p>Figura 80 - Janela de Credenciação</p>

Tabela 32 – Caso de uso de Selecionar Empresa

<i>Nome</i>	Selecionar Empresa
<i>ID</i>	UC_WVB_0002
<i>Descrição</i>	Selecionar uma empresa.
<i>Atores</i>	Utilizador
<i>Pré-condições</i>	O utilizador está autenticado.
<i>Cenário Principal</i>	<p>Ao abrir a janela da empresa (c.f. Figura 81), apresenta-se uma lista de empresas associadas à aplicação WinVending e, automaticamente, seleciona-se a última empresa escolhida.</p> <p>Pesquisa-se por uma empresa através do código, ou do nome da empresa.</p> <p>A seleção concretiza-se através de um duplo <i>click</i>, na linha da empresa, ou através do botão de confirmação.</p> <p>Pode atribuir-se a opção, para que a janela seja apresentada, sempre que o utilizador entra na aplicação, ou optar-se por entrar diretamente, sem apresentar esta janela. Neste último caso, o sistema atribui, automaticamente, a última empresa selecionada pelo utilizador.</p>
<i>Exceções</i>	<p>EX 1 Não se selecionou qualquer linha de empresa</p> <ol style="list-style-type: none"> <li>1. O sistema notifica o utilizador para a necessidade de selecionar uma empresa.</li> <li>2. Retorna à janela inicial de seleção de empresa</li> </ol>

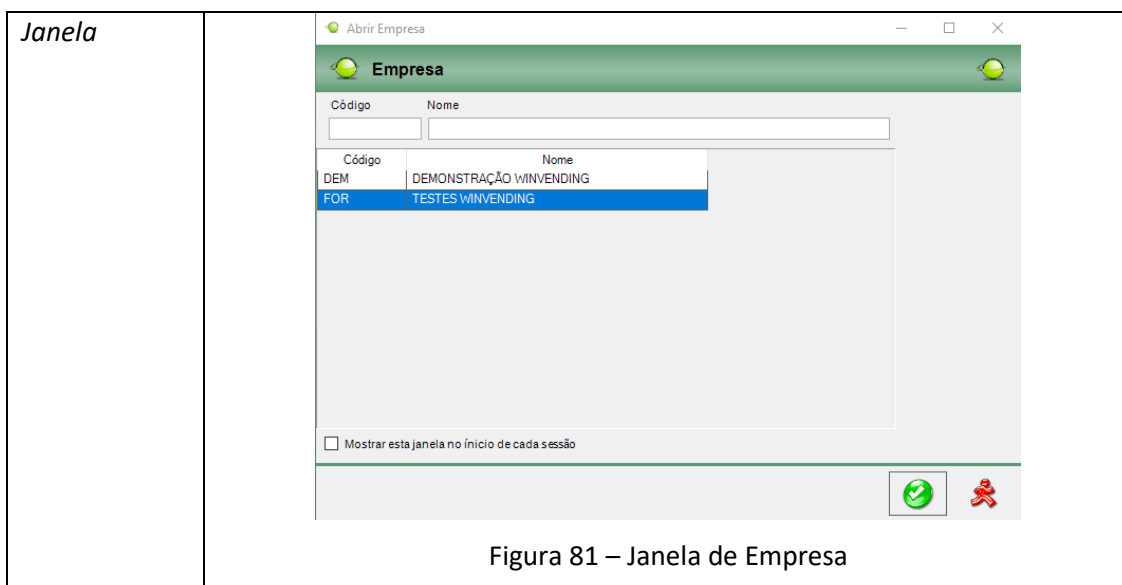


Tabela 33 – Caso de uso de Consultar Painel de Bordo

<i>Nome</i>	Consultar Painel de Bordo
<i>ID</i>	UC_WVB_0003
<i>Descrição</i>	Visão geral da informação operacional da empresa: progressão das rotas dos vendedores do dia, listagem das últimas comunicações por máquina e listagem das anomalias agrupadas por categoria.
<i>Atores</i>	Utilizador (todos)
<i>Pré-condições</i>	O utilizador está autenticado e tem permissões de visualização da janela.
<i>Cenário Principal</i>	<p>Ao abrir o painel de bordo (c.f. Figura 82), o utilizador tem acesso a um conjunto de informações relevantes para a gestão operacional da empresa, separadas em três secções: (i) rotas do dia dos vendedores, (ii) listagem das comunicações e (iii) listagem das anomalias.</p> <p>A seção das rotas encontra-se, por sua vez, dividida em grupos, onde cada um representa a rota de um vendedor. A rota do vendedor resume o seu estado atual com as informações: da visita (atual, ou próxima, ou última), do progresso (número de máquinas a visitar / número de máquinas visitadas) e da listagem de todas as máquinas a serem visitadas no dia. Por cada visita, são visíveis as informações: nome do estabelecimento, hora de início / fim de visita e um indicador para sinalizar se se realizou, ou não, fora de rota.</p> <p>No fundo deste separador, representa-se a progressão geral de todas as</p>

	<p>rotas (número de todas as máquinas a visitar / número de todas as máquinas visitadas).</p> <p>Na secção de comunicações, são listadas, de modo predefinido, todas as últimas comunicações por máquina “pendentes ou com falhas”, por ordem decrescente de data / hora. O utilizador pode alterar a opção para visualizar: todas as comunicações, apenas aquelas que contêm falhas, ou apenas as pendentes. Em cada linha de comunicação, descreve-se: o tipo (estatística, contabilística, configuração, inicialização), o estado (pendente, falha, anulada, sucesso), o número da máquina, o nome do estabelecimento e a data / hora da comunicação.</p> <p>Na secção de anomalias, são listadas todas as anomalias agrupadas por categoria. O utilizador pode especificar qual o tipo de anomalia a consultar, através da seleção do seu tipo (só são mostrados os tipos que têm anomalias associados). Em cada linha de anomalia, visualiza-se: o código da máquina, o nome do estabelecimento, a descrição das anomalias e o número de ocorrências dessa anomalia.</p> <p>Os dados apresentados sofrem uma atualização automática, de X em X segundos, onde X é um parâmetro da ficha de utilizador. Também é possível a atualização manual, através do botão de atualizar, situado no canto superior esquerdo. No canto superior direito, mostra-se a data / hora da última atualização.</p>
<p><i>Requisitos</i></p> <p><i>Desempenho</i></p>	<p>De modo a garantir disponibilidade da aplicação, carregam-se, em segundo plano, as listas de comunicações e anomalias.</p>

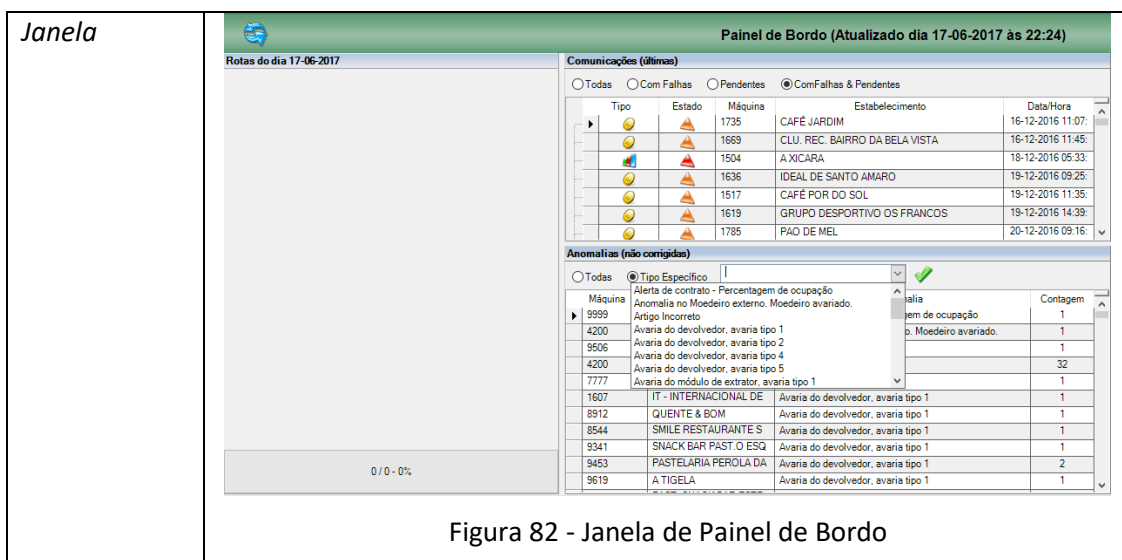


Tabela 34 – Caso de uso de Consultar Localização de Máquinas

<i>Nome</i>	Consultar Localização das Máquinas
<i>ID</i>	UC_WVB_0004
<i>Descrição</i>	Visualizar a posição geográfica das máquinas com a possibilidade de navegar diretamente para a ficha da máquina.
<i>Atores</i>	Utilizador (Administrativo)
<i>Pré-condições</i>	O utilizador está autenticado numa empresa, cujas máquinas são geridas por telemetria e tem permissão de visualização.
<i>Cenário Principal</i>	Ao abrir a janela de localizações (c.f. Figura 83) visualizam-se, no mapa, todas as máquinas de todas as rotas a visitar, no dia seguinte. Detalha-se, para cada máquina: a identificação e a posição da máquina e os respetivos estados dos sensores (porta, ligada / desligada, cabo ligado / desligado / cobertura de rede). Disponibiliza-se um filtro para apresentação das máquinas por: (i) rota completa, (ii) dia específico da rota, (iii) máquina(s) específica(s) e (iv) apenas as máquinas fora do raio de segurança.

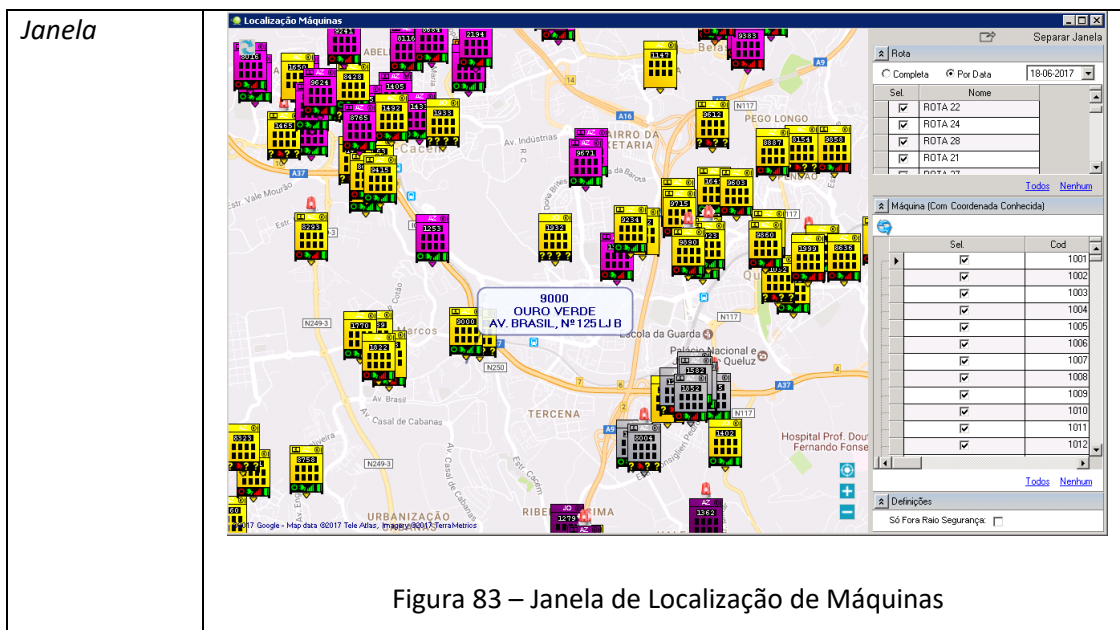


Tabela 35 – Caso de uso de Manter Análises Personalizáveis

<i>Nome</i>	Manter análises personalizadas
<i>ID</i>	UC_WVB_0005
<i>Descrição</i>	Criar, alterar e remover grupos e análises personalizáveis.
<i>Atores</i>	Utilizador (Administrador)
<i>Pré-condições</i>	O utilizador está autenticado e tem permissões de visualização da janela.
<i>Cenário Principal</i>	<p>Ao abrir a janela de análises personalizáveis (c.f. Figura 84), visualiza-se uma estrutura em árvores, onde os ramos são compostos pelos grupos e as folhas pelas análises. O grupo é identificado na árvore por um nome e, entre parênteses, por um número que corresponde ao número de análises que contém. A análise é composta por: (i) um nome, (ii) uma expressão SQL, (iii) uma lista de parâmetros e (iv) uma lista de sumários.</p> <p>Ao editar ou criar uma nova análise, preenche-se os campos pretendidos, em particular, aqueles que são obrigatórios (e.g. nome). Dá-se ênfase ao preenchimento do campo “expressão”, também ele obrigatório. Atribui-se o valor desse campo de acordo com as regras da linguagem SQL e identificam-se os parâmetros a incluir na instrução, pelo seu número antecedido do carácter “@”.</p>

	<p>Os parâmetros da análise podem ser criados, alterados e removidos. A sua numeração é gerada automaticamente, no momento da criação, através do incremento de uma unidade ao último número. A numeração pode ser alterada sob a responsabilidade e gestão do utilizador. Um parâmetro é, ainda, constituído por: um nome, tipo (numérico, texto, data / hora, booleano) e máscara de edição do valor.</p> <p>Os sumários representam as funções agregadoras a colocar como última linha, aquando da apresentação dos resultados. Um sumário é composto por: um nome de uma coluna, presente na expressão SQL, um campo indicador (se é ou não para ser apresentado como sumário) e um tipo (contagem, somatório, média, máximo, mínimo).</p> <p>Posteriormente à manutenção da análise personalizável, o utilizador pode visualizá-la, através da opção para esse efeito, que se encontra no menu. Ao pressionar essa opção e caso existam parâmetros, invoca-se a janela de introdução dos parâmetros da análise (UC_WVB_0006), caso contrário, abre-se diretamente a janela de visualização dos resultados (UC_WVB_0007) (c.f. Figura 16).</p>
<i>Exceções</i>	<p>EX 1 Falta o preenchimento de um campo obrigatório (nome do grupo ou análise, expressão de SQL), nome / tipo do parâmetro e nome / tipo do sumário), ou existe uma numeração repetida nos parâmetros.</p> <ol style="list-style-type: none"> <li>1. O sistema notifica o utilizador para a obrigatoriedade de preenchimento do campo;</li> <li>2. Posiciona o cursor no campo em falta;</li> <li>3. Retorna, em modo de edição, para o ponto em que se encontrava.</li> </ol>

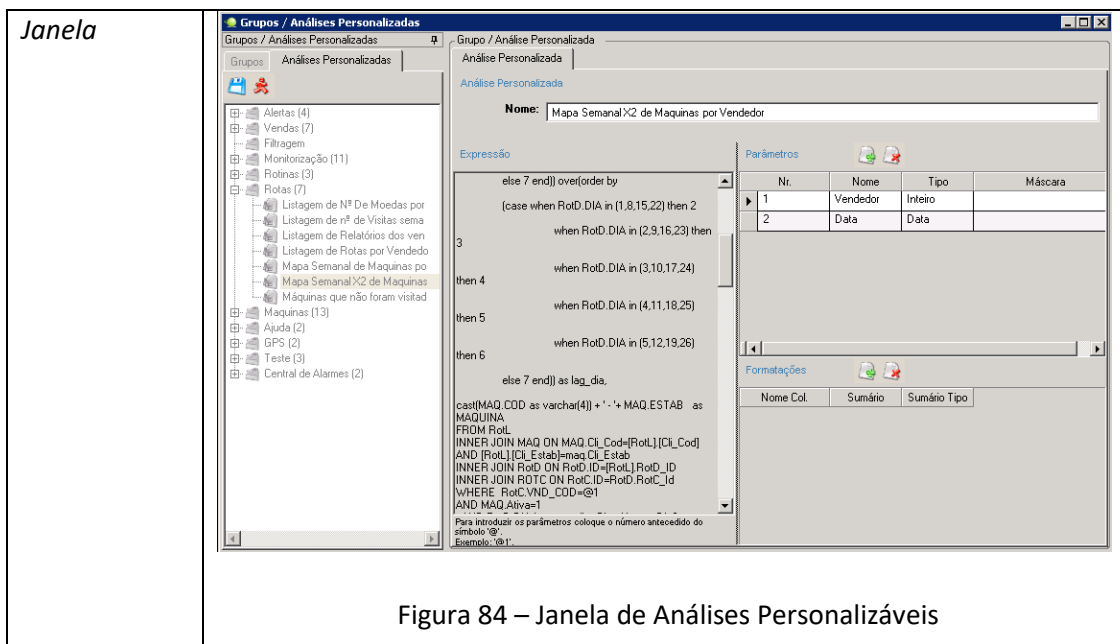


Figura 84 – Janela de Análises Personalizáveis

Tabela 36 – Caso de uso de Introduzir Parâmetros da Análise Personalizável

<i>Nome</i>	Introduzir Parâmetros da Análise Personalizável
<i>ID</i>	UC_WVB_0006
<i>Descrição</i>	Introduzir os parâmetros da análise personalizável.
<i>Atores</i>	Utilizador (todos)
<i>Pré-condições</i>	O utilizador está autenticado, tem permissões para executar a análise e a análise contém parâmetros.
<i>Cenário Principal</i>	<p>Ao entrar na janela de introdução de parâmetros da análise personalizável (c.f. Figura 85), mostram-se os parâmetros da análise, com o nome e o controlo de introdução de dados, em acordo com o tipo e máscara atribuídos.</p> <p>O utilizador preenche os parâmetros, conforme a sua preferência e confirma a sua ação. Também dispõe de uma opção para limpar o conteúdo de todos os parâmetros.</p>
<i>Exceções</i>	<p>EX 1 Tipo do parâmetro incorreto (e.g. introduzir texto num campo numérico)</p> <ol style="list-style-type: none"> <li>1. O sistema notifica o utilizador da falha ocorrida;</li> <li>2. Retorna para a janela de Introdução de Parâmetros</li> </ol>

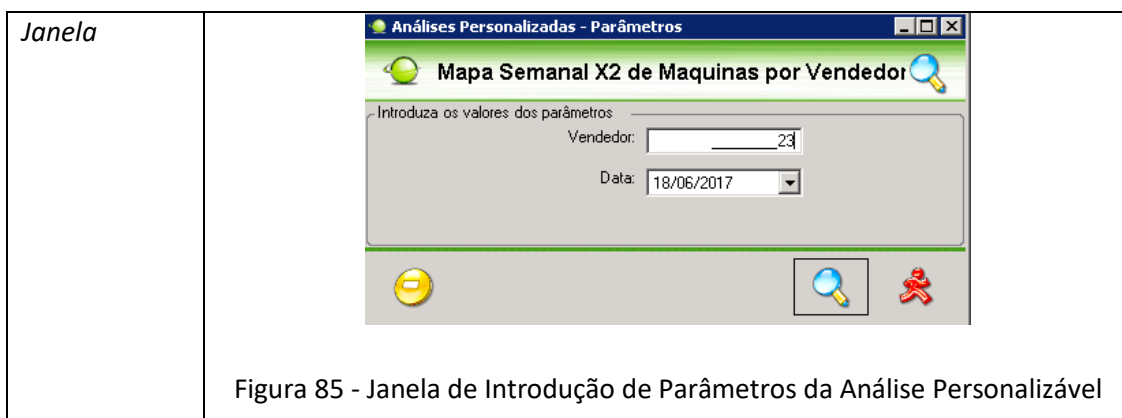


Tabela 37 – Caso de uso de Visualizar Análise Personalizável

<i>Nome</i>	Visualizar análise personalizável
<i>ID</i>	UC_WVB_0007
<i>Descrição</i>	Visualizar uma análise personalizável.
<i>Atores</i>	Utilizador (todos)
<i>Pré-condições</i>	O utilizador está autenticado, tem permissões para executar a análise e estão introduzidos corretamente todos os parâmetros necessários para a execução da análise.
<i>Cenário Principal</i>	<p>Ao abrir a janela de visualização da análise personalizável (c.f. Figura 86), apresentam-se: os parâmetros enviados, o resultado da análise, sob uma forma tabular, com todos os campos colocados, na expressão SQL e respetivas colunas de sumário, caso atribuídas.</p> <p>Concede-se ao utilizador a possibilidade de exportar os resultados para a área de transferência, ou para uma folha de Excel. Pode-se acionar a opção de imprimir parâmetros, de modo a que se incluam juntamente com o resultado.</p>
<i>Exceções</i>	<p>EX 1 Falha na execução da expressão SQL (e.g. erro de sintaxe ou falta de parâmetros)</p> <ol style="list-style-type: none"> <li>1. O sistema notifica o utilizador da falha ocorrida;</li> <li>2. Retorna para a janela de análises personalizadas.</li> </ol>

**Janela**

**Análises Personalizadas - Resultado**

**Mapa Semanal X2 de Maquinas por Vendedor**

Parâmetros  
 Vendedor: 23  
 Data: 18-06-2017

Imprimir

Resultado

SEM	MAQUINAS 2ª Feira	MAQUINAS 3ª Feira	MAQUINAS 4ª Feira	MAQUINAS 5ª Feira	MAQUINAS 6ª Feira
1	1253 - L.A. CATTY	1028 - PAPELARIA C	1005 - O PREGUINH	1230 - PASTELARIA	1028 - PAPELARIA C
1	1405 - CAFÉ CARTO	1587 - GRUPO BAND	1403 - CAFÉ PALMEI	1234 - O GULO SO D	1269 - PAST. TOJA
1	1810 - O RETIRO DE	1731 - PARA AI BAR	1504 - A XICARA	1369 - SNACK BAR	1356 - TASC A DO C
1	2025 - BURGER AND	1767 - CAFE LIMAO	1732 - MIMINHO DA	1415 - REST. A RAM	1731 - PARA AI BAR
1	2048 - FATIAS E ME	1771 - CAFÉ NO CA	1851 - DOCES VICIU	1433 - O ALEMAD C	1815 - PONTO DE E
1	2052 - PÉROLA DO T	2034 - DOM PALADA	1856 - CENTRAL DE	1473 - CAFÉ MILAG	1876 - ARIZONA - PA
1	2106 - PAST. ANCO	2078 - ONPRESS	1869 - CAFÉ COM L	1506 - TABERNA DO	2034 - DOM PALADA
1	2130 - CAFE FRANC	2154 - MARI E MART	2050 - ARIZONA - SE	1566 - CAFÉ ANTONI	2078 - ONPRESS
1	2187 - CAFÉ CENTR	2171 - PASTELARIA	2055 - CAFÉ MIRAD	1605 - 3 CALHAUS	2088 - OPTIMISTIC
1	2194 - PAST. LONDR	2192 - CAFE TOZE	2080 - CAFÉ SOL NA	1699 - O TIJOLO	2192 - CAFE TOZE
1	2230 - BAR THE GE	8266 - TORRADINHA	2101 - PASTELARIA	1736 - O PIKKAS	8266 - TORRADINHA
1	8191 - REST. SOLDA	8332 - PONTO E VIR	2161 - CANTINHO D	2106 - PAST. ANCO	8356 - DELICIA DE B
1	8509 - OS RECREIO	8356 - DELICIA DE B	2253 - CAFÉ E COM	2231 - CAFÉ EURDP	8412 - PANICOLHO
1	8512 - LUA NOVA	8369 - O FORNO DE	8016 - ROSA DO VE	8116 - A BOLINHA D	8798 - ROCK BAR

Figura 86 – Janela de Visualização de Análises Personalizáveis

Tabela 38 – Caso de uso de Comunicar com Máquinas

<i>Nome</i>	Comunicar com Máquinas
<i>ID</i>	UC_WVB_0008
<i>Descrição</i>	Assistente de pedidos de comunicação para múltiplas máquinas
<i>Atores</i>	Utilizador (Administrativo)
<i>Pré-condições</i>	O utilizador está autenticado e tem permissões para comunicar com as máquinas.
<i>Cenário Principal</i>	Ao abrir o assistente de comunicação com as máquinas, o utilizador entra automaticamente na etapa número um do assistente de comunicação, a qual permite a seleção das máquinas com que se pretende comunicar (UC_WVB_0009). Posteriormente, avança-se para a segunda e última etapa do assistente, onde se efetivam os pedidos de comunicação para as máquinas eleitas (UC_WVB_0010) (c.f. Figura 17).

Tabela 39 – Caso de uso de Selecionar Máquina a Comunicar


<i>Nome</i>	Selecionar Máquina a Comunicar
<i>ID</i>	UC_WVB_0008
<i>Descrição</i>	Assistente de pedidos de comunicação para múltiplas máquinas
<i>Atores</i>	Utilizador (Administrativo)
<i>Pré-condições</i>	O utilizador está autenticado e tem permissões para comunicar com as máquinas.
<i>Cenário Principal</i>	<p>Na etapa um do assistente de comunicação com as máquinas (c.f. Figura 87), o utilizador pode selecionar as máquinas com que pretende comunicar, de dois modos distintos: (i) por rota e (ii) por máquina específica. Ao selecionar uma rota, selecionam-se, automaticamente, as máquinas pertencentes à rota, para a data definida. Alternativamente, ou como complemento, seleciona-se cada máquina individual.</p> <p>O tipo de comunicação predefinido é o “estatístico”, no entanto, pode alterar-se para “teste” ou “configuração”.</p>
<i>Exceções</i>	<p>EX 1 Não é selecionada qualquer máquina</p> <p>1. O sistema notifica o utilizador que deve selecionar, pelo menos, uma máquina e não avança para a etapa seguinte.</p>
<i>Janela</i>	 <p>Figura 87 – Janela Assistente de Comunicação Máquina – Etapa 1 – Selecionar Máquinas</p>

Tabela 40 – Caso de uso de Executar Pedidos de Comunicação

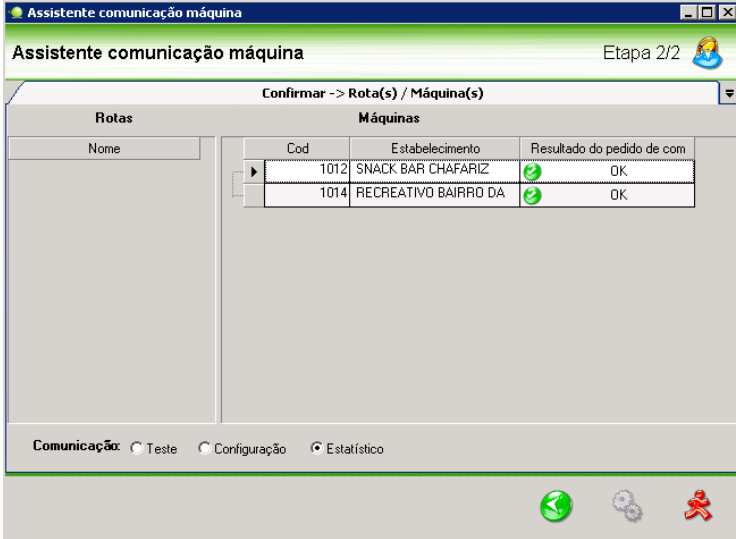
<i>Nome</i>	Executar Pedidos de Comunicação
<i>ID</i>	UC_WVB_0010
<i>Descrição</i>	Efetuar pedidos de comunicação às máquinas selecionadas
<i>Atores</i>	Utilizador (Administrativo) e sistema Middleware
<i>Pré-condições</i>	O utilizador está autenticado, tem permissões para comunicar com as máquinas e existem máquinas selecionadas para se efetuar pedidos de comunicação.
<i>Cenário Principal</i>	Na etapa dois (c.f. Figura 88), verificam-se as comunicações selecionadas para comunicar e, em caso de concordância, procede-se à execução dos pedidos de comunicação, junto do sistema Middleware. Posteriormente, analisam-se os resultados apresentados na coluna “Resultado do Pedido de Comunicação”.
<i>Exceções</i>	EX 1 A máquina não se encontra disponível (e.g. desligada, cabo de dados desligado, pouco sinal de rede)  1. O sistema notifica o utilizador, diretamente na lista de pedidos, através de uma nova coluna que apresenta o estado do pedido.
<i>Requisitos Desempenho</i>	De modo a garantir disponibilidade da aplicação, efetuam-se os pedidos de comunicação, em segundo plano e apresentam-se os resultados, assim que conhecidos.
<i>Janela</i>	 <p>Figura 88 – Janela Assistente de Comunicação Máquina – Etapa 2 – Efetuar Pedidos de Comunicação</p>

Tabela 41 – Caso de uso de Notificar Ocorrências

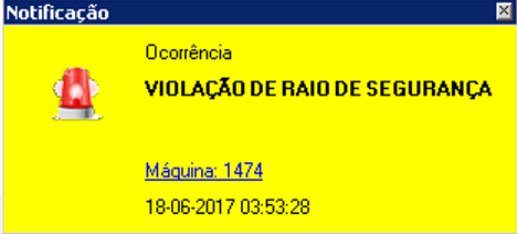
<i>Nome</i>	Notificar Ocorrências
<i>ID</i>	UC_WVB_0011
<i>Descrição</i>	Notificar a ocorrência de um alarme de violação do raio de segurança de uma máquina.
<i>Atores</i>	Utilizador (todos)
<i>Pré-condições</i>	O utilizador está autenticado.
<i>Cenário Principal</i>	<p>Após a autenticação do utilizador, o sistema verifica, de X em X segundos, a existência de uma nova ocorrência (X é um parâmetro configurável da empresa). Caso detete uma nova ocorrência, verifica qual o seu tipo (e.g. abertura de porta, máquina desligada) e se se trata de uma violação de raio de segurança. Quando tal se confirma, é apresentada uma janela instantânea (c.f. Figura 89), cerca de 5 segundos (a partir dos 3 segundos a janela desvanece-se até ao seu fecho) e, moderadamente cintilante, a informar o código da máquina e a data e hora da ocorrência da violação de raio de segurança. Fornece-se, também, uma ligação direta ao separador de localização da ficha da máquina, ao pressionar o código da máquina.</p> <p>Quando ocorrem múltiplas ocorrências, em simultâneo, as que surgem a seguir aguardam pelo fecho da que se encontra em visualização. Posteriormente, é apresentada a próxima mensagem, e assim sucessivamente, por ordem da data e hora da ocorrência.</p>
<i>Janela</i>	 <p>Figura 89 – Notificação de Violação de Raio de Segurança</p>

Tabela 42 – Caso de uso de Consultar Tabelas

<i>Nome</i>	Consultar Tabelas
<i>ID</i>	UC_WVB_1000
<i>Descrição</i>	Consultar, sob ponto de vista tabular, as entidades, localizações, produtos, rotas e tabelas específicas do negócio de Vending.
<i>Atores</i>	Utilizador (Administrador e Administrativo)
<i>Pré-condições</i>	O utilizador está autenticado e tem permissões de visualização para a tabela a consultar.
<i>Cenário Principal</i>	<p>A janela “Tabelas” (c.f. Figura 90) encontra-se dividida em duas partes. A primeira é composta por um menu, em formato harmónico, onde se encontram agrupados, por secções, os diferentes tipos de tabela (e.g. grupo “Entidades” / tabela de “Clientes”). A segunda contém uma lista com todos os elementos da tabela a consultar e, na sua parte superior, disponibilizam-se dois modos de pesquisa: por código, ou por nome do elemento. Ao introduzir um texto numa das caixas de pesquisa, efetua-se, automaticamente, uma pesquisa na tabela de registos, começados pelo texto introduzido e seleciona-se a primeira ocorrência.</p> <p>O utilizador tem acesso a um menu de contexto da tabela e a um conjunto de opções que permitem a invocação da janela de manutenção do respetivo registo, no modo pretendido (e.g. criação / alteração / remoção). É-lhe, ainda, facultada uma opção para atualizar a lista de registos. Esta opção existe para precaver situações onde outros utilizadores manipulam os registos em visualização pelo utilizador atual.</p>

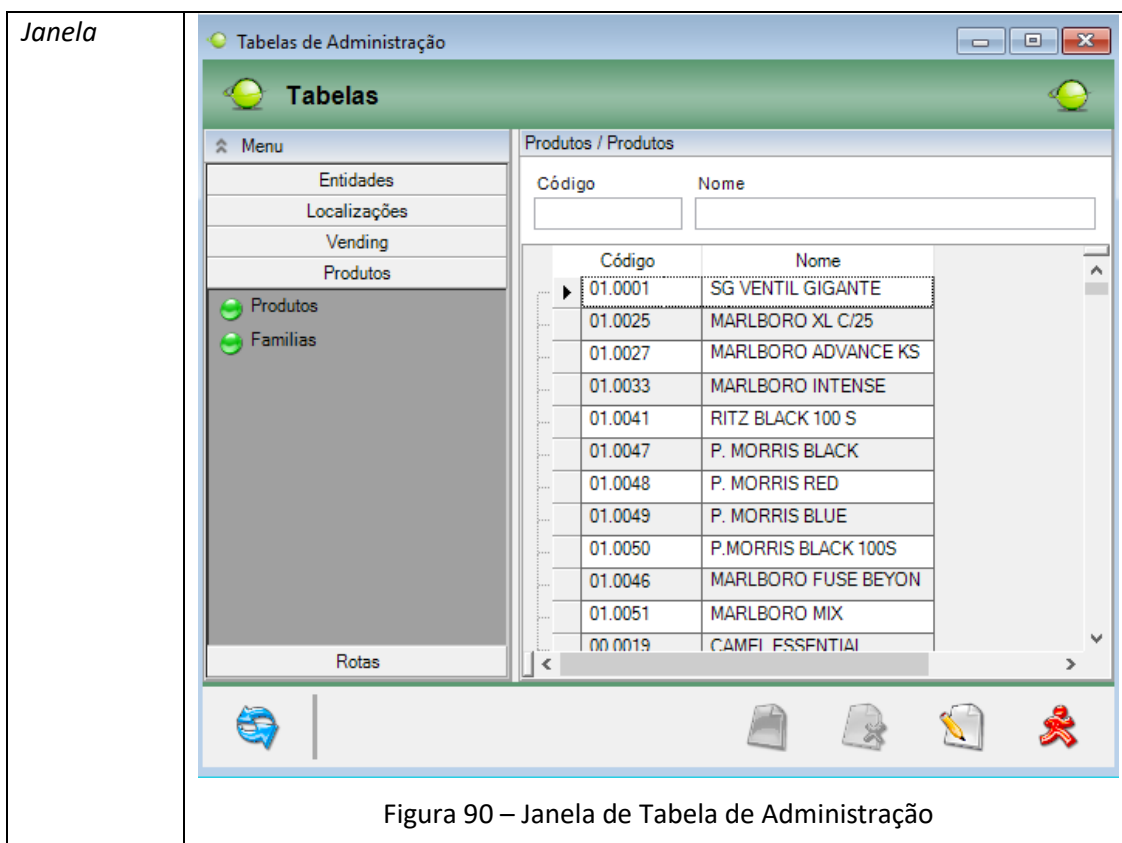


Tabela 43 – Caso de uso de Manter Clientes

<i>Nome</i>	Manter Clientes
<i>ID</i>	UC_WVB_1001
<i>Descrição</i>	Criar, alterar e remover um cliente ou estabelecimento
<i>Atores</i>	Utilizador (Administrativo, Administrador)
<i>Pré-condições</i>	O utilizador está autenticado e tem permissões para executar as diferentes operações (criar, alterar, remover).
<i>Cenário Principal</i>	Ao abrir a janela de “Clientes” (c.f. Figura 93) visualizam-se, no topo, os dados identificativos do cliente (código, número de estabelecimento, nome e número de contribuinte) e uma opção para o inativar. No centro da janela, encontra-se um conjunto de separadores cuja visibilidade depende do facto de se tratar de um cliente, ou estabelecimento. Por se tratar de informação relativa a cada estabelecimento particular, consideram-se três separadores comuns: (i) dados principais, (ii) central de alarmes e (iii) segmentação de

	<p>clientes (UC_WVB_1003). Especificam-se os próximos separadores, para uso exclusivo do cliente: (i) comercial e (ii) estabelecimentos.</p> <p>No separador “Dados Principais”, encontram-se os seguintes campos: morada do cliente, horário de trabalho / funcionamento, vendedor, fornecedor, periodicidade e segmento. No separador comercial, define-se a moeda e um conjunto de condições de pagamento (e.g. máximo de documentos por pagar). No separador central de alarmes, definem-se: (i) a entidade a notificar (proprietário, estabelecimento e autoridade), (ii) a empresa de segurança a contactar, por meio do protocolo definido entre o cliente e a empresa (e.g. via serviço <i>web</i>) e (iii) a autoridade a contactar, no caso da ocorrência de um alarme que o justifique. Os separadores “Estabelecimentos” e “Segmentação” são expostos, nos respetivos casos de uso (UC_WVB_1002 e UC_WVB_1003).</p>
<i>Exceções</i>	<p>EX 1 Falta o preenchimento de campos obrigatórios (e.g. código e número de estabelecimento, nome e número de contribuinte)</p> <ol style="list-style-type: none"> <li>1. O sistema notifica o utilizador que o campo é de preenchimento obrigatório;</li> <li>2. Posiciona o cursor no campo em falta;</li> <li>3. Retorna, em modo de edição, para o ponto em que se encontrava</li> </ol> <p>EX 2 Horário de funcionamento encontra-se fora dos limites do horário de trabalho;</p> <ol style="list-style-type: none"> <li>1. O sistema notifica o utilizador que o horário está incorreto;</li> <li>2. Posiciona o cursor no campo em falta;</li> <li>3. Retorna, em modo de edição, para o ponto em que se encontrava.</li> </ol>
<i>Diagrama conceptual de classes</i>	<p>Na Figura 91, verificam-se as relações do cliente com as várias entidades. Deteta-se que o cliente tem uma associação própria representativa dos vários estabelecimentos que pode deter. Caso possua máquinas, estas associam-se a si e, caso se faça a monitorização de alarmes com o recurso à solução WinGuard, pode atribuir-se uma central de alarmes.</p> <p>Para efeitos de gestão de rota, pode atribuir-se uma periodicidade de visita ao cliente e indicar-se qual o vendedor habitual.</p>

Adicionalmente, pode indicar-se: (i) o fornecedor associado, (ii) o país, (iii) a moeda e (iv), para efeitos de segmentação, o canal e os segmentos relacionados.

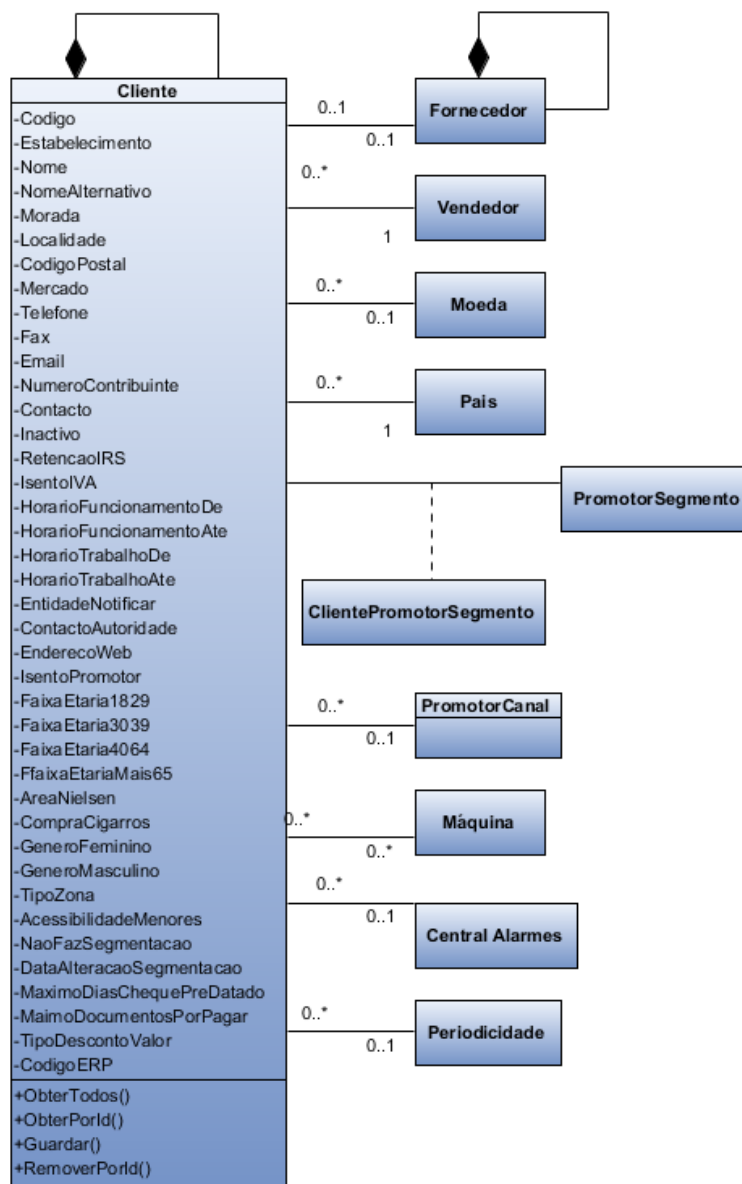


Figura 91 – Diagrama conceptual de classes da visão de *cliente*

Modelo de dados

Representa-se, na Figura 92, a tabela principal dos dados do cliente (denominada de “Cli”) e as tabelas diretamente relacionadas: (i) vendedor principal (denominada de “Vnd”), (ii) periodicidade, (iii) central de alarmes, (iv) país, (v) moeda (moe), (vi) canal (denominada de “PrmCan”), (vii) tabela de associação (denominada “Cli\_PrmSgm”) que possibilita a associação de um cliente a vários segmentos (denominada “PrmSgm”) e (viii) uma tabela

de associação entre o cliente e o fornecedor (denominada “Cli\_Frn”). Embora a relação entre o cliente e fornecedor seja de um para um, de modo a evitar inconsistências no modelo de dados, opta-se pela criação de uma tabela de associação entre as entidades. Assim, ao remover-se uma linha desta tabela, afeta-se, simultaneamente, ambas as tabelas associadas.

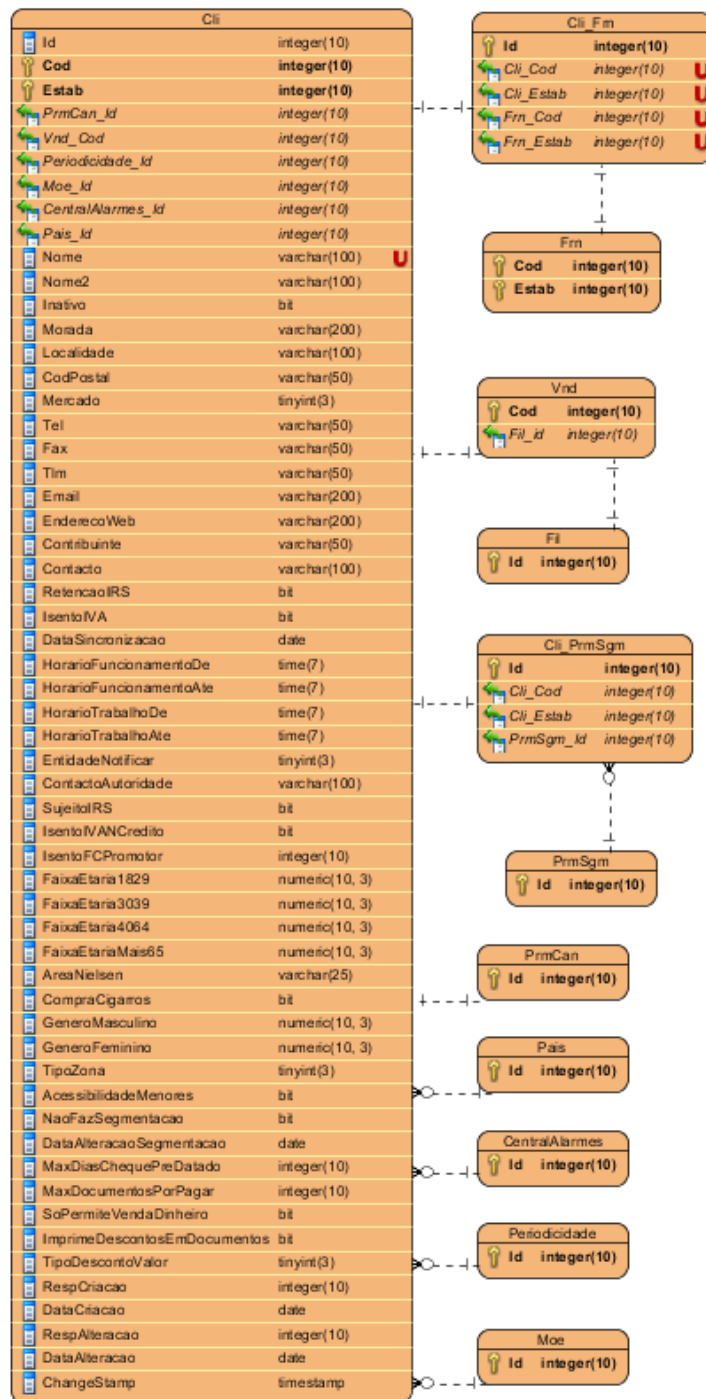


Figura 92 – Modelo de dados da visão do cliente

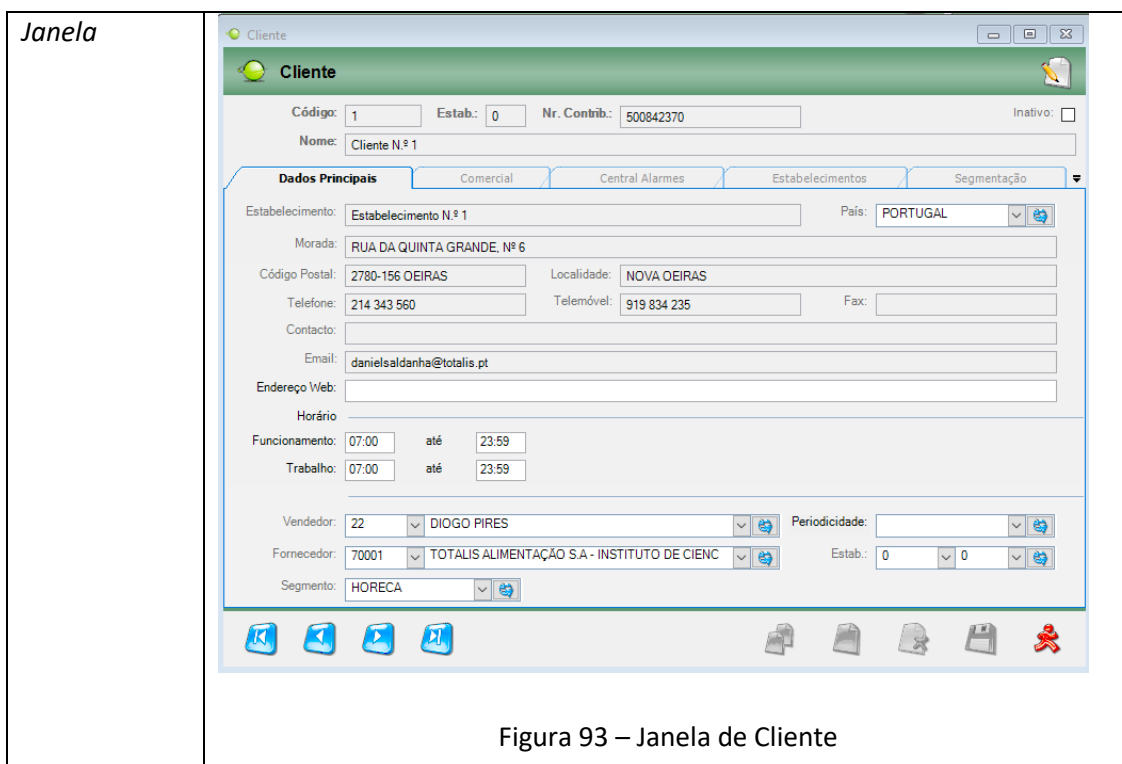


Tabela 44 – Caso de uso de Manter Estabelecimentos Cliente

<i>Nome</i>	Manter Estabelecimentos Cliente
<i>ID</i>	UC_WVB_1003
<i>Descrição</i>	Criar, alterar e remover um estabelecimento
<i>Atores</i>	Utilizador (Administrativo, Administrador)
<i>Pré-condições</i>	O utilizador está autenticado e tem permissões para executar as diferentes operações (criar, alterar, remover).
<i>Cenário Principal</i>	Ao seleccionar o separador “Estabelecimentos” (c.f. Figura 94 – Janela Cliente – Separador Estabelecimentos) apresenta-se uma tabela com os dados do estabelecimento: nome, vendedor, código, morada, telefone e telemóvel. Permite-se ao utilizador, além da visualização, a criação, alteração, ou remoção de um registo, através do menu lateral da tabela, ou por meio de um menu de contexto.

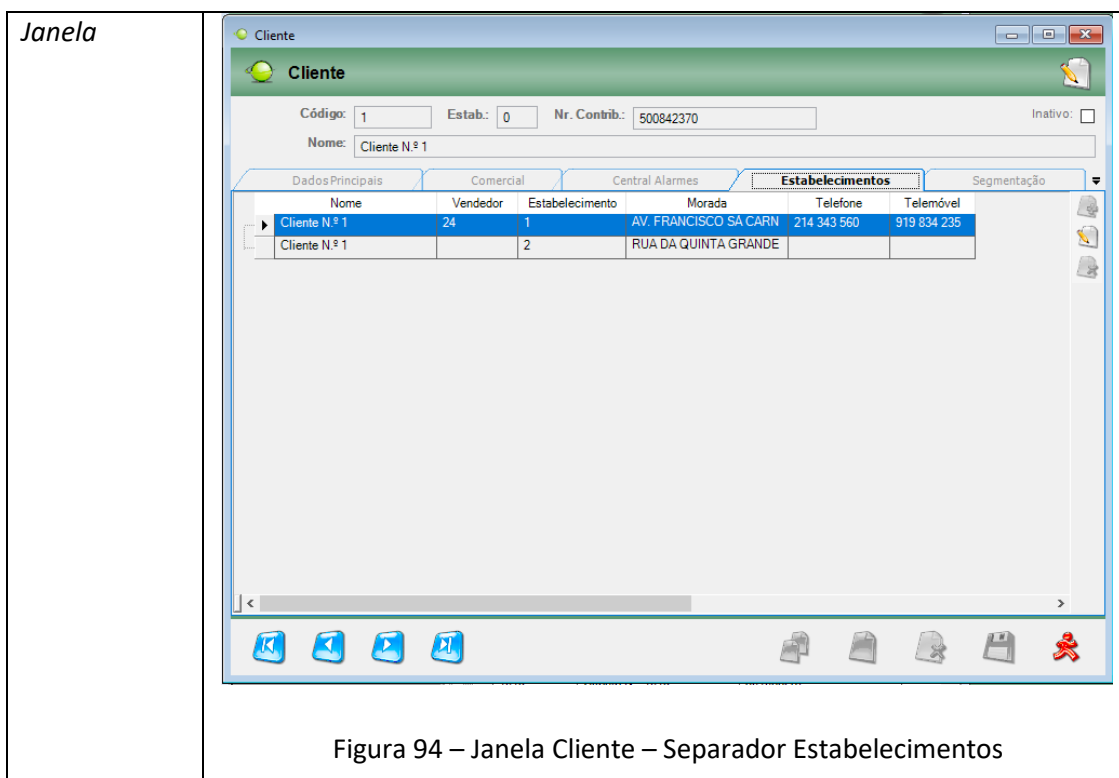
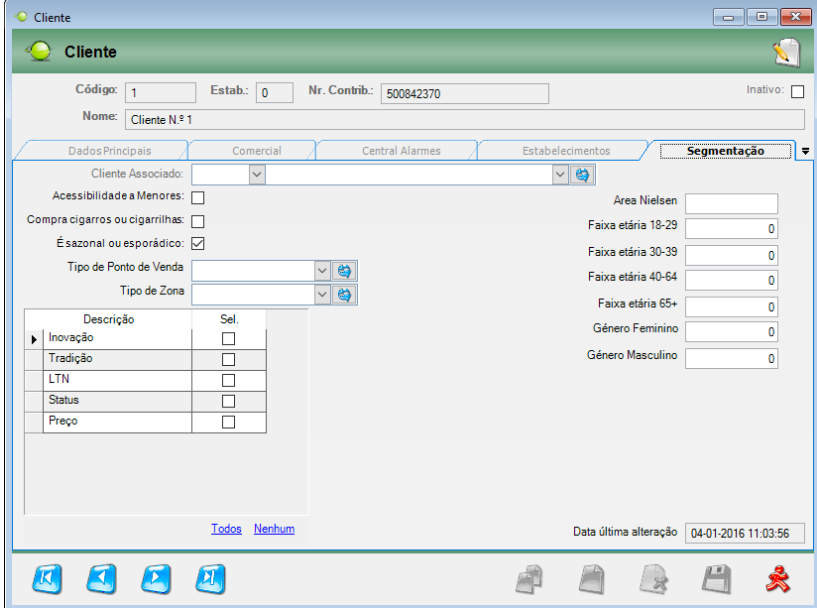


Tabela 45 – Caso de uso de Manter Segmentação do Cliente

<i>Nome</i>	Manter Segmentação Cliente
<i>ID</i>	UC_WVB_1004
<i>Descrição</i>	Editar os dados relativos à segmentação dos clientes.
<i>Atores</i>	Utilizador (Administrativo, Administrador)
<i>Pré-condições</i>	O utilizador está autenticado e tem permissões de edição.
<i>Cenário Principal</i>	Ao seleccionar o separador “Segmentação” (c.f. Figura 95), apresentam-se os dados estatísticos para o controlo da segmentação do estabelecimento: acessibilidade a menores, compra cigarros ou cigarrilhas, vendas sazonais ou esporádicas, tipo de ponto de venda (e.g. tabacaria, quiosque, horeca <sup>160</sup> ), tipo de zona (rural ou urbano), área Nielsen <sup>161</sup> , percentagem de clientes por

<sup>160</sup> “Horeca é um acrónimo de ho[téis]+re[staurantes]+ca[fés]” (Infopédia 2017).

<sup>161</sup> “A Nielsen é um líder mundial em pesquisa e informação, estabelecida em mais de 100 países.” (Nielsen 2017)

	faixa etária, percentagem de clientes por género (feminino / masculino), segmentos (inovação <sup>162</sup> , tradição <sup>163</sup> , status <sup>164</sup> , preço <sup>165</sup> , LTN) e data / hora da última alteração da segmentação.
Exceções	<p>EX 1 Falha na validação dos campos de percentagem (e.g. fora dos limites de 0 a 100)</p> <ol style="list-style-type: none"> <li>1. O sistema notifica qual o intervalo de valores permitido;</li> <li>2. Posiciona o cursor no campo em falha;</li> <li>3. Retorna, em modo de edição, para o ponto em que se encontrava.</li> </ol>
Janela	 <p>Figura 95 – Janela Cliente de Separador Segmentação</p>

<sup>162</sup> Inovação é o tipo de clientes que experimenta novos produtos.

<sup>163</sup> Tradição é o tipo de clientes que se mantém fiel a um produto.

<sup>164</sup> Status é o tipo de clientes que compra produtos mais caros.

<sup>165</sup> Preço é o tipo de clientes que compra produtos mais baratos.

Tabela 46 – Caso de uso de Manter Fornecedores

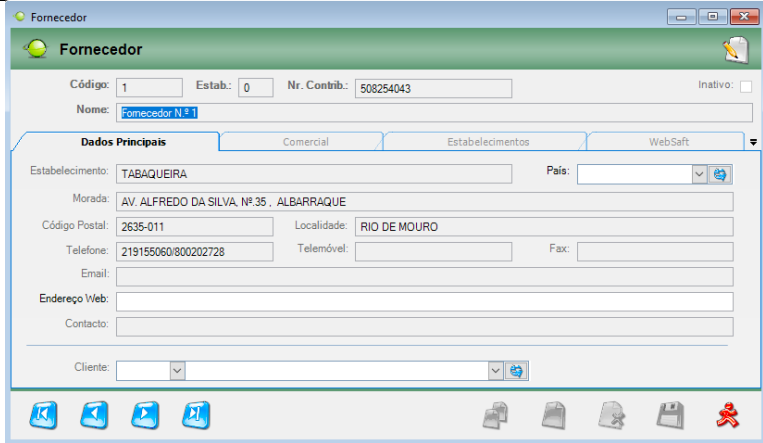
<i>Nome</i>	Manter Fornecedores
<i>ID</i>	UC_WVB_1010
<i>Descrição</i>	Criar, alterar e remover um fornecedor ou estabelecimento
<i>Atores</i>	Utilizador (Administrativo, Administrador)
<i>Pré-condições</i>	O utilizador está autenticado e tem permissões para executar as diferentes operações (criar, alterar, remover).
<i>Cenário Principal</i>	Ao abrir a janela de “Fornecedores” (c.f. Figura 96), visualizam-se, no topo, os dados identificativos do cliente (código, número de estabelecimento, nome e número de contribuinte) e uma opção para o inativar. No centro da janela, encontra-se um conjunto de separadores, cuja visibilidade depende do facto de se tratar de um fornecedor, ou estabelecimento. Por se tratar de informação relativa a cada estabelecimento particular, considera-se o separador “Dados Principais” como o único separador comum. Especificam-se os próximos separadores, para uso exclusivo do fornecedor: (i) “Estabelecimentos” e (ii) “WebSAFT”.
<i>Exceções</i>	EX 1 Falta o preenchimento de campos obrigatórios (e.g. código e número de estabelecimento, nome e número de contribuinte) <ol style="list-style-type: none"> <li>1. O sistema notifica o utilizador que o campo é de preenchimento obrigatório;</li> <li>2. Posiciona o cursor no campo em falta;</li> <li>3. Retorna, em modo de edição, para o ponto em que se encontrava.</li> </ol>
<i>Janela</i>	 <p>Figura 96 – Janela Fornecedor – Separador Dados Principais</p>

Tabela 47 – Caso de uso de Manter Estabelecimentos Fornecedor

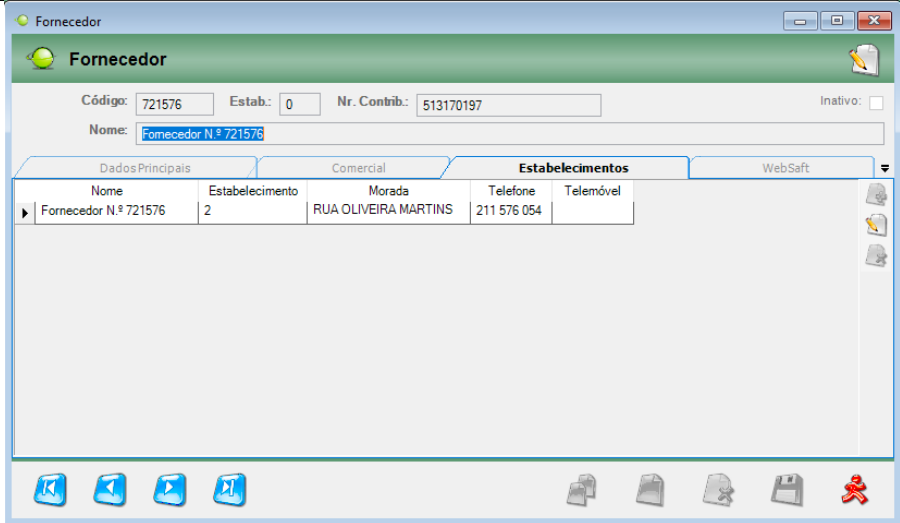
<i>Nome</i>	Manter Estabelecimentos Fornecedor
<i>ID</i>	UC_WVB_1011
<i>Descrição</i>	Criar, alterar e remover um estabelecimento
<i>Atores</i>	Utilizador (Administrativo, Administrador)
<i>Pré-condições</i>	O utilizador está autenticado e tem permissões para executar as diferentes operações (criar, alterar, remover).
<i>Cenário Principal</i>	Ao seleccionar o separador “Estabelecimentos” (c.f. Figura 97), apresenta-se uma tabela com os dados do estabelecimento: nome, código, morada, telefone e telemóvel. Além de visualizar, o utilizador pode, através do menu lateral da tabela, ou por meio de um menu de contexto, efetuar uma das seguintes ações: criar, alterar ou remover um registo.
<i>Janela</i>	 <p>Figura 97 – Caso de uso de Separador Estabelecimentos Fornecedor</p>

Tabela 48 – Caso de uso de Manter WebSAFT

<i>Nome</i>	Manter WebSAFT
<i>ID</i>	UC_WVB_1012
<i>Descrição</i>	Editar os dados relativos à configuração do WebSAFT.
<i>Atores</i>	Utilizador (Administrativo, Administrador)
<i>Pré-condições</i>	O utilizador está autenticado e tem permissões de edição.

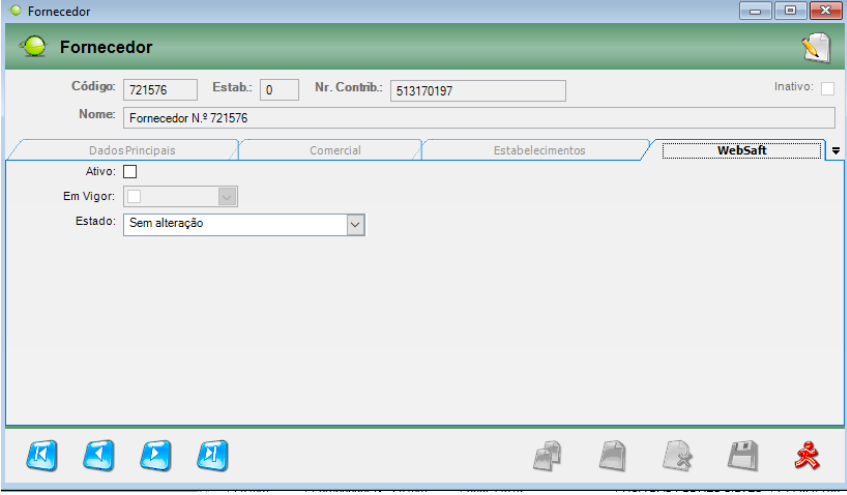
<i>Cenário</i>	Ao selecionar o separador “WebSAF” (c.f. Figura 98), apresentam-se os dados relativos à configuração do WebSAFT: ativar / inativar o fornecedor nessa solução, data de entrada em vigor da ativação e, em modo de leitura, o estado de alteração do fornecedor (e.g. “Sem Alteração”, “Pendente Criação”).
<i>Janela</i>	 <p style="text-align: center;">Figura 98 – Janela Fornecedor – Separador WebSAFT</p>

Tabela 49 – Caso de uso de Manter Modelos Máquina

<i>Nome</i>	Manter Modelos
<i>ID</i>	UC_WVB_1060
<i>Descrição</i>	Criar, alterar e remover um modelo de máquina
<i>Atores</i>	Utilizador (Administrativo, Administrador)
<i>Pré-condições</i>	O utilizador está autenticado e tem permissões para executar as diferentes operações (criar, alterar, remover).
<i>Cenário Principal</i>	<p>O separador “Dados Principais” da janela “Modelo Máquina” (c.f. Figura 101) compõe-se por três separadores: (i) “Dados Principais”, (ii) “Desenho” e (iii) “Máquinas”.</p> <p>No separador de “Dados Principais”, mantém-se os dados do modelo: nome, fabricante, indicador de comunicação por telemetria, modelo e protocolo de comunicação (apenas visível em telemetria), tipo de contabilidades (por canal ou seleção), indicador de apagar contabilidades, indicador da capacidade de programar a máquina, indicador de habilidade para</p>

	<p>programar múltiplas configurações de produto, número de seleções e canais, número da primeira seleção e canal, número de linhas e colunas, descrição do número de teclas indisponíveis (utiliza-se o separador “,” para distinguir diferentes números), número de moedeiros, indicador de presença de noteiros e orientação da numeração do <i>facing</i> da máquina.</p>
<i>Exceções</i>	<p>EX 1 Falta o preenchimento de, pelo menos, um dos campos obrigatórios como: nome, fabricante, modelo ou protocolo. Estes dois últimos campos são referidos, apenas se for permitida a comunicação por telemetria.</p> <ol style="list-style-type: none"> <li>1. O sistema notifica o utilizador para a obrigatoriedade do preenchimento do campo;</li> <li>2. Posiciona o cursor no campo em falta;</li> <li>3. Retorna, em modo de edição, para o separador “Dados Principais”.</li> </ol> <p>EX 2 Número de linhas a multiplicar pelo número de colunas é superior ao número de , ou igual a zero.</p> <ol style="list-style-type: none"> <li>1. O sistema notifica que o número de linhas / colunas é superior ao número de seleções, ou igual a zero;</li> <li>2. Posiciona o cursor no campo em falta;</li> <li>3. Retorna, em modo de edição, para o separador “Dados Principais”.</li> </ol> <p>EX 3 Formato incorreto do texto de teclas indisponíveis (e.g. conter um número de seleção superior ao número de seleções).</p> <ol style="list-style-type: none"> <li>1. O sistema notifica que o número da seleção deve ser maior que zero e menor, ou igual, ao número de seleções e que o carater separador de teclas é a vírgula;</li> <li>2. Posiciona o cursor no campo “teclas indisponíveis”;</li> <li>3. Retorna, em modo de edição, para o separador “Dados Principais”.</li> </ol>
<i>Diagrama conceptual de classes</i>	<p>Constata-se, na Figura 99, que uma máquina tem um modelo e, por sua vez, o modelo tem um fabricante e pode, ou não, ter um modelo base associado. Na prática, um modelo, que permita comunicação por telemetria, deve ter atribuído, obrigatoriamente, um modelo base.</p>

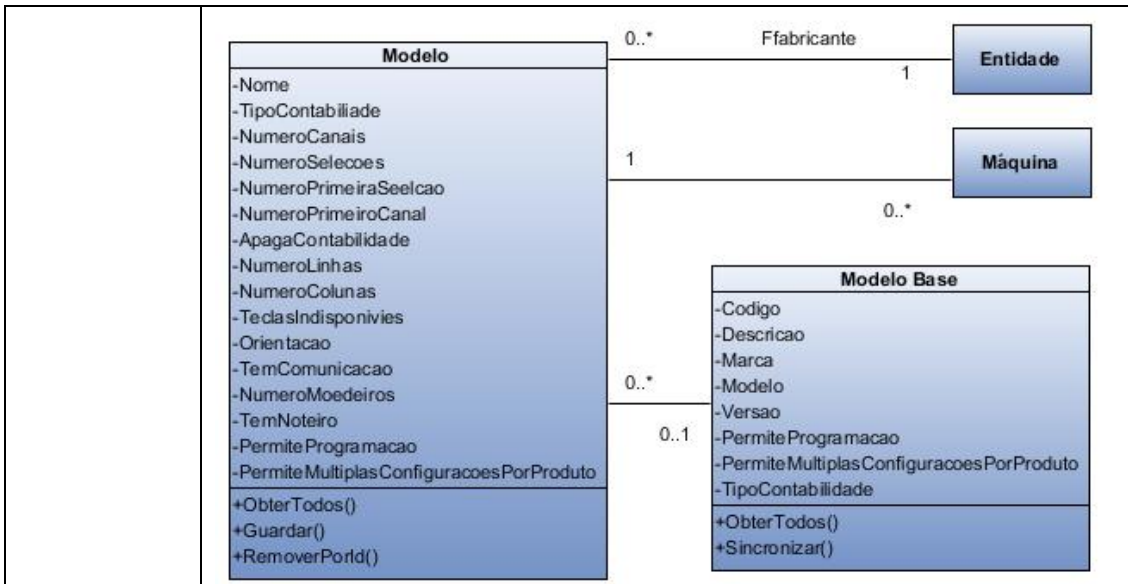


Figura 99 – Diagrama conceptual de classes da visão de *modelo*

*Modelo de dados* Destaca-se, na Figura 100, a tabela do modelo (denominada “Tpm”) e as associadas: (i) entidade fabricante (denominada “Ent”) e (ii) tipo de modelo base (denominada de “Tmb”). Observa-se, também, a relação entre a máquina (denominada de “Maq”) e o modelo.



Figura 100 – Modelo de dados da visão do modelo

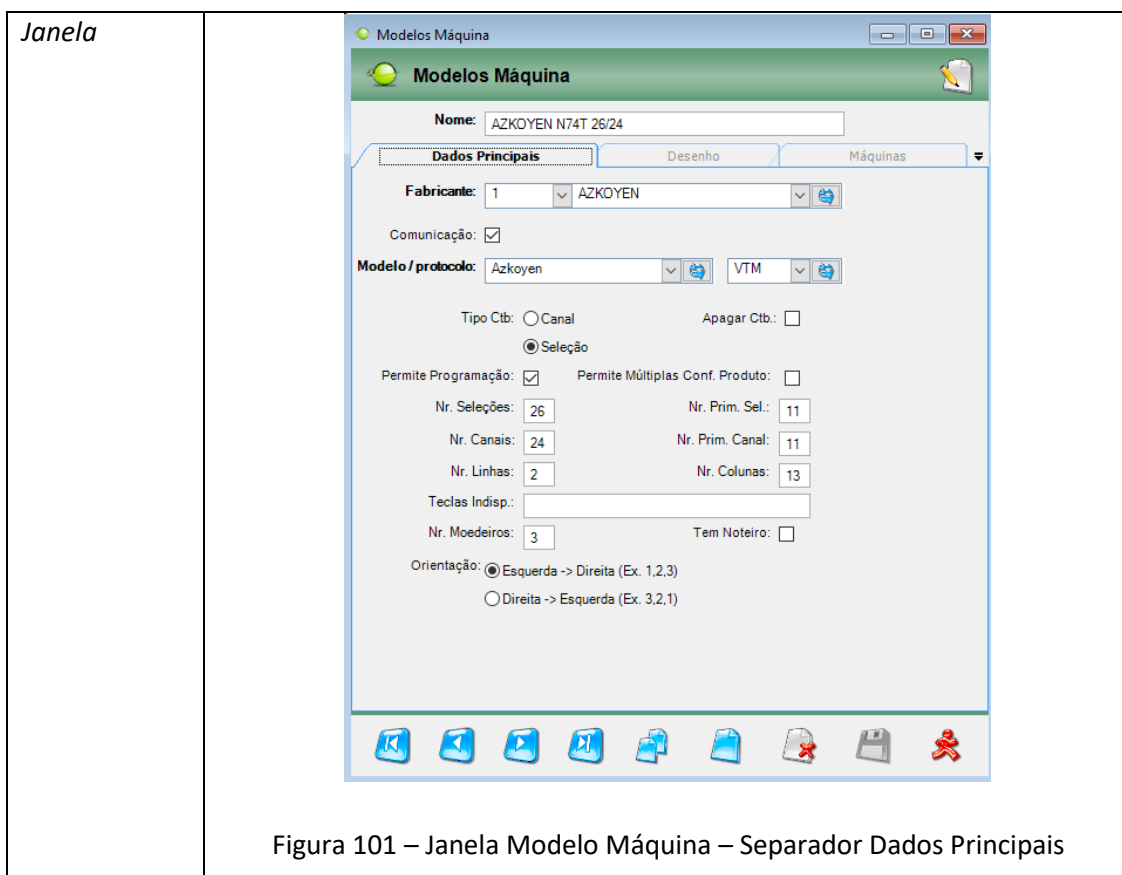


Tabela 50 – Caso de uso de Consultar *Facing* Modelo

<i>Nome</i>	Consultar <i>Facing</i> Modelo
<i>ID</i>	UC_WVB_1061
<i>Descrição</i>	Consultar desenho de <i>facing</i> e canais do modelo
<i>Atores</i>	Utilizador (Administrativo, Administrador)
<i>Pré-condições</i>	O utilizador está autenticado e tem permissões para visualizar o modelo.
<i>Cenário Principal</i>	<p>No separador “Desenho” da janela “Modelo Máquina” (c.f. Figura 102), visualiza-se a construção automática do desenho predefinido das seleções e canais, num determinado modelo.</p> <p>Para o desenho das seleções, recorre-se aos parâmetros: número de seleções, número da primeira seleção, número de linhas e orientação da numeração. Inicia-se a numeração da construção, através do parâmetro: (i) “número da primeira seleção”, caso a orientação seja da esquerda para a direita (E-D), ou (ii) “número de seleções”, caso a orientação seja da direita</p>

para a esquerda (D-E). Posteriormente, criam-se tantas caixas de desenho quanto o “número de seleções” definido. Dependendo da orientação (E-D ou D-E), a próxima caixa pode ter, respetivamente, um número de seleção imediatamente a seguir, ou anterior ao atual. Para determinar o número de linhas, divide-se o número de seleções pelo número de linhas e o resultado inteiro equivale ao número máximo de seleções por linha.

Janela

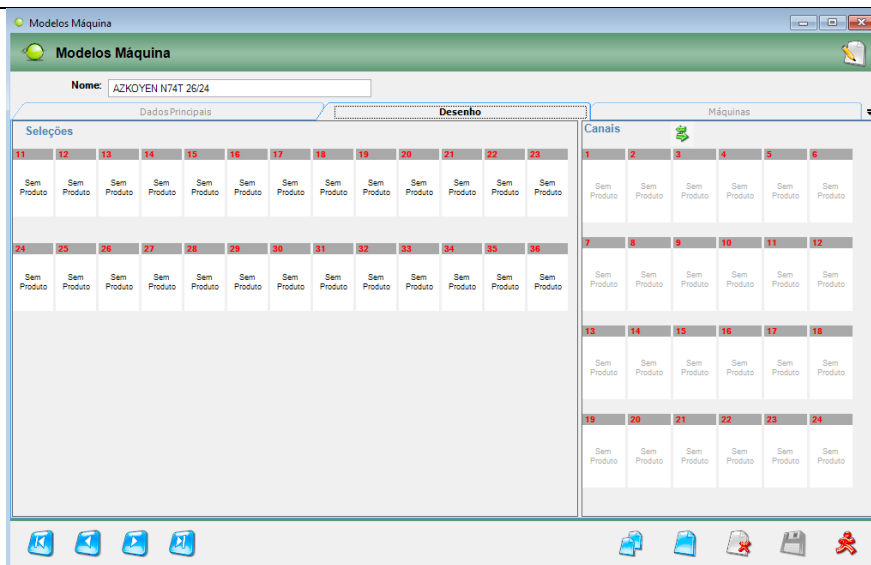


Figura 102 – Janela Modelo Máquina – Separador Desenho

Tabela 51 – Caso de uso de Consultar Máquinas Associadas ao Modelo

<i>Nome</i>	Consultar Máquinas Associadas ao Modelo
<i>ID</i>	UC_WVB_1062
<i>Descrição</i>	Consultar máquinas associadas ao modelo
<i>Atores</i>	Utilizador (Administrativo, Administrador)
<i>Pré-condições</i>	O utilizador está autenticado e tem permissões para visualizar o modelo.
<i>Cenário</i>	No separador “Máquina” da janela “Modelo Máquina” (c.f. Figura 103), visualiza-se a lista de máquinas, cujo modelo é igual ao modelo em observação. Identifica-se a máquina pelos seguintes campos: código, número do cliente e número e nome do estabelecimento.  Disponibiliza-se a capacidade de navegar diretamente para a janela da máquina, através de um duplo <i>click</i> , na linha da máquina.

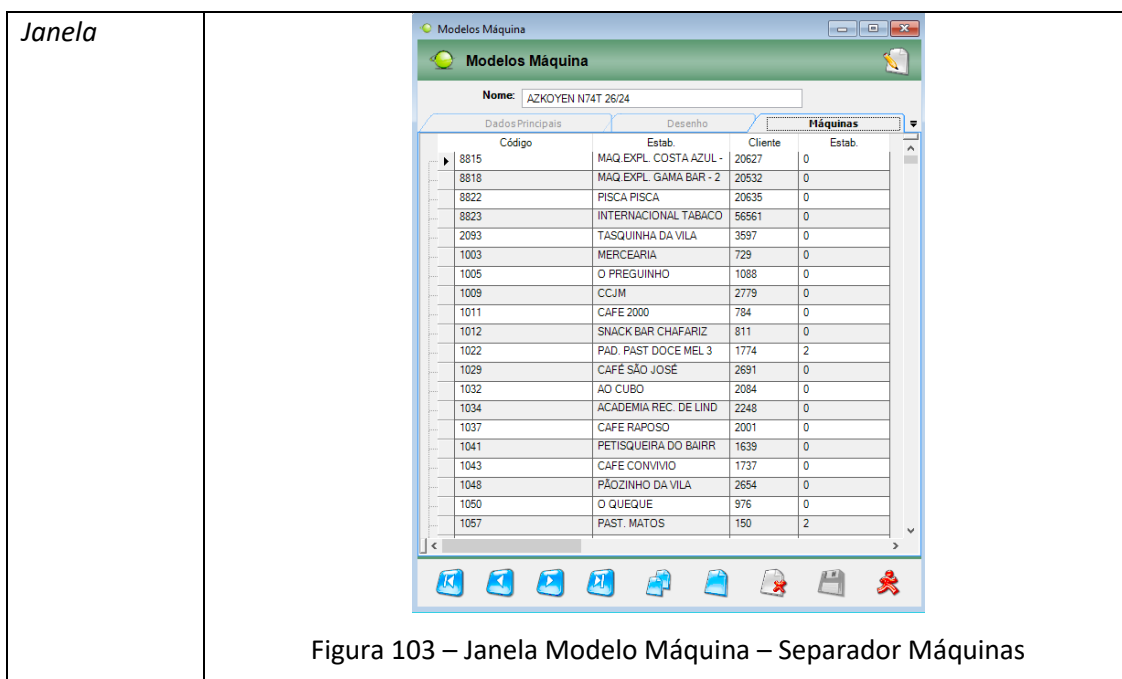


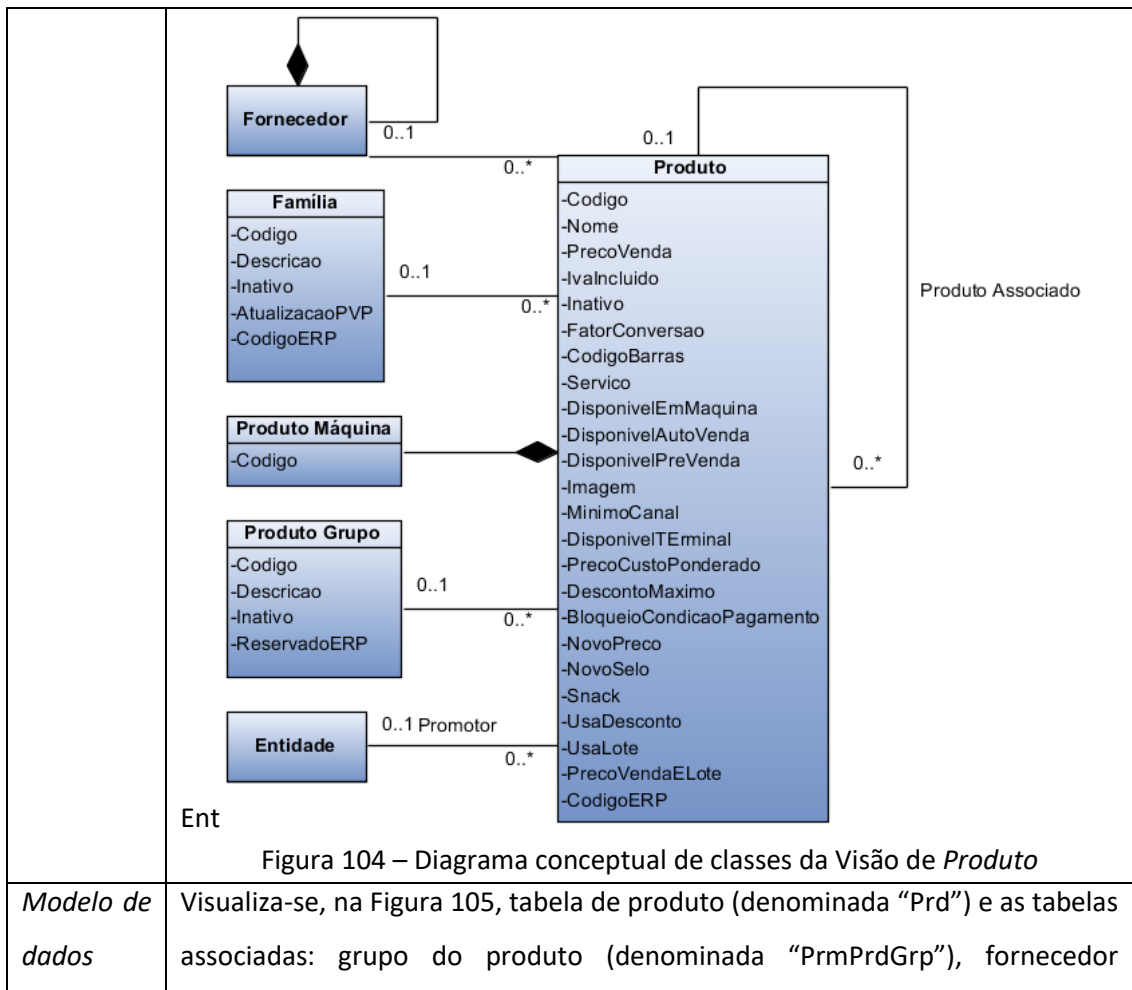
Tabela 52 – Caso de uso de Manter Produtos

<i>Nome</i>	Manter Produtos
<i>ID</i>	UC_WVB_1110
<i>Descrição</i>	Criar, alterar e remover um produto
<i>Atores</i>	Utilizador (Administrativo, Administrador)
<i>Pré-condições</i>	O utilizador está autenticado e tem permissões para executar as diferentes operações (criar, alterar, remover).
<i>Cenário</i>	<p>O separador “Principal” da janela de “Produto” (c.f. Figura 106) é composto por três separadores: (i) “Dados Principais”, (ii) “Códigos de Produto Máquina” e (iii) “Outros Dados”.</p> <p>No topo da janela, mantêm-se os dados identificadores do produto, tais como: referência, nome, código de barras e indicador, se se tratar, ou não, de um serviço. Através da opção “escolher” do menu de contexto, ou de um duplo <i>click</i> sobre a imagem do produto, associa-se uma imagem ao mesmo. Remove-se a associação através da opção “remover”, do mesmo menu de contexto. Em</p>

	<p>modo de leitura, são visíveis os códigos de produto máquina em formato CSV<sup>166</sup>. No separador de “Dados Principais”, mantêm-se os dados do produto: família, produto associado (e.g. a vinheta associa-se ao respetivo produto), promotor, fornecedor, preço de venda e compra, taxa de IVA, fator de conversão (e.g. a venda pode concretizar-se ao volume ou ao maço) e campos indicadores da disponibilidade do produto (máquina, terminal, auto venda, pré-venda). No separador “Códigos de Produto Máquina”, expõe-se no caso de uso UC_WVB_1111 (c.f. Tabela 53). No separador “Outros Dados”, mantêm-se dados complementares do produto (e.g. usa desconto, usa lote).</p>
<i>Exceções</i>	<p>EX 1 Falta o preenchimento de, pelo menos, um dos campos obrigatórios como: referência e designação.</p> <ol style="list-style-type: none"> <li>1. O sistema notifica o utilizador para a obrigatoriedade do preenchimento do campo;</li> <li>2. Posiciona o cursor no separador “Dados Principais” e no campo em falta;</li> <li>3. Retorna, em modo de edição.</li> </ol>
<i>Diagrama conceptual de classes</i>	<p>Observa-se, na Figura 104, que o produto pode ou não ter associação com as entidades: (i) “família”, (ii) “grupo de produto” e (iii) “promotor”. O “grupo de produto” identifica a categoria do produto (e.g. material publicitário). O “promotor” indica a entidade fabricante do produto (e.g. Tabaqueira).</p> <p>O produto pode ser composto por um conjunto de códigos de produto máquina que possibilita, para modelos distintos de máquinas, serem utilizados números, também, diferentes.</p> <p>A referência circular, visível no produto, justifica-se pela necessidade de identificar que, por exemplo, o produto “Vinheta do produto ABC” diz respeito ao produto “ABC”.</p>

---

<sup>166</sup> Comma-separated values (CSV) é um termo que indica um formato de texto separado por um delimitador de campo específico (e.g. vírgula) (Comma-separated values 2017).



(denominada “Frn”), família (denominada “Fam”) e entidade promotor (denominada “Ent”).

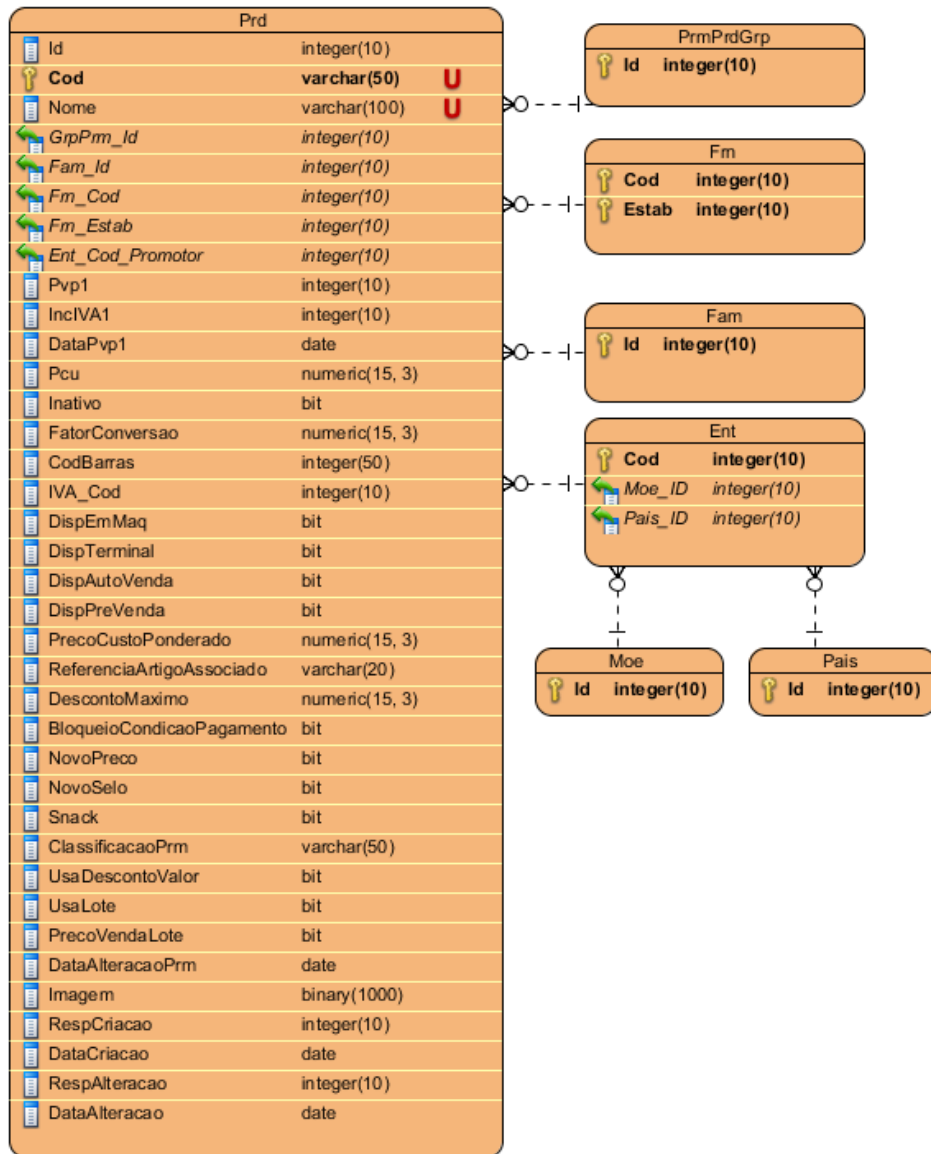


Figura 105 – Modelo de dados da visão de produto

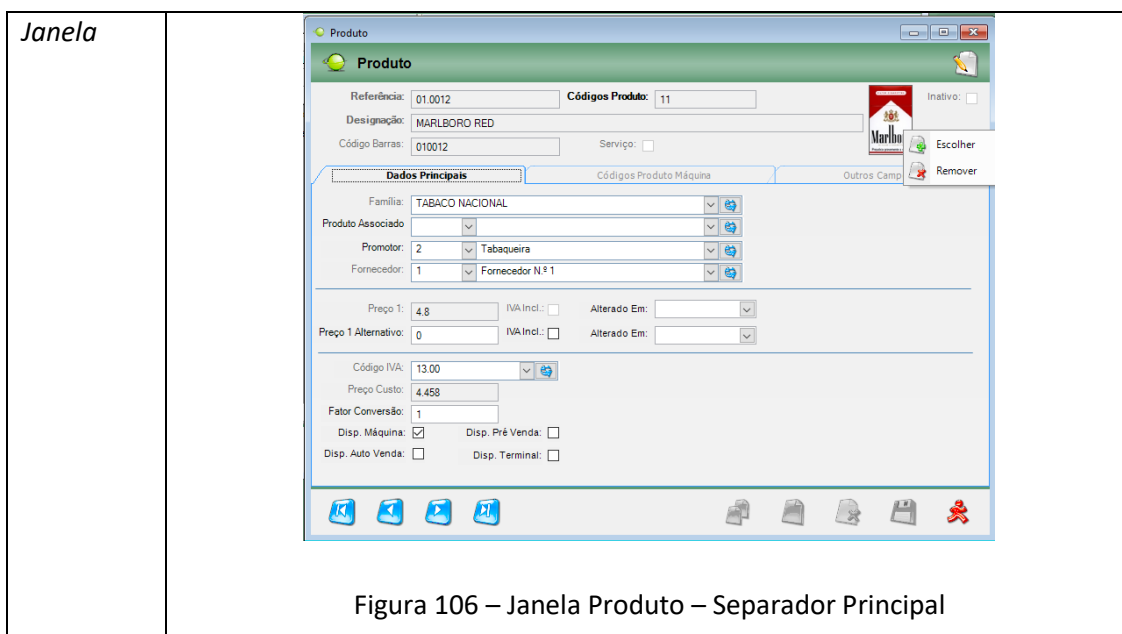


Tabela 53 – Caso de uso de Manter Códigos de Produto Máquina

<i>Nome</i>	Manter Códigos de Produto Máquina
<i>ID</i>	UC_WVB_1111
<i>Descrição</i>	Criar, alterar e remover os códigos de produto máquina
<i>Atores</i>	Utilizador (Administrativo, Administrador)
<i>Pré-condições</i>	O utilizador está autenticado e tem permissões para executar as diferentes operações (criar, alterar, remover).
<i>Cenário</i>	No separador “Códigos Máquina” da janela “Produto” (c.f. Figura 107), adiciona-se, remove-se ou altera-se um código de produto máquina. Executam-se as operações de adição e remoção, através do menu de contexto, ou diretamente no menu lateral da tabela de produtos máquina. Ao criar um novo código, atribui-se, automaticamente, o primeiro código de produto máquina livre. A edição do número é efetuada diretamente, na célula da tabela.
<i>Exceções</i>	<p>EX 1 O código de produto máquina encontra-se repetido na lista em manutenção, ou encontra-se associado a outra máquina.</p> <p>1. O sistema notifica o utilizador para o facto de o campo código de produto máquina ter de ser único no sistema;</p>

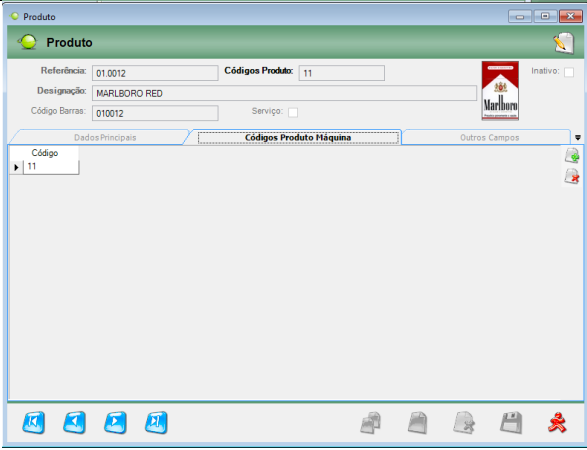
	<ol style="list-style-type: none"> <li>2. Posiciona o cursor no separador “Dados Principais” e na linha onde ocorre o erro;</li> <li>3. Retorna, em modo de edição.</li> </ol>
Janela	 <p>Figura 107 – Janela Produto – Separador Códigos Máquina</p>

Tabela 54 – Caso de uso de Manter Periodicidades

<i>Nome</i>	Manter Periodicidades
<i>ID</i>	UC_WVB_1120
<i>Descrição</i>	Criar, alterar e remover periodicidades
<i>Atores</i>	Utilizador (Administrativo, Administrador)
<i>Pré-condições</i>	O utilizador está autenticado e tem permissões para executar as diferentes operações (criar, alterar, remover).
<i>Cenário</i>	Na janela “Periodicidade” (c.f. Figura 109), mantêm-se, no topo, os dados: código e o nome da periodicidade. Ao centro, marca-se, diretamente no calendário, os dias da visita comercial. Efetua-se a marcação através de um duplo <i>click</i> , ou do pressionar da tecla ENTER, no dia específico. Remove-se a marcação pelos mesmos meios utilizados para adicionar.
<i>Exceções</i>	<p>EX 1 Falta o preenchimento de, pelo menos, um dos campos obrigatórios como: código e nome.</p> <ol style="list-style-type: none"> <li>4. O sistema notifica o utilizador para a obrigatoriedade do preenchimento do campo;</li> <li>5. Posiciona o cursor no separador “Dados Principais” e no campo em falta;</li> <li>6. Retorna, em modo de edição.</li> </ol>


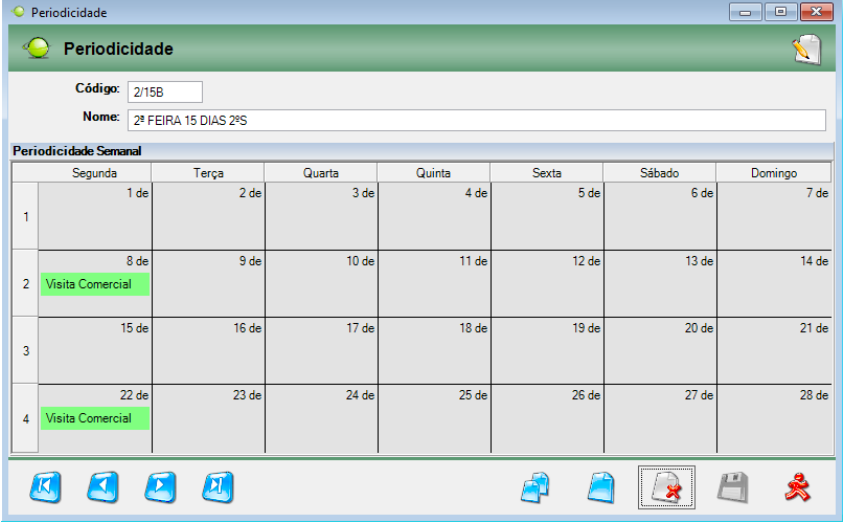
<p><i>Diagrama conceptual de classes</i></p>	<p>Na Figura 108, confirma-se que um cliente pode ter uma periodicidade associada e, por sua vez, uma periodicidade é composta por compromissos que referem o dia e o tipo de visita efetuado (e.g. comercial).</p>  <pre> classDiagram     Cliente "0..*" -- "0..1" Periodicidade     Periodicidade "0..1" *-- PeriodicidadeCompromisso     class Periodicidade {         -Codigo         -Nome     }     class PeriodicidadeCompromisso {         -Dia         -Tipo     } </pre> <p>Figura 108 – Diagrama conceptual de classes da Visão de Periodicidade</p>
<p><i>Janela</i></p>	 <p>Figura 109 – Janela Periodicidade</p>

Tabela 55 – Caso de uso de Manter Rotas

<i>Nome</i>	Manter Rotas
<i>ID</i>	UC_WVB_1130
<i>Descrição</i>	Criar, alterar e remover rotas
<i>Atores</i>	Utilizador (Administrativo, Administrador)
<i>Pré-condições</i>	O utilizador está autenticado e tem permissões para executar as diferentes operações (criar, alterar, remover).
<i>Cenário Principal</i>	Na janela “Rota” (c.f. Figura 112), mantêm-se os dados: nome da rota, vendedor, indicador de rota ativa e cor associada (utilizada para distinguir as rotas, na representação das máquinas, no mapa). Representa-se, no canto superior direito, um calendário genérico de quatro semanas, onde se assinala com cor, automaticamente, todos os dias em que se visita, pelo

	<p>menos, uma máquina e distinguem-se com as cores: (i) cor-de-laranja, os dias cujas visitas não foram confirmadas pelo utilizador e (ii) verde, os dias confirmados. Ao centro da janela, apresenta-se uma tabela com: a ordem de visita, o cliente e respetivo estabelecimento, o identificador de presença de máquina no cliente e o indicador de visita fixa (ao calcular, ou recalculer uma rota, as visitas fixas não são afetadas).</p> <p>Começa-se uma nova rota sem qualquer visita associada. Disponibilizam-se, dois modos distintos de associar visitas: (i) automática, através da execução de uma das calculadoras (mensal ou diária) e (ii) manual, onde se seleciona um dia específico genérico e se atribuem os clientes a visitar.</p> <p>No processo manual, seleciona-se um dia e acrescenta-se, diretamente na tabela, uma nova linha. Ao acrescentar-se uma linha, atribui-se, imediatamente, a próxima ordem de visita e assinala-se a linha como fixa (valor predefinido). Posteriormente, no ato de atribuição do cliente, predefine-se o valor zero ao estabelecimento e indica-se, automaticamente, se o cliente tem, ou não, alguma máquina associada.</p> <p>Na primeira etapa de execução da calculadora de rota mensal, o sistema obtém uma lista temporária de clientes associados ao vendedor e respetivos dias associados. Para cada dia da lista temporária, executam-se os passos: (i) obtém-se o dia da lista atual, caso exista, recuperam-se as visitas atualmente em vigor, se não existir, cria-se um novo dia, (ii) removem-se as linhas não fixas, cujo cliente não conste da lista temporária, (iii) recuperam-se as linhas assinaladas para remoção e cujo cliente pertença à lista temporária, (iv) criam-se as linhas que ainda não tenham representação na tabela de visitas do dia e (v) assinalam-se as opções “tem visitas” e “sugestão por confirmar”. Na segunda etapa, e para os dias que existem na lista em vigor, mas não têm representação na lista temporária, procede-se aos seguintes passos: carregam-se as visitas do dia, ainda não obtidas, marcam-se, para remoção, as visitas e o respetivo dia.</p> <p>O menu lateral contém algumas ações complementares à gestão da rota: subir e descer a ordem de uma linha, calcular a rota diária (funcionamento similar ao da calculadora mensal, mas afeta a um dia) e libertar todas as</p>
--	--

	linhas fixas, de modo a tornarem-se suscetíveis de remoção, no caso de não existirem.
<i>Exceções</i>	<p>EX 1 Falta o preenchimento de pelo menos um dos campos obrigatórios: nome e vendedor.</p> <p>7. O sistema notifica o utilizador para a obrigatoriedade do preenchimento do campo;</p> <p>8. Posiciona o cursor no separador “Dados Principais” e no campo em falta;</p> <p>9. Retorna, em modo de edição.</p>
<i>Diagrama conceptual de classes</i>	<p>Na Figura 110, visualizam-se todas as entidades envolvidas na construção da rota. Delega-se sempre uma rota a um vendedor que tem vários clientes atribuídos. Por sua vez, o cliente pode ter várias máquinas e uma periodicidade. Uma rota é constituída por um conjunto de dias e, em cada dia, podem ser visitados vários clientes.</p> <pre> classDiagram     class Periodicidade     class Máquina     class Cliente     class Vendedor     class RotaCabecalho {         -Nome         -Tipo         -Cor         -Ativa     }     class RotaDia {         -Dia         -SugestaoPorConfirmar     }     class RotaLinha {         -Ordem         -HoraInicio         -HoraFim         -TipoVisita         -Manual     }     Periodicidade "0..1" -- "0..*" Cliente     Máquina "0..*" -- "0..*" Cliente     Vendedor "1" -- "1" RotaCabecalho     RotaCabecalho "0..1" -- "0..*" RotaDia     RotaDia "0..*" -- "0..*" RotaLinha     Cliente "1" -- "0..*" RotaDia     Cliente "1" -- "0..*" RotaLinha     RotaCabecalho "1" -- "0..*" RotaDia     RotaDia "0..*" -- "0..*" RotaLinha   </pre> <p>Figura 110 – Diagrama conceptual de classes da Visão de Rota</p>
<i>Modelo de dados</i>	<p>Observam-se, na Figura 111, as três principais tabelas responsáveis por armanezar os dados da rota: (i) o cabeçalho da rota (denominada “RotC”), os dias com visitas previstas (denominada “RotD”) e as visitas previstas (denominada “RotL”). Uma vez que uma rota pertence a um vendedor, a tabela de cabeçalho da rota tem uma associação à tabela de vendedor. Assim como, a tabela de visitas prevista associa-se à tabela de clientes (denominada “Cli”) para identificar os clientes a visitar.</p>

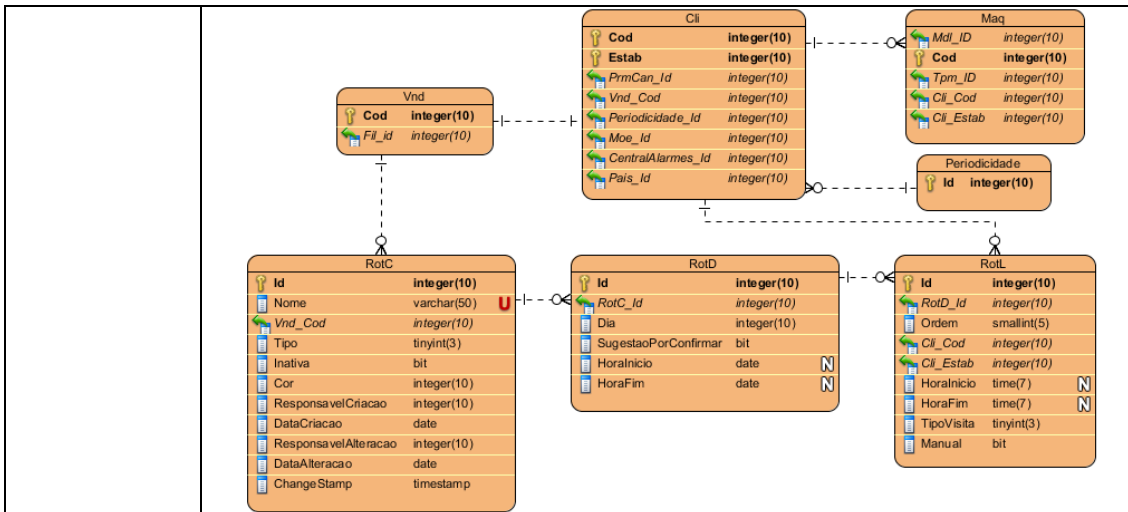


Figura 111 – Modelo de dados visão rota

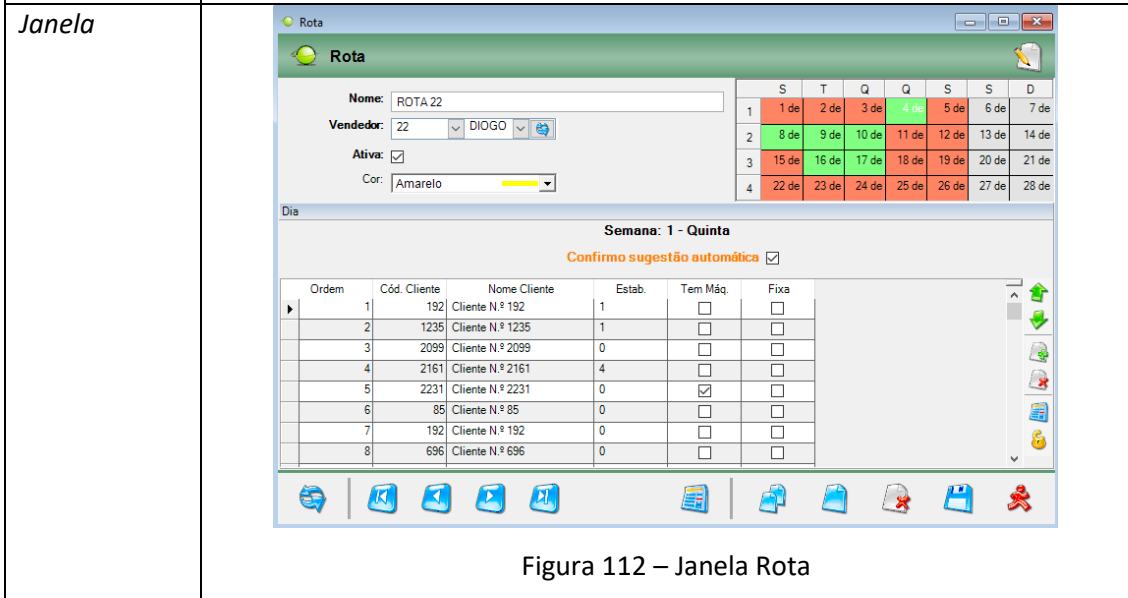


Figura 112 – Janela Rota

Tabela 56 – Caso de uso de Manter Agendamentos Comunicações

Nome	Manter Agendamentos Comunicações
ID	UC_WVB_1080
Descrição	Criar, alterar e remover agendamentos de comunicações
Atores	Utilizador (Administrativo, Administrador)
Pré-condições	O utilizador está autenticado e tem permissões para executar as diferentes operações (criar, alterar, remover).

<i>Cenário</i>	<p>No separador “Periodicidade” da janela “Agendamento” (c.f. Figura 115), mantêm-se os dados: nome e descrição do agendamento, indicador de inatividade, periodicidade (única ou diária), intervalo de validade, periodicidade, alvo, agendamento e parâmetros.</p> <p>O separador “Periodicidade” comporta-se de dois modos diferentes, consoante a periodicidade seja única ou diária. Na periodicidade única, define-se o dia e a hora de execução. A hora de execução baseia-se nas seguintes opções: (i) hora específica, (ii) hora definida na ficha da máquina, (iii) hora de abertura / fecho da filial e (iv) hora de abertura / fecho de estabelecimento. Ao definir-se uma hora, pode-se especificar um momento exato, antes ou depois de X minutos do evento selecionado. Na periodicidade diária, especifica-se a base de execução do agendamento e os respetivos parâmetros, atrás referidos. Apresentam-se os próximos separadores, nos respetivos caso de uso: (i) “Alvo” (c.f. Tabela 57 – Caso de uso de Manter Alvos de Agendamento), (ii) “Histórico” (c.f. Tabela 58 – Caso de uso de Consultar Histórico de Agendamento) e (iii) “Parâmetros” (c.f. Tabela 59).</p>
<i>Exceções</i>	<p>EX 1 Falta o preenchimento de pelo menos um dos campos obrigatórios como: nome, periodicidade, data de início de validade, base de execução e respetivos parâmetros.</p> <ol style="list-style-type: none"> <li>1. O sistema notifica o utilizador para a obrigatoriedade do preenchimento do campo;</li> <li>2. Posiciona o cursor no separador “Periodicidade” e no campo em falta;</li> <li>3. Retorna, em modo de edição.</li> </ol>
<i>Diagrama conceptual classes</i>	<p>Na Figura 113, observam-se as entidades envolvidas na elaboração do agendamento de comunicações. Assim, no cabeçalho do agendamento definem-se os parâmetros (e.g. momento de execução e critérios) para a sua execução. Nas linhas, atribuem-se os alvos que podem ser rotas ou máquinas. No histórico, registam-se os dados relativos ao agendamento efetuado em determinada data e, após o escalonamento, associam-se as comunicações ao cabeçalho do agendamento.</p>

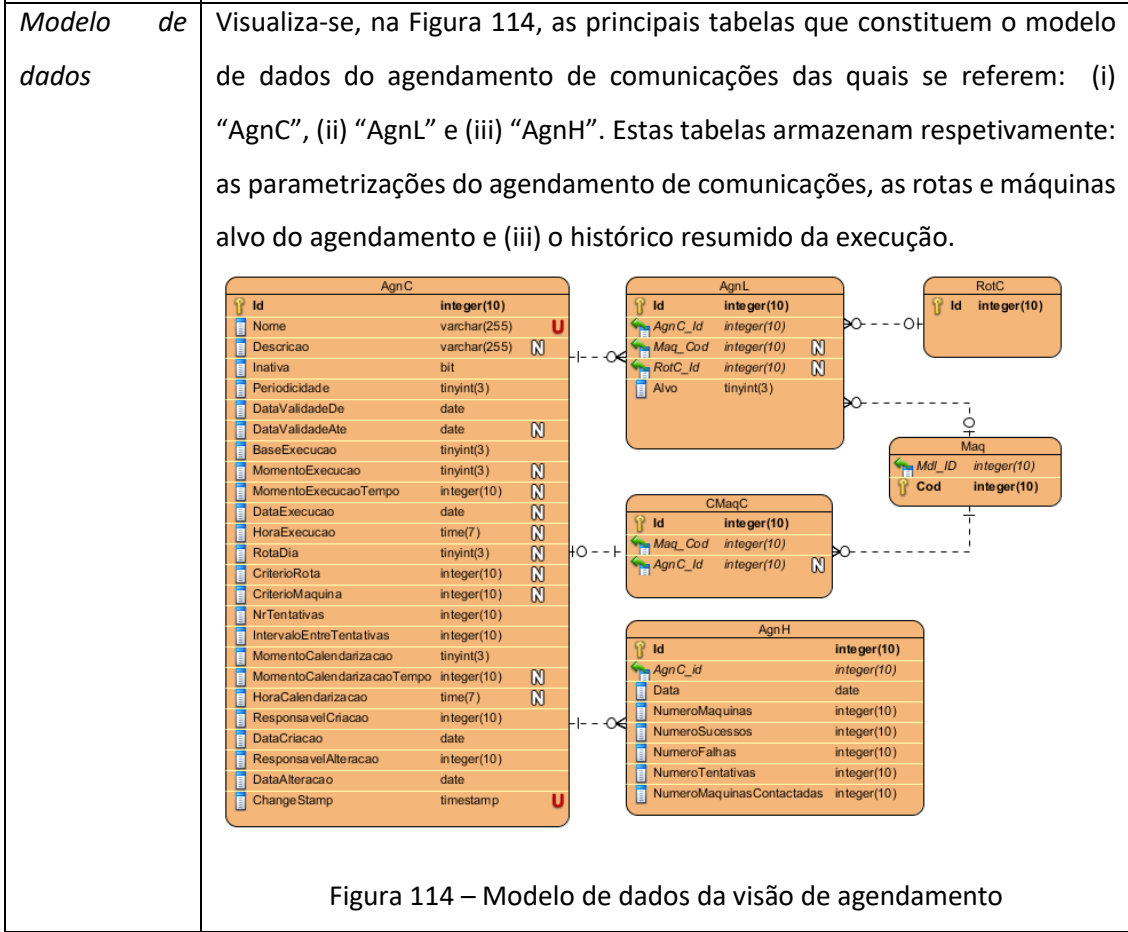
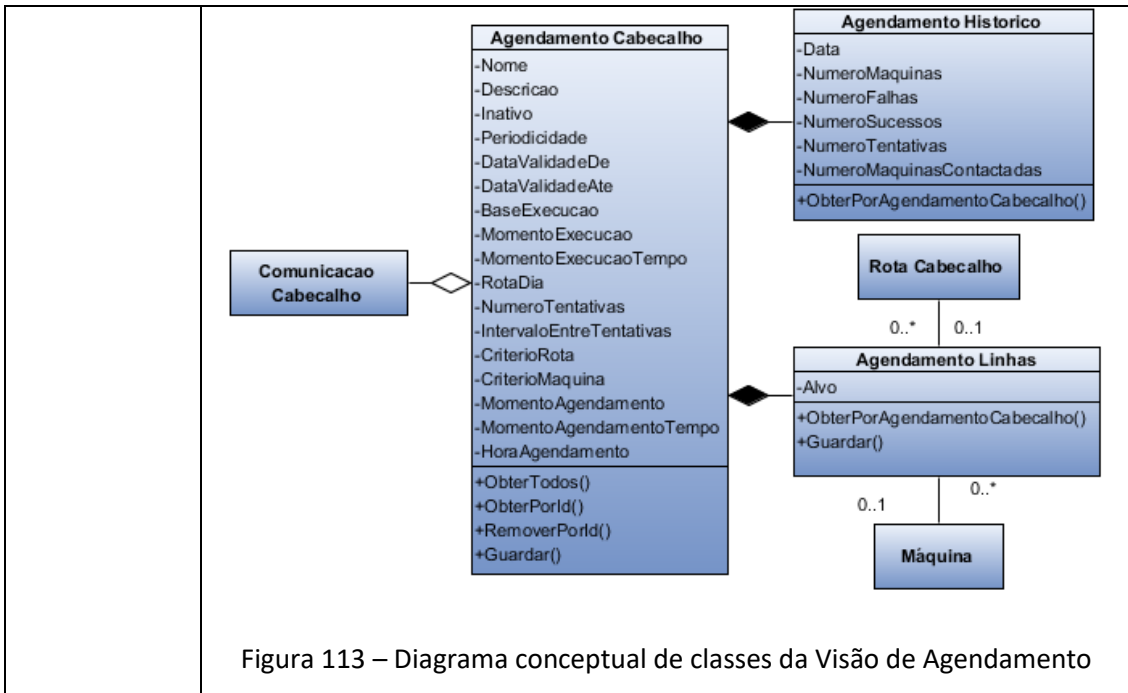




Tabela 57 – Caso de uso de Manter Alvos de Agendamento

<i>Nome</i>	Manter Alvos de Agendamento
<i>ID</i>	UC_WVB_1081
<i>Descrição</i>	Criar, alterar e remover alvos de agendamento
<i>Atores</i>	Utilizador (Administrativo, Administrador)
<i>Pré-condições</i>	O utilizador está autenticado e tem permissões de edição (criar ou alterar).
<i>Cenário</i>	No separador “Alvo” da janela “Agendamento” (c.f. Figura 116), mantêm-se os alvos de agendamento: (i) por rota e (ii) por máquina. Por rota / máquina especifica-se o critério de atuação: (i) nenhum (o critério é ignorado), (ii) todas (contemplam-se todas as rotas / máquinas) e (iii) específicas (selecionam-se as rotas / máquinas com as quais se pretende comunicar). No critério “específica(s)”, disponibilizam-se duas opções: (i) nenhum (retira-se a seleção de todos os elementos) e (ii) todos (selecionam-se todos os elementos). Quando se seleciona uma rota, pretende-se comunicar com todas as máquinas dessa rota. Caso seja selecionada uma rota e várias máquinas e, se alguma dessas

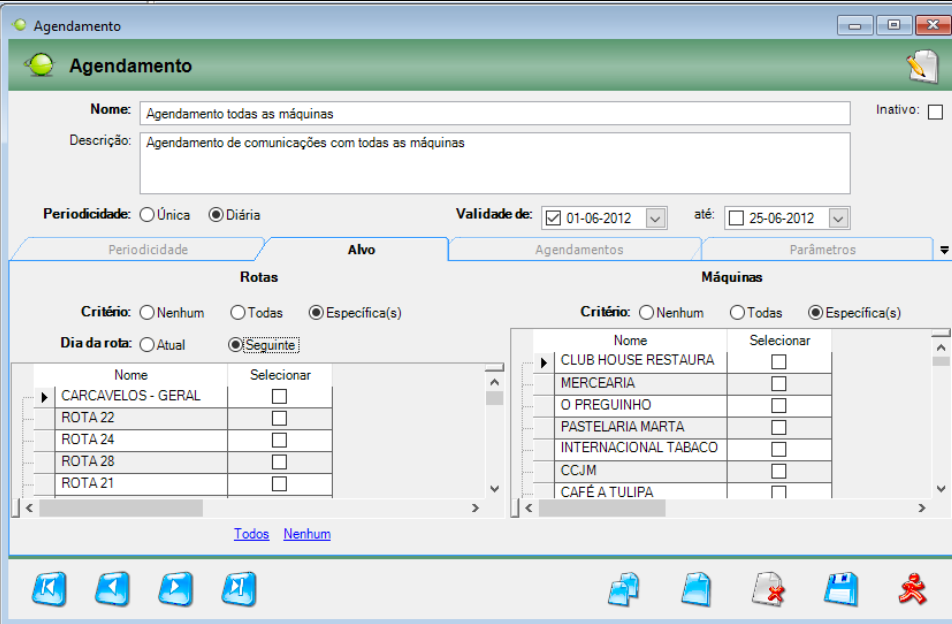
	máquinas pertencer a uma das rotas selecionadas, efetua-se apenas uma comunicação por máquina.
<i>Exceções</i>	<p>EX 1 Falta o preenchimento do critério de rota / máquina.</p> <p>EX 2 Falta o selecionar, pelo menos, uma rota ou máquina.</p> <ol style="list-style-type: none"> <li>1. O sistema notifica o utilizador para a obrigatoriedade do preenchimento do campo;</li> <li>2. Posiciona o cursor no separador “Alvo” e no campo em falha;</li> <li>3. Retorna, em modo de edição.</li> </ol>
<i>Janela</i>	 <p>Figura 116 – Janela Agendamento – Separador Alvo</p>

Tabela 58 – Caso de uso de Consultar Histórico de Agendamento

<i>Nome</i>	Consultar Histórico de Agendamento
<i>ID</i>	UC_WVB_1082
<i>Descrição</i>	Consultar o resumo dos agendamentos efetuados e o estado das comunicações solicitadas.
<i>Atores</i>	Utilizador (Administrativo, Administrador)
<i>Pré-condições</i>	O utilizador está autenticado e tem permissões de visualização.

<i>Cenário</i>	<p>No separador “Histórico” da janela “Agendamento” (c.f. Figura 117), visualiza-se o histórico de agendamento, através de uma tabela hierárquica, com dois níveis: (i) dia de agendamento e (ii) máquinas alvo da comunicação. No primeiro nível da hierarquia, observa-se: a data e hora de execução do agendamento, o número de máquinas alvo da comunicação, o número de contactos (número de máquinas com as quais se estabelece, pelo menos, um pedido de comunicação), número de sucessos (número de máquinas com as quais se estabelece uma leitura válida), número de falhas (número de comunicações com resultado negativo) e número de tentativas (número de vezes que se tenta comunicar). No segundo nível, visualizam-se os dados de agendamento de um conjunto de máquinas alvo do agendamento: número da máquina, nome do estabelecimento, estado da comunicação (e.g. completo, falha, suspenso), data de agendamento, data e hora do pedido e leitura de comunicação, número da tentativa e âmbito da tentativa. O âmbito da tentativa assume dois valores: (i) data de agendamento e (ii) pronta a comunicar. O âmbito “data de agendamento” significa que a comunicação com a máquina se efetua num determinado horário específico enquanto que, o âmbito “pronta a comunicar” implica que a comunicação se concretiza quando existir uma ocorrência indicadora de atividade da máquina (e.g. máquina ligada). O último âmbito evita que se efetuem tentativas de comunicação desnecessárias à máquina, isto porque, se a máquina está desligada, é irrelevante uma tentativa de comunicação, isto porque se conhece, de antemão, que o resultado será negativo.</p>
<i>Requisitos</i> <i>Desempenho</i>	<p>De modo a garantir disponibilidade da aplicação, carrega-se, em segundo plano, a lista de histórico de agendamento.</p>



Tabela 59 – Caso de uso de Manter Parâmetros de Agendamento

<i>Nome</i>	Manter Parâmetros de Agendamento
<i>ID</i>	UC_WVB_1082
<i>Descrição</i>	Editar parâmetros de agendamento
<i>Atores</i>	Utilizador (Administrativo, Administrador)
<i>Pré-condições</i>	O utilizador está autenticado e tem permissões de edição (criar ou alterar).
<i>Cenário</i>	No separador “Parâmetros” da janela “Agendamento” (c.f. Figura 118), mantêm-se os parâmetros do agendamento: (i) número máximo de tentativas, (ii) intervalo de tempo, em minutos, entre cada tentativa e (iii) hora da execução do agendamento.
<i>Exceções</i>	EX 1 Falta o preenchimento de, pelos menos, um dos campos obrigatórios: número tentativas, intervalo entre tentativas e hora de calendarização. <ol style="list-style-type: none"> <li>1. O sistema notifica o utilizador para a obrigatoriedade do preenchimento do campo;</li> <li>2. Posiciona o cursor no separador “Parâmetros” e no campo em falha;</li> <li>3. Retorna, em modo de edição.</li> </ol>

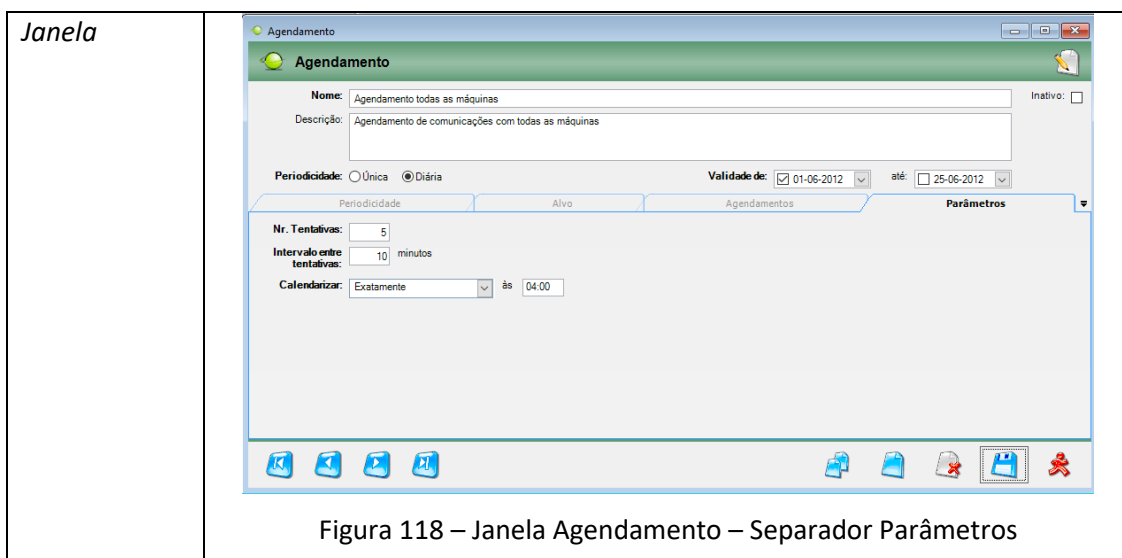


Tabela 60 – Caso de uso de Consultar Visão Geral Máquina

<i>Nome</i>	Consultar Visão Geral Máquina
<i>ID</i>	UC_WVB_2001
<i>Descrição</i>	Visualizar dados da máquina
<i>Atores</i>	Utilizador (Administrativo, Administrador)
<i>Pré-condições</i>	O utilizador está autenticado e tem permissões de visualização.
<i>Cenário</i>	Ao entrar na ficha da máquina, o utilizador tem acesso, no topo, aos dados identificadores da máquina: código, nome do estabelecimento, número de série, modelo, filial associada e indicador que assinala se está, ou não, ativa (e.g. uma máquina adquirida, ou retirada de um estabelecimento, tem o estado inativo). Ao centro da janela, conhecem-se os dados essenciais da máquina, com relevância para indícios potencialmente negativos (e.g. produtos esgotados, máquina ou cabo de dados desligado): (i) informação dos estados dos <sup>167</sup> sensores da máquina (ligado/desligado, porta aberta/fechada, cabo ligado/desligado e presença de conectividade), (ii)

<sup>167</sup> Os estados dos sensores da máquina encontram-se visíveis, no ato de abertura da ficha da máquina, se apresentarem um estado não esperado (e.g. cabo de dados desligado). No entanto, expande-se, por opção do utilizador, de modo a se visualizarem todos os estados.

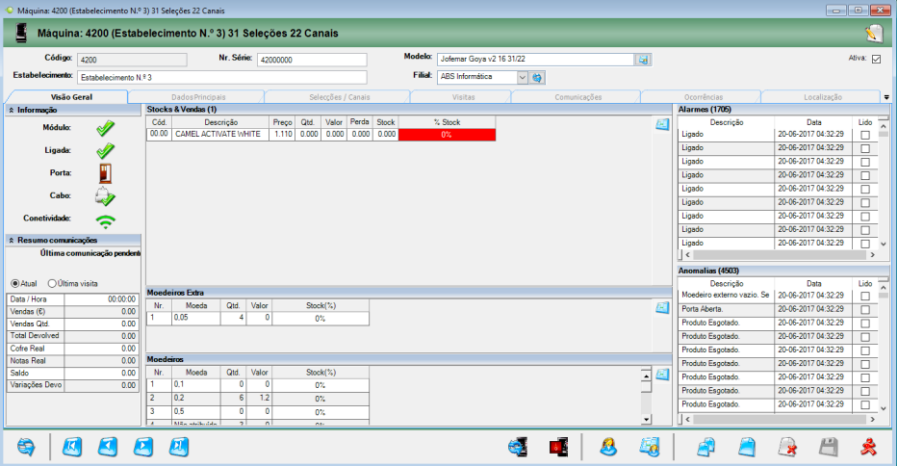
	<p>resumo dos contadores atuais da máquina, ou desde a última visita (data/hora da última comunicação, valor das vendas, quantidade de vendas, total em devolvedores, total em moedas/notas em cofre, saldo e variação de devolvedores), (iii) listagem de stocks e vendas por produto (código e descrição do produto, preço, quantidade, valor das vendas, valor das vendas perdidas, quantidade em stock<sup>168</sup>, percentagem<sup>169</sup>), (iv) listagem de moedeiros (número, moeda, quantidade, valor e percentagem<sup>170</sup>), (v) listagem de alarmes ocorridos desde a última visita à máquina (descrição, data/hora, indicador de leitura do alarme<sup>171</sup>), (vi) listagem de anomalias na máquina (descrição, data/hora<sup>172</sup>, indicador de leitura da anomalia<sup>173</sup>).</p>
<p><b>Requisitos Desempenho</b></p>	<p>De modo a garantir disponibilidade da aplicação, carregam-se, em segundo plano, as listas de: (i) stocks, (ii) alarmes, (iii) anomalias e (iv) moedeiros.</p>
<p><b>Janela</b></p>	 <p>The screenshot shows the 'Máquina: 4200' interface. It includes a top navigation bar with tabs for 'Visão Geral', 'Dados Principais', 'Seleções / Canais', 'Visitas', 'Comunicações', 'Alarmes', and 'Localização'. The 'Visão Geral' tab is active, displaying several data sections:</p> <ul style="list-style-type: none"> <li><b>Informação:</b> Status indicators for 'Módulo', 'Ligado', 'Porta', 'Cabo', and 'Conectividade'.</li> <li><b>Resumo comunicações:</b> 'Última comunicação pendente'.</li> <li><b>Stocks &amp; Vendas (1):</b> A table with columns: Cód., Descrição, Preço, Qtd., Valor, Perda, Stock, % Stock. One row is visible: '00.00 CAMEL ACTIVATE WHITE' with a 0% stock percentage.</li> <li><b>Moedeiros Estão:</b> A table with columns: Nr., Moeda, Qtd., Valor, Stock(%). One row is visible: '1 0,05 4 0 0%'.</li> <li><b>Moedeiros:</b> A table with columns: Nr., Moeda, Qtd., Valor, Stock(%). Three rows are visible, showing different coin denominations and their stock percentages.</li> <li><b>Alarmes (1700):</b> A table with columns: Descrição, Data, Lido. Multiple 'Ligado' entries are listed for the date 20-06-2017 at 04:32:29.</li> <li><b>Anomalias (400):</b> A table with columns: Descrição, Data, Lido. Entries include 'Moedeiro externo vazio: 5s', 'Porta Aberta', and 'Produto Esgotado'.</li> </ul>

Figura 119 – Janela Máquina – Separador Visão Geral

<sup>168</sup> Calcula-se a quantidade em stock da máquina, pela subtração da quantidade em stock do sistema ERP, pelo valor das vendas obtidas por telemetria.

<sup>169</sup> Define-se a percentagem de stock de um produto na máquina, pela divisão do somatório da capacidade de produtos, em cada canal onde está presente, pelo somatório da quantidade existente de cada produto, nos respetivos canais.

<sup>170</sup> Estabelece-se a percentagem de stock dos moedeiros da máquina, pela divisão da capacidade total de moedas, num moedeiro, pelo somatório de moedas presente, no mesmo.

<sup>171</sup> Ao assinalar-se um alarme como lido, retira-se, automaticamente, da listagem.

<sup>172</sup> A data / hora da anomalia é igual à data / hora da primeira ocorrência não corrigida de uma anomalia.

<sup>173</sup> Ao marcar-se uma anomalia como lida, esta mantém-se visível na listagem.

Tabela 61 – Caso de uso de Teste da Máquina

<i>Nome</i>	Testar Máquina
<i>ID</i>	UC_WVB_2002
<i>Descrição</i>	Verificar os estados dos sensores da máquina e confirmar se o módulo se encontra instalado, na máquina esperada
<i>Atores</i>	Utilizador (Administrativo, Administrador) e Sistema Middleware
<i>Pré-condições</i>	O utilizador está autenticado e tem permissões de visualização.
<i>Cenário</i>	A ação de testar a máquina consiste em dois passos: (i) efetuar-se um pedido de consulta dos estados da máquina e (ii) executar-se um pedido de comunicação. Rececionados os estados dos sensores, atualizam-se os dados relativos aos sensores, situados no separador “Visão Geral” (c.f. Tabela 60) e procede-se, caso os sensores indiquem que a máquina se encontra preparada para comunicar (i.e. máquina e cabos ligados e com conectividade), com um pedido de comunicação do tipo “Teste” ao sistema Middleware. Posteriormente, à receção da comunicação compara-se o código da máquina, enviado na comunicação, com o da ficha da máquina e procede-se de acordo com o resultado da comparação.
	<p>EX 1 Falha o pedido de comunicação com a máquina.</p> <p>O sistema notifica o utilizador da falha ocorrida;</p> <p>Retorna para a mesma janela.</p> <p>EX 2 O código enviado pela máquina é diferente do código da ficha da máquina.</p> <p>O sistema notifica o utilizador que o código da máquina é diferente do enviado na última comunicação (efetua-se a notificação no momento da ocorrência e sempre que se entre numa ficha de máquina nessas condições);</p> <p>Suspende-se a comunicação;</p> <p>Retorna para a mesma janela.</p>

Tabela 62 – Caso de uso de Manter Estado da Máquina

<i>Nome</i>	Manter Estado da Máquina
<i>ID</i>	UC_WVB_2002
<i>Descrição</i>	Ativar / desativar a componente de telemetria da máquina
<i>Atores</i>	Utilizador (Administrativo, Administrador) e Sistema Middleware
<i>Pré-condições</i>	O utilizador está autenticado e tem permissões de edição. A máquina está ativa e tem um módulo associado.
<i>Cenário principal</i>	Ao pressionar-se o botão de ativar a componente de telemetria da máquina, envia-se um pedido ao Middleware, para a ativação do módulo e efetua-se um registo no histórico de alterações com: (i) a data/hora da alteração do estado, (ii) o novo estado, (iii) o código da máquina e (iv) um comentário opcional do motivo de alteração. Caso o pedido de alteração seja bem-sucedido, altera-se o estado do botão para o modo de inativar. Com a ativação do módulo todos os alarmes da máquina são enviados até ao sistema e permite-se a comunicação e a programação com a máquina.
<i>Cenário Alternativo</i>	Pressiona-se o botão de inativar a componente de telemetria da máquina quando se pretende suspender o envio de alarmes, ou alterar o módulo de máquina. Ao efetuar-se essa ação, concretiza-se um pedido de inativação do módulo ao Middleware e registam-se os dados de histórico de alteração de estado do módulo. Ao rececionar-se a resposta positiva de inativação do módulo, substitui-se o estado do botão para o modo de ativação.
	EX 1 Falha o pedido de alteração do estado. O sistema notifica o utilizador da falha ocorrida; Retorna para a mesma janela.

Tabela 63 – Caso de uso de Manter Dados Principais Máquina

<i>Nome</i>	Manter Dados Principais Máquina
<i>ID</i>	UC_WVB_2010
<i>Descrição</i>	Visualizar dados da máquina
<i>Atores</i>	Utilizador (Administrativo, Administrador)
<i>Pré-condições</i>	O utilizador está autenticado e tem permissões de edição (criar ou alterar).
<i>Cenário</i>	No separador “Dados Principais” da janela “Máquina” (c.f. Figura 120), disponibilizam-se os seguintes dados para manutenção: (i) dados gerais do cliente, com a possibilidade de abertura da janela de manutenção de cliente (c.f. Tabela 43), (ii) vendedor, (iii) propriedade da máquina (empresa ou cliente), (iv) exploração da máquina (empresa ou cliente), (v) identificadores de chaveiros onde se guardam as chaves da máquina (e.g. porta, cofre), (vi) número de moedeiros da máquina, (vii) número de moedeiros acrescentados adicionalmente às máquinas, (viii) identificador de presença de discriminador <sup>174</sup> e noteiro, (ix) versão de <i>firmware</i> da máquina, (x) dados do agendamento (data / hora de comunicação, número de tentativas de comunicação e intervalo entre tentativas), (xi) manutenção do módulo, (xii) manutenção do código de operador, (xiii) manutenção da mensagem publicitária e (xiv) observações.
<i>Exceções</i>	<p>EX 1 Falta o preenchimento de, pelos menos, um dos campos obrigatórios: código da máquina, número de série, modelo, estabelecimento, filial, vendedor.</p> <ol style="list-style-type: none"> <li>1. O sistema notifica o utilizador para a obrigatoriedade do preenchimento do campo;</li> <li>2. Seleciona o separador “Dados Principais” e posiciona no campo em falta;</li> <li>3. Retorna, em modo de edição.</li> </ol>

---

<sup>174</sup> Discriminador é a capacidade de um moedeiro distinguir diferentes tipos de moedas (e.g. moedas de cinco e dez cêntimos, no mesmo moedeiro).



Tabela 64 – Caso de uso de Manter Código de Operador

<i>Nome</i>	Manter Código de Operador
<i>ID</i>	UC_WVB_2011
<i>Descrição</i>	Atualizar ou programar o código de operador da máquina
<i>Atores</i>	Utilizador (Administrativo, Administrador) e Sistema Middleware
<i>Pré-condições</i>	O utilizador está autenticado e tem permissões de visualização.
<i>Cenário</i>	A ação de testar a máquina consiste em dois passos: (i) efetuar-se um pedido de consulta dos estados da máquina e (ii) executar-se um pedido de comunicação. Rececionados os estados dos sensores, atualizam-se os dados relativos aos sensores situados no separador “Visão Geral” (c.f. Tabela 60) e procede-se, caso os sensores indiciem que a máquina se encontra preparada para comunicar (i.e., máquina e cabos ligados e com conectividade) com um pedido de comunicação do tipo “Teste” ao sistema Middleware. Posteriormente à receção da comunicação, compara-se o código da máquina, enviado na comunicação, com o da ficha da máquina e procede-se de acordo com o resultado da comparação.
	EX 1 Falha o pedido de comunicação com a máquina. 1. O sistema notifica o utilizador da falha ocorrida; 2. Retorna para a mesma janela.

	<p>EX 2 O código enviado pela máquina é diferente do código da ficha da máquina.</p> <ol style="list-style-type: none"> <li>1. O sistema notifica o utilizador que o código da máquina é diferente do enviado na última comunicação (efetua-se a notificação no momento da ocorrência e sempre que se entre numa ficha de máquina nessas condições);</li> <li>2. Suspende-se a comunicação;</li> <li>3. Retorna para a mesma janela.</li> </ol>
--	---

Tabela 65 – Caso de uso de Programar Mensagem Publicitária

<i>Nome</i>	Programar Mensagem Publicitária
<i>ID</i>	UC_WVB_2012
<i>Descrição</i>	Verificar os estados dos sensores da máquina e confirmar se o módulo se encontra instalado na máquina esperada
<i>Atores</i>	Utilizador (Administrativo, Administrador) e Sistema Middleware
<i>Pré-condições</i>	O utilizador está autenticado e tem permissões de visualização.
<i>Cenário</i>	A ação de testar a máquina consiste em dois passos: (i) efetuar-se um pedido de consulta dos estados da máquina e (ii) executar-se um pedido de comunicação. Rececionados os estados dos sensores, atualizam-se os dados relativos aos sensores situados no separador “Visão Geral” (c.f. Tabela 60) e procede-se, caso os sensores indicem que a máquina se encontra preparada para comunicar (i.e., máquina e cabos ligados e com conectividade), com um pedido de comunicação do tipo “Teste” ao sistema Middleware. Após a receção da comunicação, compara-se o código da máquina enviado na comunicação com o da ficha da máquina e procede-se de acordo com o resultado da comparação.
	<p>EX 1 Falha o pedido de comunicação com a máquina.</p> <ol style="list-style-type: none"> <li>3. O sistema notifica o utilizador da falha ocorrida;</li> <li>4. Retorna para a mesma janela.</li> </ol> <p>EX 2 O código enviado pela máquina é diferente do código da ficha da máquina.</p>

	<p>4. O sistema notifica o utilizador que o código da máquina é diferente do enviado na última comunicação (efetua-se a notificação no momento da ocorrência e sempre que se entre numa ficha de máquina nessas condições);</p> <p>5. Suspende-se a comunicação;</p> <p>6. Retorna para a mesma janela.</p>
--	---

Tabela 66 – Caso de uso de Consultar Visitas Máquinas

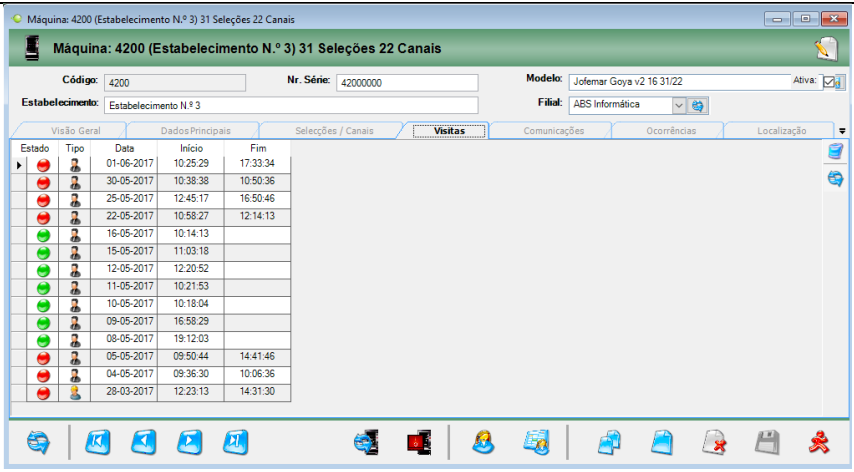
<i>Nome</i>	<i>Consultar Visitas Máquinas</i>
<i>ID</i>	UC_WVB_2020
<i>Descrição</i>	Visualizar visitas às máquinas
<i>Atores</i>	Utilizador
<i>Pré-condições</i>	O utilizador está autenticado e tem permissões de edição (criar ou alterar).
<i>Cenário</i>	<p>No separador de “Visitas” da janela “Máquina” (c.f. Figura 121 – Janela Máquina – Separador Visitas), consulta-se uma listagem de visitas com o ênfase nos dados: (i) estado da visita (agendada, em curso, fechada), (ii) tipo (comercial ou técnica), (iii) data, (iv) hora de início e (v) hora de fim.</p> <p>Inicialmente, visualizam-se todas as visitas agendadas e em curso. Caso se pretenda, é possível visualizar todas as visitas, ou atualizar a lista atual.</p>
<i>Requisitos</i>	De modo a garantir disponibilidade da aplicação carrega-se, em segundo plano, a lista de visitas.
<i>Janela</i>	 <p>The screenshot shows a software window titled 'Máquina: 4200 (Estabelecimento N.º 3) 31 Seleções 22 Canais'. It features a search bar with 'Código: 4200' and 'Nr. Série: 42000000'. Below the search bar are tabs for 'Visão Geral', 'Dados Principais', 'Seleções / Canais', and 'Visitas' (which is selected). The 'Visitas' tab displays a table with columns: Estado, Tipo, Data, Início, and Fim. The table contains 15 rows of visit data, with some rows having red status icons and others having green status icons. The Windows taskbar is visible at the bottom of the screenshot.</p>
	<p>Figura 121 – Janela Máquina – Separador Visitas</p>

Tabela 67 – Caso de uso de Consultar Visita Máquina

<i>Nome</i>	<i>Consultar Visita Máquina</i>
<i>ID</i>	UC_WVB_2021
<i>Descrição</i>	Visualizar dados de uma visita à máquina
<i>Atores</i>	Utilizador
<i>Pré-condições</i>	O utilizador está autenticado e tem permissões de visualização
<i>Cenário</i>	Na janela “Visita” (c.f. Figura 122), visualizam-se os seguintes dados da visita à máquina: (i) data da visita, (ii) hora efetiva de início/fim e respetiva duração, (iii) hora prevista de início/fim e respetiva duração (concede-se uma opção para alternar entre a visão efetiva/prevista), (iv) vendedor associado, (v) data/hora do último reinício de visita, caso ocorra, (vi) número do saco onde se guardam as moedas e notas recolhidas da máquina, (vii) indicador de visita realizada fora de rota, (viii) observações auxiliares, antecessoras da visita, (ix) relatório concretizado pelo vendedor da visita, (x) lista de alarmes ocorridos desde o término da visita anterior, até ao fecho da visita em análise (descrição e data/hora do alarmes), (xi) lista de anomalias ocorridas, desde o fecho da visita anterior, até à concretização da atual (descrição, data/hora e indicador de correção ou não da anomalia), (xii) listagem de moedeiros ao iniciar/terminar-se a visita (número de moedeiro, quantidade, valor e quantidade <sup>175</sup> colocadas/retiradas) e (xiii) consulta de todas as comunicações realizadas entre o fim da visita anterior, até à conclusão da atual ().
<i>Requisitos</i> <i>Desempenho</i>	De modo a garantir disponibilidade da aplicação carregam-se, em segundo plano, as listas: alarmes, anomalias, moedeiros e comunicações.

---

<sup>175</sup> A gestão de trocos da máquina implica que, no ato da visita, se realizam operações de colocar ou retirar moedas, de modo a regularizar o nível de moedas por moedeiro.

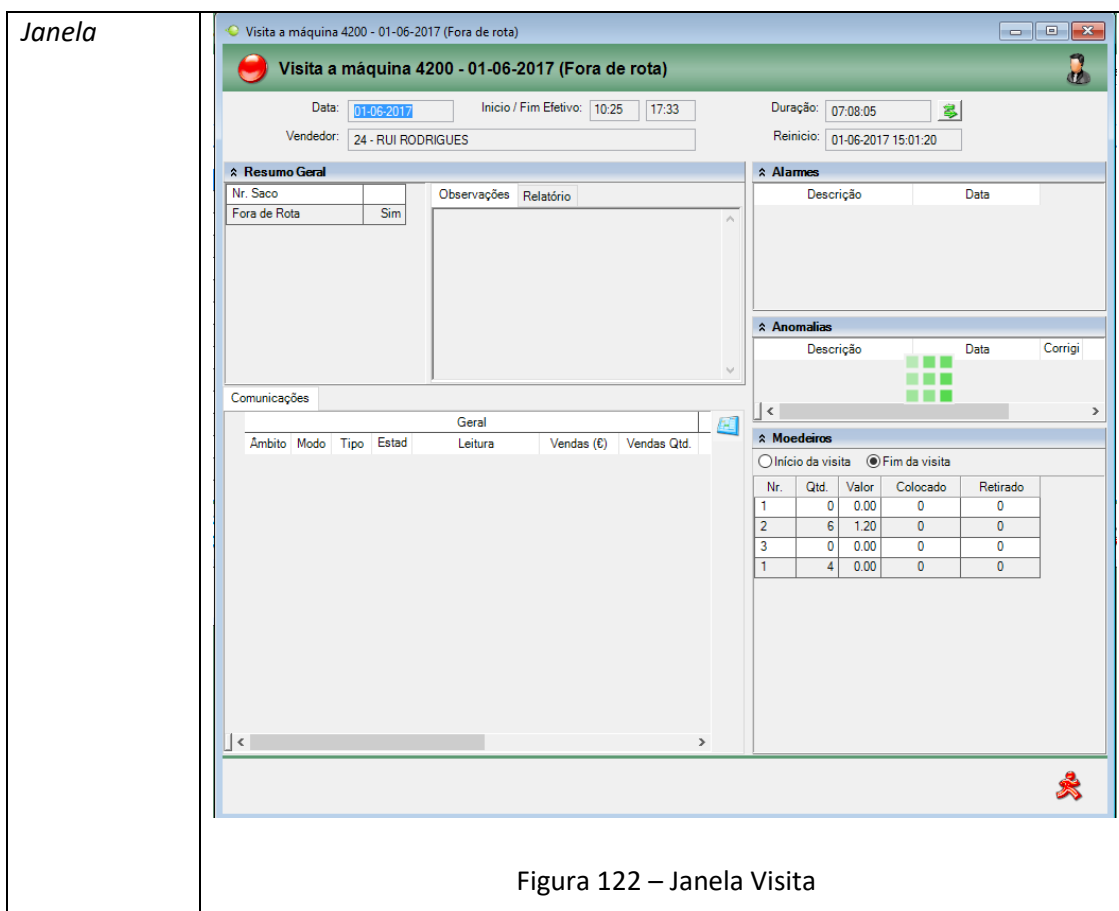


Tabela 68 – Caso de uso de Manter Consulta Canal

<i>Nome</i>	Manter Consulta Canal
<i>ID</i>	UC_WVB_2031
<i>Descrição</i>	Visualizar seleções & canais sob o ponto de vista de tabela
<i>Atores</i>	Utilizador
<i>Pré-condições</i>	O utilizador está autenticado e tem permissões de visualização.
<i>Cenário</i>	No separador “Seleções / Canais” da janela “Máquina” (c.f. Figura 123), visualizam-se, em formato de tabela, os dados da última configuração da máquina, no que diz respeito às seleções e canais. Na lista de seleções, apresentam-se: o número da seleção, o código, a descrição, o preço e o código identificador do produto na máquina. Na listagem de canais, consulta-se: o número, o código e a descrição do produto e os dados do <i>stock</i> por canal (e.g. ótimo, máximo e mínimo).

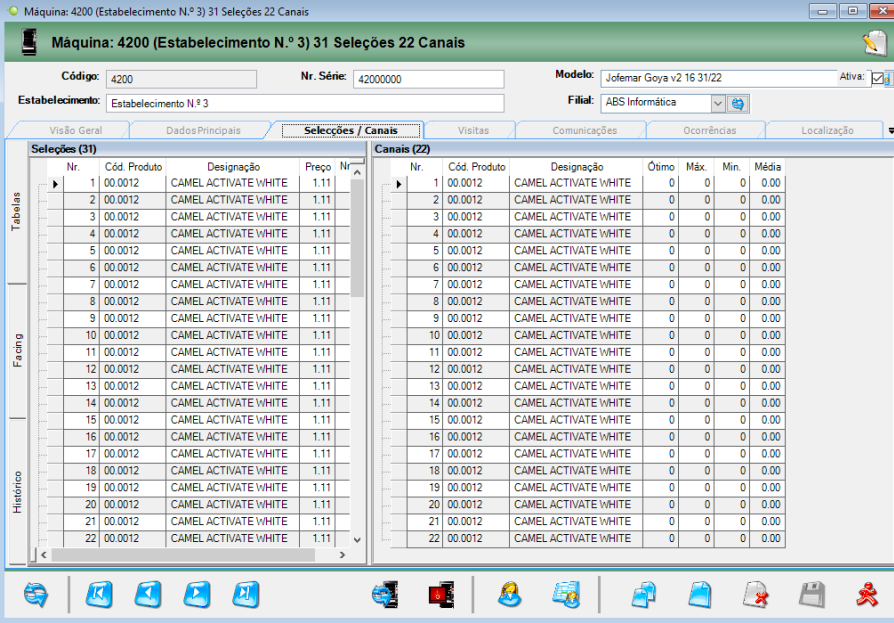
<b>Requisitos</b>  <b>Desempenho</b>	De modo a garantir disponibilidade da aplicação, carregam-se, em segundo plano, o <i>facing</i> da máquina e as listas de: (i) seleções, (ii) canais e (iii) histórico de programação.
<b>Janela</b>	 <p style="text-align: center;">Figura 123 – Janela Máquina – Separador Seleções / Canais</p>

Tabela 69 – Caso de uso de Consultar Seleções & Canais

<b>Nome</b>	Consultar Seleções & Canais
<b>ID</b>	UC_WVB_2032
<b>Descrição</b>	Visualizar o <i>facing</i> da máquina e a associação com os respetivos canais
<b>Atores</b>	Utilizador
<b>Pré-condições</b>	O utilizador está autenticado e tem permissões de visualização.
<b>Cenário</b>	Na visão de “Facing” do separador “Seleções & Canais” da janela “Máquina” (c.f. Figura 124), visualizam-se as seleções e canais da máquina, de acordo com o desenho efetuado no modelo (c.f. Tabela 50). O desenho de uma seleção tem os próximos componentes: número da seleção, imagem do produto, ou, na sua ausência, o seu nome, o preço do produto e os canais associados, no formato CSV. O desenho do canal constitui-se por: número do canal, imagem do produto, ou, na sua ausência, pelo seu nome e seleções associadas, no formato

CSV. De modo a adaptar-se a diferentes desenhos da máquina, permite-se alternar, a visão das seleções e canais, de horizontal para vertical e vice-versa.

Janela

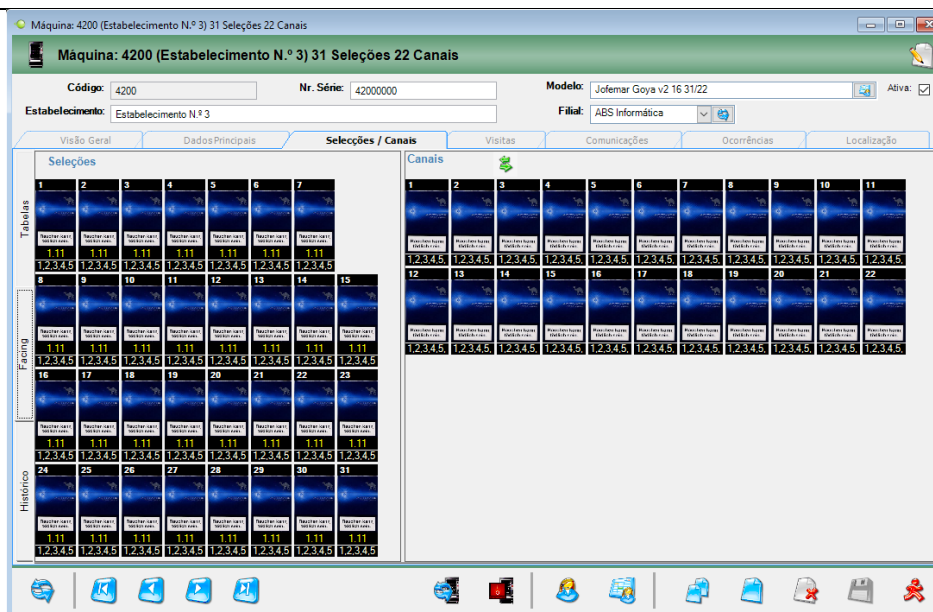


Figura 124 – Janela Máquina – Separador Facing

Tabela 70 – Caso de uso de Consultar Histórico Programação

Nome	Consultar Histórico Programação
ID	UC_WVB_2033
Descrição	Visualizar histórico de programação
Atores	Utilizador
Pré-condições	O utilizador está autenticado e tem permissões de visualização.
Cenário	No separador “Histórico” contido no separador “Seleções / Canais” da janela “Máquina” (c.f. Tabela 90), listam-se as programações efetuadas na máquina, com os dados: âmbito (visita ou assistente), modo (Bluetooth ou telemetria), ação (programação ou configuração), estado (pendente, atual, histórico, anulada), data efetiva de programação, data de alteração de estado, tipo de agendamento (última visita, automaticamente), data de entrada em vigor e origem (vazio ou com base noutra máquina). Ao expandir a programação, visualizam-se as seleções e canais definidos com os dados: número de

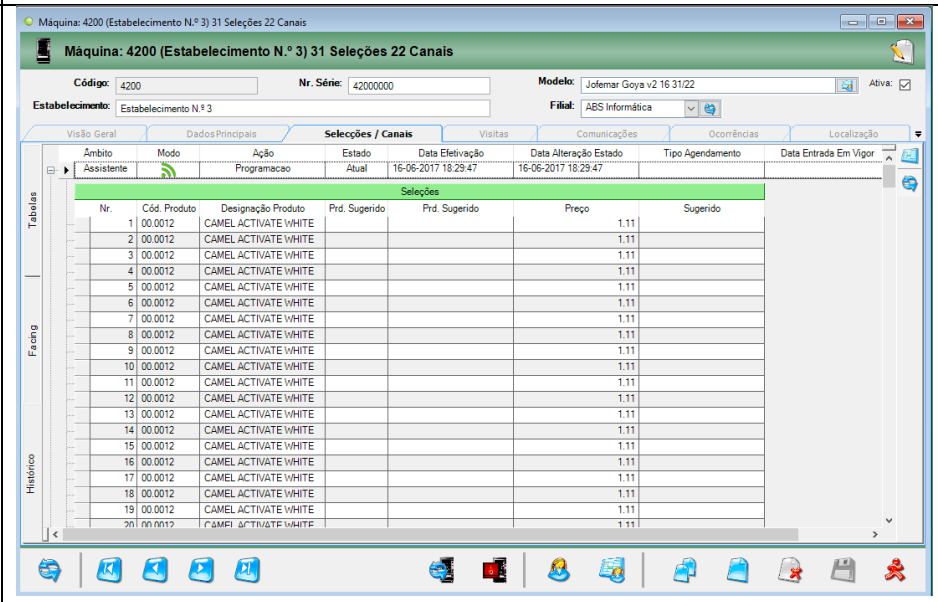
	<p>seleção/canal, código, descrição e preço do produto, código, descrição e preço do produto sugerido (utilizado em situações de agendamento).</p> <p>Concede-se uma visão alargada para consulta de dados mais técnicos como: código identificador de comunicação com o Middleware, código da máquina origem (utilizado em caso de cópia de seleções/canais da máquina), código da comunicação origem (utilizado em caso de cópia com base numa comunicação) e modelo da máquina. A nível do detalhe das seleções, observa-se: o código de produto máquina e o identificador do produto na máquina.</p>
<p><b>Janela</b></p>	 <p>Figura 125 – Janela Máquina – Separador Histórico Programação</p>

Tabela 71 – Caso de uso de Consultar Comunicações Máquina

<i>Nome</i>	Consultar Comunicações Máquina
<i>ID</i>	UC_WVB_2050
<i>Descrição</i>	Visualizar dados da máquina
<i>Atores</i>	Utilizador
<i>Pré-condições</i>	O utilizador está autenticado e tem permissões de visualização.
<i>Cenário</i>	Ao abrir-se o separador “Comunicações” da janela “Máquina” (c.f. Tabela 91), visualiza-se uma lista com todas as comunicações relevantes (Pendentes/Completas/Suspensas/Parciais) e cuja data seja superior à

	<p>última visita à máquina. Essa lista constitui-se por cabeçalho, linhas de seleção/canal (consoante o modelo de máquina), linhas de moedeiros (caso se aplique) e linhas de incidências (caso se aplique).</p> <p>O cabeçalho da comunicação contém campos agrupados do seguinte modo: geral, valores reais, valores acumulados e dados técnicos. O grupo geral é constituído pelas colunas: âmbito (visita, assistente, agendamento), modo (Bluetooth ou telemetria), tipo (inicialização, configuração, estatística, contabilística, teste), estado (pendente, completo, parcial, falha, anulada), data/hora de pedido, data/hora de leitura, valor e quantidade das vendas. Os valores reais representam-se pelos valores: colocado, retirado, cofre, notas, <sup>176</sup>crédito e saldo. Os valores acumulados são: colocado, retirado, cofre, notas, valores para devolvedores, variação de devolvedores e troco. Visualizam-se os seguintes valores técnicos: código do cliente e estabelecimento, código do pedido ao Middleware, código da máquina, indicador de comunicação técnica, indicador de comunicação irrelevante, âmbito da tentativa de comunicação (data de agendamento e pronta a comunicar) e número de tentativas. As linhas de comunicação constituem-se por quatro grupos: geral, valores reais, valores acumulados atuais e valores acumulados anteriores. O grupo geral contém as colunas: número de seleção/canal, código e descrição do produto. O grupo valores reais apresenta os valores das vendas: quantidade, valor, valor perdido e preço. O grupo valores acumulados atuais representa os resultados acumulados enviados pela máquina, no momento da comunicação. Já os acumulados anteriores representam uma cópia dos valores acumulados da comunicação anterior (valor, quantidade e valor perdido). A lista de incidência constitui-se por: um código, uma descrição e um texto com dados adicionais. A lista de moedeiros compõe-se por: número, quantidade, valor e unidade da moeda.</p>
--	--

---

<sup>176</sup> O conceito de crédito utiliza-se em determinados modelos de máquina, normalmente utilizado em estações de serviço, onde não se efetua a gestão do pagamento; em vez disso, a máquina fica com um valor denominado de crédito, que é deduzido em cada compra, enquanto o seu valor for superior a zero.

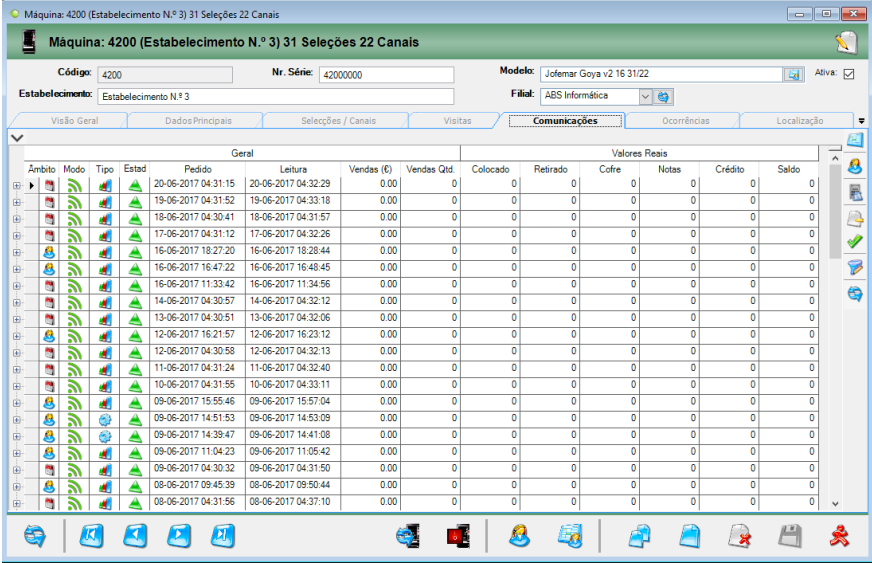
	<p>Adicionalmente, disponibilizam-se as funcionalidades: (i) alternar entre visão alargada e reduzida (constituída apenas pelos grupos “geral” e “valores reais”), (ii) invocação do assistente de comunicação (Tabela 38 – Caso de uso de Comunicar com Máquinas), (iii) visualizar o ficheiro de suporte da comunicação (aplica-se nas comunicações efetuadas por Bluetooth e visa facilitar o rastreamento das comunicações), (iv) anular / aceitar uma comunicação suspensa (aplica-se apenas em comunicações suspensas e não são revertíveis), (v) filtrar as comunicações por um conjunto de critérios definidos e (vi) atualizar a lista de comunicações consoante o último filtro selecionado.</p>
<p><b>Requisitos Desempenho</b></p>	<p>De modo a garantir disponibilidade da aplicação, carrega-se, em segundo plano, a lista de comunicações.</p>
<p><b>Janela</b></p>	 <p>Figura 126 – Janela Máquina – Separador Comunicações</p>

Tabela 72 – Caso de uso de Consultar Ocorrências Máquina

<p><i>Nome</i></p>	<p>Consultar Ocorrências Máquina</p>
<p><i>ID</i></p>	<p>UC_WVB_2060</p>
<p><i>Descrição</i></p>	<p>Visualizar ocorrências da máquina</p>
<p><i>Atores</i></p>	<p>Utilizador (Administrativo, Administrador)</p>
<p><i>Pré-condições</i></p>	<p>O utilizador está autenticado e tem permissões de visualização.</p>

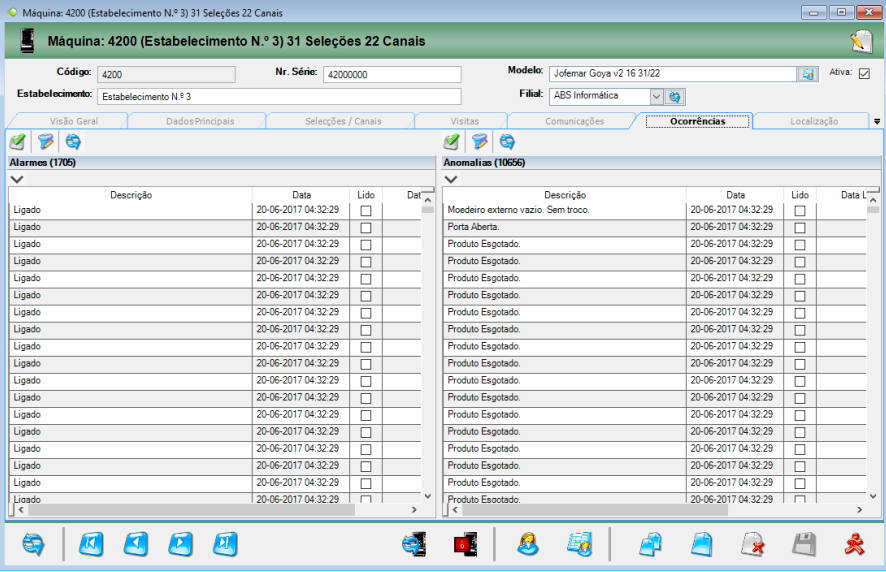
<p><i>Cenário</i></p>	<p>Na entrada do separador “Ocorrências” na janela “Máquina” (c.f. Figura 127), visualizam-se os alarmes e anomalias ocorridas, desde a última visita à máquina. Na listagem de alarmes observa-se: a descrição, a data/hora, um campo indicador de leitura do alarme e a data de leitura. A lista de anomalias contém a mesma estrutura da lista de alarmes, a que acrescem as colunas: identificação do estado corrigido e a respetiva data de correção. Possibilita-se a atualização de ambas as listas, através do menu superior, ou do menu de contexto referente a cada uma delas.</p>
<p><i>Requisitos</i> <i>Desempenho</i></p>	<p>De modo a garantir disponibilidade da aplicação carregam-se, em segundo plano, as listas de alarmes e anomalias.</p>
<p><i>Janela</i></p>	 <p>Figura 127 – Janela Máquina – Separador Ocorrências</p>

Tabela 73 – Caso de uso de Consultar *Localização da Máquina*

<p><i>Nome</i></p>	<p>Consultar Localização da Máquina</p>
<p><i>ID</i></p>	<p>UC_WVB_2070</p>
<p><i>Descrição</i></p>	<p>Visualizar a localização de instalação e atual da máquina no mapa</p>
<p><i>Atores</i></p>	<p>Utilizador</p>
<p><i>Pré-condições</i></p>	<p>O utilizador está autenticado e tem permissões de visualização.</p>

<p><i>Cenário</i></p>	<p>Ao selecionar-se o separador “Localização na janela da “Máquina” (c.f. Figura 128), ocorre uma das seguintes opções: (i) apresenta-se a coordenada atual e de instalação da máquina (se definidas), ou (ii) apresenta-se a coordenada de instalação da máquina, através dos dados da morada do estabelecimento, ou (iii) indica-se o desconhecimento das coordenadas da máquina.</p> <p>De um modo geral, visualizam-se os seguintes dados acerca da localização da máquina: (i) o desenho, no mapa, da coordenada de instalação e atual, (ii) os dados da coordenada atual (latitude, longitude e distância em relação à coordenada de instalação), (iii) os dados da coordenada de instalação (latitude e longitude), (iv) as definições do mapa (e.g. raio de segurança em metros) e a (v) listagem de histórico de localizações (data/hora, distância, velocidade, latitude, longitude e direção). Possibilitam-se as seguintes operações sobre a coordenada atual: (i) limpar e (ii) limpar a coordenada atual. Capacita-se o utilizador de atualizar a coordenada de instalação da máquina, a partir dos dados da ficha do cliente. A partir do histórico de localizações, possibilita-se a visualização do rasto da máquina e a definição de uma das coordenadas do histórico, como sendo a coordenada de instalação. O rasto da máquina caracteriza-se pelo desenho do conjunto de coordenadas presentes no histórico de localizações. Cada coordenada do histórico é representada por: um número (ordem crescente da data de captura da coordenada), uma localização e uma direção.</p> <p>Disponibiliza-se, ainda, um modo de perseguição, acionado automaticamente, caso a máquina seja deslocada para fora do seu raio de segurança (baseada na coordenada de instalação). Este modo caracteriza-se pela atualização do desenho do histórico de localizações, de X em X segundos, de forma a acompanhar, em tempo real, a localização e direção da máquina.</p> <p>Se indisponível, qualquer coordenada da máquina apresenta uma mensagem elucidativa desse facto e esconde-se o mapa. A mensagem deve aparecer, no local do mapa, em formato de hiperligação e, caso pressionada, apresenta-se o mapa com o centro na coordenada da filial da máquina e em modo de atribuição de coordenada de instalação da máquina. Nesse modo, o cursor do rato fica com a imagem do pin de instalação da máquina e, logo que</p>
-----------------------	--

pressionado qualquer ponto no mapa é, imediatamente, atribuído esse ponto como local de instalação da máquina.

Janela

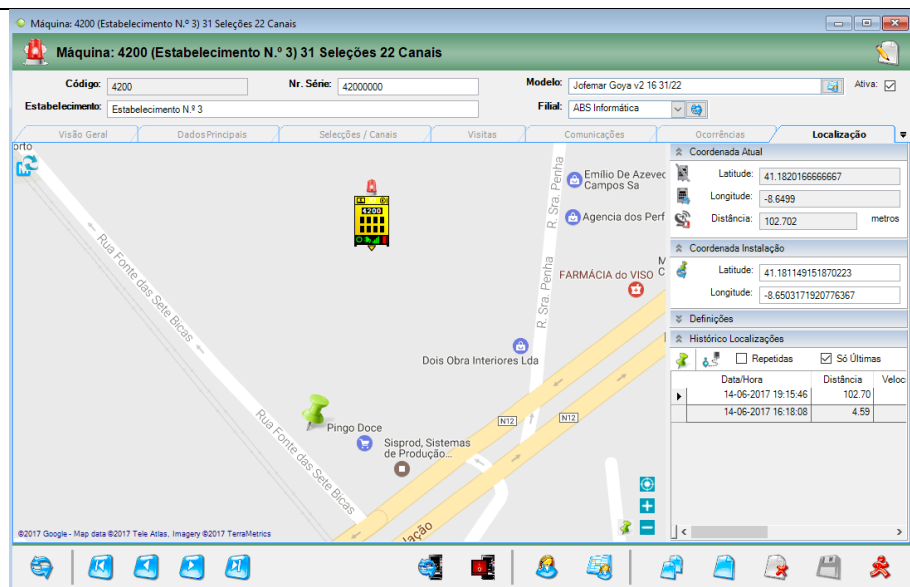


Figura 128 – Janela Máquina – Separador Localização

Tabela 74 – Caso de uso de Programar Máquina Por Assistente

<i>Nome</i>	Caso de uso de Programar Máquina Por Assistente
<i>ID</i>	UC_WVB_2040
<i>Descrição</i>	Selecionar o método de manutenção das seleções & canais da máquina
<i>Atores</i>	Utilizador (Administrativo, Administrador)
<i>Pré-condições</i>	O utilizador está autenticado e tem permissões de edição.
<i>Cenário</i>	<p>Ao entrar-se no assistente de programação da máquina (c.f. Figura 129), concede-se ao utilizador três métodos de atribuição das seleções &amp; canais da máquinas: modificar a atual configuração da máquina, copiar as seleções / canais de outra máquina, ou copiar as seleções / canais a partir da última comunicação com a máquina.</p> <p>Para prosseguir à etapa seguinte, seleciona-se uma das opções e avança-se no assistente. Se se seleciona a opção “Modificar Máquina” ou “Última Comunicação”, encaminha-se o utilizador para a etapa de atribuição de seleções &amp; canais (c.f. Tabela 76), caso contrário, para a opção “Cópia de outra máquina”, o utilizador é direcionado para a etapa que permite essa</p>


	definição (c.f. Tabela 75 – Caso de uso de Copiar Seleções & Canais de Outra Máquina).
<i>Exceções</i>	<p>EX 1 Falta a seleção de uma opção.</p> <ol style="list-style-type: none"> <li>1. O sistema notifica o utilizador para a obrigatoriedade da seleção de uma das opções disponíveis;</li> <li>2. Retorna para a mesma janela.</li> </ol> <p>EX 2 Seleciona-se a opção última comunicação e não existe qualquer comunicação válida.</p> <ol style="list-style-type: none"> <li>1. O sistema notifica o utilizador para a inexistência de qualquer comunicação válida;</li> <li>2. Retorna para a mesma janela.</li> </ol>
<i>Janela</i>	 <p>Figura 129 – Janela de Assistente de Programação Máquina – Selecionar Método de Atribuição de Seleção &amp; Canais</p>

Tabela 75 – Caso de uso de Copiar Seleções & Canais de Outra Máquina

<i>Nome</i>	Copiar Seleções & Canais de Outra Máquina
<i>ID</i>	UC_WVB_2041
<i>Descrição</i>	Selecionar uma máquina da qual se pretende copiar as seleções e/ou canais
<i>Atores</i>	Utilizador (Administrativo, Administrador)
<i>Pré-condições</i>	O utilizador está autenticado e tem permissões de edição.

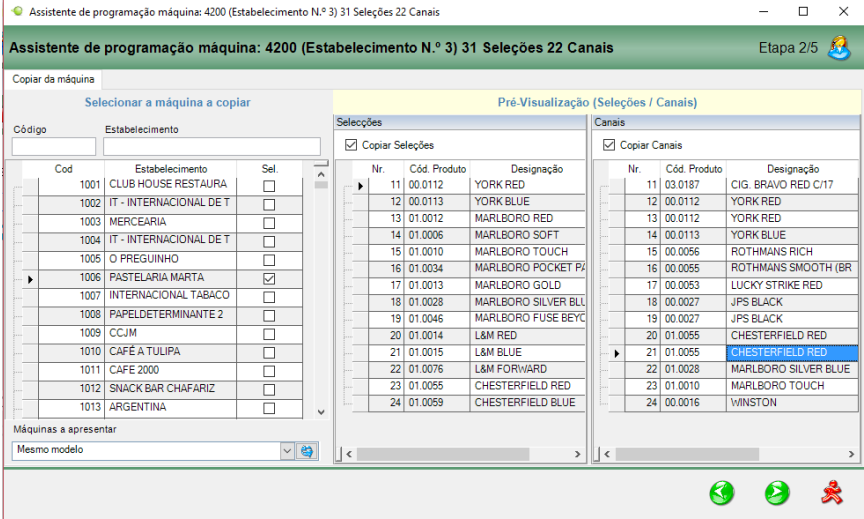
<b>Cenário</b>	Na etapa de cópia de seleções & canais de outra máquina (c.f. Figura 130), seleciona-se, em primeiro, o tipo de máquinas a visualizar (todas, mesmo modelo ou mesmo número de seleções & canais), em segundo e a partir das máquinas apresentadas, seleciona-se uma das máquinas (caso outra se encontre selecionada retira-se, automaticamente, essa seleção) e, em último, seleciona-se as secções (seleções, canais) que se pretendem copiar (predefinido, não se encontra selecionada qualquer secção).
<b>Exceções</b>	<p>EX 1 Falta a seleção de uma máquina ou da secção (seleção e/ou canal) da máquina a copiar.</p> <ol style="list-style-type: none"> <li>1. O sistema notifica o utilizador para a obrigatoriedade da seleção de uma máquina e de, pelo menos, uma das secções;</li> <li>2. Retorna para a mesma janela.</li> </ol>
<b>Janela</b>	 <p>Figura 130 - Janela de Assistente de Programação Máquina – Copiar Seleções &amp; Canais de Outra Máquina</p>

Tabela 76 – Caso de uso de Manter Seleções & Canais

<b>Nome</b>	Manter Seleções & Canais
<b>ID</b>	UC_WVB_2042
<b>Descrição</b>	Atribuir os produtos às seleções e canais
<b>Atores</b>	Utilizador (Administrativo, Administrador)
<b>Pré-condições</b>	O utilizador está autenticado e tem permissões de visualização.

<i>Cenário</i>	<p>Ao entrar-se na etapa de “Seleções/Canais”, disponibiliza-se ao utilizador três modos de atribuição de produtos às respetivas seleções e canais: (i) diretamente nas respetivas tabelas (seleções, canais) (c.f. Tabela 77), (ii) por via de uma visão de <i>facing</i> da máquina (c.f. Tabela 78), ou através de um assistente de configuração de produto (c.f. Tabela 79). A apresentação do conteúdo das seleções &amp; canais varia consoante a opção selecionada, na primeira etapa (c.f. Tabela 74): (i) para a opção “Modificar Máquina”, apresentam-se as seleções e canais atualmente configurados na máquina, (ii) para a opção “última comunicação” apresentam-se as seleções ou canais incluídos na última comunicação (não se apresentam, em simultâneo, as seleções e os canais e a escolha de um, ou outro, varia consoante o modelo da máquina) e (iii) “Cópia de outra máquina” obtêm-se as seleções e canais da máquina selecionada (caso a máquina origem tenha mais seleções ou canais, ignoram-se todos os elementos excedentes).</p> <p>Configuram-se as seleções e canais através do método, ou combinação de métodos, preferido(s) do utilizador e avança-se para a etapa seguinte (c.f. Tabela 80).</p>
<i>Exceções</i>	<p>EX 1 Falta a atribuição de um produto numa seleção ou canal.</p> <p>EX 2 Falta a atribuição do preço de um produto.</p> <p>EX 3 Não existe correspondência total entre os produtos das seleções e os produtos dos canais (todos os produtos das seleções têm de estar presentes nos canais e vice-versa).</p> <ol style="list-style-type: none"> <li>1. O sistema notifica o utilizador do motivo da falha;</li> <li>2. Seleciona a vista tabular e a linha onde ocorre a falha na tabela de seleção, ou canal, mediante o caso específico;</li> <li>3. Retorna para a mesma janela.</li> </ol>

Tabela 77 – Caso de uso de Manter Seleções & Canais – Vista Tabular

<i>Nome</i>	Manter Seleções & Canais – Vista Tabular
<i>ID</i>	UC_WVB_2043
<i>Descrição</i>	Atribuir os produtos às seleções e canais, através de uma visão tabular

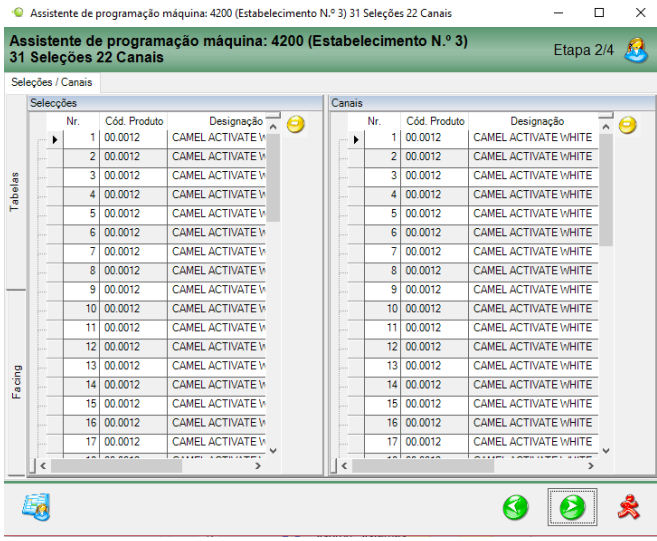
<b>Atores</b>	Utilizador (Administrativo, Administrador)
<b>Pré-condições</b>	O utilizador está autenticado e tem permissões de edição.
<b>Cenário</b>	Ao seleccionar-se o separador “Tabelas” (c.f. Figura 131), observam-se duas tabelas: (i) listagem de seleções e (ii) listagem de canais. Na tabela de seleções, atribui-se, para cada seleção, o produto, o preço e, automaticamente e de modo sequencial, o número de identificação do produto na máquina. Na tabela de canais, atribui-se o código do produto e os dados referentes ao <i>stock</i> (e.g. ótimo, máximo e mínimo). Para cada uma das tabelas concede-se a opção para limpar todos os produtos introduzidos.
<b>Requisitos Desempenho</b>	De modo a garantir disponibilidade da aplicação, carregam-se, seguidamente, as listas de seleções e canais.
<b>Janela</b>	 <p>Figura 131 – Janela de Assistente de Programação Máquina – Manter Seleções &amp; Canais – Visão Tabular</p>

Tabela 78 – Caso de uso de Manter Seleções & Canais – Vista Facing

<b>Nome</b>	Manter Seleções & Canais – Vista Facing
<b>ID</b>	UC_WVB_2044
<b>Descrição</b>	Atribuir os produtos às seleções e canais através de uma visão de <i>facing</i>
<b>Atores</b>	Utilizador (Administrativo, Administrador)
<b>Pré-condições</b>	O utilizador está autenticado e tem permissões de edição.

<p><i>Cenário</i></p>	<p>Ao selecionar-se o separador “Facing” (c.f. Tabela 78), observam-se três áreas distintas: (i) desenho das seleções, conforme definido no modelo da máquina, (ii) desenho sequencial dos canais da máquina e (iii) listagem de produtos disponíveis para inclusão em máquinas.</p> <p>Na secção de “seleções”, atribui-se para edição a primeira seleção sem produto atribuído. Para atribuir um produto é suficiente efetuar-se um duplo <i>click</i> ou pressionar-se a tecla ENTER, no produto pretendido. Caso se deseje, atribui-se, na seleção, o preço do produto e, sob o formato de texto CSV, a lista de canais associados à seleção. Ao concretizar-se a associação entre a seleção e o canal, automaticamente, atribui-se o produto ao desenho do respetivo canal e adiciona-se, no canal, a associação com a seleção em causa. Ao atribuir-se um produto, o sistema seleciona, automaticamente, a seleção seguinte.</p> <p>A secção “canais” tem um funcionamento em tudo, semelhante à secção “seleções” com a diferença de no canal não ser possível atribuir o preço do produto e a lista CSV corresponder às seleções associadas ao canal.</p> <p>Na secção “produto”, consulta-se a lista de produtos disponíveis para se associar às máquinas com base em dois critérios: (i) todos os produtos ou (ii) por promotor. Quando se seleciona o filtro de promotor, surge uma lista de promotores. Ao selecionar-se um promotor, atualiza-se, automaticamente, a lista de produtos com aqueles que pertencem ao promotor selecionado.</p>
<p><i>Exceções</i></p>	<p>EX 1 Falta a atribuição de seleção ou canal.</p> <p>EX 2 Falta a atribuição do preço de um produto.</p> <p>EX 3 Não existe correspondência total entre os produtos das seleções e os produtos dos canais (todos os produtos das seleções têm de estar presentes nos canais e vice-versa).</p> <ol style="list-style-type: none"> <li>1. O sistema notifica o utilizador do motivo da falha;</li> <li>2. Seleciona a vista tabular e a linha onde ocorre a falha na tabela de seleção ou canal, mediante o caso específico;</li> <li>3. Retorna para a mesma janela.</li> </ol>
<p><i>Requisitos Desempenho</i></p>	<p>De modo a garantir disponibilidade da aplicação carrega-se, em segundo plano, o <i>facing</i>.</p>

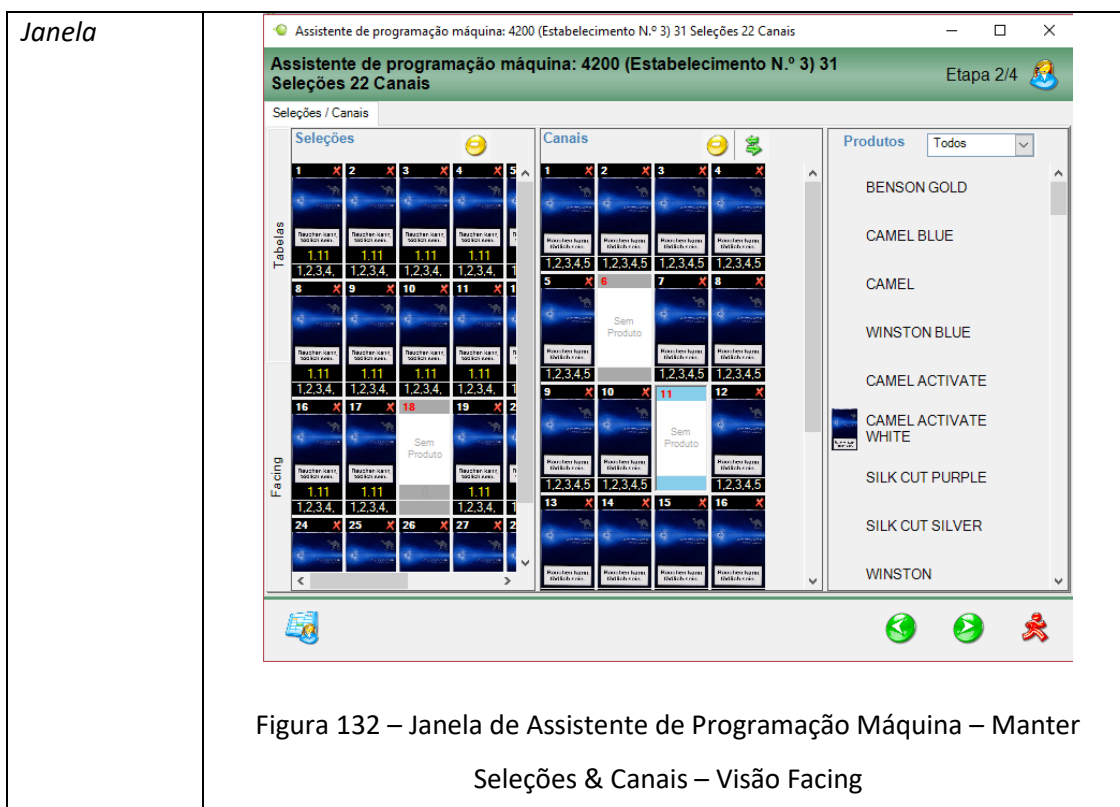


Tabela 79 – Caso de uso de Configurar Produto Por Assistente

Nome	Configurar Produto Por Assistente
ID	UC_WVB_2045
Descrição	Configurar preço, seleções e canais de um determinado produto
Atores	Utilizador (Administrativo, Administrador)
Pré-condições	O utilizador está autenticado e tem permissões de edição.
Cenário	Ao entrar-se na janela “Assistente de Configuração de Produto” (c.f. Figura 133), observam-se os dados do produto (código, descrição, foto e preço) e duas tabelas com os números de seleções e canais associados ao produto. Em cada uma das tabelas permite-se a adição, remoção e alteração da respetiva linha de seleção ou canal. Adicionalmente, disponibiliza-se uma opção para limpar todos as linhas de uma tabela.
Exceções	EX 1 A seleção ou canal já se encontra atribuída a outro produto. 1. O sistema notifica o utilizador que a seleção, ou canal, já se encontra atribuído a outro produto;

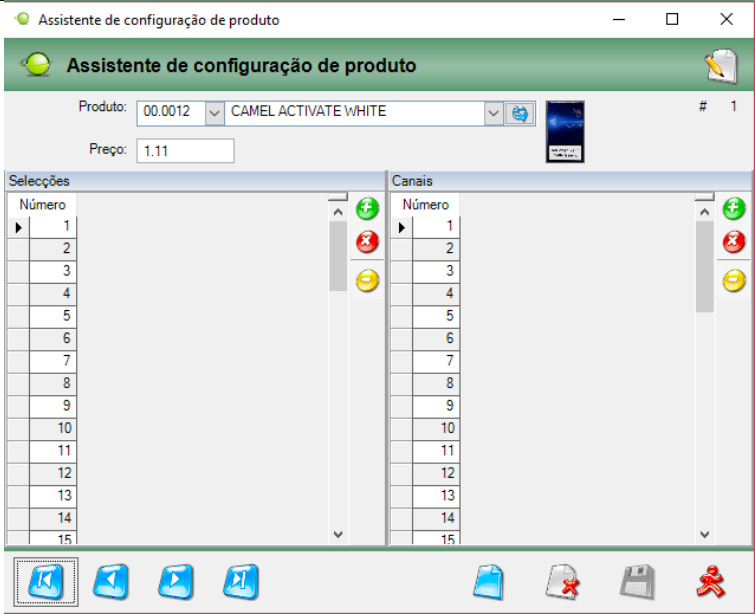
	<p>2. Selecciona a linha onde ocorre a falha na tabela de seleção, ou canal, mediante o caso específico;</p> <p>3. Retorna para a mesma janela.</p>
Janela	 <p>Figura 133 – Janela de Configuração de Produto</p>

Tabela 80 – Caso de uso de Manter Seleções & Canais – Aplicar Configuração

Nome	Manter Seleções & Canais – Aplicar Configuração
ID	UC_WVB_2046
Descrição	Selecionar a ação pretendida entre duas possíveis: (i) configurar ou (ii) programar a máquina
Atores	Utilizador (Administrativo, Administrador)
Pré-condições	O utilizador está autenticado e tem permissões de edição.
Cenário	<p>Ao entrar-se na etapa “Aplicar Configuração” (c.f. Figura 134), concede-se duas opções ao utilizador: (i) configurar ou programar a máquina.</p> <p>Para prosseguir para a etapa seguinte, selecciona-se uma das opções e avança-se no assistente. Na opção “Configurar”, encaminha-se o utilizador para a etapa final de “Configurar” (c.f. Figura 135), para a opção “Programar”, solicita-se ao utilizador a introdução das suas credenciais e,</p>

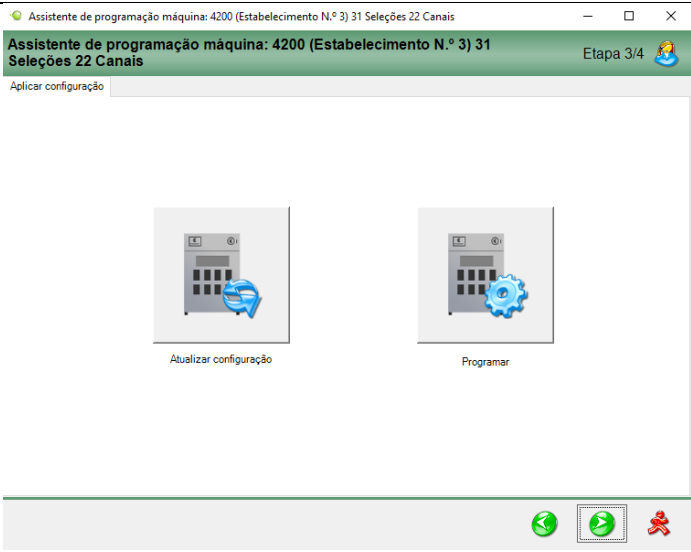
	em caso de sucesso, direciona-se para a última de “Programar” (c.f. Figura 136).
	<p>EX 1 Falta a seleção de uma opção.</p> <ol style="list-style-type: none"> <li>1. O sistema notifica o utilizador para a obrigatoriedade da seleção de uma das opções disponíveis;</li> <li>2. Retorna para a mesma janela.</li> </ol> <p>EX 2 Falha na introdução das credenciais.</p> <ol style="list-style-type: none"> <li>1. O sistema notifica o utilizador para o facto da palavra-passe estar incorreta;</li> <li>2. Retorna para a etapa de “Aplicar Configuração”.</li> </ol>
<i>Janela</i>	 <p>Figura 134 - Janela de Assistente de Programação Máquina – Aplicar Configuração</p>

Tabela 81 – Caso de uso de Atualizar Configuração da Máquina

<i>Nome</i>	Atualizar Configuração da Máquina
<i>ID</i>	UC_WVB_2047
<i>Descrição</i>	Atualizar a configuração da máquina e visualizar o resultado da configuração
<i>Atores</i>	Utilizador (Administrativo, Administrador)
<i>Pré-condições</i>	O utilizador está autenticado e tem permissões de edição.
<i>Cenário</i>	Na etapa de “Progresso de Configuração” (c.f. Figura 135), efetua-se a configuração da máquina e visualiza-se o resultado da configuração.

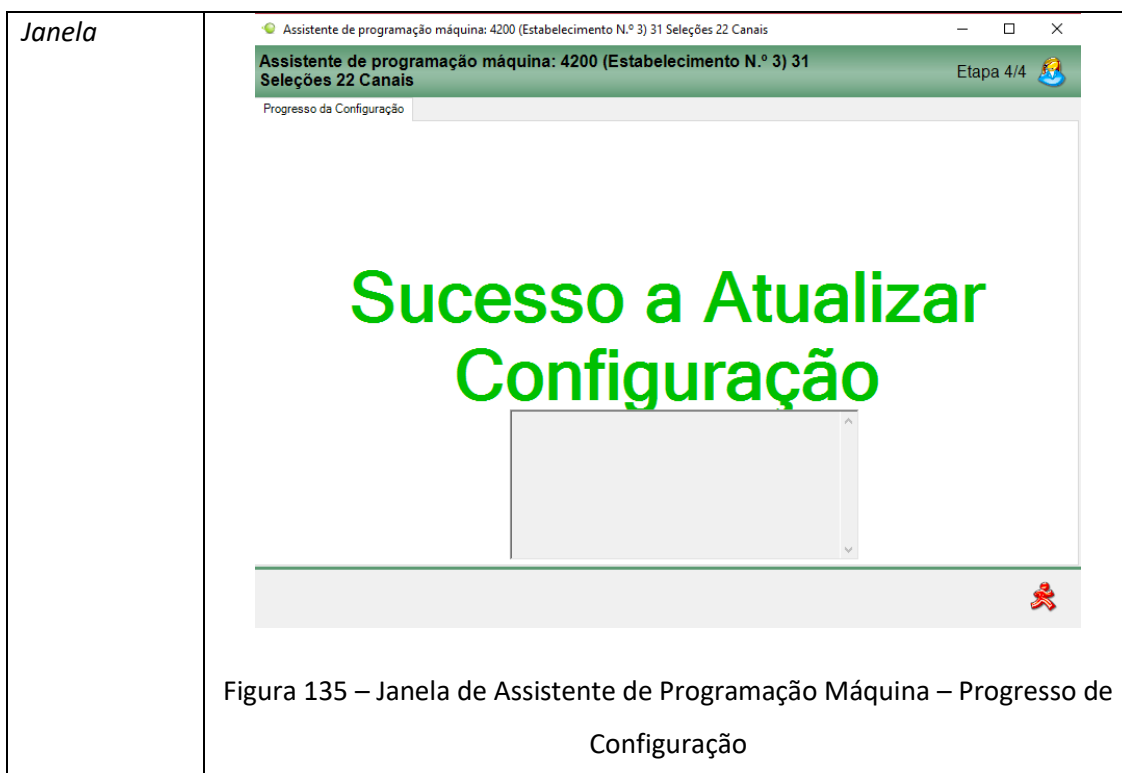
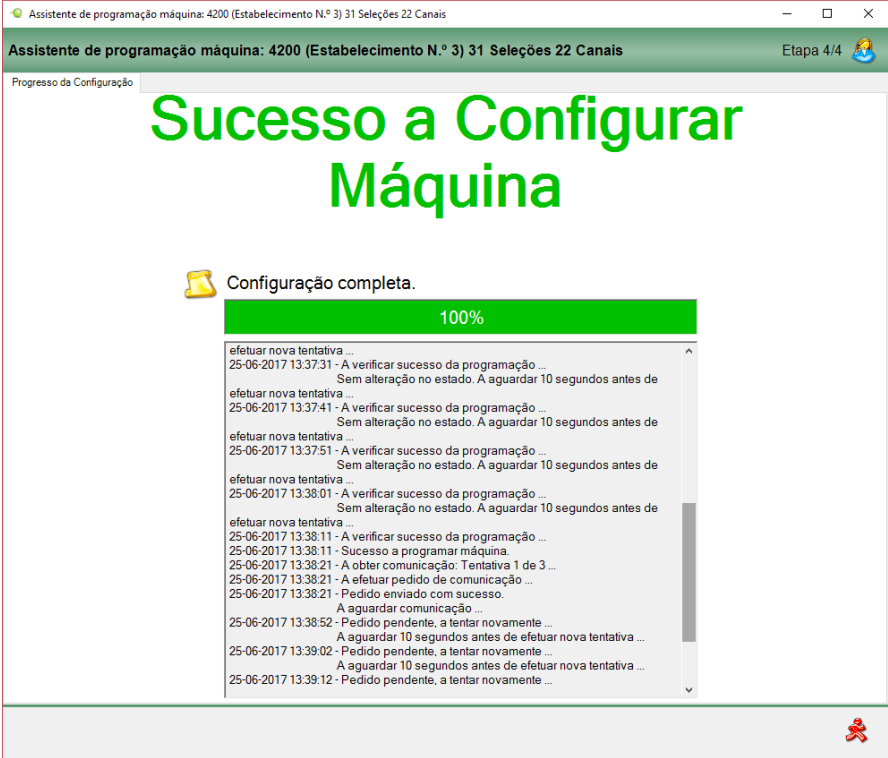


Tabela 82 – Caso de uso de Programar Máquina

<i>Nome</i>	Programar Máquina
<i>ID</i>	UC_WVB_2048
<i>Descrição</i>	Efetuar os seguintes passos principais por ordem: (i) efetuar uma comunicação estatística, (ii) programar a máquina e (iii) efetuar uma comunicação de configuração. Acompanhar o progresso da execução dos passos e visualizar o resultado final.
<i>Atores</i>	Utilizador (Administrativo, Administrador), Sistema Middleware
<i>Pré-condições</i>	O utilizador está autenticado e tem permissões de visualização.
<i>Cenário</i>	Na etapa de “Progresso de Configuração” (c.f. Figura 135), detalham-se as ações executadas: (i) efetuar um pedido de comunicação estatística, (ii) aguardar pela resposta, (iii) efetuar um pedido de programação da máquina, (iv) efetuar um pedido de comunicação de configuração, (v) aguardar pela resposta e (vi) apresentar o resultado final.

	<p>Na programação de máquinas, alteram-se dados sensíveis (e.g. preços, produtos) que afetam a contabilidade da máquina. Evita-se que essas alterações influenciem os dados contabilísticos, através da realização de uma comunicação do tipo “Estatístico” que apura as vendas ocorridas, desde a última comunicação efetuada. Inicia-se, então, o processo pelo pedido, ao Middleware, de uma comunicação estatística e, no caso do pedido ser aceite, aguarda-se pela receção da resposta. O processo de espera de uma comunicação consiste em aguardar-se X segundos (e.g. 30) pela chegada de uma resposta, posteriormente, em intervalos de Y segundos (e.g. dez), confirma-se a resposta da comunicação até Z minutos (e.g. três) de tempo limite, caso se atinja esse tempo, efetuam-se N tentativas (e.g. três) de comunicações. Ao rececionar-se uma resposta válida da comunicação, procede-se à programação da máquina. Começa-se a programação pelo envio de um pedido com as seleções, canais e preços, com a finalidade de os atribuir à máquina; posteriormente, aguarda-se pela resposta do pedido. O processo de espera de uma programação consiste em aguardar-se X segundos (e.g. 30) pela chegada de uma resposta, posteriormente, em intervalos de Y segundos (e.g. dez), confirma-se a resposta da comunicação até Z minutos (e.g. cinco) de tempo limite; caso se atinja esse tempo, efetuam-se N tentativas (e.g. três) de programação. Com a confirmação de uma programação bem-sucedida, torna-se necessário concretizar-se uma comunicação de configuração, de modo a estabelecerem-se os valores acumuladas da máquina, após a programação. Com essa finalidade, efetua-se um pedido de comunicação de configuração e espera-se pela resposta, através da aplicação do mesmo processo atrás mencionado. Por último, apresenta-se o resultado de todo o processo.</p>
	<p>EX 1 Falha nos pedidos de programação, comunicação estatística ou de configuração (e.g. máquina ou cabo desligados).</p> <p>EX 2 Falha a resposta da programação, comunicação estatística ou de configuração (e.g. dados corrompidos).</p> <p>1. O sistema notifica o utilizador para o motivo da falha, na caixa de texto que acompanha a barra da progressão, coloca um texto indicativo de falha e altera a cor da barra de progressão para vermelho;</p>

	2. Retorna para a mesma janela.
<p><i>Janela</i></p>	 <p>Figura 136 – Janela de Assistente de Programação Máquina – Programar Máquina</p>
<p><i>Diagrama conceptual de classes</i></p>	<p>Observa-se, na Figura 137, a entidade histórico de programação da máquina que regista os seguintes dados: (i) parametrizações definidas, (ii) momento de execução e (iii) estado mais recente. Compõe-se esta entidade por: (i) canais e (ii) seleções. Na atualização / programação imediata, associa-se um produto efetivo aos canais e seleções. Caso se defina uma programação por agendamento, inicialmente, o produto associado é apenas uma sugestão; mais tarde, aquando da atualização / programação, atribui-se o produto efetivo. No caso de sucesso na atualização / programação, afetam-se as entidades seleção e canal, em conformidade com o programado.</p> <p>Se o procedimento de programação se baseia numa entidade (e.g. máquina ou comunicação), regista-se a respetiva associação de origem.</p>

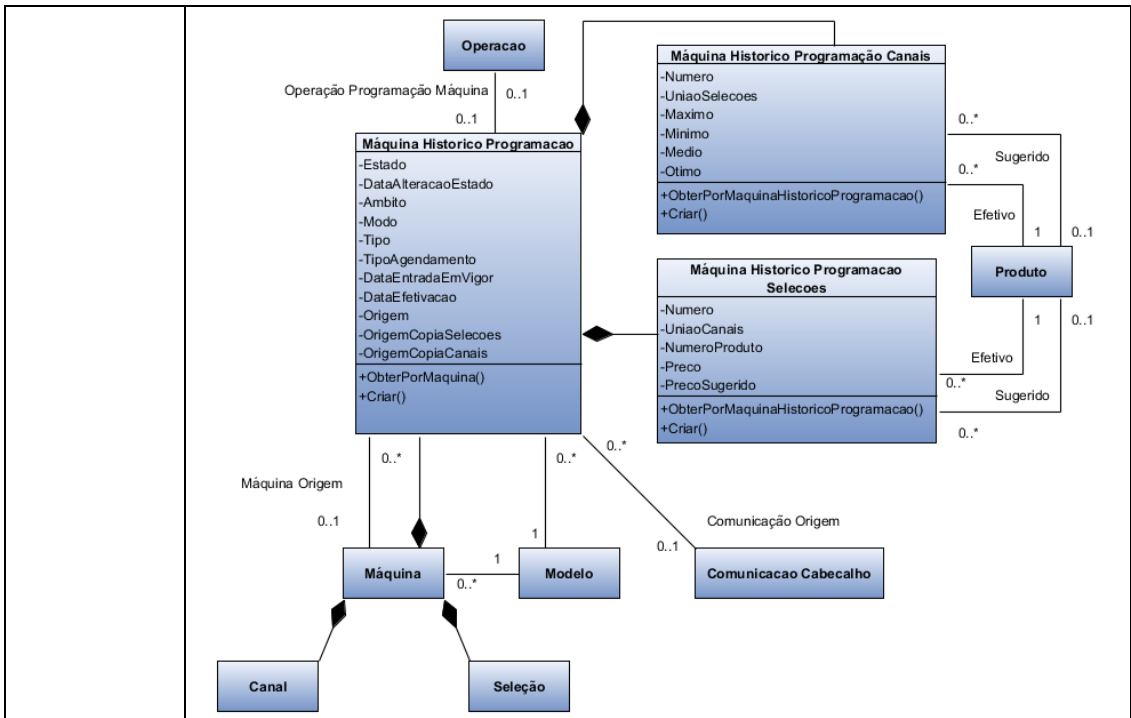


Figura 137 – Diagrama conceptual de classes da Visão de Programação Máquina

Modelo de dados

Destacam-se, na Figura 138, as tabelas que permitem o rastreamento das programações das máquinas: (i) “MaqPrg”, (ii) “MaqPrgSel” e (iii) “MaqPrgCan”. A primeira tabela guarda os dados contextualizadores da programação, a segunda e terceira guardam os produtos programados e sugeridos, nas seleções e canais, respetivamente.

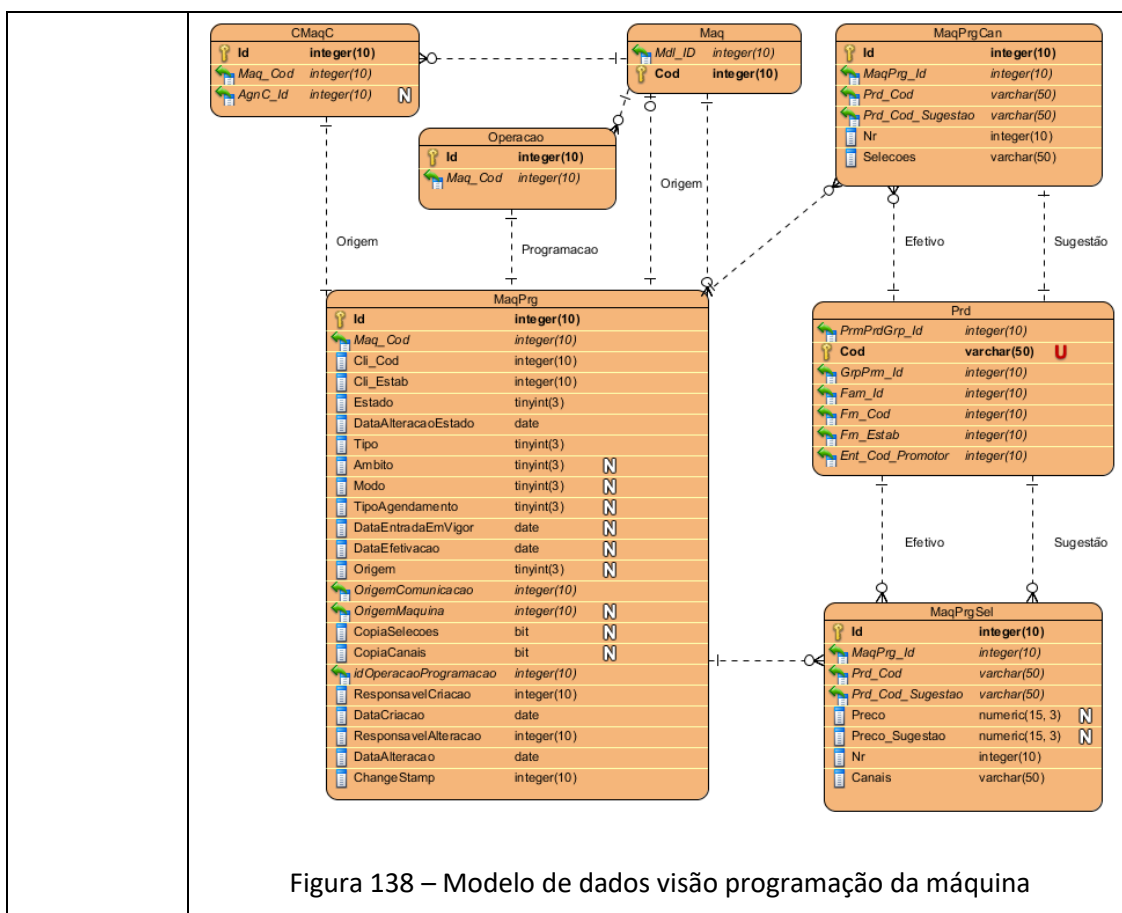
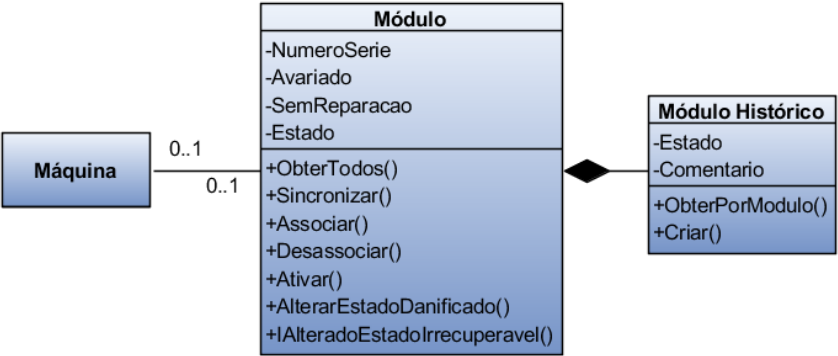


Figura 138 – Modelo de dados visão programação da máquina

Tabela 83 – Caso de uso de Manter Módulo

<b>Nome</b>	Manter Módulo
<b>ID</b>	UC_WVB_1070
<b>Descrição</b>	Manter a associação entre o módulo e a máquina. Gerir os diferentes estados do módulo.
<b>Atores</b>	Utilizador (Administrativo, Administrador) e Sistema Middleware
<b>Pré-condições</b>	O utilizador está autenticado e tem permissões de edição.
<b>Cenário</b>	Ao abrir-se a janela “Módulo” (c.f. Figura 141), visualizam-se três áreas distintas: (i) dados do módulo, (ii) dados da máquina e (iii) histórico de alterações do estado do módulo.  A janela “Módulo” assume características diferentes, mediante a janela que a invoca. Assim, ao abrir-se a atual janela a partir das tabelas administrativas, esta comporta-se como descrito no caso de uso UC_WVB_1071 (c.f. Figura 142 – Janela de Módulo – Atribuir Máquina), caso

	<p>contrário, ao entrar-se através da janela “Máquina”, assume o comportamento detalhado no caso de uso UC_WVB_1072 (c.f. Figura 143). Independentemente do modo de invocação, observa-se sempre a presença do separador “Histórico”, pormenorizado no caso de uso UC_WVB_1073 (c.f. Figura 144).</p>
<p><i>Diagrama conceptual de classes</i></p>	<p>Visualiza-se, na Figura 139, que uma máquina pode, ou não, ter um módulo (obrigatório para máquinas com telemetria) e que o módulo é composto por um conjunto de histórico de alterações de estado.</p>  <pre> classDiagram     class Máquina     class Módulo {         -NumeroSerie         -Avariado         -SemReparacao         -Estado         +ObterTodos()         +Sincronizar()         +Associar()         +Desassociar()         +Ativar()         +AlterarEstadoDanificado()         +!AlteradoEstadoIrrecuperavel()     }     class MóduloHistórico {         -Estado         -Comentario         +ObterPorModulo()         +Criar()     }     Máquina "0..1" -- "0..1" Módulo     Módulo "1" *-- "1" MóduloHistórico   </pre> <p>Figura 139 – Diagrama conceptual de classes da visão de módulo</p>
<p><i>Modelo dados</i></p>	<p>Referem-se, na Figura 140, as tabelas que possibilitam a gestão dos módulos: “Mdl” e “Maq_Mdl_Hst”. A primeira tabela contém os dados identificadores do módulo e a segunda armazena os dados de rastreamento dos módulos pelas várias máquinas, bem como, os estados dos mesmos.</p>

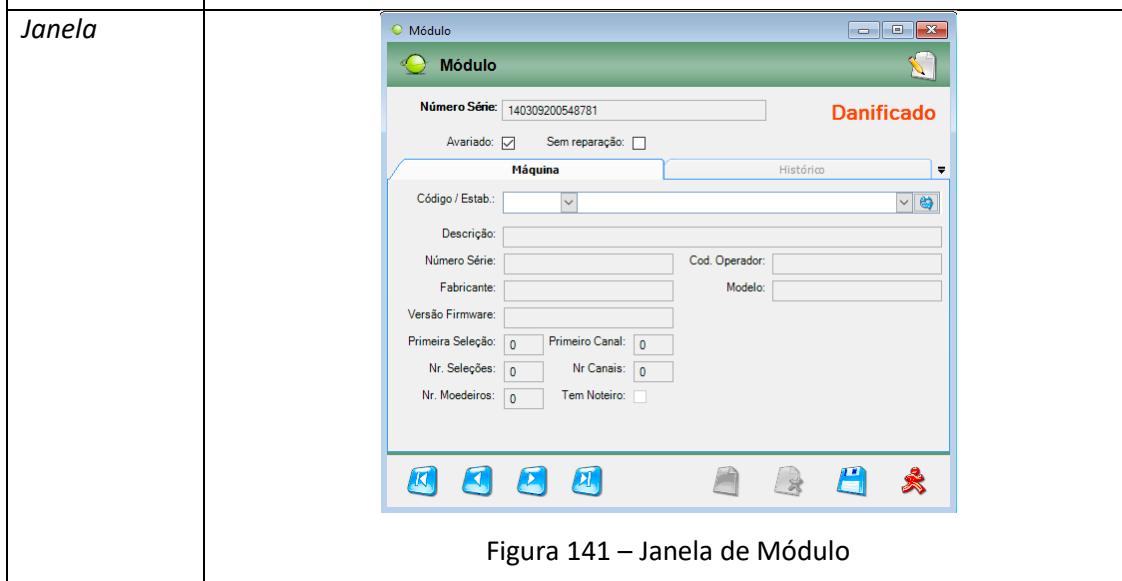
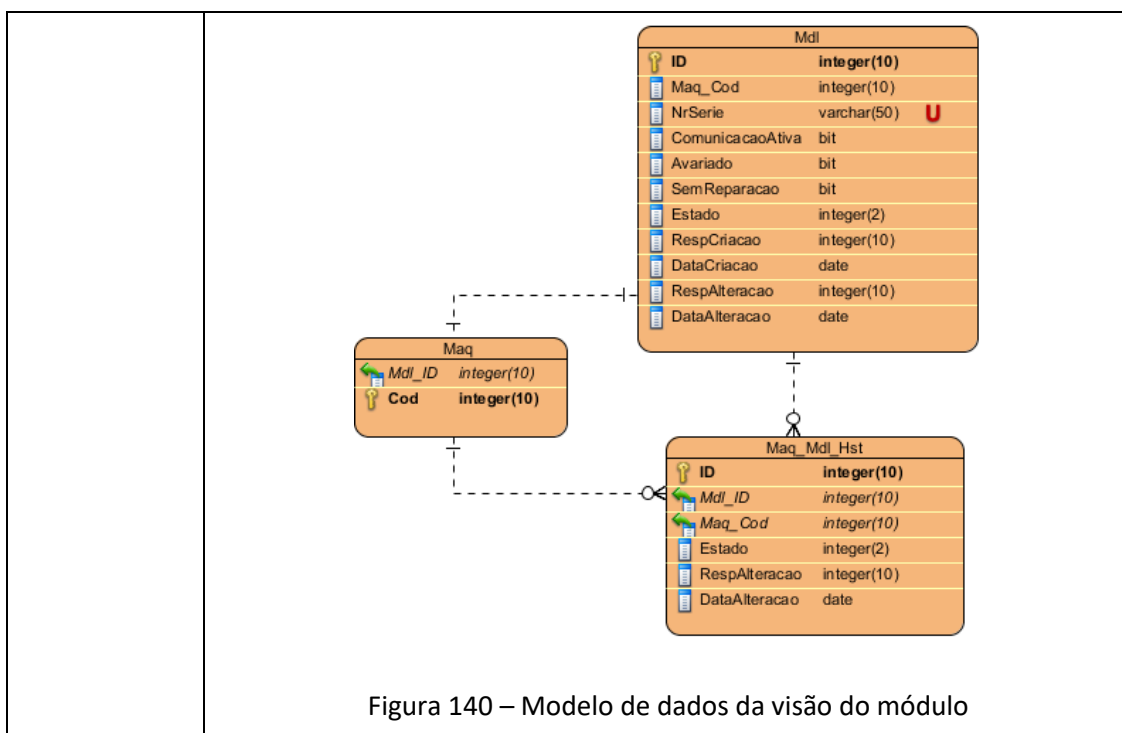


Tabela 84 – Caso de uso de Atribuir Máquina ao Módulo

Nome	Atribuir Máquina ao Módulo
ID	UC_WVB_1071
Descrição	Atribuir uma máquina ao módulo em edição.
Atores	Utilizador (Administrativo, Administrador) e Sistema Middleware
Pré-condições	O utilizador está autenticado e tem permissões de edição.

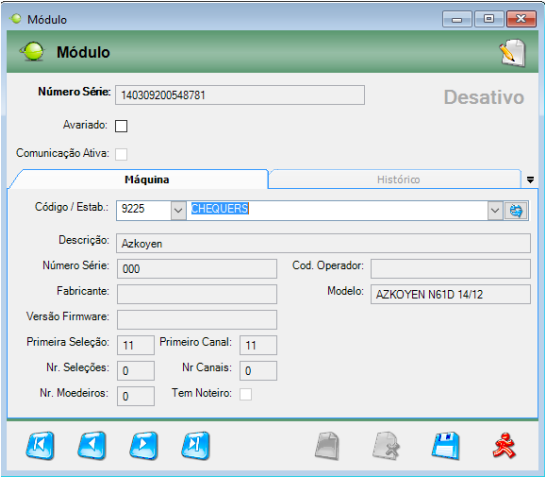
<p><i>Cenário Principal</i></p>	<p>Ao entrar-se na janela de “Módulo” a partir da janela “Tabelas” (c.f. Figura 142), permite-se a atribuição de uma máquina ao módulo, desde que o estado seja “Livre”.</p> <p>Observam-se, em modo de leitura, os seguintes dados da máquina: descrição, número de série, código de operador, fabricante, modelo, versão de <i>firmware</i>, primeira seleção / canal, número de seleções / canais, número de moedeiros e presença de noteiro.</p>
<p><i>Cenário Alternativo 1</i></p>	<p>Ao abrir-se a janela de “Módulo” no estado “Desativo”, é possível desassociar-se a máquina do módulo e ao concretizar-se essa ação o estado passa para “Livre”. Visualizam-se todos os dados em modo de leitura.</p>
<p><i>Cenário Alternativo 2</i></p>	<p>Ao abrir-se a janela de “Módulo” no estado “Ativo”, não é possível efetuar-se qualquer tarefa de manutenção. Visualiza-se, apenas, os dados em modo de leitura.</p>
<p><i>Exceções</i></p>	<p>EX 1 Falha o pedido de alteração do estado do módulo ao Middleware.</p> <ol style="list-style-type: none"> <li>1. O sistema notifica o utilizador da falha ocorrida;</li> <li>2. Retorna para a mesma janela.</li> </ol>
<p><i>Janela</i></p>	<div style="text-align: center;">  </div> <p style="text-align: center;">Figura 142 – Janela de Módulo – Atribuir Máquina</p>

Tabela 85 – Caso de uso de Atribuir Módulo à Máquina

<p><i>Nome</i></p>	<p>Atribuir Módulo à Máquina</p>
<p><i>ID</i></p>	<p>UC_WVB_1072</p>
<p><i>Descrição</i></p>	<p>Atribuir um módulo à máquina em edição.</p>
<p><i>Atores</i></p>	<p>Utilizador (Administrativo, Administrador) e Sistema Middleware</p>

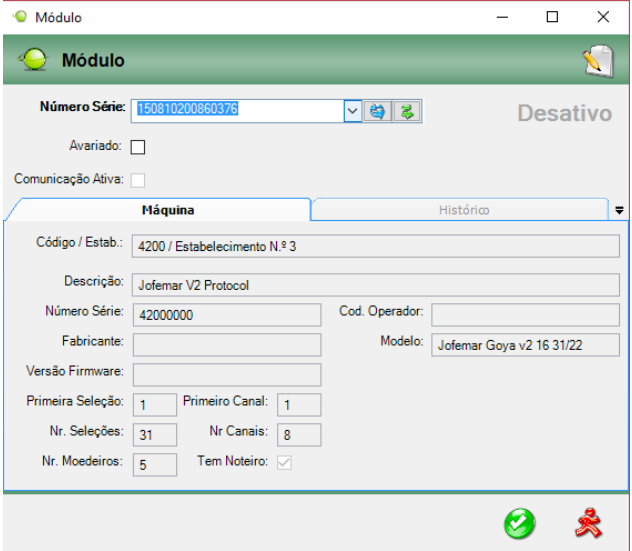
<i>Pré-condições</i>	O utilizador está autenticado e tem permissões de edição. A máquina suporta a utilização da componente da telemetria.
<i>Cenário Principal</i>	Ao entrar-se na janela de “Módulo” a partir da janela “Máquina” (c.f. Figura 143), permite-se a atribuição de um módulo à máquina, desde que o estado seja “Livre”. Observam-se, em modo de leitura, os dados da máquina.
<i>Cenário Alternativo</i>	Ao abrir-se a janela de “Módulo” no estado “Desativo”, é possível desassociar-se o módulo da máquina e, ao concretizar-se essa ação, o estado passa para “Livre”. Visualizam-se todos os dados em modo de leitura.
<i>Exceções</i>	EX 1 Falha o pedido de alteração do estado do módulo ao Middleware. 3. O sistema notifica o utilizador da falha ocorrida; 4. Retorna para a mesma janela.
<i>Janela</i>	 <p>Figura 143 – Janela de Módulo – Atribuir Módulo</p>

Tabela 86 – Consultar Histórico Módulo

<i>Nome</i>	Consultar Histórico Módulo
<i>ID</i>	UC_WVB_1073
<i>Descrição</i>	Visualizar o histórico de estados do módulo
<i>Atores</i>	Utilizador (Administrativo, Administrador)
<i>Pré-condições</i>	O utilizador está autenticado e tem permissões de visualização.
<i>Cenário</i>	Ao selecionar-se o separador “Histórico” (c.f. Figura 144), consultam-se todas as alterações realizadas ao estado de um determinado módulo: (i)

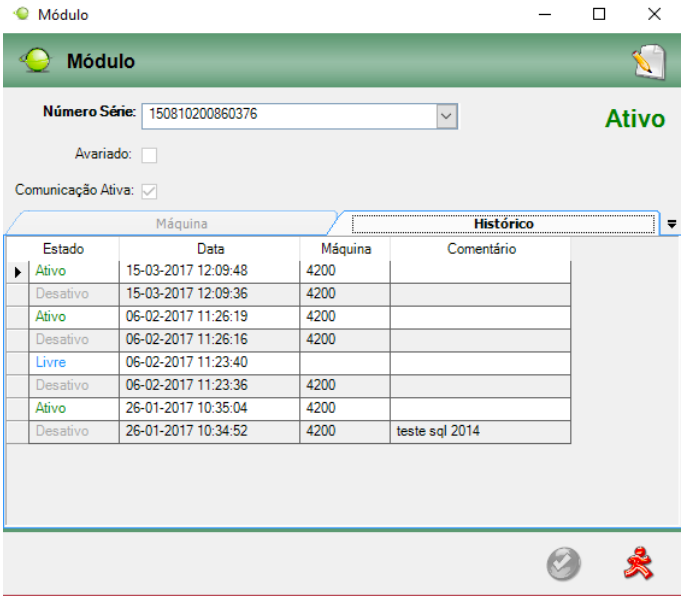
	estado, (ii) data/hora de alteração do estado, (iii) código da máquina associada, (iv) comentário introduzido pelo utilizador responsável pela mudança do estado.
	EX 1 Falha o pedido de comunicação com a máquina. 1. O sistema notifica o utilizador da falha ocorrida; 2. Retorna para a mesma janela.
Janela	 <p>Figura 144 – Janela de Módulo – Histórico</p>

Tabela 87 – Caso de uso de Notificar Alarmes

Nome	Notificar Alarmes
ID	UC_WVM_0001
Descrição	Notificar alarmes
Atores	Sistema Middleware e Sistema WinGuard
Pré-condições	O sistema está com as parametrizações definidas e as máquinas estão com a componente de telemetria ativa.
Cenário	A partir do momento em que as comunicações da máquina se encontram ativas, o sistema Middleware notifica o sistema <i>Backoffice</i> dos alarmes ocorridos na máquina (e.g. porta aberta/fechada, máquina ligada/desligada, movimento e localização da máquina).

	Aquando da notificação de um alarme de uma máquina, o sistema <i>Backoffice</i> verifica se o cliente, ligado à máquina, possui uma “Central de Alarmes” associada. No caso positivo, confirma, ainda, se o alarme ocorre fora do horário de trabalho do estabelecimento e, ao verificar-se essa condição, reencaminha-se o alarme, automaticamente, para o sistema WinGuard (c.f. Figura 28).
	EX 1 O sistema <i>Backoffice</i> não está disponível. 1. Perde-se o alarme.

Tabela 88 – Caso de uso de Notificar Operações

<i>Nome</i>	Notificar Operações
<i>ID</i>	UC_WVM_0002
<i>Descrição</i>	Notificar alarmes
<i>Atores</i>	Sistema Middleware e Sistema WinGuard
<i>Pré-condições</i>	O sistema está com as parametrizações definidas e as máquinas estão com a componente de telemetria ativa.
<i>Cenário</i>	O sistema <i>Backoffice</i> realiza pedidos de comunicação e programação (canais, seleções, preços, código de operador e mensagem publicitária) ao sistema Middleware. Após o tratamento dos pedidos o sistema Middleware envia ao sistema <i>Backoffice</i> uma resposta positiva ou negativa (c.f. Figura 29).
	EX 1 O sistema <i>Backoffice</i> não está disponível. 1. Perde-se a recepção da operação.

### 11.1.2 WinGuard

Nesta seção, consultam-se os casos de uso da solução WinGuard.

Tabela 89 – Caso de uso de Notificar Alarme

<i>Nome</i>	Notificar Alarme
<i>ID</i>	UC_WVB_001
<i>Descrição</i>	Notificar alarmes
<i>Atores</i>	Sistema WinGuard e Sistema WinVending
<i>Pré-condições</i>	O sistema está com as parametrizações definidas.
<i>Cenário</i>	O sistema <i>Backoffice</i> notifica o sistema WinGuard da existência de um novo alarme. O sistema WinGuard procede ao registo e à categorização do alarme, segundo as próximas condições.
	EX 1 O sistema <i>WinGuard</i> não está disponível. 2. Perde-se a receção da operação.

Tabela 90 – Caso de uso de Consultar Alarmes

<i>Nome</i>	Consultar Alarmes
<i>ID</i>	UC_WVG_001
<i>Descrição</i>	Consultar os alarmes sob vários critérios em formato de lista ou mapa.
<i>Atores</i>	Sistema WinGuard
<i>Pré-condições</i>	O sistema está com as parametrizações definidas.
<i>Cenário</i>	Ao abrir-se a aplicação WinGuard, observa-se, do lado direito, uma lista de alarmes e, do lado esquerdo, um mapa com a localização das respetivas máquinas (c.f. Figura 145). A listagem das máquinas encontra-se agrupada em três categorias: (i) prioritários, (ii) informativos e (iii) visão personalizada. De modo predefinido, são automaticamente apresentados os alarmes prioritários. Cada linha da lista apresenta os seguintes dados da máquina: (i) o ícone figurativo do último alarme, (ii) o nome do estabelecimento, (iii) o número da máquina, (iv) o contacto do estabelecimento, (v) a data e hora do alarme e (vi) uma ilustração dos diferentes alarmes ocorridos na máquina, com o

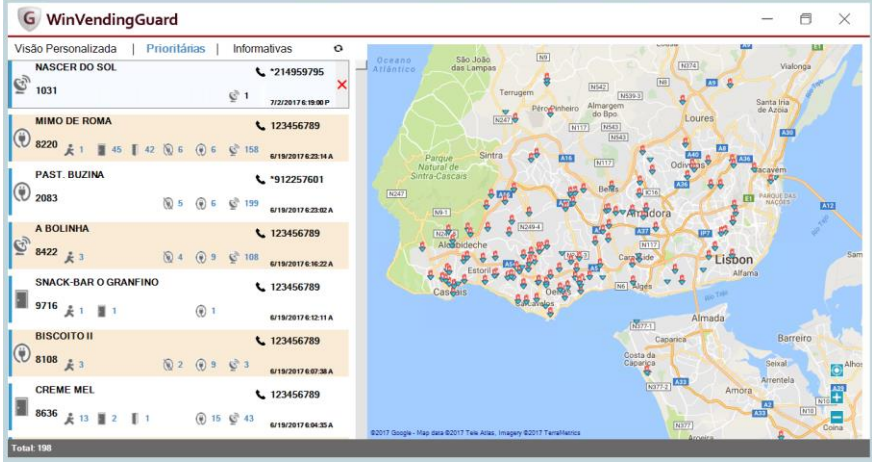
	<p>respetivo número de ocorrências. Permite-se, após selecionar-se uma máquina, o fecho de todos os alarmes associados.</p> <p>O mapa apresenta a localização de todas as máquinas listadas no separador selecionado. Ao efetuar-se uma aproximação do mapa, observa-se um ícone ilustrativo do último alarme rececionado da máquina em causa.</p>
<p><i>Janela</i></p>	 <p>Figura 145 – Janela Principal WinGuard (Visão Geral)</p>

Tabela 91 – Caso de uso de Consultar Alarmes Com Vista Personalizada

<i>Nome</i>	Consultar Alarmes Com Vista Personalizada
<i>ID</i>	UC_WVG_002
<i>Descrição</i>	Consultar os alarmes das máquinas filtrados por um conjunto de critérios.
<i>Atores</i>	Sistema WinGuard
<i>Pré-condições</i>	O sistema está com as parametrizações definidas.
<i>Cenário</i>	Com a consulta personalizada (c.f. Figura 146) permite-se a seleção de um conjunto de filtros de modo a visualizarem-se os alarmes pretendidos. Destacam-se os seguintes filtros: (i) intervalo de datas, (ii) intervalo de número de máquina, (iii) critério de fechada, (iv) critério lida e (v) listagem de tipos de ocorrência. Após a confirmação dos filtros, apresenta-se uma lista de alarmes, segundo os filtros selecionados.

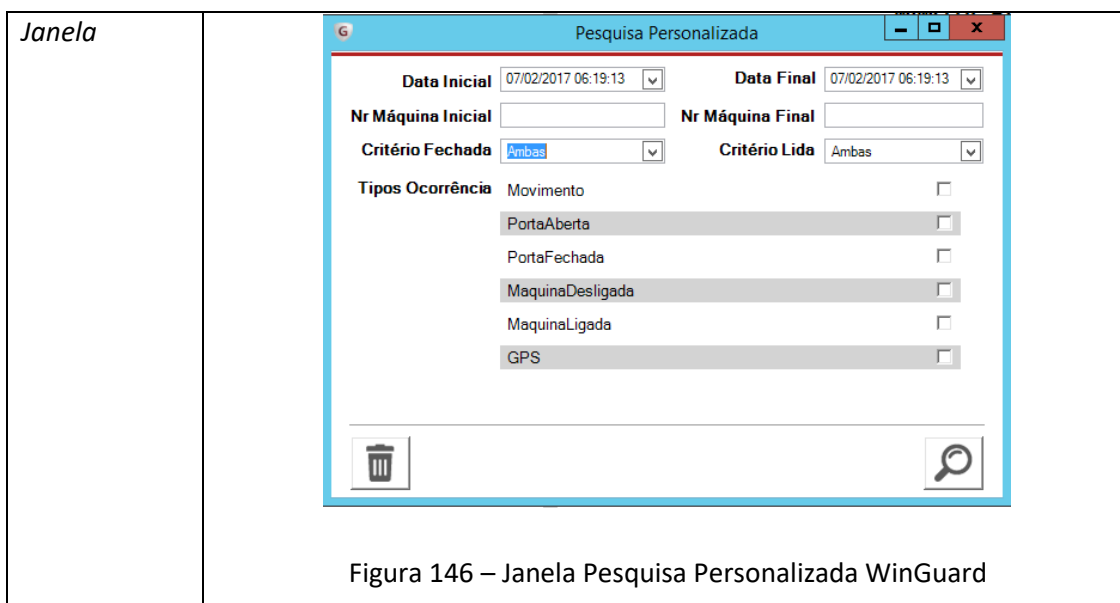


Tabela 92 – Caso de uso de Consultar Alarmes Prioritários

<i>Nome</i>	Consultar Alarmes Prioritários
<i>ID</i>	UC_WVG_003
<i>Descrição</i>	Consultar todos os alarmes prioritários.
<i>Atores</i>	Sistema WinGuard
<i>Pré-condições</i>	O sistema está com as parametrizações definidas.
<i>Cenário</i>	A consulta de alarmes prioritários é a consulta predefinida, ao abrir-se a aplicação e apresenta todos os alarmes prioritários, ainda não fechados.

Tabela 93 – Caso de uso de Consultar Alarmes Informativos

<i>Nome</i>	Consultar Alarmes Informativos
<i>ID</i>	UC_WVG_004
<i>Descrição</i>	Consultar todos os alarmes informativos.
<i>Atores</i>	Sistema WinGuard
<i>Pré-condições</i>	O sistema está com as parametrizações definidas.
<i>Cenário</i>	A consulta de alarmes informativos permite a visualização de todos os alarmes informativos não fechados.

Tabela 94 – Caso de uso de Consultar Alarmes Por Máquina

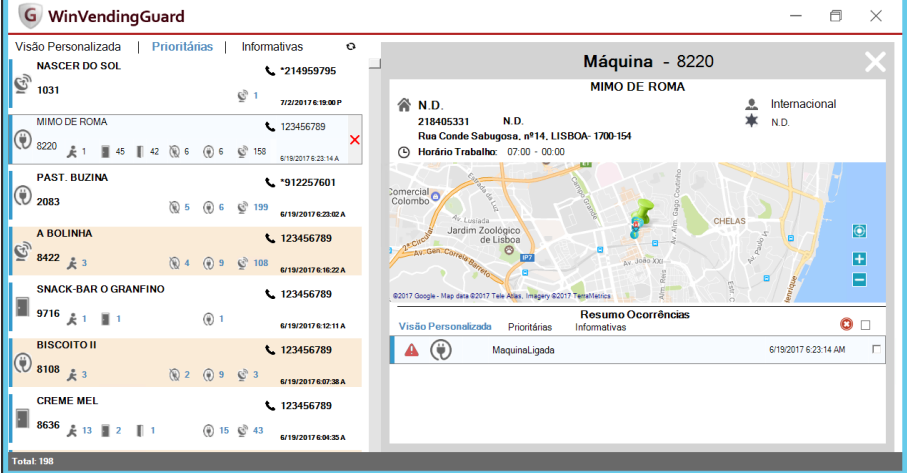
<i>Nome</i>	Consultar Alarmes Por Máquina
<i>ID</i>	UC_WVG_010
<i>Descrição</i>	Consultar alarmes de uma máquina.
<i>Atores</i>	Sistema WinGuard
<i>Pré-condições</i>	O sistema está com as parametrizações definidas.
<i>Cenário</i>	<p>Ao selecionar-se uma máquina, visualizam-se os dados completos conhecidos da máquina (c.f. Figura 147): (i) nome do estabelecimento, (ii) número da máquina, (iii) morada, (iv) número de telefone, (v) número de telemóvel, (vi) nome do depositário, (vii) identificação da autoridade a contactar em caso de emergência, (viii) horário de trabalho, (ix) mapa com a localização da máquina, (x) lista de alarmes da máquina.</p> <p>Permitem-se diferentes visões da lista de alarmes: prioritários, informativos e com uma vista personalizada. A listagem de máquinas é composta, independentemente da categoria selecionada, pelos seguintes campos: (i) a categoria do alarme, (ii) um ícone ilustrativo do tipo de alarme, (iii) a descrição do tipo de alarme e (iv) a data do alarme.</p> <p>Possibilita-se, ainda, o fecho de um alarme específico ou de um conjunto de alarmes.</p>
<i>Janela</i>	 <p>The screenshot displays the WinVendingGuard application. On the left, a list of machines is shown with columns for name, machine number, phone number, and status. The selected machine, 'MIMO DE ROMA' (ID 8220), is highlighted. On the right, a detailed view of this machine is shown, including its address, phone number, and a map of its location in Lisbon. The interface also shows a 'Resumo Ocorrências' (Summary Occurrences) section at the bottom right.</p>

Figura 147 - Janela Principal WinGuard (Visão Máquina)

Tabela 95 – Caso de uso de Fechar Alarme

<i>Nome</i>	Fechar alarmes
<i>ID</i>	UC_WVG_020
<i>Descrição</i>	Fechar alarmes das máquinas.
<i>Atores</i>	Sistema WinGuard
<i>Pré-condições</i>	O sistema está com as parametrizações definidas.
<i>Cenário Máquina</i>	Ao efetuar-se a opção de fecho, através da seleção de uma máquina, fecham-se todos os alarmes relacionados.
<i>Cenário Alarme</i>	Ao concretizar-se a operação de fecho na lista de alarmes, permite-se a seleção dos alarmes que se pretendem fechar.

## 11.2 Modelo de dados

Nesta secção, referem-se as abreviaturas de tabelas e campos (c.f. Tabela 96) do modelo de dados do sistema WinVending.

<b>Abreviatura</b>	<b>Descrição</b>
Agn	Agendamento
Arm	Armazém
Can	Canal
Cli	Cliente
Cod	Código
Ctb	Contabilidade
Desc	Descrição
Ent	Entidade
Estab	Estabelecimento
Fam	Família
Fil	Filial
Frn	Fornecedor

Grp	Grupo
Hst	Histórico
Loc	Localização
Maq	Máquina
Max	Máximo
Mdl	Módulo
Med	Média
Min	Mínimo
Moe	Moeda
Nr	Número
Obj	Objetivo
Obs	Observações
Prd	Produto
Prg	Programação
Prim	Primeir(o)(a)
Prm	Promotor
Qtd	Quantidade
Resp	Responsável
Rot	Rota
Sel	Seleção
Snc	Sincronização
Stk	Stock
Tab	Tabela
Tel	Telefone
Tlm	Telemóvel
Tmb	Tipo Modelo Base
Tp	Tipo
Tpm	Tipo Máquina
Und	Unidade
Vnd	Vendedor
WV	WinVending

Tabela 96 – Abreviaturas de tabelas e campos

## 11.3 Histórico de alterações

Nesta secção, apresenta-se uma entrada do ficheiro de registo de histórico de alterações da aplicação: WinVending. Nele, descrevem-se e categorizam-se (c.f. Tabela 97, Tabela 98 e Tabela 99) as alterações efetuadas numa determinada versão da aplicação (e.g. 17.0.5.0).

Tabela 97 – Histórico de alterações (Novas funcionalidades)

Novas Funcionalidades	
○	<b>Cliente</b>
○	Novo campo que indica se o vendedor do cliente é automaticamente atualizado nas máquinas em que está associado, no caso de alteração do vendedor na ficha do cliente ("É vendedor máquina")
○	<b>Filial</b>
○	Novos campos indicativos da localização geográfica da filial (Longitude / Latitude)
○	<b>Máquina</b>
○	<b>Cliente</b>
▪	Se cliente tem o campo "É vendedor máquina" selecionado então: <ul style="list-style-type: none"><li>• Vendedor não é editável</li><li>• Ao alterar cliente altera vendedor</li></ul>
○	<b>Localização</b>
▪	Caso as coordenadas de instalação e inicial não estejam definidas é apresentada uma mensagem, sob a forma de ligação, com o seguinte texto: "Coordenadas de instalação não definidas. Pressione nesta ligação para atribuir a posição de instalação."
○	<b>Assistente de Programação (Última comunicação)</b>
▪	Novo botão para apresentar / ocultar o número de produto nas seleções
○	<b>Comunicações</b>
▪	Novo critério de suspensão de comunicação quando comunicação inclui linhas sem produto atribuído e resultado de comunicação é parcial
○	<b>Rota</b>
○	Alteração no método de recálculo da rota de modo a assumir sempre o vendedor máquina nos casos em que a visita é realizada à máquina

Tabela 98 – Histórico de alterações (Melhoramentos)

Melhoramentos	
○	<b>Comunicação</b>
○	Apresentação do campo "Originou Abastecimento" no cabeçalho da comunicação
○	Apresentação do campo "Código Produto Máquina" nas linhas da comunicação
○	<b>Máquina</b>

- **Visita**
  - Ajuste visual das tabelas: alarmes, moedeiros e anomalias
- **Localização**
  - No caso de algum campo necessário para a obtenção automática das coordenadas de instalação não esteja atribuído é solicitada a confirmação antes de trocar de separador

Tabela 99 – Histórico de alterações (Correções)

Correções	
○ <b>Máquina</b>	<ul style="list-style-type: none"> <li>○ <b>Visita</b> <ul style="list-style-type: none"> <li>▪ Correção da verificação do estado da visita anterior (e.g. terminada)</li> </ul> </li> <li>○ <b>Assistente de Programação (Última comunicação)</b> <ul style="list-style-type: none"> <li>▪ Apresentação dos dados da última comunicação nas seleções ou canais consoante o tipo de contabilidade da última comunicação</li> <li>▪ Correção da identificação do produto máquina;</li> </ul> </li> <li>○ <b>Moedeiros</b> <ul style="list-style-type: none"> <li>▪ Correção na criação de novos moedeiros</li> </ul> </li> </ul>
○ <b>Rota</b>	<ul style="list-style-type: none"> <li>○ Ao criar / alterar cliente manualmente na tabela verifica-se se o cliente tem máquinas associadas</li> </ul>

## 11.4 Amostras

Nesta secção, visualiza-se uma amostra dos dados do número de volumes de tabaco transportados nas carrinhas, num determinado cliente, em dois períodos diferentes: antes e depois da adoção da telemetria. Realça-se que os dados, após telemetria, incluem uma reserva de stock para colmatar necessidades inesperadas.

Tabela 100 – Amostra de stock circulante antes e depois da adoção da telemetria

Antes da Telemetria	Antes Telemetria com uma Melhoria de 15 %	Após Telemetria
3892	3308	1835
3510	2983	3833
2930	2490	4193

3550	3017	3388
3620	3077	3648
3410	2898	3358
2790	2371	4183
3010	2558	3130
3810	3238	3055
3730	3170	2470
4430	3765	2589
3630	3085	3405
3280	2788	3860
3720	3162	2610
3780	3213	2655
3560	3026	2535
2850	2422	3175
2550	2167	3900
3280	2788	2890
3550	3017	2240
3440	2924	2745
2450	2082	3160
2570	2184	3590
2880	2448	2705
3220	2737	2525
2950	2507	2345
2580	2193	2340
2820	2397	3580
3530	3000	2615
3000	2550	2800
4080	3468	3130
3890	3306	4550
3590	3051	4895
3570	3034	1850
3650	3102	2220

3120	2652	3735
2520	2142	3520
3810	3238	3095
1500	1275	2180
3820	3247	3110
2680	2278	3370
3030	2575	3165
3670	3119	2510
3780	3213	2230
4030	3425	2970
2810	2388	2870
3130	2660	3195
3330	2830	2680
3520	2992	3020
3880	3298	3360
2760	2346	3075
2830	2405	4470
3510	2983	4585