

Estudo e desenvolvimento de soluções de localização/internacionalização

JOÃO PEDRO GUEDES TAVARES

Julho de 2019

Estudo e desenvolvimento de soluções de localização/internacionalização

João Pedro Guedes Tavares

**Dissertação para obtenção do Grau de Mestre em
Engenharia Informática, Área de Especialização em
Engenharia de Software**

Orientador: Nuno Filipe Fonseca Vasconcelos Escudeiro

Supervisor: Pedro Fernandes

Declaração de Confidencialidade

Nos termos do contrato de estágio curricular e do acordo de confidencialidade celebrado com a ALERT Life Sciences Computing, S.A. (“ALERT”), o presente relatório é confidencial e poderá conter referências a invenções, *know-how*, desenhos, programas de computador, segredos comerciais, produtos, fórmulas, métodos, planos, especificações, projetos, dados ou obras abrangidos por direitos de propriedade industrial e/ou intelectual da ALERT. Este relatório só poderá ser utilizado para efeitos de investigação e de ensino. Qualquer outro tipo de utilização está sujeita a autorização prévia e por escrito da ALERT.

Dedicatória

«Dedico este trabalho ao meu avô materno, o meu herói, mesmo não estando presente fisicamente sei que me acompanha todos os dias. Obrigado por tudo!»

Resumo

A empresa ALERT desenvolve sistemas de informação para a área da saúde, comercializando os seus produtos em várias regiões do globo. Para tal necessita de uma solução atual, fiável, rápida e eficaz para o processo de internacionalização e localização destes produtos.

Apesar de alguns locais do globo já usufruírem das soluções criadas pela empresa, esta, devido à atual fase de reconversão de código Flash/Flex para HTML5, pretende consolidar o processo de internacionalização e localização dos seus produtos para que estes sejam corretamente adaptados às diferentes culturas dos locais do mundo.

O documento descreve o processo desenvolvido de internacionalização e localização de *software* dos produtos ALERT®, nomeadamente do ALERT® PFH, tendo em conta os problemas existentes e já identificados relacionados com estes temas, bem como as necessidades da gama de produtos da empresa. Foram criadas soluções para dar resposta a cada um destes problemas, utilizando-se ferramentas que auxiliavam o seu desenvolvimento. Ainda, como os clientes são fatores importantes para a evolução dos produtos, é-lhes possibilitada a personalização de conteúdos relacionados com o tema, pelo que o processo em causa dá resposta a estes ajustes.

Avaliou-se o processo desenvolvido e a hipótese formulada, através de inquéritos a elementos da equipa de Arquitetura da empresa e de outros dados estatísticos. Concluiu-se que a solução de *software* desenvolvida permite dar resposta às necessidades dos produtos da ALERT, não afetando as suas funcionalidades já existentes.

Palavras-chave: ALERT, Internacionalização, Localização, Cuidados de Saúde, Local

Abstract

ALERT develops information systems for the healthcare area and provides their products to various regions of the globe. For this purpose, ALERT needs an up-to-date, reliable, agile and efficient solution that will suit the internationalization and localization process of their products.

Although some places of the world already use the solutions made by ALERT, and because of the current phase of code reversion from Flash/Flex to HTML5, the company needs to reinforce the internationalization and localization process for its products so that they are correctly adapted to the different places and cultures of the globe.

This document describes the internationalization and localization process developed for ALERT's software products, namely ALERT® PFH, taking into account the already existing and identified problems related to this subject as well as the particular needs of the company's product suite. For each of these problems a solution was created and the tools used aimed to help the development of solutions. Also, as the clients are an important factor to the evolution of the products, they are allowed to customize content, so the process in question responds to the needed adjustments.

The develop process and the formulated hypothesis were evaluated through questionnaires to elements of the company's Architecture team and through the analysis of other statistical data. It was concluded that the developed software solution corresponds to the needs of ALERT's products, without affecting its existing functionalities.

Keywords: ALERT, Internationalization, Localization, Healthcare, Local

Agradecimentos

Em primeiro lugar, quero agradecer à empresa ALERT por me permitir desenvolver este projeto aliciante junto de uma equipa fantástica. Um agradecimento especial a todos os elementos da equipa de Arquitetura por todo o suporte, ajuda, paciência, ensinamentos e disponibilidade dados. Obrigado pelo ambiente incrível e por me ajudarem a tornar num melhor profissional e numa melhor pessoa!

Seguidamente, agradecer ao Instituto Superior de Engenharia do Porto (ISEP) e aos seus docentes pelos ensinamentos transmitidos e por todo o apoio fornecido durante estes dois anos de mestrado. Em especial ao Professor Doutor Nuno Escudeiro por me ter acompanhado e ajudado durante todo o desenvolvimento deste projeto.

Um agradecimento muito especial aos meus pais, por terem tornado este percurso académico possível e sempre acreditarem em mim, bem como por me darem constantemente alento para atingir os meus objetivos. Agradeço também ao meu irmão e a toda a minha família, por serem um dos grandes pilares da minha vida. Pelos momentos que vivemos e por me moldarem e ajudarem a ser a pessoa que sou hoje. Obrigado!

Por fim, um agradecimento sentido aos meus amigos, que estiveram presentes em todos os momentos, inclusive naqueles em que mais precisei e por me proporcionarem memórias inesquecíveis.

A todos os que estão presentes e aos que já partiram deixando enorme saudade, um eterno obrigado!

Índice

Lista de Figuras	xv
Lista de Tabelas	xvii
Lista de Extratos de Código	xix
Acrónimos	xxi
1 Introdução	1
1.1 Contexto	1
1.2 Problema.....	2
1.3 Hipótese.....	3
1.4 Objetivos.....	3
1.5 Motivação Pessoal	4
1.6 Planeamento do Projeto	4
1.7 Estrutura do Documento	5
2 Estado de Arte	7
2.1 Internacionalização vs Localização.....	7
2.1.1 Globalização	7
2.1.2 Internacionalização	8
2.1.3 Localização	10
2.1.4 Problemas de Internacionalização e Localização.....	10
2.1.5 Processo para Internacionalizar e Localizar <i>Software</i>	20
2.1.6 Aplicações externas de apoio à internacionalização e localização.....	23
2.1.7 Ferramentas de Internacionalização e Localização	27
2.2 Enquadramento Tecnológico	35
3 Contexto	37
3.1 Problema.....	37
3.2 Partes Interessadas	39
3.3 Requisitos.....	40
3.3.1 Requisitos Funcionais	40
3.3.2 Requisitos Não Funcionais	42
3.4 ALERT® PFH	42
3.5 Análise de Valor	43
3.5.1 Orientação.....	45
3.5.2 Identificação e Análise Funcional	51
3.5.3 Criação de Alternativas	58
3.5.4 Análise e Avaliação	59

3.5.5	Implementação	75
4	Análise e Desenho	77
4.1	Processo de Internacionalização e Localização	77
4.1.1	Processo detalhado de Internacionalização	79
4.1.2	Processo detalhado de Localização	81
4.2	Análise do ALERT® PFH.....	83
4.3	Melhorias a efetuar	89
4.3.1	Melhorias ao projeto “labels-manager”	89
4.3.2	Outras melhorias	91
5	Desenvolvimento	95
5.1	Desenvolvimento do projeto “labels-manager”	95
5.1.1	Página inicial do “labels-manager”	95
5.1.2	Página central do “labels-manager”	97
5.1.3	Página de gestão de clientes	102
5.1.4	Página de gestão dos ficheiros JSON com traduções.....	103
5.1.5	Alterações ao ALERT® PFH	105
5.1.6	Exemplo prático - pedido de tradução de cliente	107
5.2	Desenvolvimento de outras melhorias	109
5.2.1	Armazenamento de personalizações de clientes	109
5.2.2	Implementação da solução relativa às fontes de texto	111
5.2.3	Integração da ferramenta Angular Locale	114
5.2.4	Exemplo prático - alteração do separador decimal a pedido do cliente	117
5.3	Conclusões	118
6	Experimentação e Avaliação	121
6.1	Inquérito	121
6.1.1	Guião	122
6.1.2	Caracterização da população	123
6.1.3	Apresentação do Inquérito.....	123
6.1.4	Resultado dos Inquéritos	124
6.2	Outros indicadores	133
6.3	Conclusões	135
7	Conclusão	137
7.1	Objetivos Alcançados	137
7.2	Limitações e Trabalho Futuro	138
7.3	Apreciação Final	139
	Referências.....	141
	Anexo A Suíte de produtos ALERT	149
	Anexo B Consistência dos Critérios.....	153

Anexo C	Grunt.....	157
Anexo D	Ranch	159
Anexo E	Ficheiro “angular-locale-pt-pt”	161
Anexo F	Estado de desenvolvimento de requisitos	163

Lista de Figuras

Figura 1 – Globalização	8
Figura 2 – Diferentes orientações de texto numa página HTML	12
Figura 3 – Resultado da associação de <i>labels</i>	13
Figura 4 – Posições de <i>labels</i> possíveis	14
Figura 5 – Significado de ícones na leitura/escrita da esquerda para a direita	14
Figura 6 – Significado de ícones na leitura/escrita da direita para a esquerda	14
Figura 7 – Diferenças dos símbolos de email.....	15
Figura 8 – Símbolos associados a paragem.....	15
Figura 9 – Imagem possível de ferir suscetibilidades em certas culturas.....	16
Figura 10 – Imagem genérica	16
Figura 11 – Diferenças nos tamanhos das traduções	18
Figura 12 – Questões iniciais do processo de internacionalização	20
Figura 13 – Evolução do <i>software</i> internacionalizado e sucesso/retorno associado	21
Figura 14 – Exemplo de estrutura de diretórios	23
Figura 15 – Processo de <i>Value Analysis</i>	44
Figura 16 – Processo de inovação de um produto.....	45
Figura 17 – <i>New Concept Development</i>	46
Figura 18 – <i>Customer Chain</i>	52
Figura 19 – Diagrama de árvore.....	54
Figura 20 – Casa de Qualidade	56
Figura 21 – <i>Pairwise Comparison</i> : Processo de Internacionalização e Localização	58
Figura 22 – Árvore de divisão hierárquica	60
Figura 23 – Árvore de divisão hierárquica com valores associados.....	74
Figura 24 – Separação geral do código dos Produtos ALERT®	78
Figura 25 – Processo de Internacionalização	80
Figura 26 – Processo de Localização	82
Figura 27 – Diagrama de componentes ALERT® PFH	84
Figura 28 – ALERT® PFH e componentes relacionados.....	85
Figura 29 – Labels-manager	86
Figura 30 – Processo de criação/edição/eliminação de <i>labels</i>	87
Figura 31 – Ficheiro “i18n.generic.json” e respetivos ficheiro gerados	88
Figura 32 – Escolha do local/língua.....	88
Figura 33 – Diagrama de componentes do “labels-manager” e suas relações.....	91
Figura 34 – Diagrama de sequência representativo do login de um cliente	93
Figura 35 – Escolha da versão do ALERT® PFH.....	96
Figura 36 – Página central do “labels-manager”	97
Figura 37 – Botões de criação/alteração de identificadores de traduções	97
Figura 38 – Restantes botões do cabeçalho da página central do “labels-manager”	98
Figura 39 – Árvore com os identificadores únicos para cada uma das traduções.....	99
Figura 40 – Seleção para o tipo de traduções a inserir	100

Figura 41 – Área de traduções feitas pela empresa	101
Figura 42 – Área de traduções para pedidos de clientes	102
Figura 43 – Página de gestão de clientes	103
Figura 44 – Página de gestão dos ficheiros JSON com traduções	104
Figura 45 – Uso do identificador único da tradução	106
Figura 46 – Alterações ao ALERT® PFH.....	107
Figura 47 – Interface gráfica de seleção de produto.....	108
Figura 48 – Diagrama de operações a realizar no “labels-manager”	108
Figura 49 – Alterações à interface gráfica de seleção do produto.....	109
Figura 50 – Diferença de fontes de texto	114
Figura 51 – Diferenças no sentido da direção de texto.....	116
Figura 52 – Componente “keypad-numeric” com “.” como separador decimal	117
Figura 53 – Componente “keypad-numeric” com “,” como separador decimal	118
Figura 54 – Testes unitários realizados ao “keypad-numeric”	119
Figura 55 – Curso/documentação sobre o uso do projeto “labels-manager”	119
Figura 56 – Curso/documentação sobre o uso do Angular Locale.....	120
Figura 57 – Diagrama de atividades com as várias fases do guião	122
Figura 58 – Resposta à questão 1 do inquérito.....	125
Figura 59 – Resposta à questão 2 do inquérito.....	125
Figura 60 – Resposta à questão 3 do inquérito.....	125
Figura 61 – Resposta à questão 4 do inquérito.....	126
Figura 62 – Resposta à questão 5 do inquérito.....	126
Figura 63 – Resposta à questão 6 do inquérito.....	126
Figura 64 – Resposta à questão 7 do inquérito.....	127
Figura 65 – Resposta à questão 8 do inquérito.....	127
Figura 66 – Resposta à questão 9 do inquérito.....	127
Figura 67 – Consulta da documentação/curso sobre o projeto “labels-manager”	133
Figura 68 – Consulta da documentação/curso sobre o Angular Locale	134
Figura 69 – Suíte de produtos ALERT®	149
Figura 70 – Plataforma Ranch	159

Lista de Tabelas

Tabela 1 – Diferenças nos formatos de datas	19
Tabela 2 – Diferentes formatos numéricos.....	19
Tabela 3 – Comparação de aplicações	27
Tabela 4 – Comparação de ferramentas.....	33
Tabela 5 – Benefícios e Sacrifícios.....	49
Tabela 6 – <i>Perceived Value</i>	50
Tabela 7 – Escala Fundamental – Níveis de importância de comparações	60
Tabela 8 – Comparação de prioridades dos critérios (matriz de prioridades).....	61
Tabela 9 – Somatório das colunas de cada critério	62
Tabela 10 – Comparação de prioridades dos critérios normalizados	63
Tabela 11 – Prioridades dos critérios	63
Tabela 12 – Valores IR definidos pelo Laboratório Nacional de Oak Ridge	64
Tabela 13 – Data de última contribuição e número de programadores de cada uma das ferramentas.....	66
Tabela 14 – Matriz de prioridades das alternativas para o critério de suporte por parte da comunidade	66
Tabela 15 – Somatório de cada coluna das alternativas do critério de suporte por parte da comunidade	66
Tabela 16 – Valores normalizados das alternativas do critério de suporte por parte da comunidade	67
Tabela 17 – Prioridades das alternativas do critério de suporte por parte da comunidade	67
Tabela 18 – Matriz de prioridades das alternativas para o critério de alcance de diversos tratamentos de problemas de i18n e l10n.....	68
Tabela 19 – Somatório de cada coluna das alternativas do critério de alcance de diversos tratamentos de problemas de i18n e l10n.....	68
Tabela 20 – Valores normalizados das alternativas do critério de alcance de diversos tratamentos de problemas de i18n e l10n.....	69
Tabela 21 – Prioridades das alternativas do critério de alcance de diversos tratamentos de problemas de i18n e l10n.....	69
Tabela 22 – Matriz de prioridades das alternativas para o critério de adoção de <i>standards</i> internacionais.....	70
Tabela 23 – Somatório de cada coluna das alternativas do critério de adoção de <i>standards</i> internacionais.....	70
Tabela 24 – Valores normalizados das alternativas do critério de adoção de <i>standards</i> internacionais.....	71
Tabela 25 – Prioridades das alternativas do critério de adoção de <i>standards</i> internacionais ..	71
Tabela 26 – Matriz de prioridades das alternativas para o critério de compatibilidade com diversos navegadores.....	72
Tabela 27 – Somatório de cada coluna das alternativas do critério de compatibilidade com diversos navegadores.....	72

Tabela 28 – Valores normalizados das alternativas do critério de compatibilidade com diversos navegadores	73
Tabela 29 – Prioridades das alternativas do critério de compatibilidade com diversos navegadores	73
Tabela 30 – Dados finais de resposta ao inquérito	135
Tabela 31 – Estado de desenvolvimento de requisitos.....	163

Lista de Extratos de Código

Extrato de código 1 – Exemplo da definição da direção de texto.....	12
Extrato de código 2 – Exemplo de associação de <i>labels</i>	13
Extrato de código 3 – Conteúdo do ficheiro “i18n.generic.json” com identificadores únicos de traduções	87
Extrato de código 4 – Conteúdo do ficheiro gerado em inglês.....	88
Extrato de código 5 – Exemplo com a relação entre versões	96
Extrato de código 6 – Exemplo de um ficheiro resultante do <i>merge</i>	104
Extrato de código 7 – Ficheiro “core_pt_PT.json” criado	105
Extrato de código 8 – Vista da base de dados para armazenar conteúdo do cliente.....	110
Extrato de código 9 – Função para obtenção de dados dos clientes.....	111
Extrato de código 10 – Função criada no serviço REST.....	111
Extrato de código 11 – Chamada ao serviço REST	111
Extrato de código 12 – Ficheiro com definição de fontes de texto	112
Extrato de código 13 – Configuração de fontes de texto para o cliente	112
Extrato de código 14 – Objeto com as fontes de texto.....	113
Extrato de código 15 – Identificador para português de Portugal.....	114
Extrato de código 16 – Inserção da biblioteca Angular Dynamic Locale	114
Extrato de código 17 – Alteração do caminho para os ficheiros do Angular Locale.....	115
Extrato de código 18 – Alteração do local através da biblioteca Angular Dynamic Locale	115
Extrato de código 19 – Inserção das variáveis \$locale e \$filter	115
Extrato de código 20 – Exemplo de uso de \$filter	115
Extrato de código 21 – Exemplo de uso de \$locale	115
Extrato de código 22 – Componente “ <i>keypad-numeric</i> ” sem \$locale	115
Extrato de código 23 – Componente “ <i>keypad-numeric</i> ” com \$locale	116
Extrato de código 24 – Personalização do símbolo decimal pelo cliente	116
Extrato de código 25 – Exemplo de um “Gruntfile”	157
Extrato de código 26 – Ficheiro “angular-locale-pt-pt”	162

Acrónimos

Os acrónimos, que são mencionados durante o presente documento, serão agora apresentados, com os respetivos significados, por ordem alfabética.

Acrónimo	Significado
ADW	ALERT® DATA WAREHOUSE
AHP	Analytic Hierarchy Process
ALERT	A empresa ALERT Life Sciences Computing, S.A.
ALERT®	Produtos pertencentes à empresa ALERT Life Sciences Computing, S.A.
API	Application Programming Interface – Interface de Programação de Aplicações
BI	ALERT® BUSINESS INTELLIGENCE
CLDR	Unicode Common Locale Data Repository
CSV	Comma Separated Values
EDIS	ALERT® EMERGENCY DEPARTMENT INFORMATION SYSTEM
EMR	ALERT® ELECTRONIC MEDICAL RECORD
FFE	Fuzzy Front End
FURPS+	Functionality, Usability, Reliability, Performance, Supportability
GNU	GNU's Not Unix
HIE	ALERT® HEALTH INFORMATION EXCHANGE
HL7	Health Level Seven
HTML	Hypertext Markup Language
i18n	Acrónimo da palavra “Internationalization”
IANA	Internet Assigned Numbers Authority
ICD9	International Classification of Diseases 9 th Revision
ICPC2	International Classification of Primary Care 2 nd Version
IR	Índice de Consistência
ISEP	Instituto Superior de Engenharia do Porto
ISO	International Organization for Standardization
JSON	JavaScript Object Notation
L10n	Acrónimo da palavra “Localization”
LTR	Sentido de leitura da esquerda para a direita (“Left To Right”)
MEI	Mestrado em Engenharia Informática
NCD	New Concept Development
NPD	New Product Development
NPM	Node Package Manager
OBS	ALERT® OBSERVATION ROOM
ORIS	ALERT® OPERATION ROOM INFORMATION SYSTEM
PDMS	ALERT® PATIENT DATA MANAGEMENT SYSTEM

Acrônimo	Significado
PFH	<i>ALERT® PAPER FREE HOSPITAL</i>
PHR	<i>ALERT® PERSONAL HEALTH RECORD</i>
QFD	<i>Quality Function Deployment</i>
RC	Razão de Consistência
REST	<i>Representational State Transfer</i>
RTL	Sentido de leitura da direita para a esquerda (<i>“Right To Left”</i>)
TBX	<i>TermBase eXchange</i>
TMDEI	Tese/Dissertação/Estágio
VA	<i>Value Analysis</i>
VE	<i>Value Engineering</i>
XLIFF	<i>XML Localization Interchange File Format</i>
XML	<i>Extensible Markup Language</i>

1 Introdução

Este capítulo tem como intuito apresentar o projeto desenvolvido, começando por referir o seu contexto, qual o problema que deve ser resolvido, a hipótese a testar e os objetivos a alcançar. Também, apresenta-se a motivação pessoal que levou ao seu desenvolvimento, quais foram os procedimentos adotados de forma a elaborar o mesmo, bem como a estrutura principal de todo o documento para apresentar ao leitor o seu conteúdo.

1.1 Contexto

O presente documento foi desenvolvido no âmbito da unidade curricular de Tese/Dissertação/Estágio (TMDEI), lecionada no segundo ano, do curso de Mestrado em Engenharia Informática (MEI), ramo de Engenharia de Software, no Instituto Superior de Engenharia do Porto (ISEP). TMDEI tem como intuito desafiar os estudantes na medida em que estes devem ter como encargo o desenvolvimento de um projeto complexo, tendo que encontrar uma solução para o mesmo, nunca pondo de parte as ciências e boas práticas de engenharia informática (Pereira, 2018).

Este projeto tem como tema “Estudo e desenvolvimento de soluções de localização e internacionalização” proposto pela empresa ALERT Life Sciences Computing, S.A., de agora em diante referenciada apenas como “ALERT”. Tal como o título indica, o principal objetivo do projeto é o desenvolvimento de soluções de localização e internacionalização independentes dos produtos ALERT® já existentes e que permitam que estes se adaptem às necessidades relacionadas com este tema.

O *software* ALERT® é uma suíte inovadora de documentação clínica que integra o fluxo de trabalho de diferentes profissionais de saúde que interagem com os seus pacientes. Adapta-se a diversos ambientes da saúde, nomeadamente cuidados primários, hospitalar e medicina ao domicílio, sendo utilizado tanto no setor público como no privado. O ALERT® disponibiliza o registo de informação centrado no paciente, através de uma forma inovadora de qualificação tendo em conta *standards* internacionais em medicina como o *International Classification of*

Diseases 9th Revision (ICD9) ou o *International Classification of Primary Care 2nd Version (ICPC2)*. O ALERT® é igualmente uma ferramenta interoperável com várias aplicações de *software*, utilizando *standards* internacionais para comunicação como HL7 e *Web services*. O mercado da saúde, onde a ALERT se insere, é um mercado bastante exigente (Serviço Nacional de Saúde, 2018), onde o registo clínico eletrónico do paciente desempenha um papel crucial na prestação de cuidados de saúde.

Uma das principais preocupações da ALERT é satisfazer da melhor forma todos os seus clientes, com o intuito de garantir qualidade e integridade de todos os produtos e conteúdos que disponibiliza, ao mesmo tempo que é realizada uma migração de Flash/Flex para HTML5. Assim, a ALERT pretende ter soluções de localização e internacionalização independentes dos produtos ALERT® que permitam que estes se adaptem a essas necessidades utilizando os recursos gerados pelas soluções encontradas. A localização cinge-se à adaptação em concreto do *software* a todas as exigências nos países/locais alvo, sejam elas de idioma, cultura, questões legais, data e hora, símbolos, etc. (Ishida & Miller, 2018). Internacionalização já se refere à forma de desenvolvimento de *software* que permite simplificar o processo de localização, tornando-o mais rápido, intuitivo e eficiente, tendo em conta as variações de cultura, região e idioma (Ishida & Miller, 2018).

1.2 Problema

Nos dias de hoje, a saúde é uma preocupação a nível mundial, sendo dada cada vez mais prioridade aos cuidados médicos prestados aos utentes com o intuito de melhorar a sua esperança de vida (Ortiz-Ospina & Roser, 2019). Aliado a isto existe um constante aprimoramento da tecnologia o que permite facilitar os processos relativos aos cuidados de saúde prestados. Assim, as empresas de informática relacionadas com a área da saúde têm vindo a evoluir também de forma a acompanhar os novos procedimentos e *standards* existentes nesta área em que operam. Tentam, ao máximo, fazer com que os mesmos se integrem da melhor forma com os produtos que disponibilizam.

Uma dessas empresas é a ALERT, que tem o propósito de disponibilizar globalmente a sua suíte de produtos. Existe, então, a necessidade de os adaptar aos diferentes requisitos locais dos países para os quais os produtos serão internacionalizados, sendo que dentro de um mesmo país os requisitos podem ser diferentes consoante as necessidades/pedidos dos clientes.

Na atualidade esta suíte de produtos já se encontra disponível em vários pontos do globo. Contudo, como a tecnologia e o mercado em que a empresa se insere está sempre a evoluir, a ALERT pretende possuir um processo concreto de internacionalização e localização de desenvolvimento de *software*, para que não necessite de estar constantemente a reagir aos problemas com que se vai deparando nos locais alvo de internacionalização. Então, a empresa pretende precaver-se de tais acontecimentos que poderão vir a surgir, melhorando/avaliando as soluções que possui para disponibilizar o *software* noutros países. Também, atualmente, a empresa encontra-se numa fase de reconversão de código Flash/Flex para HTML5, pelo que o

processo deve ter esta última tecnologia em conta e deve-se ajustar às suas características com o intuito de que seja disponibilizado aos clientes um produto devidamente adaptado aos requisitos locais.

Todo o processo de localização e internacionalização é complexo devido às variadas diferenças existentes entre os diferentes países, desde logo relativos aos cuidados de saúde disponibilizados, mas também relacionados com a cultura dos mesmos.

1.3 Hipótese

Tendo em conta o contexto e o problema identificados neste projeto, elaborou-se uma hipótese que se pretende avaliar: atendendo a que a ALERT comercializa os seus produtos a nível global, assim como as necessidades de internacionalização e localização que deste facto decorrem, é possível implementar um processo para internacionalizar e localizar os seus produtos de *software* para um determinado local do globo. Processo esse que deve ser mais simples e eficaz que as soluções implementadas pela empresa para adaptar os produtos a outras regiões do mundo, resolvendo as dificuldades e os problemas identificados.

1.4 Objetivos

Tendo em consideração o processo de localização e internacionalização da suíte de produtos ALERT®, um conjunto de objetivos foram estabelecidos, de acordo com os requisitos da empresa:

- **Identificação das necessidades da suíte de produtos ALERT®:** estudo dos produtos ALERT®, nomeadamente do produto ALERT® PFH, e das suas necessidades para tornar mais fácil, eficaz, rápido e intuitivo o processo de localização e internacionalização;
- **Estudo do estado da arte de metodologias e soluções de localização e internacionalização:** pesquisa e análise de diversas soluções de localização e internacionalização para possível integração com os produtos disponibilizados pela empresa;
- **Desenvolvimento e adaptação das soluções às necessidades dos produtos ALERT®:** tendo em consideração o estudo tanto do estado de arte relativo à localização e internacionalização, como dos produtos disponibilizados pela ALERT, desenvolver uma solução intuitiva e eficaz para os requisitos identificados;
- **Desenvolvimento e adaptação das soluções independentes dos produtos ALERT®:** forma de facilitar o processo requerido pela empresa, tendo em atenção que os processos de localização e internacionalização devem ser realizados independentemente das versões dos produtos ALERT® para que assim existam normas a seguir para a exportação dos produtos a outros países ou locais do globo;

- **Documentação das metodologias e soluções abordadas:** documentar os desenvolvimentos relacionados com as metodologias que se utiliza para a realização do projeto, bem como quais as soluções que se encontrou para os problemas do mesmo;
- **Documentação da arquitetura das soluções encontradas:** documentar a arquitetura desenvolvida para melhor perceção das soluções que foram encontradas.

Assim, todos os objetivos anteriormente descritos devem ser alcançados de forma a permitir que a suíte de produtos ALERT® seja internacionalizada e localizada de forma eficiente, concedendo uma mais valia à empresa, reduzindo o tempo e os custos que esta despende para disponibilizar os seus produtos em diferentes partes do mundo.

1.5 Motivação Pessoal

No que diz respeito à motivação pessoal, este projeto possui como âmbito de investigação um tema que desperta curiosidade ao nível de determinação de quais os requisitos que um sistema deve ter em conta para que seja realizada adequadamente a localização e internacionalização de uma plataforma *web*. Esta localização e internacionalização é um processo que deve ser realizado desde o início da concessão da plataforma, sendo que no caso de tal não acontecer, todo o processo se complica mais (Ishida & Miller, 2018). Permite a investigação de diferenças a nível de países e de locais e de como o *software* deve ser disponibilizado nesses sítios, dando um maior conhecimento sobre os mesmos e sobre as suas culturas.

Contudo, a sua investigação não se cinge apenas ao processo de internacionalização e de localização, mas como a ALERT se insere no ramo da saúde há que ter em conta todas as especificações impostas por esta área nos diversos países alvo do processo de internacionalização. Assim, esta é uma oportunidade para obtenção de conhecimento e experiência tanto na área da localização e internacionalização de uma plataforma como também abrange conhecimentos relativos aos cuidados de saúde.

Também, o facto de o projeto estar integrado na equipa de Arquitetura da empresa torna o projeto mais aliciente pois permite o contacto com diferentes profissionais, adquirindo mais conhecimento com estes.

1.6 Planeamento do Projeto

No início do desenvolvimento do projeto, foi definida uma metodologia de trabalho e um planeamento, com o intuito de se alcançar os objetivos e os requisitos definidos, bem como para melhor organizar o trabalho a ser desenvolvido.

O projeto foi dividido em duas grandes fases, tendo em conta as entregas a serem realizadas no âmbito da unidade curricular onde este projeto se insere (TMDEI). Para além disso, também foram consideradas as necessidades da empresa relativamente a tempos de desenvolvimento

da solução e conforme reuniões realizadas com o orientador da empresa. Este planeamento foi sofrendo ajustes ao longo do projeto considerando o *feedback* que se ia recebendo por parte da ALERT. Todo o planeamento foi desenvolvido segundo a metodologia Scrum (Scrum.org, 2019), com o auxílio do Git (Git, 2019) e do Jira (Atlassian, 2019).

Assim, numa primeira fase, planeou-se, organizou-se e investigou-se o projeto, tendo-se definido como aspetos mais importantes a considerar os seguintes:

- Estudar e diferenciar todos os conceitos associados ao projeto, nomeadamente a diferença entre internacionalização e localização;
- Investigar diversos problemas relacionados com a internacionalização e a localização de plataformas *web*;
- Estudar e avaliar processos de internacionalização e localização bem como de ferramentas que auxiliam o desenvolvimento deste processo;
- Analisar a suíte de produtos ALERT® e as suas necessidades ao nível de localização e internacionalização, dando maior importância ao ALERT® PFH;
- Analisar o problema proposto pela empresa com o intuito de corretamente se definir uma solução para o mesmo;
- Definir os requisitos do projeto;
- Analisar e avaliar o valor associado ao desenvolvimento do projeto;
- Elaborar, num primeiro rascunho, o desenho da solução e respetiva análise;
- Determinar uma forma de avaliar o projeto desenvolvido.

Numa segunda fase, já mais relacionada com o desenvolvimento do projeto tendo em conta a investigação realizada na primeira fase, foram definidas as seguintes tarefas mais relevantes:

- Elaborar o desenho final da solução, auxiliando-se da pesquisa realizada na primeira fase do planeamento;
- Analisar e desenvolver os requisitos definidos na primeira fase tendo em conta o desenho elaborado para a solução;
- Experimentar e avaliar os requisitos desenvolvidos;
- Analisar os resultados das experiências;
- Aferição da satisfação dos clientes com o projeto desenvolvido.

1.7 Estrutura do Documento

Excetuando o presente capítulo, o documento possui mais 6 capítulos, sendo eles os seguintes:

- **Estado de Arte:** este capítulo é iniciado pela distinção dos diferentes conceitos associados ao projeto, nomeadamente a diferença existente entre a internacionalização e a localização de *software*, bem como problemas associados a estes conceitos e ferramentas que auxiliam o processo; também se dá uma noção do que é a globalização e como esta está relacionada com o projeto, para além de se abordar o processo para se internacionalizar e localizar *software*; apresenta-se algumas aplicações existentes no mercado que facilitam a internacionalização e localização; é

ainda neste capítulo que se apresenta o âmbito onde a suíte de produtos ALERT® se insere, ou seja, o facto de os produtos estarem inseridos numa reconversão de código;

- **Contexto:** neste capítulo é aprofundado e detalhado o problema que se pretende resolver, assim como quem são as partes interessadas na elaboração da solução para o problema para além de quais são os requisitos, funcionais e não funcionais, que se deve considerar para o desenvolvimento da solução. É dado um contexto relativamente ao produto ALERT® PFH, produto alvo do processo de internacionalização e localização. Por fim é realizada uma análise de valor ao projeto;
- **Análise e Desenho:** este capítulo exhibe a análise que foi realizada tendo em conta o problema que se pretende resolver, assim como também é apresentado o desenho mais apropriado para a solução que se pretende implementar;
- **Desenvolvimento:** tendo em consideração a fase de análise e desenho da solução, neste capítulo são apresentadas as funcionalidades desenvolvidas para assegurar o sucesso do processo de internacionalização e de localização da suíte de produtos ALERT®; são também apresentadas as decisões tomadas mais arquitecturalmente relevantes para o desenvolvimento de cada uma das funcionalidades identificadas;
- **Experimentação e Avaliação:** este capítulo referencia o processo pelo qual o produto final passa para ser experimentado e avaliado consoante a perspetiva das partes interessadas no seu desenvolvimento;
- **Conclusão:** neste último capítulo é descrito o estado de desenvolvimento dos requisitos identificados, limitações que foram encontradas aquando da realização do projeto, o trabalho a ser realizar no futuro e algumas últimas considerações relativamente a todo o desenvolvimento do projeto.

2 Estado de Arte

No presente capítulo especifica-se o que é a internacionalização e localização, os seus problemas e sugestões de resolução dos mesmos. O conceito de globalização é também abordado. Ainda se referencia alguns processos para internacionalizar e localizar produtos de *software* e se apresenta algumas ferramentas que auxiliam este processo de internacionalização e localização, nomeadamente bibliotecas e *frameworks*. A par disso, são também apresentadas algumas aplicações disponíveis no mercado que facilitam o processo. Por fim refere-se o porquê de no momento os produtos ALERT® estarem a sofrer uma reconversão de código.

2.1 Internacionalização vs Localização

Com o intuito de disponibilizar um produto globalmente, existe a necessidade de o adaptar aos diferentes requisitos dos locais. Para tal, é necessário criar um processo que permita adaptar um produto a novos locais do globo, de forma gradual, com o intuito de se resolver potenciais problemas que possam surgir (Collins, 2002) (Moen, Gavlen, & Endresen, 2004).

Todo este processo de internacionalização e localização tem a sua complexidade. É, então, desde logo fundamental contextualizar e distinguir estes dois termos para uma melhor interpretação daquilo que deve ser feito durante o desenvolvimento de *software*, para além de ser importante entender quais os problemas associados ao tema em questão.

2.1.1 Globalização

A internacionalização e a localização são abrangidas por um conceito denominado globalização que deve ser explicitado para melhor perceção do enquadramento em que estes dois termos iniciais se encontram.

A globalização engloba os diversos processos de internacionalização e de localização pelos quais um produto passa para ser disponibilizado ao redor do mundo, para além de ter as suas próprias tarefas a desenvolver (Figura 1) (Bhatia, Tomar, & Sharma, 2013).



Figura 1 – Globalização

Como tarefas, a globalização é responsável por (Bhatia, Tomar, & Sharma, 2013):

- Identificar e analisar potenciais locais para a internacionalização dos produtos da empresa;
- Selecionar e validar os potenciais locais, tendo em consideração aqueles que mais benefício/lucro trarão à empresa;
- Preparação e suporte dos processos de internacionalização e localização;
- Identificação e análise dos diferentes requisitos globais aos quais o produto deve conseguir responder;
- Identificação e análise de *standards* internacionais;
- Estudo de mercados semelhantes com o intuito de ser possível reutilizar os processos desenvolvidos (p. ex. as tecnologias escolhidas ou as campanhas de publicidade para promoção do produto).

2.1.2 Internacionalização

Internacionalização, também conhecida por i18n (Ishida & Miller, 2018), é um processo de desenvolvimento que facilita a posterior localização de um produto. A internacionalização tem em atenção que o *software* não deve conter nenhum elemento que identifique um local ou região com o intuito de não ser necessária nenhuma reestruturação posterior no código. Assim, um *software* internacionalizado é independente da cultura de um local (Bhatia, Tomar, & Sharma, 2013).

Para que ocorra a i18n, existem algumas considerações a ter em conta (Bhatia, Tomar, & Sharma, 2013) (Ishida & Miller, 2018), como por exemplo:

- Os componentes gráficos da interface do utilizador têm de estar configurados para a possibilidade de expansão de texto devido às diferentes traduções;
- Aquando da codificação das interfaces do utilizador não se deve colocar conjuntos de caracteres textuais (variáveis de texto) nas mesmas, em vez disso deve-se colocar em arquivos separados e, quando necessário, aceder a esses conjuntos de caracteres;
- Não internacionalizar objetos textuais fixos, como por exemplo o nome de marcas;
- Devido à possibilidade de traduções, não devem ser utilizadas condições “if” para avaliar conteúdos de uma variável de texto;
- De forma a ser possível a existência de vários tipos de caracteres internacionais, deve-se utilizar o *standard* Unicode uma vez que este suporta um grande conjunto de *scripts* (linguagens);
- Especificar adequadamente o intuito de cada uma das variáveis existentes no código do produto desenvolvido, para que não existam dúvidas sobre a funcionalidade associada;
- Possibilitar a adaptação da orientação da leitura textual da interface do utilizador, ou seja, permitir que a leitura do texto, consoante o local, seja feita da esquerda para a direita, da direita para a esquerda ou na vertical;
- Separar elementos textuais de elementos gráficos; por exemplo, numa ilustração, o texto deve estar devidamente separado da própria imagem para no futuro ser possível utilizar a mesma imagem em locais diferentes, sendo só necessário a tradução do texto;
- Tentar, na medida do possível, evitar problemas futuros de localização ou implantação internacional do *software*;
- Adaptar o código para que facilmente se altere os requisitos dos locais alvo da internacionalização, como por exemplo o formato da data e da hora;
- Aquando da escolha do *software/hardware* para o desenvolvimento do produto, há que ter em atenção os mais utilizados pelos potenciais clientes; por exemplo, disponibilizar o produto em diversos tipos de navegadores.

Tendo em consideração estas tarefas, é ainda necessário referir que a internacionalização pode não abranger a localização, ou seja, pode perfeitamente existir internacionalização sem que ocorra a localização uma vez que a internacionalização é considerada uma boa prática no desenvolvimento de *software* (Ishida & Miller, 2018).

No desenvolvimento de um produto, há que considerar a internacionalização, se possível desde logo no seu início, para que posteriormente não seja necessário reestruturar o produto para que uma localização ocorra. Aliás, esta reestruturação pode ser sobejamente complexa caso o produto esteja demasiado dependente de características locais/regionais. Tendo em conta a internacionalização inicial do produto, poupanças a nível de tempo e de custos acontecem, o que beneficia a empresa (Ishida & Miller, 2018).

2.1.3 Localização

Localização, também conhecida por l10n (Ishida & Miller, 2018), é o processo pelo qual um produto de *software* deve passar de forma a se adaptar a características ao nível de cultura e linguagem de diferentes locais. Idealmente, o processo de localização é realizado numa aplicação que já tenha passado anteriormente por um processo de internacionalização de *software*. Assim, tendo a i18n efetuada, toda a adaptação aos requisitos dos locais alvo, como por exemplo a linguagem, é simplificada e realizada mais rapidamente (Bhatia, Tomar, & Sharma, 2013).

A l10n abrange/inclui algumas características (Ishida & Miller, 2018) (Bhatia, Tomar, & Sharma, 2013):

- Tradução dos conteúdos textuais das interfaces disponibilizadas ao utilizador, tendo em conta a linguagem utilizada pelo mesmo;
- Ajuste dos elementos gráficos (p. ex. imagens e ícones) para que estes se adequem e sejam percetivos na cultura do local alvo;
- Formatação de dados numéricos, para além das datas e horas dos locais corretamente adaptadas, conforme o que é utilizado onde a aplicação é disponibilizada;
- Alteração da moeda consoante o local;
- Tornar os símbolos, ícones e cores apropriados culturalmente de forma a não criar interpretações erróneas;
- Controlo da qualidade do conteúdo disponibilizado ao nível da localização;
- Ter atenção ao texto e gráficos disponibilizados aos utilizadores que possam conter referências a ideias, estereótipos ou ações que podem ser consideradas inapropriadas;
- Atender às diversas questões legais conforme o local;
- Cuidados relativamente a dependências de recursos, nomeadamente nas traduções; por exemplo um recurso referencia outro;
- Aquando da tradução, ter em atenção a existência de dados dinâmicos e regras gramaticais que podem colocar em causa a respetiva tradução.

Assim, verifica-se que a localização é uma tarefa fundamental para o desenvolvimento de *software* tendo em vista a disponibilização correta deste em diversos locais do globo.

2.1.4 Problemas de Internacionalização e Localização

Como processos complexos, tanto a internacionalização como a localização possuem alguns problemas que devem ser devidamente tratados. Estes, podem ir surgindo à medida que o desenvolvimento do *software* ocorre, contudo, é necessário tentar pensar previamente em potenciais problemas para o *software* estar devidamente preparado no caso de estes surgirem.

Os problemas mais relevantes e que se deve ter em consideração aquando da adaptação do *software* para o mundo são: codificação de caracteres, as fontes e estilos de texto, direção do texto, *labels*, ícones e imagens, cores, traduções, formatos de data e hora e formatos numéricos.

As soluções dadas, para cada um dos problemas referidos, devem ter em consideração os locais alvo onde será disponibilizado o *software*.

É de referir que estes problemas estão direta ou indiretamente relacionados com as interfaces visuais disponibilizadas aos utilizadores das aplicações.

2.1.4.1 Codificação de Caracteres

Consoante o local/linguagem no qual se disponibilizará a aplicação, os caracteres alfabéticos utilizados variam, sendo que a aplicação deve estar apta a estas variações. Por exemplo os caracteres alfabéticos ingleses (p. ex. “A”) são diferentes dos caracteres alfabéticos no chinês simplificado (p. ex. “该”), pelo que a aplicação deve saber lidar com estas diferenças.

Para tal, existe o *standard* Unicode. O Unicode é um conjunto universal de caracteres onde, independentemente de qual seja a linguagem, estes caracteres são representados como um identificador único (Unicode, Inc, 2017). Antes da existência do Unicode, já cada caractere era representado, contudo não existia nenhum codificador de caracteres que contivesse todos os caracteres a nível global (Unicode, Inc, 2017). Como exemplo da codificação Unicode temos o seguinte: o carácter “A” é representado em Unicode como “0041”.

Assim, utilizando Unicode é possível disponibilizar a aplicação em várias linguagens, tendo cada uma delas caracteres diferentes.

2.1.4.2 Fontes e Estilos de Texto

Aquando do desenvolvimento de um *software*, as páginas de interação com o utilizador normalmente disponibilizam conteúdos textuais. Estes conteúdos são apresentados consoante um tipo de fonte de texto pré-definido pelos desenvolvedores de *software*, sendo que no geral o utilizador não está à espera de conseguir alterar este tipo de fonte, apesar de poder acontecer em determinadas situações (Microsoft, 2016a) .

Como visto anteriormente, o Unicode permite utilizar vários tipos de caracteres, contudo, nem todas as fontes de texto conseguem lidar com os diferentes tipos de caracteres (Microsoft, 2016a). Atualmente, a maior parte dos sistemas de interpretação possuem como funcionalidade determinar se a fonte utilizada irá ou não conseguir representar todos os caracteres. Em caso de não conseguir representar algum dos caracteres esta irá realizar *fallback* (Microsoft, 2016a). *Fallback* é o processo de alterar a fonte predefinida para uma outra compatível com os caracteres que se quer representar (Microsoft, 2016a).

Para além da solução do *fallback* existe a opção de *linking* de várias fontes. *Linking* permite que se especifique uma fonte base e que depois se atribuam outras fontes chamadas de “*linked fonts*”. Assim, caracteres que não são suportados pela fonte base podem ser representados utilizando uma das “*linked fonts*” (Microsoft, 2016a).

O tamanho de cada uma das fontes também deve ser considerado e possível de ser alterado pois quando se altera a fonte utilizada normalmente o tamanho por omissão para apresentação do texto é mantido.

Para escolha da fonte deve-se considerar o seguinte (Microsoft, 2016a): não assumir que a fonte escolhida irá ser suportada no cliente nem que essa fonte interprete todos os tipos de caracteres; saber como funciona o *fallback* e o *linking* com o intuito de definir concretamente as fontes; não definir um tamanho fixo para todos os tipo de fontes e sim ser possível variar conforme a fonte.

2.1.4.3 Direção do Texto

Tal como existem diferentes tipos de linguagens ao redor do mundo, a direção de leitura e escrita de textos também varia. Nalguns casos a leitura e escrita é realizada da esquerda para a direita (p. ex. no português e no inglês) enquanto que noutros casos pode ser da direita para a esquerda (caso da hebraica e da árabe) (Figueiredo, Canhadas, & Rojas, 2004). Também acontece ler/escrever na vertical (p. ex. mongol tradicional). Portanto, aquando do desenvolvimento de *software* internacional, há que ter em conta a possibilidade de definir o sentido de escrita/leitura, ou seja o *layout* da página.

Apesar da escrita/leitura poder ser da direita para a esquerda, em alguns casos (como no árabe) os números são escritos da esquerda para a direita uma vez que a sua leitura é feita a partir do algarismo menos significativo para o mais significativo, ou seja, o número “1 345”, em árabe é lido como “cinco e quarenta e trezentos e mil” (I18nGuy, 2005) .

Para resolver este problema, tanto utilizando HTML como CSS, é possível definir o *layout* da página como se exemplifica no extrato de código HTML seguinte (Extrato de código 1):

```
<div>
  <p dir="LTR">He 123 said "שלום" (shalom] to me.</p>

  <p dir="RTL">He 123 said "שלום" (shalom] to me.</p>
</div>
```

Extrato de código 1 – Exemplo da definição da direção de texto

O resultado do código anterior é o apresentado na Figura 2:



Figura 2 – Diferentes orientações de texto numa página HTML

Para além de se especificar a orientação do texto por defeito, existe a possibilidade de se definir uma orientação automática que é reconhecida através do conteúdo textual disponibilizado ou inserido pelo utilizador.

2.1.4.4 Labels

Os elementos “<label>” são utilizados para associar um determinado texto a um campo *input* do utilizador numa página da interface com o intuito de que o utilizador ao preencher tal *input* se encontre ciente da informação que deve preencher (Wood, 2015-2018).

Portanto, este elemento *label* deve aparecer associado a um outro elemento, normalmente um *input*, contudo existe a possibilidade de aparecer relacionado com outros tipos de elementos. Existem duas formas de realizar essa associação (Extrato de código 2 e Figura 3), o atributo “*for*” do elemento *label* deve ser associado ao atributo “*id*” do elemento associado ou então dentro do elemento *label* no HTML é definido o elemento a ele associado (Wood, 2015-2018) (Mozilla and individual contributors, 2018).

```
<div>
  <label for="label-associada"> Elemento associado</label>
  <input id="label-associada">
</div>
<div>
  <label> Elemento incluído
    <input>
  </label>
</div>
```

Extrato de código 2 – Exemplo de associação de *labels*



Figura 3 – Resultado da associação de *labels*

É possível não definir *labels* associadas a elementos como o *input*, contudo utilizar *labels* é considerada boa usabilidade uma vez que o utilizador ao clicar sobre a *label* é diretamente redirecionado para o campo que deve preencher.

Estas *labels* devem ser geridas da melhor forma quando temos em mente a localização e a internacionalização de *software* uma vez que a sua posição relativamente ao campo a preencher tem de ser analisada e avaliada.

A *label*, normalmente, é colocada numa das seguintes posições, sendo a orientação do texto da esquerda para a direita (Figura 4):

1. Por cima do campo;
2. O mais à esquerda possível;
3. À esquerda, mas junto ao campo associado;
4. Dentro do campo;
5. Como *tool tip*.



Figura 4 – Posições de *labels* possíveis. Adaptado de Enders (2014)

Cada uma destas posições tem as suas vantagens e desvantagens e já estudos foram efetuados quanto à sua posição (Penzo, 2006) (Enders, 2014). Retira-se destes estudos que colocar a *label* na posição número 1 da Figura 4 é a mais adequada na maioria dos casos, ainda mais quando equacionada a internacionalização e a localização do *software*. Isto porque, com as diferenças de traduções do texto, a localização da *label* pode desformatar o *layout* da interface do utilizador.

2.1.4.5 Ícones e Imagens

A escolha de ícones e imagens para uma aplicação é mais um dos problemas que podem surgir quando se quer internacionalizar. Assim como no caso da direção da leitura e escrita, os próprios ícones podem mudar o seu significado quando se altera a direção da mesma. Por exemplo, os seguintes ícones na leitura/escrita da esquerda para a direita possuem os seguintes significados:



Figura 5 – Significado de ícones na leitura/escrita da esquerda para a direita (l18nGuy, 2005)

Enquanto que da direita para a esquerda já possuem os seguintes significados:



Figura 6 – Significado de ícones na leitura/escrita da direita para a esquerda (l18nGuy, 2005)

Para além do significado dos ícones poder variar consoante a orientação da leitura/escrita, também o seu significado se altera conforme os locais ao redor do mundo. Como tal, é necessário ter cuidado com os ícones disponibilizados pois facilmente podem ser percebidos incorretamente.

Assim, quando se escolhem os ícones ou as imagens para uma página de interface do utilizador, devemos evitar os seguintes pontos (Microsoft, 2017d):

- Evitar exemplos específicos de uma cultura – por exemplo o símbolo antigo do email era, normalmente, apenas perceptível nos Estados Unidos da América devido à sua cultura, sendo que atualmente o símbolo mudou para ser entendido mais facilmente ao redor do mundo;



Figura 7 – Diferenças dos símbolos de email (à esquerda o antigo e à direita um mais atual)
(Microsoft, 2017d)

- Evitar utilizar ícones que mostrem cor da pele ou partes do corpo – por exemplo, para indicar “paragem”, em vez de utilizarmos uma mão aberta, utilizamos diretamente o símbolo de “STOP”;

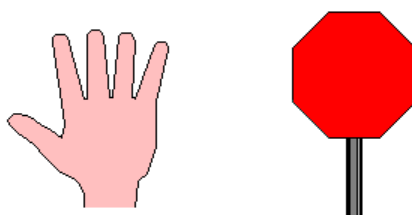


Figura 8 – Símbolos associados a paragem (Microsoft, 2017d)

- Ter cuidado com os papéis específicos e estereótipos étnicos – em certas culturas, o papel do homem e o da mulher na sociedade é diferente, sendo que utilizar ícones e imagens que demonstrem ações e comportamentos de cada um dos géneros pode causar suscetibilidades; por exemplo na Figura 9 verificamos clara diferenciação a nível de género nas pessoas representadas, sendo que tal facto pode não ser bem aceite em determinada cultura; assim, uma solução passaria por utilizar ícones e imagens genéricas, sem diferenciar géneros e estereótipos étnicos como demonstrado na Figura 10;



Figura 9 – Imagem possível de ferir suscetibilidades em certas culturas (Microsoft, 2017d)



Figura 10 – Imagem genérica (Microsoft, 2017d)

- Evitar ícones ou imagens com referências religiosas;
- Evitar ícones ou imagens com símbolos políticos;
- Evitar o uso de ícones de bandeiras para identificar as diferentes linguagens de uma aplicação – devido às variações da língua (p. ex. o português de Portugal e o português do Brasil, ou as diferentes variações da língua inglesa), representar uma certa linguagem com base num ícone pode não ser esclarecedor para o utilizador.

Existe a opção de definir para cada local um conjunto de ícones e imagens diferentes (Glezes, 2018), contudo tal não é indicado pois um mesmo utilizador que utilize a aplicação em diferentes locais do globo pode estar à espera de realizar determinada tarefa consoante o ícone a ela associado. Logo, não utilizar os mesmos ícones em locais diferentes do globo complica a usabilidade da aplicação.

Para solucionar, pode-se recorrer aos símbolos definidos pela *ISO (International Organization for Standardization)* com o intuito de que os símbolos sejam percetíveis na maioria dos locais do globo (ISO, 2018).

2.1.4.6 Cores

As cores, tal como os ícones e as imagens, podem variar o seu significado dependendo do local em que é visualizada a aplicação. Por exemplo, nos Estados Unidos da América a cor branca é a mais associada a casamentos enquanto que a preta está relacionada com funerais. O contrário

acontece no Japão, onde a cor branca é mais utilizada em funerais e a cor preta em casamentos pois é tida como uma cor que atrai sorte (Microsoft, 2017d). Logo, há que ter cuidado aquando da escolha da paleta de cores da aplicação para disponibilizar ao utilizador.

Apesar dos significados das cores variarem de local para local, é indicado que se mantenha a mesma paleta de cores para todos os locais onde se disponibiliza a aplicação, sendo que existem cores que comumente se encontram associados a ações. Por exemplo a cor vermelha está, na informática, normalmente associada a erros, a amarela a alertas e a verde a sucessos (Apache OpenOffice, 2018). Para a paleta de cores, é aconselhável que esta seja o mais sóbria possível, não utilizando cores demasiado fortes para não incomodar a visão do utilizador que utiliza o *software* regularmente no seu dia-a-dia.

2.1.4.7 Traduções

Quando se refere traduções, automaticamente se pensa em texto traduzido de uma linguagem para outra. Estas traduções podem trazer diversos problemas para as equipas de desenvolvimento de *software*, sendo que se deve lidar com o seguinte:

- Não traduzir conteúdos que não devem ser traduzidos (p. ex. nomes de marcas);
- Ter atenção aos limites de caracteres devido à expansão de texto;
- Separar imagens do texto para mais facilmente se traduzir;
- Ter atenção a texto dependente de outro;
- Problemas com *placeholders* que podem gerar incompatibilidades;
- Não incluir diretamente conteúdos textuais (varáveis de texto) da interface do utilizador diretamente no código-fonte.

Como referido, a expansão de texto é algo que merece bastante atenção quando se desenha a interface do utilizador porque, normalmente, um texto traduzido do inglês para uma outra língua aumenta consideravelmente o seu tamanho. A interface não deve, então, ser estática, mas sim deve adaptar-se facilmente ao texto que é traduzido (Wang, 2016) (Saito, 2017).

Algumas diferenças relativamente ao tamanho das traduções são representadas na imagem seguinte (Figura 11), onde se traduz a palavra “partilhar” para diferentes línguas, sendo que a língua inglesa é utilizada como termo de comparação com as restantes:

Língua	Palavra	Tamanho
Inglês	Share	1
Alemão	Freigeben	1,8
Francês	Partager	1,5
Espanhol	Compartir	1,8
Italiano	Condividi	1,7
Português	Partilhar	1,5
Japonês	共有	0,8
Coreano	공유	0,8

Figura 11 – Diferenças nos tamanhos das traduções. Adaptado de Wang (2016)

Se não se tiver em atenção estes fatores de tradução, pode-se acabar por ter uma interface de utilizador pouco amigável e com elementos de texto cortados.

Outro fator relacionado com as traduções são as variáveis de texto que se quer traduzir, o indicado será não concatenar variáveis de texto e se possível indicar o contexto no qual a variável alvo de tradução se insere na aplicação para não se correr o risco de se obter conteúdo inadequado. Em vez de utilizar-se concatenação de variáveis de texto (seja “corFavorita” uma variável, a concatenação ficaria: “A minha cor favorita é:” + corFavorita) uma boa opção será utilizar *placeholders* (“A minha cor favorita é: {0}”, sendo que {0} representa a cor favorita e pode ser colocado em qualquer lugar da frase para que, na tradução, esta faça sentido gramaticamente) (DevToolsGuy, 2017).

2.1.4.8 Formatos de Data e Hora

O formato da data e da hora varia em larga escala consoante o país e, por vezes, num mesmo país este formato pode sofrer alterações, apesar de, no geral, a data apresenta o dia, o mês e o ano enquanto que as horas mostram a hora, minutos e segundos. A formatação varia na ordem em que os campos da data/hora são apresentados, sendo que os próprios separadores entre os campos podem também variar (Microsoft, 2017a) (Microsoft, 2016c).

Podemos referir os seguintes exemplos de datas (Microsoft, 2017a):

Tabela 1 – Diferenças nos formatos de datas (Microsoft, 2017a)

	Formato extenso	Formato abreviado
Estados Unidos da América	Tuesday, October 12, 1954	10/12/54
México	Martes 12 de octubre de 1954	12/10/54
Japão	1954 年 10 月 12 日	54/10/12

Verificando a Tabela 1 constata-se que existem diferentes formatos das datas pelo que, ao nível de internacionalização de *software*, há que ter especial cuidado com este campo disponibilizado ao utilizador. Por exemplo a data 07/04/01 nos Estados Unidos da América refere-se a “4 de julho de 2001”, no México a “7 de abril de 2001” e no Japão a “1 de abril de 2007” (Microsoft, 2017a).

Relativamente ao formato da hora, existem locais que visualizam as horas utilizando o sistema de 24 horas e outros que visualizam no sistema de 12 horas. Para além disso, os caracteres utilizados para separar as horas, os minutos e os segundos também variam. Em alguns casos é utilizado o caractere “:” e noutros o caractere “h”, “m” e “s” para indicar as horas, minutos e segundos respetivamente (Microsoft, 2016c).

2.1.4.9 Formatos Numéricos

Os formatos dos números são outro fator a ter em conta quando se internacionaliza *software*. Para além de que, consoante o idioma, estes serem representados de maneira diferente, também a sua formatação varia (Figueiredo, Canhadas, & Rojas, 2004).

Na tabela seguinte (Tabela 2) são apresentadas as diferenças entre as formatações numéricas em três países (Estados Unidos da América, Alemanha e Portugal) para o número “mil e vinte e cinco unidades e sete décimas” (Microsoft, 2017b):

Tabela 2 – Diferentes formatos numéricos

	Formato	Separador dos milhares	Separador decimal
Estados Unidos da América	1,025.7	vírgula	ponto
Alemanha	1.025,7	ponto	vírgula
Portugal	1 025,7	espaço	vírgula

Como se verifica, quando se disponibilizar, na interface do utilizador, um número há que ter em atenção qual o formato numérico utilizado no local para que este não cause dúvidas de interpretação ao utilizador.

2.1.5 Processo para Internacionalizar e Localizar *Software*

Uma empresa que tenha como intuito disponibilizar o seu *software* em vários locais do mundo, deve iniciar o seu processo de internacionalização respondendo a determinadas questões.

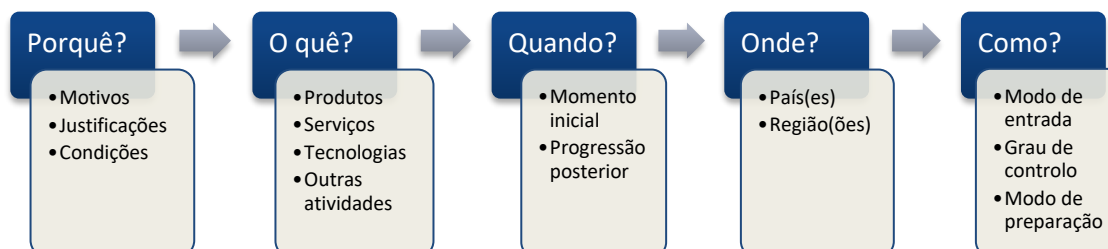


Figura 12 – Questões iniciais do processo de internacionalização. Adaptado de Carneiro & DIB (2007)

A Figura 12 apresenta questões relacionadas com a internacionalização de uma empresa, contudo pode ser facilmente adaptada para auxiliar uma empresa relativamente ao processo de internacionalização de *software*. Apesar de estarem representadas de forma linear, não é preciso seguir a ordem especificada (Carneiro & DIB, 2007).

Relativamente à questão “Porquê?”, a empresa deve ter em consideração quais são os motivos que a levam a querer internacionalizar os seus produtos, sendo que estes motivos devem ser devidamente justificados e todas as condições passíveis de surgir durante o processo devem ser equacionadas. Quanto à questão “O quê?”, se a empresa possuir mais do que um produto/serviço, deve pensar se fará sentido internacionalizar todos ou apenas alguns, tendo em mente as vantagens e desvantagens que isso acarretará para a empresa. Quando internacionalizar é algo que dependerá do mercado para o qual pretendem disponibilizar os seus serviços como também está relacionado com a situação atual do mercado onde a empresa se insere. Para além disso, sendo um produto de *software* e tendo desde o início o intuito de um dia mais tarde internacionalizar, há que ter em conta que o desenvolvimento de *software* deve estar apto a internacionalizar quando for o momento certo. Posteriormente a empresa deve equacionar relativamente a qual ou quais os locais para os quais pretende internacionalizar, sendo que deve avaliar os pontos fortes e fracos que esse local trará à empresa. Como internacionalizar o *software* já é algo mais complexo pois não existe um método que garanta o sucesso do processo e variará consoante as exigências dos locais alvo de internacionalização (Carneiro & DIB, 2007) (Dal-Sato, Alves, Bulé, & Amarante, 2015).

Tendo todas estas questões analisadas e respondidas é altura de pensar relativamente a como deve ser feita a internacionalização do *software*. É aconselhável que o processo de internacionalização e localização dos produtos seja iniciado o mais cedo possível, se exequível desde logo o início do desenvolvimento de *software* para não ser necessário reestruturações posteriores. Assim é possível reduzir custos a nível monetário e de tempo (Ishida & Miller, 2018).

Um *software* bem internacionalizado e localizado pode estar diretamente relacionado com o seu sucesso e o retorno financeiro obtido. De seguida, apresenta-se um diagrama que simula

esta evolução, sendo que o lucro vai crescendo bem como o risco associado (Figueiredo, Canhadas, & Rojas, 2004):

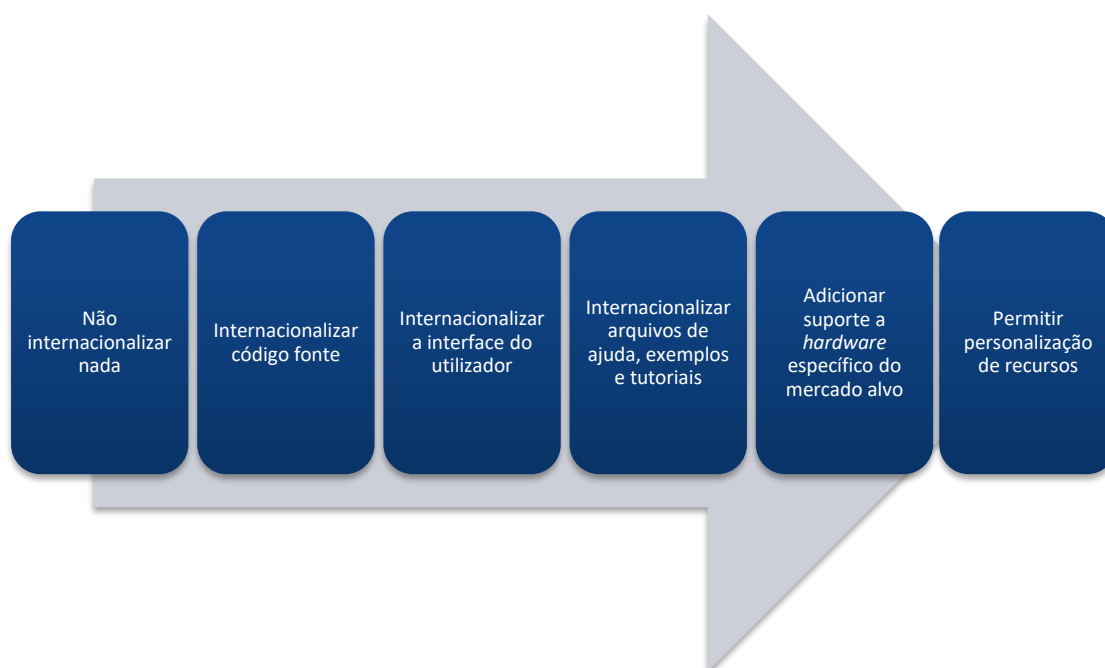


Figura 13 – Evolução do *software* internacionalizado e sucesso/retorno associado. Adaptado de Figueiredo, Canhadas & Rojas (2004)

Analisando a Figura 13, ter um *software* sem internacionalização de recursos resulta num menor lucro para a empresa mas está associado a um menor risco, enquanto que quanto mais internacionalizado está um produto, até ao ponto de ser possível personalizar recursos, maior será o lucro associado e o número de riscos que corre.

Com o intuito de reduzir os riscos, a Microsoft (2007) aconselha que o processo de internacionalização de *software* ocorra de forma faseada a par do desenvolvimento de *software* e não de uma forma “*Big Bang*”. Esta forma faseada permite, assim, reduzir o risco bem como permite que os desenvolvedores de *software* isolem e resolvam mais facilmente problemas.

Para que tal desenvolvimento faseado ocorra, um bom planeamento do processo de internacionalização e localização deve ser definido, tendo em consideração quatro fases (Microsoft, 2016b): planear, desenvolver, traduzir e iterar.

Na fase de planeamento, os desenvolvedores de *software* devem analisar as diferenças existentes nos locais para os quais vão internacionalizar o *software*, para que posteriormente estejam cientes de como devem desenvolver o código (Microsoft, 2016b). Ainda devem considerar questões legais existentes nos locais alvo, planear adequadamente a interface do utilizador e definir recursos específicos de determinado local (Figueiredo, Canhadas, & Rojas, 2004).

Na fase de desenvolvimento, tendo um bom planeamento efetuado, quem desenvolve o *software* deve desenhar o mesmo de forma a corresponder a todos os requisitos dos locais para os quais vão internacionalizar. Se possível, utilizar *standards* e ferramentas de internacionalização com o intuito de mais facilmente ocorrer o desenvolvimento. Também, devem ter cuidado quanto ao facto de ser possível traduzir os conteúdos disponibilizados aos utilizadores facilmente (Microsoft, 2016b). Nesta fase deve-se considerar todos os problemas de internacionalização e localização referidos na secção 2.1.4. Um bom desenvolvimento produz uma série de vantagens, nomeadamente a inexistência de diferentes equipas de desenvolvimento consoante o local. As atualizações de *software* serão, assim, mais fáceis pois todas as alterações são realizadas sobre um mesmo código-fonte comum a todos os locais e todos eles poderão receber as alterações simultaneamente (Figueiredo, Canhadas, & Rojas, 2004).

A fase de tradução já não se encontra relacionada com os desenvolvedores de *software*, mas sim com uma equipa especializada em tradução que traduz todos os textos que a aplicação disponibiliza ao utilizador. Apesar de esta fase não estar relacionada diretamente com os desenvolvedores de *software*, cabe-lhes a eles transmitir à equipa de tradução todo o conhecimento necessário e contexto associados aos conteúdos alvos de tradução para que se obtenha conteúdo corretamente adaptado às funcionalidades disponibilizadas (Microsoft, 2016b).

Por fim, na fase de iteração, e como os produtos não são estáticos, é necessário continuar a desenvolver o *software* e adaptá-lo a potenciais alterações que surjam, pelo que todas as fases anteriores poderão ocorrer novamente, formando-se assim um ciclo de atividades a desenvolver (Microsoft, 2016b).

Conforme a evolução do processo de desenvolvimento, não se deve colocar de lado questões relativas ao suporte fornecido pela empresa aos diferentes clientes. Assim, deve estar sempre na mente da empresa se a aplicação possui as características necessárias para suportar sempre os idiomas dos locais alvo, se os recursos de *hardware* associados ao *software* continuam a estar operacionais e se as alterações/versões que são realizadas continuam a funcionar devidamente nos clientes (Figueiredo, Canhadas, & Rojas, 2004).

De forma a facilitar a internacionalização e localização do *software*, existe um requisito que merece ser referido. Segundo a Microsoft (2018) é considerado um grave erro, para além de uma perda de tempo e espaço de disco existirem diferentes ficheiros de código-fonte para diferentes locais do globo. Assim, a solução para se utilizar um único código-fonte passa por separar todos os elementos identificadores de um local, conjuntos de caracteres de texto, ícones, imagens, etc., do código-fonte. Posteriormente é aconselhável criar uma estrutura de diretórios onde são definidos os locais e lá colocados arquivos com os elementos distintivos dos locais (Figura 14) (Microsoft, 2018):

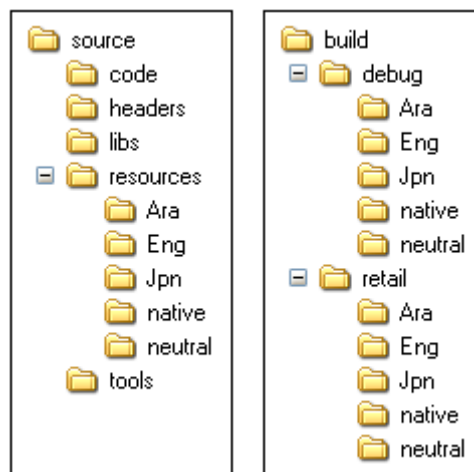


Figura 14 – Exemplo de estrutura de diretórios (Microsoft, 2018)

Assim, consoante o local, a aplicação acede aos elementos que necessita para disponibilizar corretamente os conteúdos aos utilizadores.

À medida que o processo de internacionalização e de localização de *software* vai ocorrendo, devem-se ir realizando testes de forma a verificar se todas as funcionalidades implementadas funcionam adequadamente (Microsoft, 2007).

2.1.6 Aplicações externas de apoio à internacionalização e localização

Com a exportação cada vez maior dos negócios, nomeadamente dos *softwares* desenvolvidos pelas empresas, existem no mercado aplicações que ajudam a que estes *softwares* se adaptem mais facilmente a diversos pontos do globo.

Algumas destas aplicações serão, de seguida, analisadas de uma forma geral para que se fique a conhecer qual o intuito de cada uma delas e que problemas de internacionalização e localização resolvem.

2.1.6.1 Alchemy CATALYST

O Alchemy CATALYST é descrito como uma aplicação que permite, visualmente, suportar diversos aspetos relacionados com o processo de localização de *software* (Alchemy Software Development, 2018a). Tem como principal intuito ajudar as organizações a disponibilizar o seu *software* mais rapidamente, focando-se no processo de tradução de textos que as aplicações disponibilizam (Alchemy Software Development, 2018a).

Para realização das traduções, recorre a uma tecnologia de memória de tradução (*Translation Memory*) avançada para que não seja necessário traduzir duas vezes a mesma frase, pois as

traduções feitas anteriormente são armazenadas num repositório (Alchemy Software Development, 2018a).

Como principais características, o Alchemy CATALYST possui uma interface adaptável a diferentes línguas para melhor realização das traduções. Estas traduções são realizadas tendo sempre um contexto associado para que a frase traduzida não perca o seu verdadeiro significado. Também, esta aplicação permite a conexão com outras ferramentas de tradução como o caso do Google Translate. Suporta vários *standards* relacionados com o propósito de armazenar dados relativos a localização e, assim, disponibiliza tais dados em diferentes fases do processo de localização de *software* como o XML Localization Interchange File Format (XLIFF) e o TermBase eXchange (TBX) (Alchemy Software Development, 2018a). Para além disso, esta aplicação suporta traduções de texto que estejam em diferentes formatos de ficheiros como por exemplo JSON, HTML, Microsoft Visual Studio Formats, formatos de imagens, entre muitos outros (Alchemy Software Development, 2018a).

Para além das características nomeadas anteriormente, o Alchemy CATALYST possui mais algumas que beneficiam o cliente, nomeadamente ferramentas de apoio e gestão de projetos, diminuição de custos relativos a traduções e maior qualidade das mesmas (Alchemy Software Development, 2018a). Por fim, possibilita o redimensionamento e reposição de elementos da interface gráfica através do ambiente visual que disponibiliza (Alchemy Software Development, 2018b).

2.1.6.2 SDL Passolo

A aplicação SDL Passolo disponibiliza processos inteligentes que auxiliam a localização de diversos *softwares* num ambiente visual e comum a todos os utilizadores de uma empresa, ou seja, um ambiente de localização centralizado (SDL, 2019). Disponibiliza ferramentas de gestão de projetos como o caso do Scrum (SDL, 2019).

Como principal característica, esta aplicação permite a realização de traduções à medida que se vai desenvolvendo *software*, permitindo, assim, chegar a diferentes mercados mais rapidamente. Para além disso permite também visualizar as traduções efetuadas em menus e caixas de texto com o intuito de verificar o tamanho que a tradução ocupa. Por fim, é importante realçar que a aplicação permite analisar as interfaces visuais do *software* desenvolvido com o objetivo de ser encontrado texto truncado e inconsistências em traduções criadas (SDL, 2019).

2.1.6.3 PhraseApp

PhraseApp é uma plataforma que permite a gestão de traduções (PhraseApp, 2019b). Esta plataforma disponibiliza uma API para ser integrada no *software* das empresas que pretendem internacionalizar os seus produtos para que se concretizem as traduções que são necessárias. Suporta diferentes formatos de ficheiros de texto para tradução, como XML, JSON, CSV, entre

outros. Gestão de projetos também é uma das funcionalidades disponibilizadas por esta plataforma (PhraseApp, 2019b).

Relativamente à característica principal disponibilizada pela plataforma, traduções, estas são realizadas automaticamente recorrendo a traduções feitas pela máquina (*Machine Translation* – *software* automático de traduções em diversas línguas (Globalization and Localization Association | GALA, 2019)) e traduções de memória (*Translation Memory*) (PhraseApp, 2019a). Contudo, permite também que estas aplicações sejam realizadas por uma equipa especializada (PhraseApp, 2019b).

É importante salientar que esta plataforma disponibiliza um conjunto de ferramentas que geram estatísticas para auxiliar a perceção de como estão a decorrer as traduções, nomeadamente a percentagem de traduções realizadas para uma língua (PhraseApp, 2019c).

2.1.6.4 Lingoport

Lingoport é uma suite de produtos que possuem o intuito de auxiliar o processo de desenvolvimento de *software* tendo em conta a sua disponibilização a diferentes culturas. Possibilita a gestão de projetos, assim como providencia um conjunto de métricas de forma a validar o desenvolvimento realizado (Lingoport, 2019).

Um dos produtos disponibilizados pelo Lingoport é o Globalyzer, que é um *software* que permite a descoberta de problemas relacionados com a internacionalização e localização. Como problemas que alerta, temos o caso da concatenação de variáveis de texto, métodos sensíveis a uma determinada região do globo, ficheiros estáticos, entre outros casos. Para além de alertar, auxilia a resolução destes problemas (Lingoport, 2019).

Um outro produto é o InContext que permite contextualizar as traduções a realizar, facilitando assim o trabalho a ser realizado pela equipa de traduções (Lingoport, 2019).

2.1.6.5 Lokalise

Lokalise é um sistema de gestão de traduções criado para facilitar o processo de localização de *software* (Lokalise, 2019). Facilita, também, a gestão de projetos pois permite a integração com ferramentas como o GitHub, Bitbucket, JIRA e outros (Lokalise, 2019).

A característica principal do Lokalise é a realização de traduções, sendo que consegue trabalhar com diversos formatos de ficheiros. Para além disso, é capaz de dar um contexto visual às traduções que devem ser realizadas pela equipa de traduções, nomeadamente através da partilha de capturas de ecrã (Lokalise, 2019).

De uma forma geral, Lokalise disponibiliza uma API para inserção de ficheiros de texto com conteúdo para traduzir, sendo de seguida pré-traduzidos automaticamente através de *Machine*

Translation e *Translation Memory*. Posteriormente a equipa responsável pelas traduções irá editar os ficheiros para melhor traduzir o texto (Lokalise, 2019).

2.1.6.6 Transifex

Transifex é uma plataforma baseada na *cloud*, sendo que as traduções de texto das aplicações são automaticamente sincronizadas com um repositório de conteúdos global que pode ser acedido em qualquer local e momento (Transifex, 2019a).

Transifex permite que à medida que se vai criando conteúdo textual para as aplicações, se vá traduzindo o mesmo em tempo real pois pode ser integrado no processo de desenvolvimento de *software* (Transifex, 2019a).

Em termos de funcionamento, Transifex disponibiliza duas soluções, a “*File*” e a “*Live*”. A “*File*” é mais adequada para aplicações mais complexas e para conteúdo digital, enquanto que a “*Live*” se adequa mais a páginas *web*. A grande diferença entre ambas é que a primeira trabalha com ficheiros que lhe são enviados através da API, enquanto que a segunda permite que à medida que se vai criando conteúdo textual, este seja traduzido (Transifex, 2019b).

Para além disso, a Transifex disponibiliza uma plataforma onde se pode armazenar os ficheiros traduzidos, permitindo, assim, que todas as equipas responsáveis por traduções trabalhem num único local (Transifex, 2019c).

Também disponibiliza um conjunto de métricas de forma a que se consiga verificar o estado das traduções desenvolvidas (Transifex, 2019c).

2.1.6.7 Smartling

Smartling permite automatizar, gerir e traduzir conteúdos de um *software*. É capaz de traduzir conteúdos em mais de 150 línguas diferentes, sendo adequada a páginas *web*, aplicações móveis, conteúdo publicitário, documentação de produtos e legendas de vídeos. Recorre à *Translation Memory* para realizar as diversas traduções, sendo que possibilita também que uma equipa dedicada de tradutores realize as devidas traduções (Smartling, 2019b).

Para além da realização de traduções, disponibiliza relatórios de análise aos conteúdos traduzidos, possibilitando, assim, uma maior e melhor gestão das equipas (Smartling, 2019b). Providencia dados em tempo real sobre o processo de tradução com o intuito de ser possível definir estratégias para realização das traduções, assim como é capaz de sugerir determinadas ações tendo em conta estes relatórios (Smartling, 2019c).

Possibilita, também, a conexão com repositórios de *software* para que seja acedido mais facilmente o conteúdo a traduzir, para além de que é possível indicar um contexto relativamente a esse conteúdo (Smartling, 2019c).

Para que seja integrado com diversos *softwares*, Smartling disponibiliza uma API para o efeito (Smartling, 2019a).

2.1.6.8 Conclusões

Tendo em conta a análise realizada a cada uma das aplicações externas, pode-se concluir que a maioria destas soluções se focam predominantemente no problema relacionado com as traduções do conteúdo textual do *software*, não atacando outros problemas que foram referidos na secção 2.1.4 do presente documento.

Em alguns casos, para além da tradução de conteúdo, disponibilizam também ferramentas de gestão de projetos que auxiliam as equipas no desenvolvimento de *software*, como também geram métricas e relatórios para facilmente ser possível analisar a evolução das traduções.

É apresentada, de seguida, a tabela síntese com a comparação das aplicações (Tabela 3), onde se verifica que a aplicação Lingoport é a que disponibiliza um maior conjunto de características.

Tabela 3 – Comparação de aplicações

	Traduções	Tratamento de outros problemas de i18n & l10n	Gestão de projetos	Disponibilização de métricas
Alchemy CATALYST	Efetua	Suporta	Possibilita	Não disponibiliza
SDL Passolo	Efetua	Suporta	Possibilita	Não disponibiliza
PhraseApp	Efetua	Não suporta	Não possibilita	Disponibiliza
Lingoport	Efetua	Suporta	Possibilita	Disponibiliza
Lokalise	Efetua	Não suporta	Não possibilita	Não disponibiliza
Transifex	Efetua	Não suporta	Não possibilita	Disponibiliza
Smartling	Efetua	Não suporta	Não possibilita	Disponibiliza

2.1.7 Ferramentas de Internacionalização e Localização

Apesar de existirem aplicações que facilitam o desenvolvimento da localização e internacionalização como se referiu na secção anterior (secção 2.1.6), existe também um outro conjunto de ferramentas que facilitam este processo. Estas ferramentas podem ser facilmente integradas num produto (desde que sejam compatíveis com as tecnologias utilizadas no mesmo) e permitem tratar de um maior número de problemas do que as aplicações anteriormente analisadas.

Assim, foram investigadas algumas destas ferramentas de forma a verificar a utilidade de cada uma e como poderiam ser utilizadas para ajudar no desenvolvimento do projeto. Este conjunto

de ferramentas analisadas foram escolhidas tendo em consideração que são adaptáveis às linguagens utilizadas nos produtos ALERT®, nomeadamente JavaScript e HTML, bem como serem compatíveis com a *framework* Angular.js. A análise, consoante a documentação oficial fornecida pelas ferramentas, tem em conta os seguintes fatores: suporte dado por parte da comunidade (ter em conta a atividade da comunidade em relação ao desenvolvimento da ferramenta), alcance de diversos tratamentos de problemas de i18n e l10n (existência de diferentes características disponibilizadas pela ferramenta para resolução de problemas de internacionalização e localização), adoção de *standards* internacionais (atenção dada pela ferramenta à utilização de normas de internacionalização e localização adotadas a nível global) e compatibilidade com diversos navegadores (verificar a funcionalidade da ferramenta nos diferentes navegadores mais utilizados atualmente). Toda esta investigação, principalmente a relacionada com o suporte dado pela comunidade, nomeadamente o número de programadores e data da última contribuição, foi realizada no dia 3 de dezembro de 2018, pelo que esses dados são relativos a essa data.

2.1.7.1 jQuery.i18n

A biblioteca jQuery.i18n foi desenvolvida pela comunidade Wikimedia. Esta biblioteca pretende auxiliar a internacionalização de aplicações *web* de forma fácil (Wikimedia, 2018).

jQuery.i18n disponibiliza as seguintes funcionalidades/características: o formato dos arquivos é JSON, utiliza a convecção de *placeholders* (“There are \$1 cars” – “\$1” pode variar o seu conteúdo), consegue trabalhar com diferenças gramaticais como por exemplo género e tratamento de plural recorrendo ao *standard* CLDR (Unicode Common Locale Data Repository – fornece dados relativamente a diferenças de idioma/língua existentes entre diversos locais) (Unicode, 2018), cobre um largo número de linguagens, permite definir “*Fallback chains*” (no caso de não ser possível definir uma frase numa determinada linguagem, esta aparecerá numa outra, similar à primeira, definida numa lógica realizada pelo desenvolvedor de *software* – p. ex. en_AU > en_GB > en_US, onde primeiro se tenta escrever a frase em inglês da Austrália, de seguida em inglês do Reino Unido e por fim em inglês dos Estados Unidos da América), alteração dinâmica da linguagem e orientação da interface do utilizador sem haver a necessidade de recarregar a página, possibilidade de aceder a diversos ficheiros de traduções diferentes definindo o local/linguagem em que se quer disponibilizar a aplicação, entre outras funcionalidades (Wikimedia, 2018).

jQuery.i18n é compatível com diversos navegadores, como por exemplo Google Chrome, Mozilla Firefox, Safari, Opera, Microsoft Edge e Internet Explorer nas suas versões mais recentes, tendo-se testado a demonstração fornecida na página oficial da biblioteca nestes mesmos navegadores para verificar tal compatibilidade (Wikimedia, 2018).

Para finalizar, em termos de comunidade, esta biblioteca possui cerca de 26 programadores, sendo que a 8 de novembro de 2018 foi realizada a última contribuição para a biblioteca (Wikimedia, 2018).

2.1.7.2 Polyglot

Polyglot é uma biblioteca criada pela Airbnb, com cerca de 27 programadores e última contribuição feita a 10 de agosto de 2018 (Airbnb, 2018).

Esta é uma pequena biblioteca que tem o intuito de auxiliar a internacionalização de *software*, nomeadamente o *software* desenvolvido pela Airbnb que se encontra disponível em 192 países, traduzido em cerca de 30 linguagens diferentes (Airbnb, 2018).

Como principais funcionalidades/características desta biblioteca temos: gestão de conteúdo traduzido (não realiza as próprias traduções), possibilidade de definição de ficheiros com as traduções e aceder aos mesmos consoante o local/linguagem definido pelo desenvolvedor de *software*. Estas traduções podem ser divididas em diferentes frases e serem acedidas através de “chaves” a elas associadas. Permite a utilização de *placeholders* para além de que é possível lidar com o plural (Airbnb, 2018).

2.1.7.3 Globalize

Globalize é uma biblioteca de auxílio à internacionalização e localização que tem em conta os dados provenientes do Unicode CLDR, sendo que é compatível com navegadores como o Google Chrome (versão atual e anterior), Mozilla Firefox (versão atual e anterior), Safari (a partir da versão 5.1), Opera (verão atual e a anterior), Microsoft Edge (versão atual) e Internet Explorer (a partir da versão 9) (Globalize, 2018).

Esta biblioteca tem uma comunidade com 66 programadores, sendo que a última contribuição é datada de 5 de novembro de 2018 (Globalize, 2018).

Possui como funcionalidades e características principais as seguintes: tratamento de datas e números consoante os diferentes formatos existentes, formatação de moeda, tratamento de diferenças a nível de gramática como por exemplo o plural. Trabalha com *time-zones* definidas pela IANA (*Internet Assigned Numbers Authority* – disponibiliza uma base de dados que contém códigos e dados que representam locais diversos do globo (IANA, 2018)), encontra-se dividida em diversos módulos, pelo que se pode utilizar apenas os que se necessita, e permite definir ficheiros de texto com traduções, dividindo o texto em frases que são acedidas através de “chaves” consoante o local/linguagem definida na aplicação (Globalize, 2018).

2.1.7.4 i18next

i18next é uma *framework* que tem uma comunidade bastante ativa com 105 programadores tendo 22 de novembro de 2018 como a data da última contribuição (i18next, 2018).

Em termos de compatibilidade com os principais navegadores utilizados atualmente, i18next funciona no Google Chrome, Mozilla Firefox, Microsoft Edge, Opera, Safari e Internet Explorer.

Quanto a funcionalidades e características disponibilizadas, i18next é compatível com diversas outras *frameworks*, suporta várias regras gramaticais que variam consoante a linguagem como o caso do plural, possibilita a tradução de conteúdos da interface do utilizador, sendo que permite a definição de diversos ficheiros de traduções que serão acedidos consoante o local onde a aplicação será disponibilizada. Nestes ficheiros é possível definir “chaves” que permitem aceder a determinadas partes do texto que queremos disponibilizar ao utilizador na sua interface. Admite a definição de *placeholders* e de “*Fallback chains*” de diferentes linguagens, sendo também possível a formatação de datas e possuir uma conexão flexível com o *backend* da aplicação (i18next, 2018).

2.1.7.5 Angular Translate

Esta biblioteca tem uma comunidade com 130 programadores, tendo a última contribuição sido feita no dia 16 de novembro de 2018 (Angular Translate, 2018a).

Angular Translate tem como principais características a tradução dos conteúdos das páginas dos utilizadores, consegue lidar com regras gramaticas como o caso do plural, formatar texto, é capaz de carregar os dados relativos à internacionalização de forma assíncrona, permite o uso de *placeholders* para além de que é possível definir “*Fallback chains*” de linguagens (Angular Translate, 2018c). Também é possível definir diversos ficheiros de traduções para posteriormente serem acedidos consoante a linguagem definida na aplicação (Angular Translate, 2018c).

Quanto à compatibilidade com diversos navegadores, esta biblioteca funciona no Google Chrome, Mozilla Firefox, Safari, Microsoft Edge, Opera, entre outros, sendo que é relatado que tem alguns problemas quando utilizado no Internet Explorer (Angular Translate, 2018b).

2.1.7.6 Gettext

Gettext é uma ferramenta desenvolvida pela GNU (“GNU’s Not Unix”). Tem o intuito de disponibilizar um conjunto de características que permitam traduzir os textos das interfaces do utilizador tendo em vista a internacionalização e localização das aplicações. Define um conjunto de convenções de como se deve estruturar o código das aplicações de forma a facilitar todo o processo de i18n e l10n (GNU, 2016). Através desta ferramenta é possível ainda tratar problemas relacionados com a moeda, datas e números, que variam a sua formatação consoante o local do globo (GNU, 2016).

2.1.7.7 ECMA-402

ECMA-402, “ECMAScript® Internationalization API Specification”, é uma API que está completamente operacional na maioria dos navegadores mais utilizados atualmente, tendo apenas algumas funcionalidades não compatíveis com a versão 11 do Internet Explorer (ECMAScript®, 2018a).

Esta API permite resolver alguns dos problemas relacionados com a localização e internacionalização de *software*, nomeadamente trabalhar com diferentes formatos de datas utilizados a nível global, diferentes formatos numéricos e tratamento de regras gramaticais como o caso do plural. Para além disso, permite definir diferentes *time-zones* (definidas pela base de dados IANA) e mudanças relacionadas com a moeda utilizada no local onde a aplicação é disponibilizada. Suporta múltiplas linguagens e diferenças entre diversas regiões do globo. Esta API possui uma documentação bastante extensa e específica para cada um dos métodos que disponibiliza, o que permite ultrapassar potenciais problemas que possam surgir aquando da implementação do processo de internacionalização e localização (ECMAScript®, 2018b).

2.1.7.8 i18n-2

i18n-2 tem uma comunidade de cerca de 27 programadores, sendo que a última contribuição realizada foi no dia 5 de setembro de 2018 (Resig, 2018).

Como principais características e funcionalidades, i18n-2 disponibiliza as seguintes: possibilidade de trabalhar problemas gramaticais e de texto, como as regras de plural e singular, e existe a hipótese de definir vários locais (permite que o local disponibilizado pela aplicação seja definido de diferentes formas, quer pelo desenvolvedor de *software*, quer através de uma variável de sessão ou através de uma variável de ambiente do sistema). É ainda possível definir diversos ficheiros com traduções e através do local definido como linguagem da aplicação aceder ao ficheiro adequado e respetivo conteúdo. Possibilita a definição de *placeholders* (Resig, 2018).

2.1.7.9 Format.js

Format.js é um conjunto modular de bibliotecas de auxílio à internacionalização e localização. Ao encontrar-se dividida em vários módulos, permite que o desenvolvedor de *software* opte por aqueles que mais necessita para tratar dos problemas relacionados com a internacionalização e localização de *software* (Yahoo! Inc., 2014).

Os módulos existentes permitem, de entre várias funcionalidades e características, formatar o conteúdo textual das páginas de interação com o utilizador consoante o local/linguagem definido e tendo em conta o CLDR da Unicode. Formatar as datas que poderão variar consoante os locais alvo de internacionalização de *software* como também os formatos numéricos. Além disso possibilita também a definição de *placeholders* e trabalhar com o plural e singular das

frases, entre outros problemas gramaticais. É de referir também que auxilia o armazenamento de dados em *cache*, é compatível com outras bibliotecas existentes, permite a definição de vários ficheiros de tradução diferentes e pode ser utilizado tanto do lado do cliente como do lado do servidor da aplicação (Yahoo! Inc., 2014).

Ao nível de compatibilidade com navegadores, Format.js funciona em vários, como por exemplo Google Chrome, Mozilla Firefox, Microsoft Edge, Opera, Safari e Internet Explorer (Yahoo! Inc., 2014).

Relativamente à comunidade que dá suporte e atualiza a biblioteca, tendo em conta o módulo atualizado mais recentemente, a data da sua atualização foi a 25 de abril de 2018, tendo este módulo cerca de 6 programadores (Yahoo! Inc., 2014).

2.1.7.10 iLib

A biblioteca iLib foi atualizada pela última vez a 2 de março de 2017, sendo o número de programadores desta desconhecido (iLib, 2018). É compatível com vários navegadores, até mesmo com versões antigas destes, como por exemplo Google Chrome, Internet Explorer, Microsoft Edge, Safari, Opera e Mozilla Firefox (iLib, 2018).

Esta biblioteca disponibiliza um conjunto de características e funcionalidades que possibilitam que o processo de internacionalização e localização de *software* ocorra de maneira significativamente mais fácil, nomeadamente: permite realizar traduções de texto da interface do utilizador, permite formatar datas, números e moedas, formatar números de telefone e lidar com diversos locais do globo (provenientes do Unicode CLDR). Para além de ser possível definir diversas *time-zones* (definidas pela base de dados IANA), é também capaz de suportar diversos tipos de calendários, incluindo o gregoriano, árabe e hebreu. Permite a definição de *placeholders*, converter e formatar unidades de medida e formatar endereços de locais, assim como lidar com regras gramaticais como por exemplo o plural e singular das frases (iLib, 2018).

2.1.7.11 Angular Locale

A ferramenta Angular Locale foi criada pelo AngularJS de forma a permitir que os desenvolvedores de *software* que utilizam esta tecnologia para criarem projetos possam tratar de determinados problemas relacionados com a internacionalização e localização.

Assim, o Angular Locale permite simplificar alguns destes problemas, nomeadamente o tratamento de números (p. ex. qual o caractere relativo ao separador decimal e dos milhares), percentagens, formatos de datas, horas e moedas, entre outros. Pode-se referir que também permite lidar com variações a nível gramatical como o caso de regras de plural e singular (AngularJS, 2018).

Esta ferramenta disponibiliza um conjunto de ficheiros JavaScript onde estão definidas todas as regras a utilizar em cada um dos locais do globo (tendo em conta o Unicode CLDR), sendo que facilmente se acedem aos mesmos numa aplicação desenvolvida em Angular como é o caso do projeto da empresa (AngularJS, 2018).

Estes ficheiros têm, no nome, o identificador do local do globo correspondente (p. ex. “pt_PT” para Portugal), pelo que basta que no projeto se defina qual o local que se quer aceder para usufruir das possibilidades disponibilizadas pela ferramenta (AngularJS, 2018).

Relativamente a compatibilidades com diferentes navegadores, esta ferramenta pode ser utilizada nas diversas versões do Google Chrome, Mozilla Firefox, Safari, Opera (desde a versão 15), Microsoft Edge e Internet Explorer (desde a versão 9).

Quanto à comunidade, esta ferramenta tem cerca de 58 programadores e foi atualizada, pela última vez, no dia 4 de outubro de 2018 (Angular, 2018).

2.1.7.12 Conclusões

Depois de analisadas as diversas ferramentas que auxiliam a internacionalização e a localização, apresenta-se uma tabela resumo com uma breve comparação das mesmas (Tabela 4). Em alguns casos, os valores atribuídos para comparação variam entre o desconhecido (quando não se obteve informação oficial sobre o fator a analisar) e o elevado (quando apresenta um grande número de características que dão resposta aos fatores em análise) – desconhecido < baixo < alto < elevado.

Tabela 4 – Comparação de ferramentas

	Suporte por parte da comunidade	Alcance de diversos tratamentos de i18n & l10n	Adoção de <i>standards</i> internacionais	Compatibilidade com diversos navegadores
jQuery.i18n	26 programadores; Elevada atualização	Elevado	CLDR; ISO	Elevada
Polyglot	27 programadores; Alta atualização	Alto	Desconhecido	Alta
Globalize	66 programadores; Elevada atualização	Elevado	CLDR; IANA	Elevada
i18next	105 programadores;	Elevado	Desconhecido	Alta

	Suporte por parte da comunidade	Alcance de diversos tratamentos de i18n & l10n	Adoção de <i>standards</i> internacionais	Compatibilidade com diversos navegadores
	Elevada atualização			
Angular Translate	130 programadores; Elevada atualização	Alto	Desconhecido	Alta (problemas no Internet Explorer)
Gettext	Desconhecido	Baixo	Desconhecido	Desconhecido
ECMA-402	Desconhecido	Baixo	IANA	Alta (problemas no Internet Explorer)
i18n-2	27 programadores; Alta atualização	Baixo	Desconhecido	Desconhecido
Format.js	6 programadores; Baixa atualização	Elevado	CLDR	Alta
iLib	Número de programadores desconhecido; Muito baixa atualização	Elevado	CLDR; IANA; ISO	Elevada
Angular Locale	58 programadores; Elevada atualização	Elevado	CLDR	Elevada

Tendo em conta todas as ferramentas analisadas, e respetiva comparação, é de dar destaque a seis delas devido às funcionalidades disponibilizadas que auxiliam o desenvolvimento de *software* para além de se ter tido em conta os fatores definidos para investigação sobre cada uma delas:

- jQuery.i18n;
- Globalize;
- i18next;
- Format.js;
- iLib;
- Angular Locale.

Estas seis ferramentas são alvo de uma Análise de Valor na secção 3.5 do presente documento, com o intuito de determinar aquela, ou aquelas, que podem ser utilizadas para auxiliar o desenvolvimento do processo de internacionalização e localização dos produtos ALERT®.

2.2 Enquadramento Tecnológico

Atualmente, os produtos ALERT® encontram-se em fase de reconversão de código Flash/Flex para HTML5. Assim, e como o projeto utiliza estas tecnologias, de seguida apresenta-se as mesmas.

O Adobe Flash Player é um *plugin* multiplataforma (Adobe, 2018) que permite a reprodução de conteúdos interativos como vídeos e jogos em páginas *web*, desenvolvidos em Flash e Flex (Adobe Corporate Communications, 2017).

Tanto o Flash como o Flex foram, durante vários anos, uma grande ajuda para os programadores de *software*, na medida em que permitiam que estes criassem páginas *web* dinâmicas, o que faria com que a interface do utilizador fosse mais apelativa, cativante e interativa. Aliás, utilizadores do navegador Google Chrome em 2014 chegavam, diariamente, a aceder a cerca de 80% de páginas *web* que utilizavam Flash no seu desenvolvimento (Laforge, 2017). Contudo esse valor tem reduzido drasticamente, sendo que em 2017 desceu para 17%, continuando a descer até aos dias de hoje (Laforge, 2017).

Atualmente existem alguns “*open standards*” que disponibilizam funcionalidades que o Flash possui, como o caso do HTML5, o que levou a que a Adobe, em conjunto com alguns dos seus parceiros tecnológicos (p. ex. Apple, Google, Facebook, Microsoft e Mozilla) decidisse descontinuar o Flash Player, com data prevista para fins do ano de 2020 (Adobe Corporate Communications, 2017). Um dos grandes problemas que levou a Adobe a decidir tal, foi o caso dos navegadores mais utilizados atualmente integrarem capacidades que eram anteriormente disponibilizadas pelo *plugin* em questão (Adobe Corporate Communications, 2017).

Assim, a Adobe tomou a iniciativa de alertar os seus clientes para tal fato, dando uma margem de cerca de três anos para que estes convertam os seus produtos para um dos “*open standards*” que existem atualmente no mercado. Além disso, a Adobe continuará a dar suporte ao Flash Player até essa data para além de que se encontra empenhada em contribuir ativamente para o desenvolvimento do *standard* HTML5 (Adobe Corporate Communications, 2017).

Como solução para a descontinuidade do Flash existe o HTML5. O HTML5 é a versão cinco, a mais recente e atual, do HTML (*Hypertext Markup Language*). Esta é uma linguagem definida pelo World Wide Web Consortium projetada para descrever semanticamente documentos e aplicações (World Wide Web Consortium, 2017).

Um documento HTML consiste numa árvore de elementos e de texto. Estes elementos são definidos através de “*tags*” (p. ex. “<body>”) e podem possuir determinadas características/atributos que mudam a estética da página disponibilizada ao utilizador.

Em termos de características o HTML5 é a solução mais adequada para substituir o Flash uma vez que é compatível com diversos navegadores e permite facilmente migrar conteúdos Flash para HTML5 (Pudeřek, 2017), para além de ser uma solução mais rápida, leve e segura que o Flash (Smedberg, 2017).

3 Contexto

Neste capítulo é apresentado, em pormenor, o problema associado ao projeto. Também se faz referência às partes interessadas no desenvolvimento do mesmo, bem como são enumerados os requisitos, quer funcionais, quer não funcionais. É, igualmente, descrito o produto ALERT® PFH, alvo do processo de internacionalização e localização. Por fim é apresentada uma análise de valor que é realizada ao projeto.

3.1 Problema

A empresa ALERT pretende que os seus produtos sejam disponibilizados em vários locais do mundo com o intuito de que estes permitam melhorar os cuidados de saúde em diferentes locais do globo. Atualmente, estes produtos já se encontram disponibilizados em alguns países. Contudo, o método de desenvolvimento de *software* utilizado para tal internacionalização deve ser estudado para, assim, ser melhorado. A empresa para internacionalizar adaptou-se aos problemas com que se ia deparando, tendo em consideração os requisitos locais que existiam. Em vez disso, a empresa deve analisar previamente uma solução para o processo de internacionalização e localização que lhe possibilite precaver dos problemas de *software* associados a este tema, resultando na criação de um processo mais fácil de utilizar, rápido e lucrativo.

Assim, o problema do presente projeto passa pela elaboração de um processo de desenvolvimento de *software* que facilite a internacionalização e a localização da suíte de produtos ALERT®.

Este processo deve considerar as necessidades que os produtos da empresa apresentam, para que assim seja encontrada a melhor solução, não descuidando a facilidade de integração da mesma nos produtos, bem como o facto de permitir uma manutenção adequada.

É de salientar que os produtos da ALERT se encontram em fase de reconversão de código para HTML5, o que impacta o desenvolvimento do processo de internacionalização e de localização uma vez que tem de ser considerada esta tecnologia no seu desenvolvimento.

Este processo deve ter em consideração os diversos requisitos locais que variam de país para país e por vezes até mesmo entre diferentes regiões de um mesmo país. Também existe a possibilidade de diferentes instituições de saúde, numa mesma região, necessitem/queiram modificações específicas nos produtos ALERT® que utilizam. Assim, há que ter em atenção todos os problemas relacionados com a internacionalização e localização de uma plataforma *web* bem como estes requisitos definidos em concreto pelas instituições de saúde.

O processo deve dar solução às necessidades de todas as partes interessadas no seu desenvolvimento, partes interessadas essas que vão desde a equipa de desenvolvimento de *software* da empresa até aos próprios clientes ALERT chegando aos profissionais de saúde e utentes, pois são estes que utilizam o *software* desenvolvido pela empresa. Todas estas partes interessadas são abordadas mais em concreto na secção 3.2 deste documento.

Com o processo desenvolvido, as partes interessadas têm acesso a um *software* que corresponde aos seus requisitos e, acima de tudo, a um *software* internacionalizado e localizado de forma correta e com as necessidades adaptadas aos utilizadores dos produtos.

O processo de internacionalização e localização de *software*, para além de respeitar os requisitos definidos pelas partes interessadas, deve ter em atenção os diversos problemas associados a este tema, como as traduções do texto e formatos de data e hora, já referidos na secção 2.1.4 do presente documento.

Em termos de processo, o mais indicado seria que a empresa, quando desenvolve um novo produto assuma logo desde o início que este deve, facilmente, adaptar-se a outros requisitos locais, no caso de uma internacionalização posterior do mesmo. Quando tal não acontece é necessário efetuar-se uma reengenharia na aplicação devido a esta não se encontrar adaptável a outros locais do globo que irão ter acesso aos produtos. Assim, com tal reengenharia, o processo de internacionalização e localização deve permitir que os requisitos, ao nível de funcionalidades já implementadas, se mantenham operacionais, mas desenvolvendo um produto adaptado ao local alvo de internacionalização, principalmente no que diz respeito à sua camada de apresentação.

Para além dos problemas de internacionalização e localização existentes, e anteriormente referidos, foi definido que a solução deve ter em consideração, aquando do seu desenvolvimento, os seguintes aspetos, que correspondem a algumas das principais necessidades dos produtos ALERT®:

- **Gestão de conteúdos relativos a locais especificados pelo cliente** – corresponde à forma como deve ser feita a gestão personalizada que os clientes ALERT pedem para ser desenvolvido/apresentado na aplicação e que esteja relacionado com a internacionalização e localização do *software*;

- **Gestão do mecanismo de carregamento de fontes** – as fontes disponibilizadas pela aplicação podem ser definidas consoante exigências por parte dos clientes ALERT; para isso, existe um mecanismo de carregamento de fontes, que é um ficheiro JSON, onde se define a hierarquia de fontes a utilizar para disponibilizar a aplicação nesse cliente; esta hierarquia tem de ter em atenção se determinada fonte suporta ou não um determinado conjunto de caracteres utilizados pelo cliente, pelo que a sua gestão deve ser realizada da melhor forma possível; contudo, não se pretende que o produto possua dados dos clientes no seu código fonte;
- **Gestão de diferenças existentes entre os diversos locais do globo** – alterações ao nível de diferenças existentes e variante de local para local, nomeadamente os relacionados com os problemas de internacionalização e localização existentes (como por exemplo formatos de datas); esta gestão deve ser realizada consoante o local/país onde a aplicação é disponibilizada;
- **Fazer com que o projeto das *labels* seja independente de versões** – a gestão de *labels* é a criação de ficheiros de tradução usados no âmbito da nova *framework* HTML5; assim, as *labels*, tal como o que acontece com diferentes outros conteúdos, podem ser personalizadas pelos clientes ALERT; esta personalização implica que, posteriormente, se consiga entender quais é que foram as alterações efetuadas pelos clientes para além de quais foram as novas configurações realizadas pela equipa de desenvolvimento da empresa; o objetivo é que depois de uma *label* alterada pelo cliente, quando sai uma nova versão da aplicação, este não volte a ter de efetuar a alteração realizada anteriormente e que a informação seja disponibilizada de forma adequada; para além disso, o facto de o cliente necessitar do lançamento de uma nova versão do produto para ter acesso às novas traduções por ele pedidas é um ponto a considerar como alteração a efetuar.

Como se verifica, vários são os problemas que se devem resolver. A abordagem para a resolução dos problemas encontra-se nos capítulos 4 e 5 do presente documento.

3.2 Partes Interessadas

Considerando o problema descrito anteriormente e os produtos ALERT® alvo de internacionalização e localização, são identificadas as seguintes partes interessadas no desenvolvimento do projeto:

- **Equipa de desenvolvimento** – é considerada a parte interessada principal de todo o desenvolvimento do processo de internacionalização e localização dos produtos uma vez que é aquela que lida diretamente com estes; pretende que o tempo que dispõe para preparar os produtos para o mundo seja reduzido e que facilmente se resolvam os problemas de adaptação aos diferentes locais para os quais os produtos serão disponibilizados;
- **Empresa** – detentora dos produtos alvo de internacionalização e localização; possui o interesse de melhorar a qualidade dos seus produtos com o intuito de disponibilizar a

melhor versão possível dos mesmos a um mercado global; para além disso pretende que o processo de internacionalização e localização ocorra de forma mais rápida e fácil o que implicará a redução de custos associada, bem como do tempo despendido a disponibilizar os produtos para o mundo;

- **Cientes ALERT** – são as instituições de saúde (hospitais, centros de saúde. ...) que compram os produtos para os disponibilizar aos profissionais que trabalham nesses locais. Demonstram o interesse de que os produtos que adquirem estejam corretamente adaptados à sua cultura para que, assim, consigam melhorar os serviços que disponibilizam, tendo um produto com qualidade, seguro, de fácil utilização e perceptível para todos os utilizadores;
- **Profissionais de saúde** – são os utilizadores finais dos produtos e, indiretamente, interessam-se pelos mesmos uma vez que irão utilizá-los diariamente nas suas atividades; assim, pretendem que estes estejam aptos a responder a todas as necessidades de forma fácil e intuitiva para que não surjam dúvidas na sua utilização (p. ex. o agendamento de uma consulta numa determinada data).

Como visto, estas partes interessadas são entidades que demonstram ou possuem algum interesse no negócio em desenvolvimento, sendo que são direta ou indiretamente afetadas por este.

3.3 Requisitos

Nesta secção são apresentados os requisitos para o desenvolvimento do projeto, começando por serem apresentados os requisitos funcionais em forma de *user stories* e posteriormente os requisitos não funcionais. Estes requisitos têm em consideração as partes interessadas definidas na secção anterior do documento.

3.3.1 Requisitos Funcionais

User Stories são representações de alto-nível de requisitos definidos pelos clientes, que auxiliam o desenvolvedor de *software* a entender mais rapidamente o que deve ser realizado (Ambler, 2018). As *user stories* ajudam também a perceber qual é o benefício associado a cada requisito (Ambler, 2018), ou seja, o porquê de o cliente querer o desenvolvimento de um requisito. Assim, consegue-se mais facilmente entender o intuito do requisito a desenvolver e associar-lhe uma prioridade.

Os *user stories* são definidos seguindo a seguinte formatação: “Como (cliente) pretendo (ação) para (benefício)” (Ambler, 2018). Cliente, ação e benefício variam consoante as necessidades de quem define o *user storie*. Como o cliente de todos os *user stories* é a equipa de

desenvolvimento da empresa¹, optou-se por retirar essa informação dos mesmos, para não dificultar a leitura e percepção. De seguida são apresentados todos os *user stories* definidos:

- Alterar facilmente o local em vigor nos produtos para poder disponibilizá-los segundo os requisitos desse novo local;
- Verificar que todos os caracteres que se introduz nos produtos são bem representados para que a informação introduzida seja disponibilizada adequadamente;
- Todos os textos da interface do utilizador sejam disponibilizados com os caracteres adequados para que o utilizador final utilize os produtos corretamente;
- A fonte de texto disponibilizada seja adequada ao local onde os produtos são utilizados para que o utilizador final consulte adequadamente o texto;
- A direção do texto apresentada nos produtos seja a mais indicada, consoante o local/país onde são disponibilizados, para que os utilizadores finais consigam os utilizar da forma correta;
- As *labels* apresentadas estejam corretamente posicionadas sobre o campo de *input* para que o utilizador final da aplicação considere o *layout* dos produtos o mais apelativo possível;
- Consoante o local/país onde são disponibilizados os produtos, o texto apresentado esteja corretamente traduzido para que o utilizador final o consiga perceber e assim utilizar adequadamente os produtos;
- A data e hora apresentada nos produtos esteja de acordo com o formato utilizado no local/país onde estes são disponibilizados para que o cliente final consulte estes dados consoante aquilo que espera verificar;
- Os números apresentados nos produtos sejam mostrados num formato numérico utilizado no local/país onde os produtos são disponibilizados para que não cause confusão e má interpretação na sua leitura aos utilizadores finais;
- O separador decimal e dos milhares dos dados numéricos sejam mostrados de acordo com o que é utilizado no local/país onde os produtos são disponibilizados para que não cause má interpretação aos utilizadores;
- O símbolo monetário apresentado seja o adequado ao local/país onde o produto é consultado para que o utilizador final consiga entender os dados disponibilizados;
- Ter um processo que auxilie a internacionalização e a localização dos produtos para que o tempo necessário para realizar tal atividade diminua;
- Permitir que os clientes personalizem características relacionadas com a internacionalização e localização como o caso das traduções para que estes se encontrem o mais satisfeitos possível com o produto disponibilizado;
- As personalizações exigidas pelos clientes da empresa não sejam eliminadas quando é lançada uma nova versão dos produtos para não se perder a consistência dos mesmos;
- Alterar facilmente a camada de apresentação dos produtos para que seja disponibilizada ao cliente a versão mais correta destes.

¹ Por desenvolver o produto e querer possuir soluções que facilitem a internacionalização e localização do mesmo.

3.3.2 Requisitos Não Funcionais

Os requisitos não funcionais que se deve ter em conta para o desenvolvimento da solução são apresentados de seguida, tendo sido adotado o modelo FURPS+ para tal (Eeles, 2005).

Em termos de funcionalidade, o cliente pretende que o produto esteja corretamente adaptado ao local alvo, sendo que todas as diferenças existentes, e variantes de região para região do globo, estejam conforme o local em questão. Também é requerido que o produto ao ser convertido siga um fluxo de carregamento de conteúdos o mais correto possível.

A usabilidade da interface do utilizador deve ser apelativa, intuitiva e de fácil utilização, sendo que em termos de consistência deve ser considerado que, devido às diferentes traduções e fontes de texto, a estética do produto deve estar sempre corretamente adaptada.

Em termos de fiabilidade/confiabilidade, o produto deve estar de acordo com os requisitos personalizados e pedidos pelos clientes para que o mesmo lhes seja apresentado na forma mais adequada e intuitiva de utilizar. Já relativamente ao desempenho, o produto, ao alterar o local/país onde é disponibilizado, deve alterar a camada de apresentação rapidamente e com todos os elementos corretamente apresentados. Também, não deve colocar em causa o tempo de resposta do sistema ao ser alterado o local/país em que o produto é disponibilizado.

Quanto à suportabilidade, o produto deve ser compatível com os navegadores mais comuns e mais utilizados atualmente, como o caso do Google Chrome, Mozilla Firefox, Safari, Opera, Microsoft Edge e Internet Explorer.

Por fim, relativamente às restrições de implementação, a solução desenvolvida deve considerar os requisitos personalizados definidos pelos clientes ALERT para que o produto lhes seja disponibilizado consoante as suas exigências. A camada de apresentação deve ter em conta a reconversão atual de código Flash/Flex para HTML5, pelo que o processo de internacionalização e de localização deve estar ajustado a esta última tecnologia. Devem ser adotados *standards* utilizados globalmente para aplicação no desenvolvimento do processo de internacionalização e de localização.

3.4 ALERT® PFH

O presente projeto tem como objetivo internacionalizar e localizar o produto ALERT® *PAPER FREE HOSPITAL* (ALERT® PFH). O ALERT® PFH é um dos muitos produtos que a empresa ALERT possui, como pode ser visto no Anexo A.

O ALERT® PFH é uma solução de *software* que permite informatizar os processos que ocorrem num hospital, privado ou público. Torna possível documentar, rever e integrar todas as informações e estado clínico dos pacientes desse mesmo hospital, em tempo real. Esta solução encontra-se dividida em diferentes módulos, todos eles relacionados com os diferentes ambientes de um hospital. Apesar de todas estes módulos, a camada de apresentação do

ALERT® PFH é comum a todos eles. Assim, ao ser considerado como uma única solução, a reconversão de código para HTML5 ocorre em todo o produto.

Os diferentes módulos do ALERT® PFH são: ALERT® *TRIAGE* (*software* de triagem para o serviço de urgência), ALERT® *EMERGENCY DEPARTMENT INFORMATION SYSTEM* (ALERT® EDIS – solução completa para os serviços de urgência), ALERT® *OBSERVATION ROOM* (ALERT® OBS – para registar informação relativa a um episódio de internamento ou de observação num serviço de urgência), ALERT® *PATIENT DATA MANAGEMENT SYSTEM* (ALERT® PDMS – permite recolher dados de dispositivos de monitorização clínica e associa-los a um paciente), ALERT® *WAITING ROOM* (aplicação para gestão das filas de espera de um hospital como por exemplo as do serviço de urgências), ALERT® *OUTPATIENT* (*software* clínico e de gestão de unidades de consulta externa), ALERT® *INPATIENT* (*software* clínico e de gestão de unidades de internamento), ALERT® *OPERATING ROOM INFORMATION SYSTEM* (ALERT® ORIS – solução utilizada em blocos operatórios), ALERT® *PHARMACY* (permite a prescrição, administração, validação, aviamento e gestão de fármacos) e ALERT® *GP PORTAL* (solução que permite a comunicação entre médicos de clínica geral e outros profissionais de saúde de diversas instituições) (ALERT® Life Sciences Computing, 2017).

O ALERT® PFH, atualmente, encontra-se disponibilizado em mercados como o Brasil, Chile, França, Itália, Kuwait, México, Portugal, Suíça, Reino Unido e Estados Unidos da América (ALERT® Life Sciences Computing, 2017).

Relativamente à análise arquitetural deste produto, a mesma é apresentada na secção 4.2 do presente documento.

3.5 Análise de Valor

No presente capítulo é abordada a Análise de Valor efetuada ao produto, tendo sido adotada a abordagem de *Value Engineering* (VE). *Value Engineering* tem em consideração as fases iniciais de um projeto, desde a sua ideia até à sua implementação, sendo que aplica a estas fases os mesmos princípios e técnicas que a *Value Analysis* (VA) (Rich & Holweg, 2000).

A VA tem como principal objetivo a determinação da forma de como será possível aumentar o valor de um produto/serviço disponibilizado a um cliente, tendo em mente a manutenção ou aumento da qualidade e a redução de custos, se possível. Por vezes, VA, é vista como uma forma de redução de custos, contudo tal não é representativo do fundamento da VA. Os custos podem por vezes aumentar, mas este aumento tem de acontecer também, e em quantidade superior, no valor que a aplicação do custo trará para o produto/serviço, logo, o valor do produto/serviço terá de ser em larga escala maior que o custo associado (Rich & Holweg, 2000).

Assim, a VA utiliza um determinado conjunto de técnicas, conhecimentos e aprendizagens para que o valor do produto/serviço seja aumentado através da sua análise. A análise tenta descobrir que custos são desnecessários, ou seja, quais os custos existentes no momento e que não

produzem nem qualidade, nem uso, nem requisitos para o funcionamento normal do produto/serviço. Também, analisa se o produto/serviço vai ao encontro das necessidades do cliente/utilizador, nunca descurando o fim último para o qual o produto/serviço foi desenvolvido. O objetivo pelo qual o produto/serviço foi desenvolvido também implica que as novas especificações que poderão vir a ser estabelecidas não fujam deste âmbito (Rich & Holweg, 2000).

O valor de um produto/serviço é aumentado tendo em consideração os seguintes elementos:

- A utilidade do produto (*Utility Value*) – o quão funcional o produto/serviço tem o intuito de ser (Rich & Holweg, 2000);
- A posse (*Esteem Value*) – este elemento representa o valor que o cliente/utilizador atribui ao produto/serviço disponibilizado, não diretamente contribuindo para a utilidade do mesmo, mas mais relacionado com o seu valor estético e subjetivo (Rich & Holweg, 2000);
- O valor de mercado (*Market Value*) – está relacionado com o preço pelo qual o mercado está disposto a pagar para poder utilizar o produto/serviço. Consiste no somatório entre o valor da utilidade e o valor da posse (*Utility Value + Esteem Value*) (Rich & Holweg, 2000).

Esta *Value Analysis* é um processo sistemático, formal e organizado de análise e avaliação, pelo qual o seguinte gráfico é idealizado:

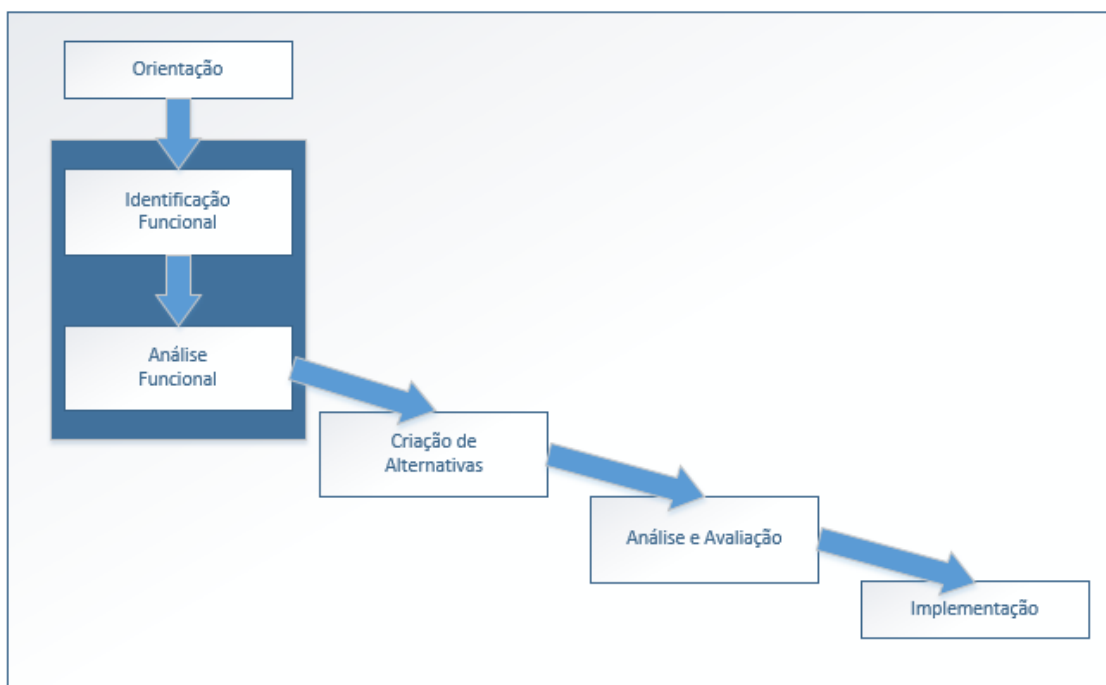


Figura 15 – Processo de *Value Analysis*. Adaptado de Rich & Holweg (2000)

Como se pode verificar na Figura 15, a análise é constituída por seis diferentes fases: Fase da Orientação, Fase da Identificação Funcional, Fase da Análise Funcional, Fase da Criação de

Alternativas, Fase de Análise e Avaliação e a Fase de Implementação. Apesar das seis fases, a Identificação Funcional e a Análise Funcional serão realizadas em conjunto.

De uma maneira geral, considerando o problema do projeto, as fases podem ser descritas da seguinte forma:

- **Orientação** – consiste na seleção, preparação e apresentação do produto que será adotado durante o processo da análise de valor (Rich & Holweg, 2000); aplica-se o modelo *Fuzzy Front End* e a tabela de Benefícios e Sacrifícios;
- **Identificação e Análise Funcional** – início da análise do produto, onde se identifica e avalia as suas principais funções tendo em consideração os clientes (Rich & Holweg, 2000); aplica-se o modelo *Quality Function Deployment* (QFD) e o *Pairwise Comparison*;
- **Criação de Alternativas** – esta fase permite a identificação de soluções com o intuito de ser possível atingir as funções do produto (Rich & Holweg, 2000);
- **Análise e Avaliação** – é avaliado tanto o custo quanto o benefício de cada uma das soluções encontradas na fase anterior com o intuito de identificar aquela que se adequa melhor ao projeto (Rich & Holweg, 2000); aplica-se o método *Analytic Hierarchy Process* (AHP);
- **Implementação** – descreve, sucintamente, a solução que será utilizada para a implementação do produto tendo como base as conclusões retiradas deste processo (Rich & Holweg, 2000).

3.5.1 Orientação

Nesta primeira fase foi adotado o *New Concept Development* (NCD) para a apresentação do ambiente no qual o produto se insere. O NCD é uma representação em linguagem comum e definição dos componentes chave do modelo *Fuzzy Front End* (FFE), que se encontra no processo de inovação de um produto (Koen, et al., 2002).

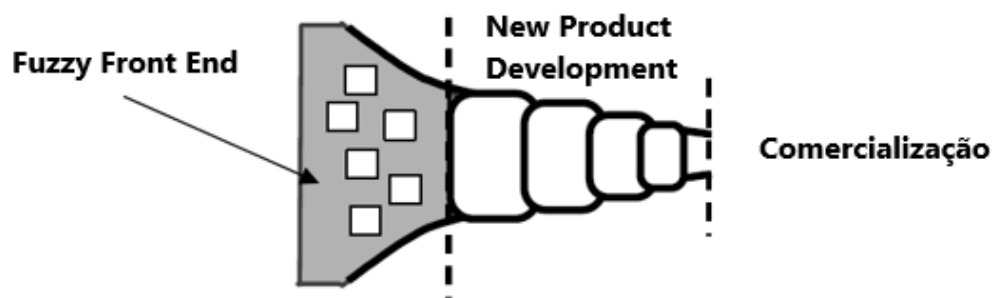


Figura 16 – Processo de inovação de um produto. Adaptado de Koen, et al. (2002)

Este processo de inovação é composto por três fases: (1) Fuzzy Front End composto por atividades desorganizadas, imprevisíveis e não estruturadas, (2) *New Product Development* (NPD) cujas atividades se encontram bem estruturadas e por fim (3) Comercialização do produto (Koen, et al., 2002).

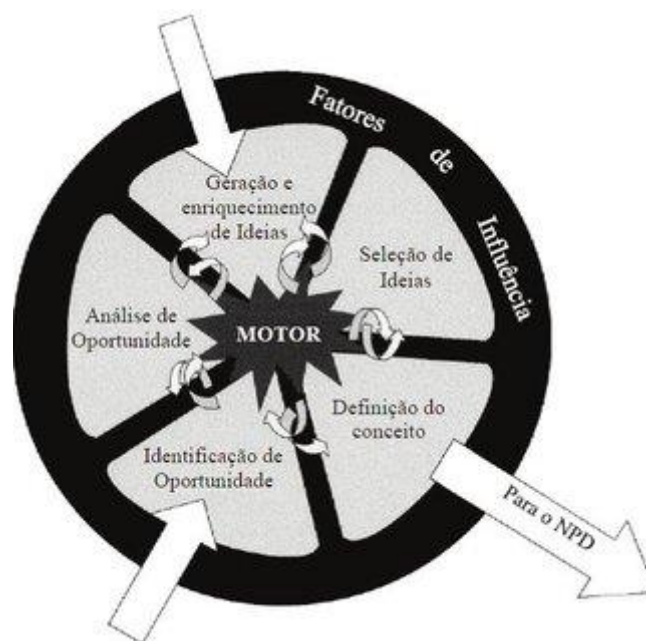


Figura 17 – *New Concept Development*. Adaptado de Koen, et al. (2001)

O modelo NCD, como referido anteriormente, adapta de forma perceptível as atividades que devem ser desenvolvidas no FFE. Encontra-se dividido em três partes: o motor, os elementos (identificação da oportunidade, Análise da Oportunidade, Ideias e por fim a Definição do Conceito) e os fatores de influência. Os elementos não seguem nenhuma ordem em específico, pelo que podem ser realizados no momento mais apropriado.

3.5.1.1 Fatores de influência

Estes fatores de influência dizem respeito ao ambiente que envolve a empresa que disponibiliza o produto, ou seja, consiste nas estratégias de negócio adotadas, fatores competitivos, as capacidades da organização e o nível de maturidade das tecnologias utilizadas (Koen, et al., 2001).

Tendo em conta o projeto alvo desta análise, os fatores de influência detetados são:

- as capacidades organizacionais da ALERT;
- as tecnologias existentes no mercado (variantes de país para país) e que têm algum tipo de impacto no desenvolvimento de produtos tecnológicos ligados ao ramo da saúde;
- os clientes e competição direta em cada um dos países onde a ALERT atua;
- influências externas que afetam a área da saúde.

A empresa, ALERT, atua no ramo da tecnologia da saúde, ramo esse que faz com que seja imperioso o acompanhamento constante de descobertas, problemas, oportunidades e inovações. Assim, a empresa pode-se ir atualizando à medida que estes fatores se atualizem, proporcionando um crescimento da mesma. Também, com este acompanhamento, os próprios clientes da empresa saem beneficiados pois os produtos disponibilizados

encontram-se atualizados, o que providencia uma melhor qualidade de saúde. Toda esta informação é analisada e avaliada para que, no caso de necessidade, os produtos evoluam.

A tecnologia evolui muito rapidamente, sendo que nos últimos anos esta evolução tem sido cada vez maior. Também, o número de dados relacionados com a saúde aumenta rapidamente, o que necessita de uma cuidada gestão, e cujos custos são elevados. Assim, o uso da tecnologia é um facto a ter em conta pois esta permite proporcionar uma melhor gestão de dados o que implica redução de custos. Portanto, a tecnologia existente no mercado deve ser avaliada para que a empresa adote as mais adequadas para o desenvolvimento dos seus produtos (Pinochet, 2011). Estas tecnologias podem incitar mudanças a nível de desenho, arquitetura e funcionalidades dos produtos. Por exemplo, o projeto alvo desta análise insere-se num ambiente de reconversão de código de Flash/Flex para HTML5. Tal ocorre devido à empresa detentora do produto Adobe Flash Player, a Adobe, ter decidido descontinuar o mesmo com data prevista para fins do ano de 2020 como referido na secção 2.2 (Adobe Corporate Communications, 2017). Esta reconversão permitirá que os produtos acompanhem mais eficazmente tanto as inovações tecnológicas, como a possibilidade de evolução e criação de requisitos disponibilizados aos clientes, uma vez que o HTML5 é uma solução mais rápida, leve e segura que o Flash (Smedberg, 2017).

Outro dos fatores a ter em conta são os clientes e a competição existente. O cliente é a peça fundamental do negócio, pelo que a sua satisfação é tida como de extrema importância. É ele um dos fatores que impulsiona o desenvolvimento de novas funcionalidades para os produtos, pois o seu interesse pelo mesmo é sinónimo de crescimento da empresa. No momento, a empresa possui uma larga escala de clientes espalhados pelo mundo, como hospitais, clínicas e outros serviços de saúde. Existe uma grande quantidade de concorrência, ainda para mais sendo os produtos da empresa internacionalizados, o que implica que estes produtos necessitem de se destacar da melhor forma possível.

Por fim, existem influências externas que afetam a área da saúde, pelo que modificações na legislação ou alterações em *standards* adotados para o desenvolvimento dos produtos impliquem no desenvolvimento e manutenção destes.

3.5.1.2 Motor

O Motor presente neste modelo NCD, diz respeito à liderança, à cultura da empresa e a estratégias de negócio. A diferença entre uma empresa que se inova mais eficazmente e outra cujo processo de inovação não é tão eficaz deve-se à correta utilização dos três aspetos referidos anteriormente (Koen, et al., 2002). Uma liderança, cultura e estratégias de negócio apropriadas asseguram um conjunto de novos produtos e processos que geram valor à empresa.

A missão da ALERT é “Melhorar a saúde e prolongar a vida, alcançar rentabilidade para benefício da sociedade e inspirar outros para a excelência através do nosso exemplo” (ALERT® Life Sciences Computing, 1999). Com esta missão, denota-se que a cultura existente na empresa se encontra bem definida, o que indica que a inovação e o desenvolvimento de produtos, que

possibilitem que os profissionais de saúde possam cuidar de uma forma mais eficaz os seus pacientes, são fatores importantes para a ALERT. Estes produtos devem ser desenvolvidos com qualidade, seguindo boas práticas de desenvolvimento e normas internacionais, não pondo de parte a segurança e a escalabilidade destes. Assim, a empresa encontra-se organizada em diferentes equipas, orientadas a tarefas específicas, e que interagem umas com as outras para atingirem com sucesso o fim a que se propõem.

3.5.1.3 Identificação da Oportunidade

O elemento da Identificação da Oportunidade permite, tal como o nome indica, a identificação de oportunidades que a empresa tenciona perseguir/adotar (Koen, et al., 2001).

A empresa, cuja missão é permitir melhorar a saúde e prolongar a vida, pretende que os seus produtos sejam disponibilizados ao redor do globo para que assim mais pessoas possam lucrar/melhorar a sua saúde (ALERT® Life Sciences Computing, 1999). No momento, estes produtos já se encontram em vários locais do mundo a funcionar. Contudo, a empresa deve avaliar/melhorar o seu modelo e diretrizes de desenvolvimento de *software* para internacionalizar os seus produtos, para que, consoante o local alvo de internacionalização, o produto não sofra alterações, por vezes profundas, no seu código fonte. Um dos aspetos que mais sofre alterações são as páginas de interação com o utilizador devido às grandes diferenças de cultura, língua, numeração, etc., existentes de países para países e por vezes até de locais para locais dentro de um país. Este processo de adaptação ao local alvo não deve retirar muito tempo e consequentemente lucro à empresa, pelo que a necessidade de adotar um modelo que agilize o processo surge como uma oportunidade a ter em conta, e como um grande problema a resolver.

3.5.1.4 Análise da Oportunidade

Tendo em conta a oportunidade identificada anteriormente, nesta fase procede-se à sua análise para se verificar que realmente será uma oportunidade que vale a pena ter em conta para o cliente (Koen, et al., 2002).

Nos dias atuais, os cuidados de saúde tem sofrido substanciais melhorias que podem ser constatadas com o aumento da esperança média de vida da população mundial (Lopes, 2018). Este valor ronda os 70 anos, sendo que em alguns países mais desenvolvidos já ultrapassa os 80 anos de esperança de vida à nascença como o caso de Espanha (83,5), Itália (83,4) e França (82,7) (PORDATA, 2019) (Ortiz-Ospina & Roser, 2019). Para verificarmos como este valor tem vindo a aumentar temos o caso de Portugal que em 1960 tinha uma esperança de vida de 64,0 anos, enquanto que, segundo dados de 2016, este valor se situava nos 81,3 anos (PORDATA, 2019) (Ortiz-Ospina & Roser, 2019). O aumento destes valores está relacionado com o acesso que a população tem aos cuidados de saúde e como estes investem na própria saúde, para além das constantes descobertas e inovações na área da medicina. Apesar desta melhoria, ainda

grande parte da população mundial não tem acesso a estes cuidados de saúde (DN/Lusa, 2017), sendo este um facto a ter em conta.

Existindo vários *softwares*, desenvolvidos por diversas empresas, que permitem auxiliar os cuidados de saúde. Importa referir que estas, ao disponibilizarem os seus produtos ao redor do mundo, poderiam fazer com que a saúde chegasse a locais que ainda não possuem infraestruturas e equipamentos que propiciem e melhorem a saúde da população. Também o facto de que a própria internacionalização de *software* leva a que uma empresa possa crescer e expandir-se, encontrando um maior número de clientes e, por conseguinte, aumente os seus lucros (Rocha, Andrade, & Leitão, 2012).

Tendo em conta os factos definidos anteriormente e de forma a avaliar a oportunidade, é necessário colocar-se na “pele” do cliente com o intuito de identificar os seus benefícios e sacrifícios no desenvolvimento da oportunidade. No caso do projeto, a entidade cliente mais interessada é a equipa de desenvolvimento de *software* da empresa, pois é ela que tem o interesse de encontrar soluções para a localização e internacionalização dos seus produtos desenvolvidos, para que o tempo utilizado para a internacionalização reduza e que seja mais fácil internacionalizar (Araújo, 2016). Também, o facto de o processo de internacionalização e localização gerar muitos problemas devido aos diferentes requisitos existentes nos locais alvo do globo onde se pretende disponibilizar o *software*, leva a que a equipa de desenvolvimento necessite de um processo que os auxilie a realizar tal (Microsoft, 2017c). Esta análise, tendo em conta a perspetiva do cliente, é muito importante para analisar o valor da oportunidade uma vez que quem desenvolve o projeto geralmente tem uma perceção diferente do valor associado do que o próprio cliente (Nicola, 2018b). Os benefícios e os sacrifícios identificados baseiam-se na Tabela 5 realizada por Woodall (2003) na qual se encontram os valores percebidos pelos clientes (*perceived value*) (Nicola, 2018b) consoante aquilo que lhe é oferecido e os custos relacionados.

Tabela 5 – Benefícios e Sacrifícios. Adaptado de Woodall (2003)

BENEFÍCIOS		SACRIFÍCIOS
Atributos	Resultados	
Qualidade procurada	Benefícios funcionais	Preço
Qualidade do produto	Utilidade	Preço de mercado
Qualidade	Função de utilização	Custos monetários
Qualidade do serviço	Função estética	Finanças
Qualidade técnica	Benefícios operacionais	Custos
Qualidade funcional	Economia	Custos de utilização
Qualidade de desempenho	Benefícios logísticos	Custos percebidos
Desempenho do serviço	Benefícios do produto	Custos de pesquisa
Serviço	Benefícios estratégicos	Custos de aquisição
Suporte do serviço	Benefícios financeiros	Custos de oportunidade

BENEFÍCIOS		SACRIFÍCIOS
Atributos	Resultados	
Aspetos especiais do serviço	Resultado para o cliente	Custos de distribuição e instalação
Serviços adicionais	Benefícios sociais	Custos de reparação
Solução principal	Segurança	Custos de formação e de manutenção
Personalização	Conveniência	Custos não monetários
Confiança	Satisfação	Custos não financeiros
Características do produto	Apreciação por parte dos utilizadores	Custos de relação
Atributos do produto	Conhecimento, humor	Custos psicológicos
Características	Expressão própria	Tempo
Desempenho	Benefícios pessoais	Recursos humanos
	Associação com grupos sociais	Esforço
	Entusiasmo afetivo	

Tendo a tabela anterior em conta, foi então elaborada uma outra em específico para a análise da oportunidade, a qual se encontra dividida entre benefícios e sacrifícios para três áreas. As áreas são o produto, o serviço que será disponibilizado e a relação com o cliente.

Tabela 6 – *Perceived Value*

	Produto	Serviço	Relação
Benefícios	Melhoria da qualidade dos produtos ALERT®; Capacidade de personalizar os produtos ALERT®; Melhoria de desempenho;	Flexibilidade de resposta aos pedidos dos clientes e respetiva adaptação; Serviço prestado de confiança;	Redução do tempo e custos associados ao processo de internacionalização; Aumento da satisfação do cliente; Fortificação da imagem da empresa;
Sacrifícios	Custos de tempo e mão-de-obra associados à investigação e implementação;	Custos de tempo e mão-de-obra;	Formação da equipa de desenvolvimento;

Verificando a Tabela 6, no que ao Produto diz respeito observa-se que este possui algumas mais valias para o cliente uma vez que permite que o processo de internacionalização dos seus produtos melhore ao nível da qualidade, da capacidade de personalização e respetivo desempenho. Apesar disso, estas melhorias necessitam que a empresa invista quer a nível de tempo quer de mão-de-obra associada ao desenvolvimento do processo.

Relativamente ao valor associado ao Serviço prestado pelo produto, este beneficia o cliente através da sua flexibilidade de resposta e do seu serviço de confiança uma vez que a investigação associada ao produto tem como base a deteção da melhor forma de internacionalizar um *software*. Contudo, também existe custo de tempo e de mão-de-obra relacionados com a aprendizagem para posterior utilização do serviço prestado pelo produto.

Por fim, a Relação entre cliente e produto sai beneficiada no sentido em que existe uma redução de tempo associada ao processo de internacionalização da suíte de produtos ALERT®. Esta redução implica um aumento da satisfação dos clientes para além de que a imagem que a empresa transmite para o exterior se fortifica. Porém, a formação do cliente para uma correta utilização do processo de internacionalização é um sacrifício a ter em conta.

Concluindo, para a equipa de desenvolvimento de *software* da empresa, a oportunidade relacionada com o desenvolvimento de um processo de internacionalização dos produtos do cliente acaba por ser uma mais valia tendo em conta todos os benefícios associados em comparação com os sacrifícios. De forma geral, o produto permitirá reduzir tempo e aumentar a qualidade da internacionalização, o que é em larga escala superior quando comparado com os custos de mão-de-obra e de tempo associados às diferenças de requisitos impostos a nível de internacionalização dos produtos ALERT®.

3.5.1.5 Ideia

Tendo em conta todas as fases anteriores do modelo *New Concept Development*, a ideia associada a este projeto passa pela internacionalização eficaz dos produtos ALERT®, de forma a ser possível que um maior número de locais ou países consigam beneficiar das soluções ao nível dos cuidados de saúde criadas pela ALERT. É de referir que as evoluções das tecnologias e dos *standards* ao nível da saúde têm de ser acompanhadas para, assim, se disponibilizar ao mundo a melhor solução possível e, posteriormente, melhorar a saúde e prolongar a vida aos clientes da empresa.

3.5.1.6 Conceito

Como conclusão do NCD, o conceito pode ser definido como a implementação de um processo intuitivo e eficiente de internacionalização e localização da suíte de produtos ALERT®, para que, aquando de uma internacionalização, os produtos sejam facilmente adaptados ao local alvo, reduzindo os custos de desenvolvimento associados aos mesmos.

3.5.2 Identificação e Análise Funcional

A fase que se segue na *Value Analysis* é a fase de Identificação e Análise funcional, onde será realizada a análise do produto tendo em consideração os requisitos e funções principais que

este disponibiliza ao cliente. Uma função pode ser definida como a utilidade e o valor de estima (*esteem value*) que esta transmite ao cliente (Rich & Holweg, 2000).

Esta fase encontra-se dividida em duas partes, a Identificação Funcional e a Análise Funcional propriamente dita. Na primeira, são identificados e descritos os requisitos e funções principais do produto, enquanto que na segunda é realizada uma análise aos requisitos e uma comparação entre as diversas funções com o intuito de as avaliar.

3.5.2.1 Identificação Funcional

Na Identificação Funcional pretende-se dar resposta à questão “Quais as funções que este produto deve dar resposta?” (Rich & Holweg, 2000). Para tal é necessário ter em conta os requisitos que o cliente pretende ver respondidos através do desenvolvimento do produto. Assim, é importante que o cliente consiga transmitir os seus ideais de forma a que o levantamento de requisitos não seja ambíguo e no fim se consiga determinar em concreto as funções que o produto deve, ou não, disponibilizar. Por vezes, determinar qual ou quais são as funções mais importantes para o produto não é imediato, pelo que esta fase deve/pode ser morosa com o intuito de chegar à função pretendida (Rich & Holweg, 2000).

Para se determinar os requisitos do sistema, recorre-se à utilização do *Quality Function Deployment* (QFD) (Warwick Manufacturing Group, 2007). O QFD é um sistema que permite projetar um produto ou serviço tendo como base os requisitos que os clientes pretendem. O QFD permite que se determine os requisitos e funções de maior qualidade, ao menor custo, disponibilizadas pelo produto. Com o QFD é mais perceptível aquilo que o cliente pretende ver na solução desenvolvida, sendo que o intuito é que este traga valor, reputação e aumento do negócio para o cliente (Warwick Manufacturing Group, 2007).

O QFD consegue atingir estes resultados através de uma profunda análise aos requisitos propostos pelo cliente, sendo que cada requisito é segmentado e são identificadas formas de atingir cada segmento.

Para dar início ao QFD elaborar-se-á uma *Customer Chain* (Warwick Manufacturing Group, 2007) onde são identificados em concreto a cadeia de clientes do produto, mesmo que estes clientes estejam indiretamente ligados ao produto desenvolvido.

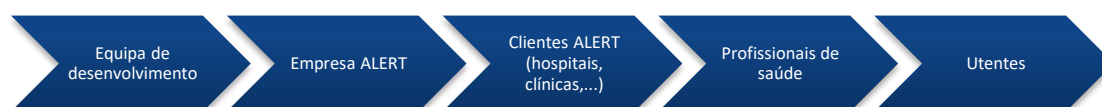


Figura 18 – Customer Chain

Assim, avaliando a Customer Chain da Figura 18 constata-se que o cliente principal do produto é a equipa de desenvolvimento da empresa, sendo que indiretamente existem outros clientes como por exemplo a própria ALERT. Para além disso, os próprios clientes da empresa são

clientes do produto em desenvolvimento, não esquecendo que os profissionais de saúde e os próprios utentes também são clientes finais do produto alvo de internacionalização e localização.

Agora, com os nossos clientes devidamente identificados, é necessário reconhecer quais são os requisitos de qualidade mais importantes impostos por eles. Tendo em conta as características gerais dos requisitos definidos na secção 3.3 deste documento, foram criados os seguintes requisitos de qualidade:

- Facilidade do processo;
- Capacidade de personalização;
- Confiança no processo;
- Redução de tempo;
- Abrangência do processo.

Cada um destes requisitos de qualidade foram avaliados pela equipa de desenvolvimento em relação ao grau de importância que se deveria atribuir a cada um deles. Para tal foram realizados questionários, onde cada um dos membros da equipa de desenvolvimento atribuía uma pontuação a cada um dos critérios, sendo que o somatório da pontuação dada por cada elemento teria de dar 100. Assim, depois de todas as respostas recolhidas, calculou-se a média atribuída a cada critério, tendo-se obtido os seguintes resultados aproximados:

- Facilidade do processo – 15;
- Capacidade de personalização – 30;
- Confiança no processo – 20;
- Redução de tempo – 10;
- Abrangência do processo – 25.

Para além dos requisitos solicitados pelo cliente, também é necessário identificar quais são os requisitos a nível funcional para melhor se avaliar e atingir a qualidade do produto:

- Tratamento de problemas de internacionalização;
- Tratamento de problemas de localização;
- Adoção de normas internacionais;
- Existência de um processo de localização e internacionalização pré-definido.

Tendo todos os requisitos principais determinados, é possível identificar quais as funções mais importantes do produto. Para tal, desenvolveu-se o seguinte diagrama de árvore:

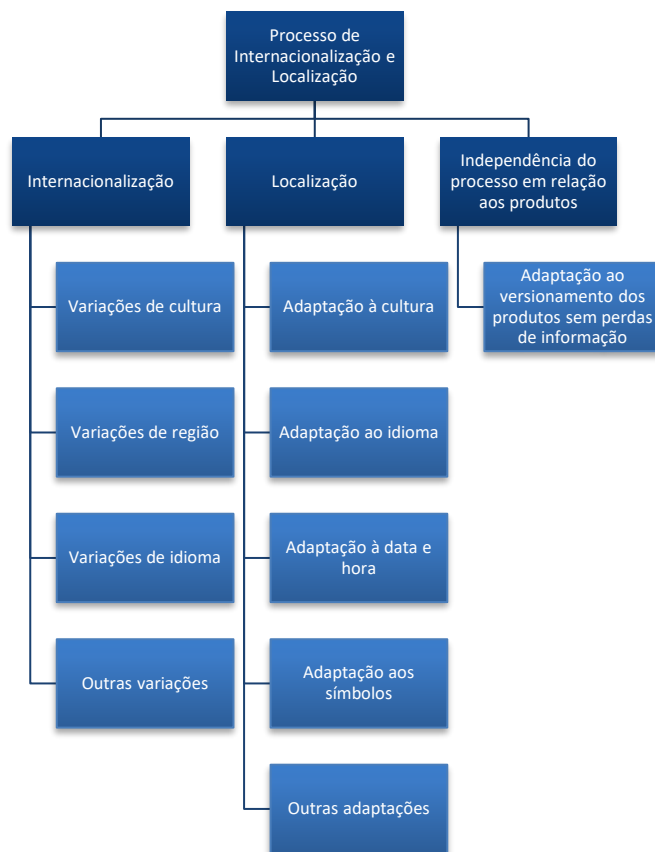


Figura 19 – Diagrama de árvore

O diagrama de árvore da Figura 19 distingue três tipos de funções importantes para o processo de Internacionalização e Localização:

- Internacionalização – processo que fará com que os produtos sejam facilmente internacionalizados sem necessidade de reengenharia;
- Localização – processo que fará com que os produtos se adaptem ao local alvo de internacionalização, como o idioma, data e hora entre outros;
- Independência do processo em relação aos produtos – o processo de internacionalização e de localização não deve ser comprometido aquando da existência de várias versões dos produtos ALERT®.

3.5.2.2 Análise Funcional

Através da Identificação Funcional, onde se identificou tanto os requisitos de qualidade do sistema solicitados pelo cliente como os requisitos funcionais, elaborar-se-á agora uma “Casa de Qualidade” tendo em conta estes requisitos identificados. Na “Casa de Qualidade” os dois tipos de requisitos (qualidade e funcionais) são comparados, sendo também avaliadas as relações existentes entre os diferentes requisitos funcionais para ser possível identificar relações positivas e negativas. Os graus de importância associados aos requisitos de qualidade definidos pelo cliente também são tidos em conta na realização da “Casa de Qualidade”. Por

fim, identificam-se valores/objetivos a atingir e um grau de dificuldade para cada um dos requisitos funcionais (Warwick Manufacturing Group, 2007).

Na Figura 20 encontra-se a “Casa de Qualidade” desenvolvida para este projeto.

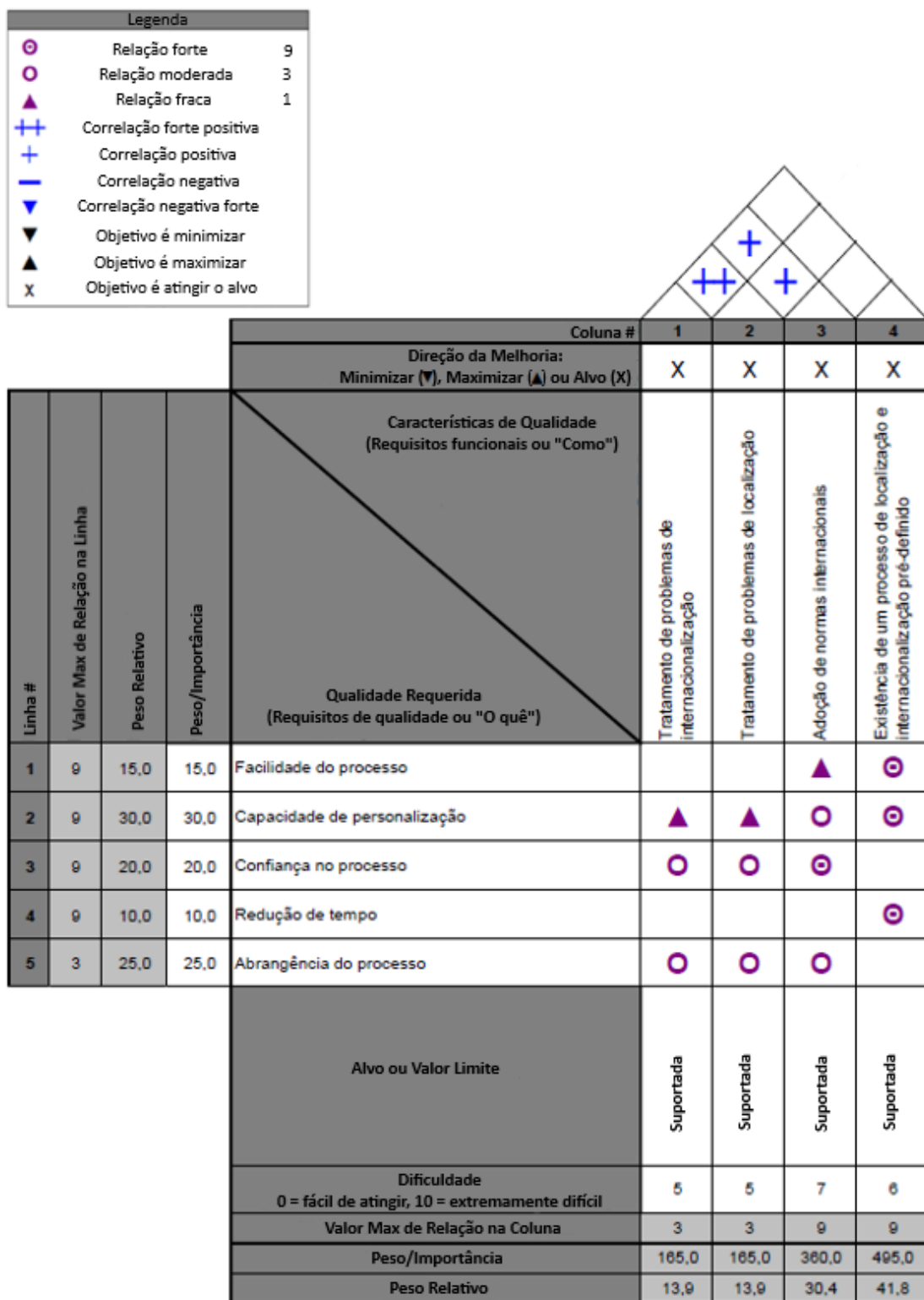


Figura 20 – Casa de Qualidade

Analisando a Figura 20 conclui-se o seguinte:

- De todos os requisitos de qualidade definidos pelo cliente, a capacidade de personalização do processo é o que apresenta uma maior importância, sendo que a abrangência do processo surge logo de seguida. O cliente pretende, assim, que o processo seja desenvolvido de forma a que este seja personalizado tendo em conta os locais alvo e que seja possível que o processo adapte os produtos ALERT® a todos os potenciais locais de internacionalização que surjam no futuro. Contudo não se pode descurar os outros requisitos, apesar de apresentarem uma menor importância, por serem essenciais para o processo.
- Em termos de comparação entre os requisitos de qualidade e os requisitos funcionais, foram definidos três tipos de graus de força relativamente à relação entre eles (relação forte, relação moderada e relação fraca). Com um grau de força forte, o grau mais importante, temos as relações entre:
 - Facilidade do processo e existência de um processo de localização e internacionalização pré-definido;
 - Capacidade de personalização e existência de um processo de localização e internacionalização pré-definido;
 - Confiança no processo e adoção de normas internacionais;
 - Redução de tempo e existência de um processo de localização e internacionalização pré-definido;

Existem outras relações, mas de menor importância quando comparadas com as anteriores.

- Comparando os diferentes requisitos funcionais entre eles nota-se que existe uma correlação fortemente positiva entre o tratamento de problemas de internacionalização e o tratamento de problemas de localização. Também existe uma correlação positiva entre o tratamento tanto de problemas de localização como de internacionalização com a adoção de normas internacionais. Em termos de correlações negativas não existe nada a apontar entre os diferentes requisitos funcionais.
- Todos os requisitos funcionais devem ser implementados no processo para se atingir a melhor qualidade possível deste, tendo diferentes níveis de dificuldade associados, sendo a adoção de normas internacionais o mais complexo de ser executado.

Por fim, com os requisitos analisados e com as funções principais identificadas na Identificação Funcional, procede-se agora à comparação entre as funções com o intuito de determinar qual será a mais significativa. Para tal recorre-se ao método “*Pairwise Comparison*” (Rich & Holweg, 2000). Nesta comparação, cada função é comparada com todas as outras, sendo utilizada uma tabela para auxiliar tal comparação. É identificado o que é mais importante em cada uma das funções, sendo que na comparação deve-se sempre decidir qual das funções é mais relevante e não deixar que ocorra um “empate” (Rich & Holweg, 2000).

Considera-se para comparação os seguintes valores:

- 1 ponto – diferença de importância menor, sendo o primeiro termo um pouco mais importante que o segundo (Rich & Holweg, 2000);
- 2 pontos – diferença de importância média, sendo o primeiro termo mais importante que o segundo (Rich & Holweg, 2000);

- 3 pontos – diferença de elevada importância, sendo o primeiro termo muito mais importante que o segundo (Rich & Holweg, 2000).

No fim de todas as funções comparadas, calcula-se o somatório de cada uma, sendo que a que tiver um valor mais elevado é considerada como a função mais relevante do projeto e a qual deve ser tida em maior conta no desenvolvimento (Rich & Holweg, 2000).

Sendo assim, de seguida são apresentadas as comparações entre as diferentes funções:

Pairwise Comparison: Processo de Internacionalização e Localização			
Funções:	A	Internacionalização	
	B	Localização	
	C	Independência do processo em relação aos produtos	
A vs B	A: 2	Pontuação:	
A vs C	A: 3	A:	5 pontos
B vs C	B: 3	B:	3 pontos
		C:	0 pontos
Pontos:	1	Um pouco mais importante / melhor	
	2	Significativamente mais importante / melhor	
	3	Muito mais importante	

Figura 21 – *Pairwise Comparison*: Processo de Internacionalização e Localização

Os resultados obtidos e demonstrados na Figura 21 permitem argumentar que a função de maior pertinência para o processo é a Internacionalização, que engloba as variações de cultura, região e idioma e que fará com que não seja necessário uma reengenharia aos produtos ALERT®. Seguidamente, surge a função de Localização onde se inserem as diferentes adaptações necessárias para que, efetivamente, o produto se adeque ao local alvo. Por fim, a função de independência do processo em relação aos produtos ALERT® é a que se revela como a menos relevante, mas que merece na mesma a atenção no seu desenvolvimento.

3.5.3 Criação de Alternativas

Nesta fase serão apresentadas algumas alternativas para dar resposta às funções do produto.

O processo de internacionalização e de localização é bastante extenso tendo diversos pontos a abordar e problemas a resolver, como por exemplo as diferenças entre formatos de datas nos diversos locais do globo.

Assim, e de forma a auxiliar este processo, foram identificadas algumas ferramentas existentes que ajudam a implementar soluções para os diferentes problemas relativos a

internacionalização e localização e que estão relacionados com o contexto do projeto e requisitos que este deve solucionar:

- JQuery.i18n;
- Globalize;
- i18next;
- Format.js;
- ILib;
- Angular Locale.

Cada uma destas ferramentas, estudadas na secção 2.1.7 do presente documento, possui determinadas características que são analisadas na fase seguinte, Análise e Avaliação.

3.5.4 Análise e Avaliação

Na fase de Análise e Avaliação é adotado o método *Analytic Hierarchy Process* (AHP), definido por Thomas L. Saaty, com o intuito de comparar as diferentes ferramentas definidas na fase de Criação de Alternativas segundo alguns critérios e, assim, determinar qual será a melhor para se utilizar no projeto.

O método AHP é um método de decisão multicritério discreto, sendo que permite que se divida o problema em análise em vários níveis hierárquicos para mais facilmente se avaliar e retirar conclusões (Nicola, 2018a).

Os critérios utilizados para avaliação das ferramentas são os seguintes:

- Suporte por parte da comunidade – verificar a atividade efetuada pela comunidade em relação ao desenvolvimento da ferramenta;
- Alcance de diversos tratamentos de problemas de i18n e l10n – existência de diferentes características disponibilizadas pela ferramenta para resolução de problemas de internacionalização e localização;
- Adoção de *standards* internacionais – atenção dada pela ferramenta à utilização de normas de internacionalização e localização adotadas a nível global;
- Compatibilidade com diversos navegadores – verificar a funcionalidade da ferramenta nos diferentes navegadores mais utilizados atualmente.

Para dar início à utilização do método AHP e com os critérios enunciados anteriormente, define-se a árvore de divisão hierárquica onde o problema a resolver é a escolha de uma das ferramentas.

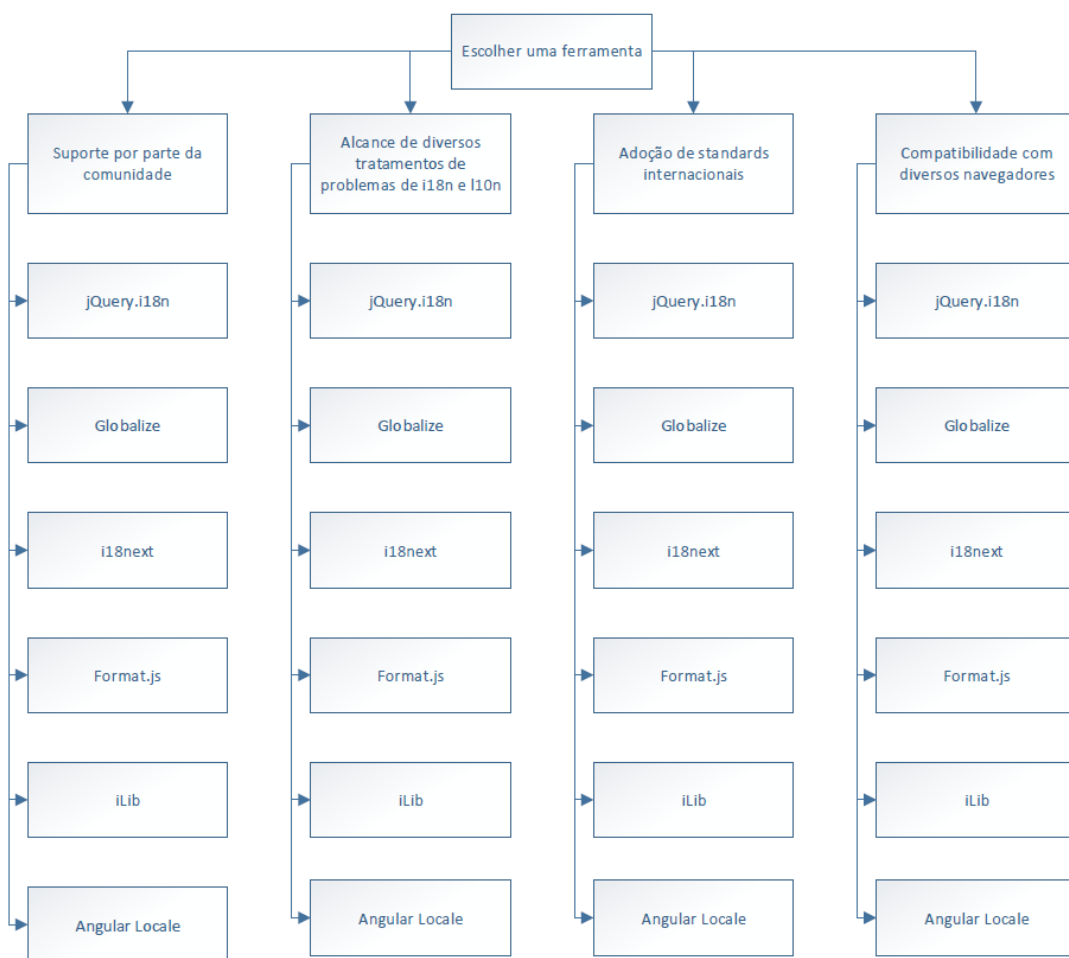


Figura 22 – Árvore de divisão hierárquica

Como se verifica pela figura anterior (Figura 22), cada um dos quatro critérios possui seis alternativas, sendo estas as ferramentas definidas na fase de Criação de Alternativas do *Value Analysis*.

Seguidamente, e adotando a Escala Fundamental definida pelo autor do método AHP (Tabela 7), é elaborada uma matriz de comparação para estabelecer prioridades entre os diferentes critérios.

Tabela 7 – Escala Fundamental – Níveis de importância de comparações (Saaty, 1991)

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

Nível de importância	Definição	Explicação
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
9	Importância absoluta	A evidência favorece uma atividade em relação a outra com o mais alto grau de certeza
2,4,6,8	Valores intermédios	Quando se procura uma condição de compromisso entre duas definições

Considerando os valores da tabela anterior, atribui-se agora prioridades a cada um dos critérios quando comparado com um outro.

Tabela 8 – Comparação de prioridades dos critérios (matriz de prioridades)

	Suporte por parte da comunidade	Alcance de diversos tratamentos de problemas de i18n e l10n	Adoção de <i>standards</i> internacionais	Compatibilidade com diversos navegadores
Suporte por parte da comunidade	1	$\frac{1}{3}$	2	2
Alcance de diversos tratamentos de problemas de i18n e l10n	3	1	3	3
Adoção de <i>standards</i> internacionais	$\frac{1}{2}$	$\frac{1}{3}$	1	$\frac{1}{2}$
Compatibilidade com diversos navegadores	$\frac{1}{2}$	$\frac{1}{3}$	2	1

O critério de alcance de diversos tratamentos de problemas de i18n e l10n é aquele que no geral, quando comparado com um dos outros três critérios, apresenta maior prioridade (p. ex. o alcance de diversos tratamentos de problemas de i18n e l10n tem uma importância levemente favorecida quando comparado com o critério de adoção de *standards* internacionais). Este critério é de facto o mais prioritário uma vez que a ferramenta deve ser utilizada para que se consiga resolver alguns dos problemas que se encontram quando se quer internacionalizar *software*.

Em segundo, ao nível geral de prioridades, o critério de suporte por parte da comunidade aparece uma vez que a ferramenta para ser utilizada convém que tenha uma comunidade ampla e ativa para mais facilmente serem detetados e corrigidos problemas e assim ter a ferramenta o mais atualizada possível.

Por fim, aparecem os critérios de compatibilidade com diversos navegadores e a adoção de *standards* internacionais, dando-se mais prioridade ao primeiro, uma vez que a ferramenta necessita de funcionar em diversos navegadores para que o cliente possa escolher o navegador com que está mais habituado a trabalhar. O critério de adoção de *standards* internacionais não pode ser colocado de lado pois a ferramenta deve adotar padrões utilizados a nível global para resolver os problemas existentes quer de internacionalização quer de localização.

Para dar continuidade ao processo AHP, calcula-se agora o somatório de cada coluna da tabela anterior para que posteriormente se consiga normalizar a tabela, onde cada valor da prioridade é dividido pelo somatório da coluna onde se encontra.

Tabela 9 – Somatório das colunas de cada critério

	Suporte por parte da comunidade	Alcance de diversos tratamentos de problemas de i18n e l10n	Adoção de <i>standards</i> internacionais	Compatibilidade com diversos navegadores
Suporte por parte da comunidade	1	$\frac{1}{3}$	2	2
Alcance de diversos tratamentos de problemas de i18n e l10n	3	1	3	3
Adoção de <i>standards</i> internacionais	$\frac{1}{2}$	$\frac{1}{3}$	1	$\frac{1}{2}$
Compatibilidade com diversos navegadores	$\frac{1}{2}$	$\frac{1}{3}$	2	1
Soma	5	2	8	$\frac{13}{2}$

Tabela 10 – Comparação de prioridades dos critérios normalizados

	Suporte por parte da comunidade	Alcance de diversos tratamentos de problemas de i18n e l10n	Adoção de <i>standards</i> internacionais	Compatibilidade com diversos navegadores
Suporte por parte da comunidade	$\frac{1}{5}$	$\frac{1}{6}$	$\frac{1}{4}$	$\frac{4}{13}$
Alcance de diversos tratamentos de problemas de i18n e l10n	$\frac{3}{5}$	$\frac{1}{2}$	$\frac{3}{8}$	$\frac{6}{13}$
Adoção de <i>standards</i> internacionais	$\frac{1}{10}$	$\frac{1}{6}$	$\frac{1}{8}$	$\frac{1}{13}$
Compatibilidade com diversos navegadores	$\frac{1}{10}$	$\frac{1}{6}$	$\frac{1}{4}$	$\frac{2}{13}$

Agora, define-se a ordem de importância de cada um dos critérios através da média aritmética dos valores de cada uma das linhas.

Tabela 11 – Prioridades dos critérios

	Suporte por parte da comunidade	Alcance de diversos tratamentos de problemas de i18n e l10n	Adoção de <i>standards</i> internacionais	Compatibilidade com diversos navegadores	Prioridades
Suporte por parte da comunidade	$\frac{1}{5}$	$\frac{1}{6}$	$\frac{1}{4}$	$\frac{4}{13}$	0,23
Alcance de diversos tratamentos de problemas de i18n e l10n	$\frac{3}{5}$	$\frac{1}{2}$	$\frac{3}{8}$	$\frac{6}{13}$	0,48
Adoção de <i>standards</i> internacionais	$\frac{1}{10}$	$\frac{1}{6}$	$\frac{1}{8}$	$\frac{1}{13}$	0,12

	Suporte por parte da comunidade	Alcance de diversos tratamentos de problemas de i18n e l10n	Adoção de <i>standards</i> internacionais	Compatibilidade com diversos navegadores	Prioridades
Compatibilidade com diversos navegadores	$\frac{1}{10}$	$\frac{1}{6}$	$\frac{1}{4}$	$\frac{2}{13}$	0,17

A Tabela 11 permite retirar o seguinte vetor de prioridades (ou vetor próprio):

$$\begin{bmatrix} 0,23 \\ 0,48 \\ 0,12 \\ 0,17 \end{bmatrix}$$

Este vetor indica que o critério de alcance de diversos tratamentos de problemas de i18n e l10n é aquele que apresenta maior prioridade, sendo seguido pelo critério de suporte por parte da comunidade. As últimas duas posições são preenchidas pelos critérios de compatibilidade com diversos navegadores e de adoção de *standards* internacionais respetivamente.

Deve-se, agora, verificar a consistência das prioridades. Para tal, calcula-se a Razão de Consistência (RC) (Nicola, 2018a) para se avaliar se as prioridades atribuídas a cada um dos critérios, inicialmente, foram consistentes ou não. O resultado do RC deve ser inferior a 0,1 para as prioridades serem consideradas consistentes. O RC é obtido através da seguinte fórmula:

$$RC = \frac{IC}{IR}$$

, onde IC é o Índice de Consistência e IR o Índice Aleatório obtido através da tabela definida pelo Laboratório Nacional de Oak Ridge dependendo do número de critérios analisados.

Tabela 12 – Valores IR definidos pelo Laboratório Nacional de Oak Ridge (Nicola, 2018a)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0,00	0,00	0,58	0,90	1,12	1,24	1,32	1,41	1,45	1,49	1,51	1,48	1,56	1,57	1,59

Como visto anteriormente, para se obter o RC é necessário antes se calcular o IC (Nicola, 2018a), sendo este obtido pela fórmula:

$$IC = \frac{\lambda_{\max} - n}{n - 1}$$

, onde λ_{max} é o valor próprio e n é o número de critérios avaliados.

λ_{max} é calculado com base na matriz de prioridades inicial e no vetor próprio obtido anteriormente:

$$Matriz\ prioridades \times Vetor\ próprio \cong \lambda_{max} \times Vetor\ próprio$$

$$\Leftrightarrow \begin{bmatrix} 1 & 4 & 3 & 2 \\ \frac{1}{4} & 1 & \frac{1}{2} & \frac{1}{3} \\ \frac{1}{3} & 2 & 1 & \frac{1}{3} \\ \frac{1}{2} & 3 & 3 & 1 \end{bmatrix} \times \begin{bmatrix} 0,23 \\ 0,48 \\ 0,12 \\ 0,17 \end{bmatrix} \cong \lambda_{max} \times \begin{bmatrix} 0,23 \\ 0,48 \\ 0,12 \\ 0,17 \end{bmatrix}$$

$$\Leftrightarrow \begin{bmatrix} 0,97 \\ 2,04 \\ 0,48 \\ 0,69 \end{bmatrix} \cong \lambda_{max} \times \begin{bmatrix} 0,23 \\ 0,48 \\ 0,12 \\ 0,17 \end{bmatrix}$$

$$\Leftrightarrow \lambda_{max} \cong 4,08$$

Tendo o valor do λ_{max} calcula-se agora o valor do IC:

$$IC = \frac{\lambda_{max} - n}{n - 1} = \frac{4,13 - 4}{4 - 1} = 0,043$$

Por fim, calcula-se o RC onde o valor de IR, segundo a tabela definida pelo Laboratório Nacional de Oak Ridge, é de 0,90 devido à avaliação de 4 critérios:

$$RC = \frac{IC}{0,90} = \frac{0,04}{0,90} \cong 0,04$$

Como se verifica, o valor da Razão de Consistência (0,04) é inferior a 0,1 pelo que os valores das prioridades são consistentes.

Tendo validado a consistência das prioridades, calcula-se agora, para cada um dos critérios, a importância relativa para cada uma das alternativas associadas. Para tal é necessário construir a matriz de comparação das alternativas de cada critério, de seguida calcular o somatório de cada coluna da matriz para posteriormente normalizar a mesma, com o intuito de obter o vetor próprio para cada critério (tal como aconteceu com a matriz das prioridades dos critérios – da Tabela 8 à Tabela 11). A consistência da atribuição das prioridades de cada um dos critérios avaliados encontra-se no Anexo B, sendo que todos os critérios são consistentes.

- Critério de Suporte por parte da comunidade:

Antes de se calcular o vetor próprio deste critério, é necessário referir que para a avaliação das prioridades, no dia 3 de dezembro de 2018 verificou-se a data da última contribuição para a ferramenta, bem como o número de programadores associados:

Tabela 13 – Data de última contribuição e número de programadores de cada uma das ferramentas

	Data da última contribuição	Número de programadores
jQuery.i18n	8 de novembro de 2018	26
Globalize	5 de novembro de 2018	66
i18next	22 de novembro de 2018	105
Format.js	25 de abril de 2018	6
iLib	2 de março de 2017	desconhecido
Angular Locale	4 de outubro de 2018	58

Tabela 14 – Matriz de prioridades das alternativas para o critério de suporte por parte da comunidade

	jQuery.i18n	Globalize	i18next	Format.js	iLib	Angular Locale
jQuery.i18n	1	$\frac{1}{3}$	$\frac{1}{4}$	5	6	$\frac{1}{3}$
Globalize	3	1	$\frac{1}{3}$	6	7	2
i18next	4	3	1	7	8	3
Format.js	$\frac{1}{5}$	$\frac{1}{6}$	$\frac{1}{7}$	1	2	$\frac{1}{6}$
iLib	$\frac{1}{6}$	$\frac{1}{7}$	$\frac{1}{8}$	$\frac{1}{2}$	1	$\frac{1}{7}$
Angular Locale	3	$\frac{1}{2}$	$\frac{1}{3}$	6	7	1

Tabela 15 – Somatório de cada coluna das alternativas do critério de suporte por parte da comunidade

	jQuery.i18n	Globalize	i18next	Format.js	iLib	Angular Locale
jQuery.i18n	1	$\frac{1}{3}$	$\frac{1}{4}$	5	6	$\frac{1}{3}$
Globalize	3	1	$\frac{1}{3}$	6	7	2
i18next	4	3	1	7	8	3
Format.js	$\frac{1}{5}$	$\frac{1}{6}$	$\frac{1}{7}$	1	2	$\frac{1}{6}$

	jQuery.i18n	Globalize	i18next	Format.js	iLib	Angular Locale
iLib	$\frac{1}{6}$	$\frac{1}{7}$	$\frac{1}{8}$	$\frac{1}{2}$	1	$\frac{1}{7}$
Angular Locale	3	$\frac{1}{2}$	$\frac{1}{3}$	6	7	1
Soma	$\frac{341}{30}$	$\frac{36}{7}$	$\frac{367}{168}$	$\frac{51}{2}$	31	$\frac{93}{14}$

Tabela 16 – Valores normalizados das alternativas do critério de suporte por parte da comunidade

	jQuery.i18n	Globalize	i18next	Format.js	iLib	Angular Locale
jQuery.i18n	$\frac{30}{341}$	$\frac{7}{108}$	$\frac{42}{367}$	$\frac{10}{51}$	$\frac{6}{31}$	$\frac{14}{279}$
Globalize	$\frac{90}{341}$	$\frac{7}{36}$	$\frac{56}{367}$	$\frac{4}{17}$	$\frac{7}{31}$	$\frac{28}{93}$
i18next	$\frac{120}{341}$	$\frac{7}{12}$	$\frac{168}{367}$	$\frac{14}{51}$	$\frac{8}{31}$	$\frac{14}{31}$
Format.js	$\frac{6}{341}$	$\frac{7}{216}$	$\frac{24}{367}$	$\frac{2}{51}$	$\frac{2}{31}$	$\frac{7}{279}$
iLib	$\frac{5}{341}$	$\frac{1}{36}$	$\frac{21}{367}$	$\frac{1}{51}$	$\frac{1}{31}$	$\frac{2}{93}$
Angular Locale	$\frac{90}{341}$	$\frac{7}{72}$	$\frac{56}{367}$	$\frac{4}{17}$	$\frac{7}{31}$	$\frac{14}{93}$

Tabela 17 – Prioridades das alternativas do critério de suporte por parte da comunidade

	jQuery.i18n	Globalize	i18next	Format.js	iLib	Angular Locale	Prioridades
jQuery.i18n	$\frac{30}{341}$	$\frac{7}{108}$	$\frac{42}{367}$	$\frac{10}{51}$	$\frac{6}{31}$	$\frac{14}{279}$	0,12
Globalize	$\frac{90}{341}$	$\frac{7}{36}$	$\frac{56}{367}$	$\frac{4}{17}$	$\frac{7}{31}$	$\frac{28}{93}$	0,23
i18next	$\frac{120}{341}$	$\frac{7}{12}$	$\frac{168}{367}$	$\frac{14}{51}$	$\frac{8}{31}$	$\frac{14}{31}$	0,39
Format.js	$\frac{6}{341}$	$\frac{7}{216}$	$\frac{24}{367}$	$\frac{2}{51}$	$\frac{2}{31}$	$\frac{7}{279}$	0,04
iLib	$\frac{5}{341}$	$\frac{1}{36}$	$\frac{21}{367}$	$\frac{1}{51}$	$\frac{1}{31}$	$\frac{2}{93}$	0,03
Angular Locale	$\frac{90}{341}$	$\frac{7}{72}$	$\frac{56}{367}$	$\frac{4}{17}$	$\frac{7}{31}$	$\frac{14}{93}$	0,19

O vetor próprio para o critério de suporte por parte da comunidade é o seguinte:

$$\begin{bmatrix} 0,12 \\ 0,23 \\ 0,39 \\ 0,04 \\ 0,03 \\ 0,19 \end{bmatrix}$$

- Critério de Alcance de diversos tratamentos de problemas de i18n e l10n:

Tabela 18 – Matriz de prioridades das alternativas para o critério de alcance de diversos tratamentos de problemas de i18n e l10n

	jQuery.i18n	Globalize	i18next	Format.js	iLib	Angular Locale
jQuery.i18n	1	$\frac{1}{4}$	$\frac{1}{3}$	$\frac{1}{4}$	$\frac{1}{2}$	$\frac{1}{4}$
Globalize	4	1	$\frac{1}{2}$	$\frac{1}{2}$	2	$\frac{1}{2}$
i18next	3	2	1	$\frac{1}{3}$	2	$\frac{1}{2}$
Format.js	4	2	3	1	3	2
iLib	2	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{3}$	1	$\frac{1}{2}$
Angular Locale	4	2	2	$\frac{1}{2}$	2	1

Tabela 19 – Somatório de cada coluna das alternativas do critério de alcance de diversos tratamentos de problemas de i18n e l10n

	jQuery.i18n	Globalize	i18next	Format.js	iLib	Angular Locale
jQuery.i18n	1	$\frac{1}{4}$	$\frac{1}{3}$	$\frac{1}{4}$	$\frac{1}{2}$	$\frac{1}{4}$
Globalize	4	1	$\frac{1}{2}$	$\frac{1}{2}$	2	$\frac{1}{2}$
i18next	3	2	1	$\frac{1}{3}$	2	$\frac{1}{2}$
Format.js	4	2	3	1	3	2
iLib	2	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{3}$	1	$\frac{1}{2}$
Angular Locale	4	2	2	$\frac{1}{2}$	2	1

	jQuery.i18n	Globalize	i18next	Format.js	iLib	Angular Locale
Soma	18	$\frac{31}{4}$	$\frac{22}{3}$	$\frac{35}{12}$	$\frac{21}{2}$	$\frac{19}{4}$

Tabela 20 – Valores normalizados das alternativas do critério de alcance de diversos tratamentos de problemas de i18n e i10n

	jQuery.i18n	Globalize	i18next	Format.js	iLib	Angular Locale
jQuery.i18n	$\frac{1}{18}$	$\frac{1}{31}$	$\frac{1}{22}$	$\frac{3}{35}$	$\frac{1}{21}$	$\frac{1}{19}$
Globalize	$\frac{2}{9}$	$\frac{4}{31}$	$\frac{3}{44}$	$\frac{6}{35}$	$\frac{4}{21}$	$\frac{2}{19}$
i18next	$\frac{1}{6}$	$\frac{8}{31}$	$\frac{3}{22}$	$\frac{4}{35}$	$\frac{4}{21}$	$\frac{2}{19}$
Format.js	$\frac{2}{9}$	$\frac{8}{31}$	$\frac{9}{22}$	$\frac{12}{35}$	$\frac{2}{7}$	$\frac{8}{19}$
iLib	$\frac{1}{9}$	$\frac{2}{31}$	$\frac{3}{44}$	$\frac{4}{35}$	$\frac{2}{21}$	$\frac{2}{19}$
Angular Locale	$\frac{2}{9}$	$\frac{8}{31}$	$\frac{3}{11}$	$\frac{6}{35}$	$\frac{4}{21}$	$\frac{4}{19}$

Tabela 21 – Prioridades das alternativas do critério de alcance de diversos tratamentos de problemas de i18n e i10n

	jQuery.i18n	Globalize	i18next	Format.js	iLib	Angular Locale	Prioridades
jQuery.i18n	$\frac{1}{18}$	$\frac{1}{31}$	$\frac{1}{22}$	$\frac{3}{35}$	$\frac{1}{21}$	$\frac{1}{19}$	0,06
Globalize	$\frac{2}{9}$	$\frac{4}{31}$	$\frac{3}{44}$	$\frac{6}{35}$	$\frac{4}{21}$	$\frac{2}{19}$	0,15
i18next	$\frac{1}{6}$	$\frac{8}{31}$	$\frac{3}{22}$	$\frac{4}{35}$	$\frac{4}{21}$	$\frac{2}{19}$	0,16
Format.js	$\frac{2}{9}$	$\frac{8}{31}$	$\frac{9}{22}$	$\frac{12}{35}$	$\frac{2}{7}$	$\frac{8}{19}$	0,32
iLib	$\frac{1}{9}$	$\frac{2}{31}$	$\frac{3}{44}$	$\frac{4}{35}$	$\frac{2}{21}$	$\frac{2}{19}$	0,09
Angular Locale	$\frac{2}{9}$	$\frac{8}{31}$	$\frac{3}{11}$	$\frac{6}{35}$	$\frac{4}{21}$	$\frac{4}{19}$	0,22

O vetor próprio para o critério de alcance de diversos tratamentos de problemas de i18n e l10n é o seguinte:

$$\begin{bmatrix} 0,06 \\ 0,15 \\ 0,16 \\ 0,32 \\ 0,09 \\ 0,22 \end{bmatrix}$$

- Critério de Adoção de *standards* internacionais:

Tabela 22 – Matriz de prioridades das alternativas para o critério de adoção de *standards* internacionais

	jQuery.i18n	Globalize	i18next	Format.js	iLib	Angular Locale
jQuery.i18n	1	$\frac{1}{3}$	3	2	$\frac{1}{2}$	2
Globalize	3	1	4	3	2	3
i18next	$\frac{1}{3}$	$\frac{1}{4}$	1	$\frac{1}{2}$	$\frac{1}{3}$	$\frac{1}{2}$
Format.js	$\frac{1}{2}$	$\frac{1}{3}$	2	1	$\frac{1}{3}$	1
iLib	2	$\frac{1}{2}$	3	3	1	3
Angular Locale	$\frac{1}{2}$	$\frac{1}{3}$	2	1	$\frac{1}{3}$	1

Tabela 23 – Somatório de cada coluna das alternativas do critério de adoção de *standards* internacionais

	jQuery.i18n	Globalize	i18next	Format.js	iLib	Angular Locale
jQuery.i18n	1	$\frac{1}{3}$	3	2	$\frac{1}{2}$	2
Globalize	3	1	4	3	2	3
i18next	$\frac{1}{3}$	$\frac{1}{4}$	1	$\frac{1}{2}$	$\frac{1}{3}$	$\frac{1}{2}$
Format.js	$\frac{1}{2}$	$\frac{1}{3}$	2	1	$\frac{1}{3}$	1
iLib	2	$\frac{1}{2}$	3	3	1	3

	jQuery.i18n	Globalize	i18next	Format.js	iLib	Angular Locale
Angular Locale	$\frac{1}{2}$	$\frac{1}{3}$	2	1	$\frac{1}{3}$	1
Soma	$\frac{22}{3}$	$\frac{11}{4}$	15	$\frac{21}{2}$	$\frac{9}{2}$	$\frac{21}{2}$

Tabela 24 – Valores normalizados das alternativas do critério de adoção de *standards* internacionais

	jQuery.i18n	Globalize	i18next	Format.js	iLib	Angular Locale
jQuery.i18n	$\frac{3}{22}$	$\frac{4}{33}$	$\frac{1}{5}$	$\frac{4}{21}$	$\frac{1}{9}$	$\frac{4}{21}$
Globalize	$\frac{9}{22}$	$\frac{4}{11}$	$\frac{4}{15}$	$\frac{2}{7}$	$\frac{4}{9}$	$\frac{2}{7}$
i18next	$\frac{1}{22}$	$\frac{1}{11}$	$\frac{1}{15}$	$\frac{1}{21}$	$\frac{2}{27}$	$\frac{1}{21}$
Format.js	$\frac{3}{44}$	$\frac{4}{33}$	$\frac{2}{15}$	$\frac{2}{21}$	$\frac{2}{27}$	$\frac{2}{21}$
iLib	$\frac{3}{11}$	$\frac{2}{11}$	$\frac{1}{5}$	$\frac{2}{7}$	$\frac{2}{9}$	$\frac{2}{7}$
Angular Locale	$\frac{3}{44}$	$\frac{4}{33}$	$\frac{2}{15}$	$\frac{2}{21}$	$\frac{2}{27}$	$\frac{2}{21}$

Tabela 25 – Prioridades das alternativas do critério de adoção de *standards* internacionais

	jQuery.i18n	Globalize	i18next	Format.js	iLib	Angular Locale	Prioridades
jQuery.i18n	$\frac{3}{22}$	$\frac{4}{33}$	$\frac{1}{5}$	$\frac{4}{21}$	$\frac{1}{9}$	$\frac{4}{21}$	0,16
Globalize	$\frac{9}{22}$	$\frac{4}{11}$	$\frac{4}{15}$	$\frac{2}{7}$	$\frac{4}{9}$	$\frac{2}{7}$	0,34
i18next	$\frac{1}{22}$	$\frac{1}{11}$	$\frac{1}{15}$	$\frac{1}{21}$	$\frac{2}{27}$	$\frac{1}{21}$	0,06
Format.js	$\frac{3}{44}$	$\frac{4}{33}$	$\frac{2}{15}$	$\frac{2}{21}$	$\frac{2}{27}$	$\frac{2}{21}$	0,10
iLib	$\frac{3}{11}$	$\frac{2}{11}$	$\frac{1}{5}$	$\frac{2}{7}$	$\frac{2}{9}$	$\frac{2}{7}$	0,24
Angular Locale	$\frac{3}{44}$	$\frac{4}{33}$	$\frac{2}{15}$	$\frac{2}{21}$	$\frac{2}{27}$	$\frac{2}{21}$	0,10

O vetor próprio para o critério de adoção de *standards* internacionais é o seguinte:

$$\begin{bmatrix} 0,16 \\ 0,34 \\ 0,06 \\ 0,10 \\ 0,24 \\ 0,10 \end{bmatrix}$$

- Critério de Compatibilidade com diversos navegadores:

Tabela 26 – Matriz de prioridades das alternativas para o critério de compatibilidade com diversos navegadores

	jQuery.i18n	Globalize	i18next	Format.js	iLib	Angular Locale
jQuery.i18n	1	$\frac{1}{2}$	1	1	$\frac{1}{2}$	$\frac{1}{2}$
Globalize	2	1	2	2	$\frac{1}{2}$	$\frac{1}{2}$
i18next	1	$\frac{1}{2}$	1	1	$\frac{1}{2}$	$\frac{1}{2}$
Format.js	1	$\frac{1}{2}$	1	1	$\frac{1}{2}$	$\frac{1}{2}$
iLib	2	2	2	2	1	1
Angular Locale	2	2	2	2	1	1

Tabela 27 – Somatório de cada coluna das alternativas do critério de compatibilidade com diversos navegadores

	jQuery.i18n	Globalize	i18next	Format.js	iLib	Angular Locale
jQuery.i18n	1	$\frac{1}{2}$	1	1	$\frac{1}{2}$	$\frac{1}{2}$
Globalize	2	1	2	2	$\frac{1}{2}$	$\frac{1}{2}$
i18next	1	$\frac{1}{2}$	1	1	$\frac{1}{2}$	$\frac{1}{2}$

	jQuery.i18n	Globalize	i18next	Format.js	iLib	Angular Locale
Format.js	1	$\frac{1}{2}$	1	1	$\frac{1}{2}$	$\frac{1}{2}$
iLib	2	2	2	2	1	1
Angular Locale	2	2	2	2	1	1
Soma	9	$\frac{13}{2}$	9	9	4	4

Tabela 28 – Valores normalizados das alternativas do critério de compatibilidade com diversos navegadores

	jQuery.i18n	Globalize	i18next	Format.js	iLib	Angular Locale
jQuery.i18n	$\frac{1}{9}$	$\frac{1}{13}$	$\frac{1}{9}$	$\frac{1}{9}$	$\frac{1}{8}$	$\frac{1}{8}$
Globalize	$\frac{2}{9}$	$\frac{2}{13}$	$\frac{2}{9}$	$\frac{2}{9}$	$\frac{1}{8}$	$\frac{1}{8}$
i18next	$\frac{1}{9}$	$\frac{1}{13}$	$\frac{1}{9}$	$\frac{1}{9}$	$\frac{1}{8}$	$\frac{1}{8}$
Format.js	$\frac{1}{9}$	$\frac{1}{13}$	$\frac{1}{9}$	$\frac{1}{9}$	$\frac{1}{8}$	$\frac{1}{8}$
iLib	$\frac{2}{9}$	$\frac{4}{13}$	$\frac{2}{9}$	$\frac{2}{9}$	$\frac{1}{4}$	$\frac{1}{4}$
Angular Locale	$\frac{2}{9}$	$\frac{4}{13}$	$\frac{2}{9}$	$\frac{2}{9}$	$\frac{1}{4}$	$\frac{1}{4}$

Tabela 29 – Prioridades das alternativas do critério de compatibilidade com diversos navegadores

	jQuery.i18n	Globalize	i18next	Format.js	iLib	Angular Locale	Prioridades
jQuery.i18n	$\frac{1}{9}$	$\frac{1}{13}$	$\frac{1}{9}$	$\frac{1}{9}$	$\frac{1}{8}$	$\frac{1}{8}$	0,11
Globalize	$\frac{2}{9}$	$\frac{2}{13}$	$\frac{2}{9}$	$\frac{2}{9}$	$\frac{1}{8}$	$\frac{1}{8}$	0,17
i18next	$\frac{1}{9}$	$\frac{1}{13}$	$\frac{1}{9}$	$\frac{1}{9}$	$\frac{1}{8}$	$\frac{1}{8}$	0,11
Format.js	$\frac{1}{9}$	$\frac{1}{13}$	$\frac{1}{9}$	$\frac{1}{9}$	$\frac{1}{8}$	$\frac{1}{8}$	0,11

	jQuery.i18n	Globalize	i18next	Format.js	iLib	Angular Locale	Prioridades
iLib	$\frac{2}{9}$	$\frac{4}{13}$	$\frac{2}{9}$	$\frac{2}{9}$	$\frac{1}{4}$	$\frac{1}{4}$	0,25
Angular Locale	$\frac{2}{9}$	$\frac{4}{13}$	$\frac{2}{9}$	$\frac{2}{9}$	$\frac{1}{4}$	$\frac{1}{4}$	0,25

O vetor próprio para o critério de compatibilidade com diversos navegadores é o seguinte:

$\begin{bmatrix} 0,11 \\ 0,17 \\ 0,11 \\ 0,11 \\ 0,25 \\ 0,25 \end{bmatrix}$

Como última fase do AHP, e tendo os diversos vetores próprios dos critérios, calcula-se a prioridade composta para as alternativas com o intuito de se escolher a melhor. Para tal cria-se uma matriz com todos os vetores próprios das alternativas de cada um dos critérios para se multiplicar pelo vetor próprio dos critérios. Todos os valores necessários para este cálculo encontram-se na Figura 23.

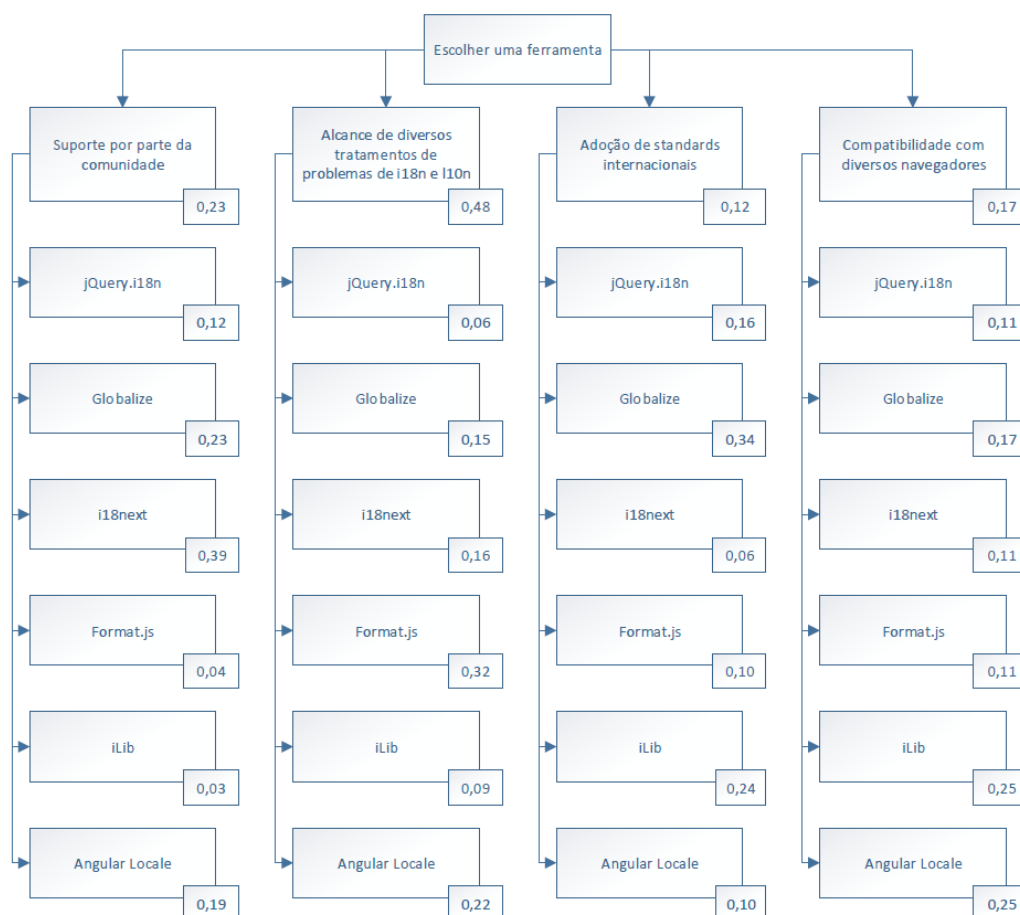


Figura 23 – Árvore de divisão hierárquica com valores associados

Cálculo:

$$\begin{bmatrix} 0,12 & 0,06 & 0,16 & 0,11 \\ 0,23 & 0,15 & 0,34 & 0,17 \\ 0,39 & 0,16 & 0,06 & 0,11 \\ 0,04 & 0,32 & 0,10 & 0,11 \\ 0,03 & 0,09 & 0,24 & 0,25 \\ 0,19 & 0,22 & 0,10 & 0,25 \end{bmatrix} \times \begin{bmatrix} 0,23 \\ 0,48 \\ 0,12 \\ 0,17 \end{bmatrix} = \begin{bmatrix} 0,094 \\ 0,195 \\ 0,192 \\ 0,194 \\ 0,121 \\ 0,204 \end{bmatrix}$$

Com o resultado obtido anteriormente, constata-se que a ferramenta mais adequada para escolha, de forma a auxiliar o processo de internacionalização e de localização, é a Angular Locale. Esta é a ferramenta que obteve um maior valor (0,204) considerando os critérios e respectivas importâncias definidas.

3.5.5 Implementação

A última fase do *Value Analysis* é a fase de implementação na qual se tem em conta os valores obtidos na fase anterior. Assim, segundo os resultados do AHP, a ferramenta escolhida é a Angular Locale, que deve ser utilizada para auxiliar a resolução de alguns dos problemas de internacionalização e de localização do processo. Contudo, outras bibliotecas, como a Globalize (segunda ferramenta com maior valor) também pode ser utilizada, principalmente para colmatar algumas deficiências que a Angular Locale possa conter.

4 Análise e Desenho

Neste capítulo são apresentados a análise e o desenho que devem ser realizados para atingir a solução do problema em questão. Inicialmente, é referida uma solução geral de como deve ser realizado o processo de internacionalização e de localização adaptado às necessidades dos produtos. De seguida, é apresentada a forma como o produto ALERT® PFH se encontrava adaptado a estes problemas no início do desenvolvimento deste projeto e o que deve ser alterado de forma a melhorar o processo em questão.

4.1 Processo de Internacionalização e Localização

De forma a preparar o *software* para o mundo, este deve passar por diversas fases de internacionalização e de localização para que seja disponibilizado da melhor forma possível.

Assim, e com o intuito de realizar tal, deve-se dar grande importância à separação do código dos produtos ALERT® em duas grandes áreas: “Código fonte” e ficheiros/recursos utilizados para representar as diferenças existentes ao nível dos diversos locais onde se pretende disponibilizar o *software* (Figura 24).

O “Código fonte” representa todo o código criado pelos desenvolvedores de *software* da empresa, que não sofre modificações quando é disponibilizado num determinado local do globo, por não conter nenhuma referência à cultura de locais. Contudo, este código fonte deve estar devidamente adaptado a ser possível utilizá-lo em conjunto com os ficheiros/recursos que especificam um local. O “Código fonte” por não possuir nenhuma referência aos locais é considerado como código internacionalizado.

A outra área na qual os produtos ALERT® devem ser divididos é em ficheiros ou recursos onde se pode definir os requisitos a serem representados nos produtos num determinado país ou local do globo.

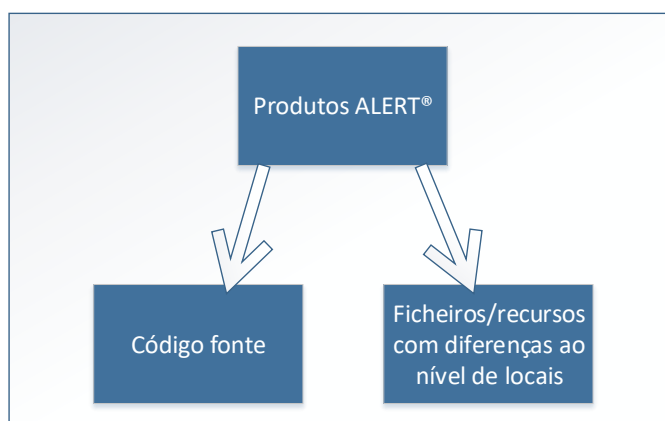


Figura 24 – Separação geral do código dos Produtos ALERT®. Adaptado de Gross (2006)

Estes ficheiros ou recursos, são divididos consoante os locais/países onde se irá disponibilizar o *software*. Ou seja, para cada local alvo de internacionalização é criado um ficheiro para representar esse local e é onde são definidas todas as soluções para os problemas a resolver para esse local, de forma a disponibilizar o *software* da melhor forma possível. Cada um destes ficheiros deve possuir conteúdo para dar resposta a cada um dos seguintes problemas:

- Cores – apresenta a paleta de cores que devem ser utilizadas para desenhar a interface gráfica do utilizador e assim disponibilizar as cores adequadas ao local para que não cause nenhum constrangimento aos utilizadores por utilização de cores menos corretas;
- Direção de texto – consoante o sentido de leitura que é utilizado nesse local do globo, define-se como os produtos ALERT® devem disponibilizar a interface ao utilizador relativamente à orientação dos caracteres textuais (p. ex. da esquerda para a direita – “LTR”² – ou da direita para a esquerda – “RTL”³);
- Formatos numéricos – como existem diferenças a nível da formatação dos números consoante os locais, define-se como se deve representar os mesmos no ficheiro do local;
- Formatos de data e hora – tal como acontece com os formatos numéricos que variam consoante os locais do globo, também surge a necessidade de definir o formato da data e da hora que deve ser utilizado para representar esses elementos;
- Traduções – neste caso existem informações de texto traduzidas para que os conteúdos textuais dos produtos sejam disponibilizados adequadamente na língua utilizada naquele local;
- Tamanho dos componentes gráficos – consoante as traduções de texto ocorridas e tendo em conta o tamanho das mesmas, define-se quais os possíveis redimensionamentos de escala que se deve fazer aos componentes da interface do utilizador, utilizando por defeito o tamanho dos componentes na língua inglesa para determinar os tamanhos nas restantes linguagens; como as traduções de texto não seguem um determinado padrão (tanto podem ser maiores ou mais pequenos que na língua inglesa), estes tamanhos devem ser analisados devidamente na interface final disponibilizada ao utilizador;

² Do inglês “Left To Right”

³ Do inglês “Right To Left”

- Fontes e estilos de texto – dependendo dos conjuntos de caracteres utilizados para representar conteúdo textual num determinado local do globo, são definidas quais as fontes e estilo de texto que devem ser utilizados para mostrar a interface ao utilizador corretamente, uma vez que existem fontes que não suportam determinados conjuntos de caracteres;
- Ícones e imagens – define-se os diversos tipos de ícones e imagens a utilizar nos produtos ALERT®, sendo que estes devem estar ajustados ao local onde o produto é disponibilizado.

Assim, com estes ficheiros definidos, os produtos ALERT® podem alterar-se mais facilmente para corresponderem às exigências dos locais alvo de internacionalização, contudo, as especificações de cada um dos ficheiros necessita de uma adequada investigação relacionada com o local utilizado.

4.1.1 Processo detalhado de Internacionalização

Apesar da internacionalização estar maioritariamente relacionada com a parte do código fonte do *software*, existem bastantes tarefas para que esse código fique corretamente adaptado, de forma a que não possua nenhuma referência que possa comprometer a internacionalização do *software*. Além disso, a interface do utilizador tem de estar facilmente recetiva a alterações ao nível de diferenças dos locais.

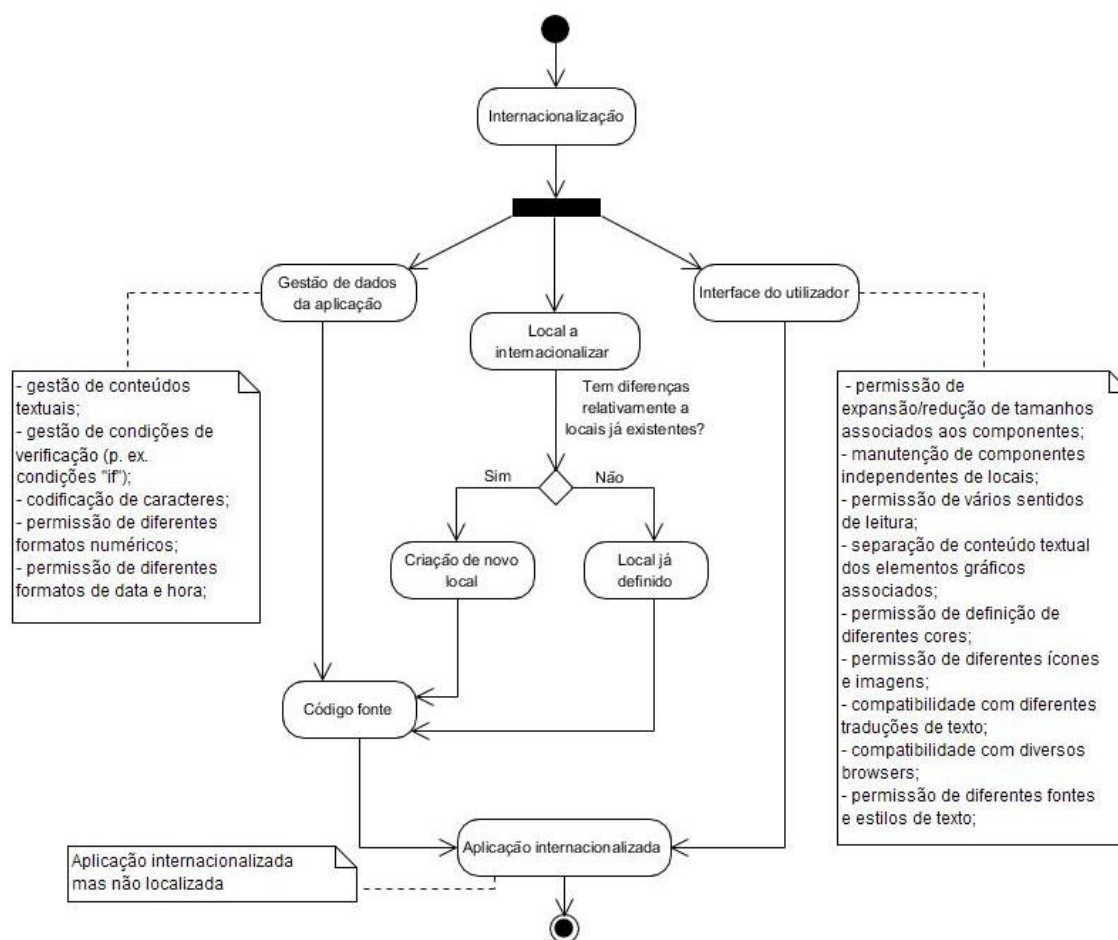


Figura 25 – Processo de Internacionalização. Adaptado de Gross (2006)

Através da análise do diagrama de atividades da Figura 25 constata-se que o *software* a internacionalizar tem de passar por diversas tarefas, como por exemplo: a escolha do local onde se pretende internacionalizar, a gestão dos dados da aplicação e a gestão de componentes gráficos relacionados com a interface do utilizador.

Relativamente ao local a internacionalizar, a empresa deve conhecer qual o local em específico no qual vai disponibilizar o *software* criado. Desta forma, é possível estabelecer uma comparação entre este e outros locais pré-definidos. Caso haja diferenças deve-se criar um ficheiro novo para armazenar os requisitos associados a esse local, caso contrário utiliza-se um pré-definido. Esta tarefa está relacionada com a localização, mas encontra-se mencionada nesta secção, devido ao facto de que ao desenvolver o código fonte já se deve ter em atenção os requisitos de diversos locais do globo, para que não seja necessário proceder a uma posterior reengenharia.

A tarefa de gestão de dados da aplicação implica a realização de diversas ações: gerir os conteúdos textuais a disponibilizar na aplicação, de forma a que todos estes conteúdos não sejam colocados no código fonte, mas sim em ficheiros separados para posterior tradução; não utilizar condições de verificação para avaliar conteúdos de variáveis que podem ser alteradas

consoante o local pois implica a alteração do código (p. ex. não utilizar condições “if” para avaliar o conteúdo de uma variável de texto); ter atenção à codificação de caracteres para que suporte os caracteres a disponibilizar na linguagem do local em que vai ser disponibilizado o *software*; ser possível definir diferentes formatos numéricos para além de ser possível mostrar a data e a hora com configurações diferentes, nomeadamente em diversos tipos de formatos.

A gestão de componentes gráficos da interface do utilizador também acarreta várias ações que devem ser realizadas, nomeadamente:

- Permitir que os componentes gráficos se expandam ou diminuam consoante o tamanho do conteúdo a eles associada;
- Deve-se ter em atenção a manutenção dos componentes gráficos que não sofrem nenhuma alteração mesmo se modificando o local onde se disponibiliza o *software* (p. ex. o símbolo de uma marca é mantido em todo o mundo);
- Ser possível consultar a interface gráfica em diversos sentidos de leitura;
- Separar o conteúdo textual de elementos gráficos como imagens e ícones para que estes elementos sejam reutilizados noutros locais;
- Ser possível que a paleta de cores seja facilmente alterada pois algumas cores podem não ser bem aceites em determinadas regiões do globo;
- Facilidade de alteração de imagens e ícones da aplicação para que estes estejam adequados à cultura do local;
- Possibilidade de facilmente se aceder a conteúdos de texto traduzidos consoante o local onde se disponibiliza a aplicação para além de que se consiga definir e apresentar o texto com diferentes fontes e estilos de acordo com os caracteres a serem disponibilizados (existem fontes de texto que não possuem suporte para determinados caracteres);
- A interface gráfica funcione da mesma forma em qualquer que seja o navegador utilizado para consultar o *software*.

Tendo todas estas tarefas em atenção, podemos ficar com uma aplicação adequadamente internacionalizada, sendo que, se tal for o intuito, conseguir-se-á apresentar o conteúdo da mesma com as especificações de um local em concreto, passando, portanto, à fase de localização do *software*.

4.1.2 Processo detalhado de Localização

Tendo o *software* corretamente internacionalizado, a localização do *software* torna-se mais fácil de acontecer. Para tal, os ficheiros relacionados com a resolução de problemas de localização devem estar corretamente definidos.

Estes ficheiros, como já referido, devem estar à parte do código fonte, sendo que devem ser facilmente acessíveis pelo *software*.

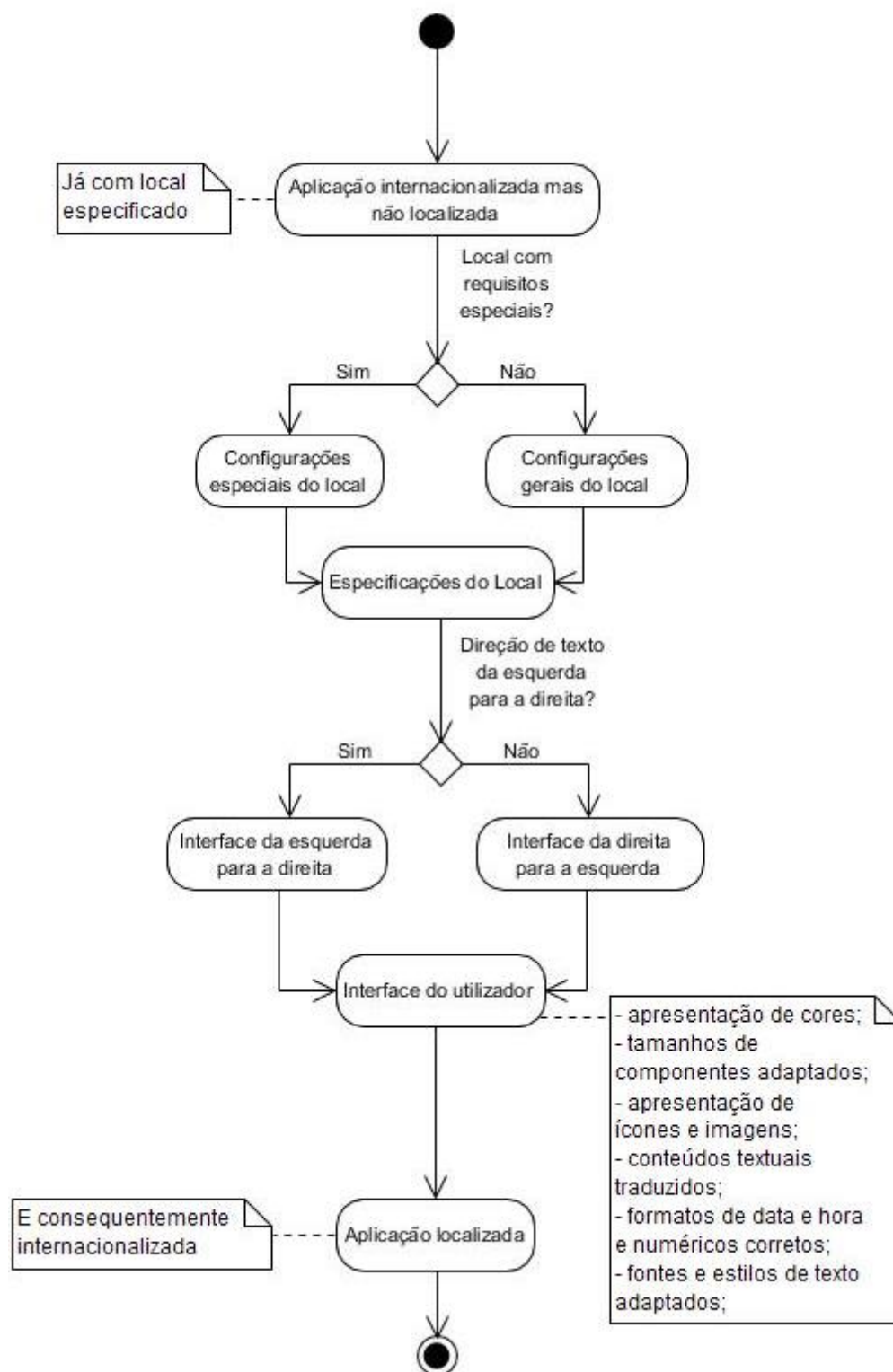


Figura 26 – Processo de Localização. Adaptado de Gross (2006)

Tal como aquilo que acontece relativamente ao processo de internacionalização de *software*, também o processo de localização passa por diversas tarefas para que seja disponibilizada ao

utilizador a melhor versão da aplicação, estando corretamente adaptada ao local onde é disponibilizada (diagrama de atividades da Figura 26).

Assim, o *software* a localizar já deve estar devidamente internacionalizado para que se inicie a fase de localização. Tendo um local especificado, há que aceder ao ficheiro onde estão definidas as características particulares do local. No caso de o local ter configurações personalizadas deve-se aceder a um ficheiro em concreto com essas especificações, caso contrário acede-se ao ficheiro com as configurações gerais do local.

Posteriormente, deve-se determinar qual é o sentido de orientação a aplicar à interface do utilizador. Consoante o que estiver especificado no ficheiro com as características do local, o sentido de leitura poderá ser da esquerda para a direita ou da direita para a esquerda, sendo que todos os componentes gráficos da interface devem estar corretamente adaptados a este sentido.

Tendo a orientação do *software* corretamente definida, deve-se, então, personalizar os diversos componentes gráficos da interface do utilizador. As cores devem ser alteradas conforme as especificações do local para além de que os componentes gráficos, consoante o tamanho das traduções, devem ser apresentados com uma dimensão adequada para que o texto não esteja nem demasiado pequeno nem demasiado grande. Também, os ícones e as imagens devem ser apresentados corretamente consoante o que foi definido para ser mostrado ao utilizador naquele local. O conteúdo do texto deve estar adequadamente traduzido, tendo uma fonte e um estilo de texto concreto para que se consiga disponibilizar todos os caracteres da aplicação sem nenhum erro. Por fim, o formato da data e da hora, bem como dos números, deve estar representado segundo as definições existentes para aquele local, para que o utilizador consulte os dados no formato que está habituado a lidar.

Com todas estas tarefas realizadas teremos um *software* corretamente adaptado ao local alvo, sendo que ao funcionar para um dado local, e alterando as configurações dos ficheiros com a resolução dos problemas de internacionalização e localização, temos um *software* que pode ser utilizado em todo o globo.

4.2 Análise do ALERT® PFH

O ALERT® PFH é o foco principal do desenvolvimento deste projeto, pelo que a análise do mesmo se revela importante para que seja possível entender a forma como se deve adaptar o processo de internacionalização e localização a esta solução de *software* e, posteriormente, ser possível utilizar o processo realizado nos restantes produtos ALERT®.

O ALERT® PFH com o intuito de permitir uma divisão modular entre os diferentes componentes, a lógica do negócio e a interface dos utilizadores, encontra-se estruturado na seguinte forma:

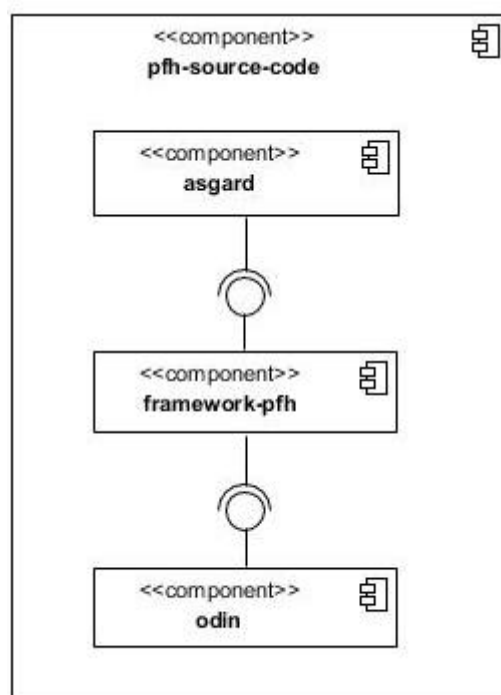


Figura 27 – Diagrama de componentes ALERT® PFH

Verificando a Figura 27 constata-se que o ALERT® PFH se encontra dividido em três componentes – “asgard”, “framework-pfh” e “odin”.

O “odin” é o componente base do ALERT® PFH. Contém um conjunto de módulos que, para além de outras operações, realiza tarefas de baixo nível como por exemplo o roteamento de eventos, passagem e manutenção de estados e deteção de eventos das interfaces do *hardware* como o teclado. O “odin” contém também todos os componentes gráficos principais que são utilizados para disponibilizar corretamente o PFH, logo muita da lógica relacionada com a internacionalização e localização encontra-se aqui. Para além disso, este componente contém ainda a *framework* AngularJS como também a biblioteca jQuery, entre outras.

A “framework-pfh” é um componente intermédio que faz a ligação entre o “odin” e o “asgard”. É responsável por adicionar lógica aos componentes gráficos disponibilizados pelo “odin” e que devem ser utilizados pelo “asgard”.

Relativamente ao “asgard”, este componente contém o código relativo à lógica de negócio⁴ associado aos componentes gráficos do “odin” acedidos através do componente “framework-pfh”. É no “asgard” que são criadas as interfaces gráficas completas das páginas disponibilizadas ao utilizador.

⁴ Área clínica.

Relacionado com o tema de internacionalização e localização de *software* e com a reconversão de código Flash/Flex para HTML5, o ALERT® PFH comunica com alguns componentes para aceder a determinados conteúdos (Figura 28).

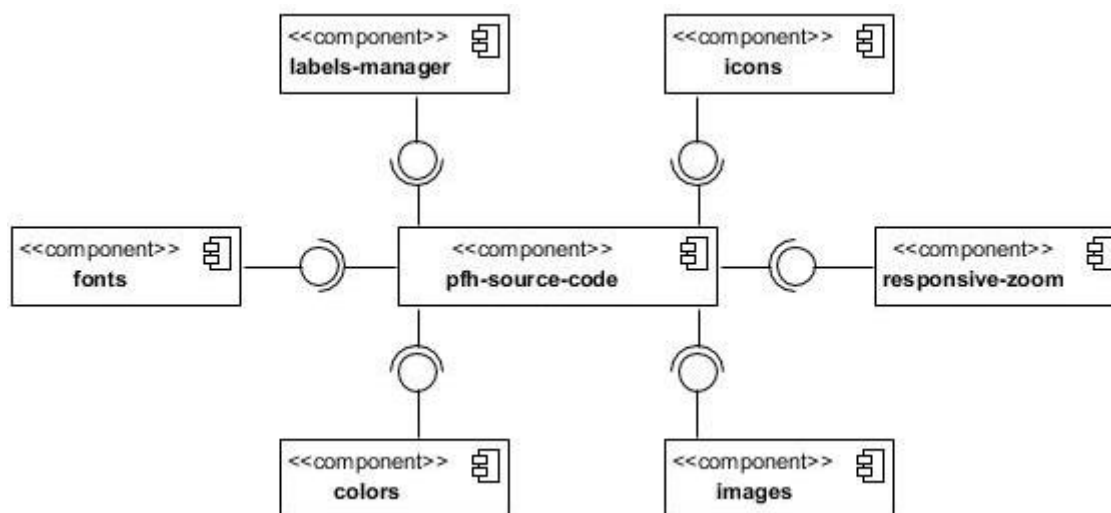


Figura 28 – ALERT® PFH e componentes relacionados

O ALERT® PFH comunica com os componentes “icons” e “images” para aceder aos ícones e às imagens, respetivamente, utilizados para serem disponibilizados na solução de *software*.

O componente “colors” possui toda a paleta de cores que é utilizada nas interfaces do utilizador do ALERT® PFH. Já o componente “fonts” possui todas as fontes necessárias para disponibilizar a solução de *software* em diversas línguas.

Relativamente ao componente “responsive-zoom”, este é responsável por não permitir que o projeto possua um desenho responsivo de forma a se adaptar a diferentes monitores automaticamente, ou seja, resoluções de ecrã diferentes. Isto deve-se ao facto de que, no atual momento, ainda se possui código Flash/Flex no produto, pelo que quando todo o código for HTML5 este componente deixará de existir. Tal acontece devido a uma limitação imposta pelo Flash que não permite que os componentes gráficos se adaptem individualmente ao tamanho do monitor (Owen, 2018). Assim, no caso do ALERT® PFH, o tamanho da interface gráfica por defeito é de 1280x1024 *píxeis*, pelo que o “responsive-zoom” irá aumentar ou diminuir toda a página de igual forma para que esta seja disponibilizada com as dimensões adequadas no ecrã do utilizador. O HTML5 já não possui esta limitação, pelo que é necessário que todos os componentes HTML5 que vão sendo criados, não difiram em termos de tamanho dos componentes ainda em Flash/Flex, permitindo assim a existência dos dois em conjunto.

Por fim, o componente “labels-manager” (Figura 29) permite a criação de ficheiros de tradução que são utilizados nesta solução de *software*.

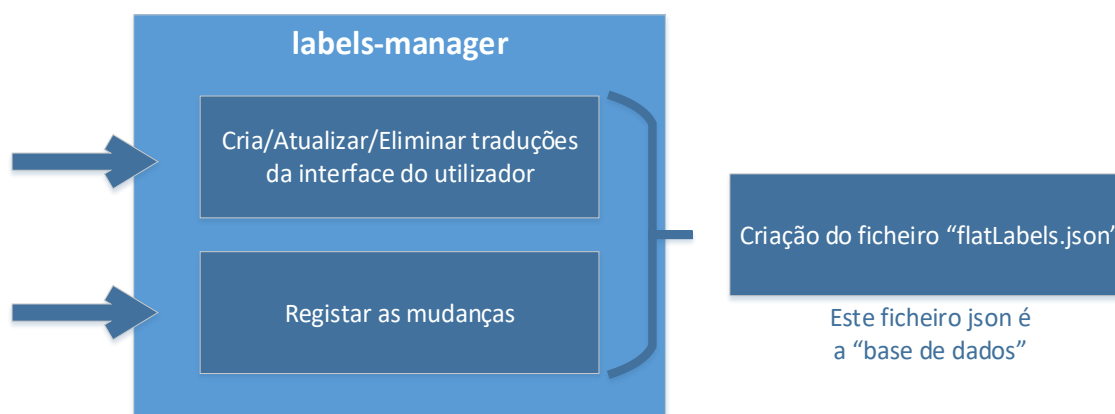


Figura 29 – Labels-manager

Antes do atual momento de reconversão de código Flash/Flex para HTML5, as traduções necessárias para disponibilizar o ALERT® PFH em diversas línguas encontravam-se numa base de dados e eram acedidas consoante o código que as identificava. Contudo, com a nova tecnologia usada para o desenvolvimento do produto, recorre-se a ficheiros estáticos (no caso ficheiros JSON), para o acesso às mesmas.

Assim, para corresponder à necessidade de criação dos ficheiros com as traduções foi desenvolvido o projeto “labels-manager”, onde todas elas se encontram definidas no ficheiro “flatLabels.json” como indica a Figura 29. Este projeto era apenas utilizado pela equipa de desenvolvimento de *software* da empresa. Em termos de tecnologias, o “labels-manager” é desenvolvido em JavaScript e recorre às ferramentas *Bower* e *NPM* para instalar as dependências do projeto, sendo executado através do *Node.js*.

Para inserir ou atualizar uma tradução deve ser realizado o seguinte processo (Figura 30) (no caso de eliminação de uma *label*⁵ os passos 1 a 3 são omitidos):

1. Um elemento da equipa de desenvolvimento de *software* entra em contacto com a equipa de traduções da empresa para pedir a tradução de conteúdos, indicando a interface do utilizador onde pretende inserir a *label* e informações do contexto relativamente ao conteúdo da *label*, para que ocorra uma tradução adequada;
2. A equipa de traduções analisa o pedido efetuado e cria as devidas traduções;
3. Depois das traduções criadas, envia as mesmas para o elemento da equipa de desenvolvimento que efetuou o pedido;
4. O elemento da equipa de desenvolvimento acede ao “labels-manager” e faz as alterações necessárias para realizar a ação que pretende (criar ou alterar ou eliminar uma *label*);
 - 4.1 A aplicação pede que o utilizador grave as alterações efetuadas;
5. O elemento da equipa de desenvolvimento confirma as alterações que efetuou;
 - 5.1 As alterações são armazenadas no ficheiro “flatLabels.json” existente nesta aplicação.

⁵ Uma *label* é identificada através de um identificador único.

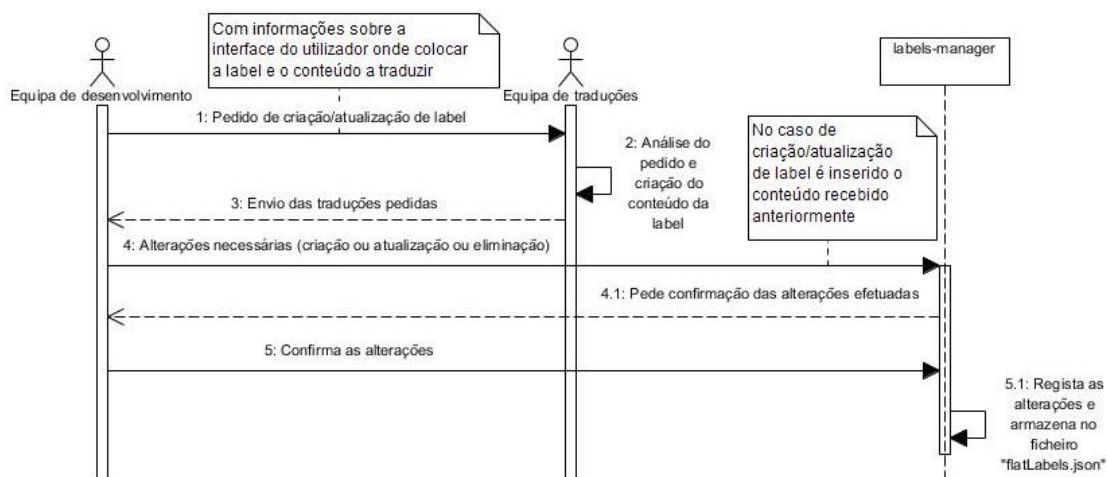


Figura 30 – Processo de criação/edição/eliminação de *labels*

O ALERT® PFH acede, posteriormente, ao ficheiro “flatLabels.json”, realizando uma cópia do mesmo para próprio uso.

Para que as traduções sejam devidamente disponibilizadas no ALERT® PFH, é necessário que este produto possua os seguintes ficheiros:

- “flatLabels.json” – é o ficheiro proveniente do projeto “labels-manager” que possui todos os identificadores únicos e respetivas traduções para diversas línguas;
- Ficheiros “i18n.*.json” – estes ficheiros apenas incluem os identificadores únicos para as traduções que necessitam de ser utilizadas em determinada interface gráfica; o símbolo “*” representa o nome da interface gráfica (p. ex. “i18n.generic.json”) (Extrato de código 3);
- Ficheiros finais “*_LOCAL.json” – estes ficheiros são o resultado da junção entre os identificadores únicos definidos em cada ficheiro “i18n.*.json” e as respetivas traduções existentes no “flatLabels.json”; são criados diversos ficheiros finais para uma mesma interface gráfica, sendo que cada ficheiro final possui as traduções para apenas uma língua; a junção é realizada através de um executador de tarefas de JavaScript – Grunt⁶ (GruntJS, 2019); o símbolo “*” no nome do ficheiro corresponde ao nome da interface gráfica e o “LOCAL” ao local/língua das traduções existentes no ficheiro (p. ex. “biometric.generic_pt_PT.json”) (Figura 31 e Extrato de código 4).

```
[
  "core.login.username",
  "core.login.password",
  "core.login.password.info"
]
```

Extrato de código 3 – Conteúdo do ficheiro “i18n.generic.json” com identificadores únicos de traduções

⁶ A descrição desta tecnologia encontra-se no Anexo C.



Figura 31 – Ficheiro “i18n.generic.json” e respetivos ficheiro gerados

```
{
  "0":{
    "core.login.username": "User",
    "core.login.password": "Password",
    "core.login.password.info": "Please enter your username and
password."
  }
}
```

Extrato de código 4 – Conteúdo do ficheiro gerado em inglês

Tendo os ficheiros finais criados, consoante o local/língua definido no produto, são disponibilizadas as traduções correspondentes.

Algo que também é importante referir é a forma como pode ser escolhido o local/língua em que se deve disponibilizar o *software* (Figura 32). Ao iniciar o produto, por defeito, deve existir um local definido tendo em conta o país onde o produto é disponibilizado. Posteriormente podem existir outras formas de configurar o local, sendo que um utilizador poderá escolher o local/língua em que pretende consultar o produto, a própria instituição médica onde o produto é disponibilizado pode especificar o local a utilizar em todo o *software* ou para cada uma das soluções do produto é escolhido um determinado local.

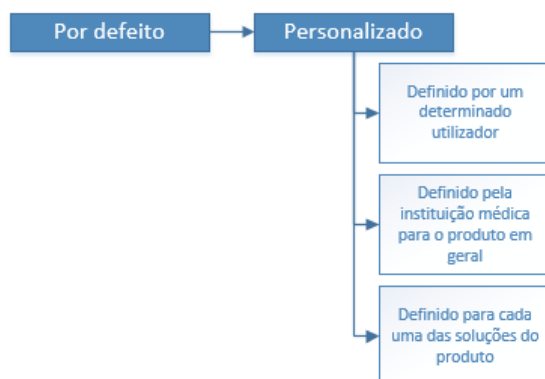


Figura 32 – Escolha do local/língua

4.3 Melhorias a efetuar

Apesar de já haver uma estrutura relacionada com a internacionalização e localização do produto ALERT® PFH, é necessário proceder a algumas melhorias de forma a que se consiga implementar um processo adequado ao problema e dar resposta aos requisitos definidos na secção 3.3.

De seguida, encontram-se apresentadas as melhorias aos produtos da empresa. Inicialmente são definidas as alterações ao projeto “labels-manager” que trata das traduções do ALERT® PFH e posteriormente as alterações a outros projetos relacionados com diferentes problemas de internacionalização e localização, bem como ao próprio ALERT® PFH para que este se adapte às modificações.

4.3.1 Melhorias ao projeto “labels-manager”

De forma a dar resposta a alguns dos requisitos definidos na secção 3.3 e a permitir que o cliente defina as suas próprias traduções, é necessário efetuar-se um conjunto de alterações ao projeto “labels-manager” para que este cumpra com tais requisitos.

Como referido na secção 4.2 do presente documento, o “labels-manager” é utilizado apenas pela equipa de desenvolvimento da empresa. Contudo, este projeto deve ser utilizado por mais equipas para que seja possível otimizar o processo de criação de traduções.

Assim, as equipas que possuem interesse na criação de traduções são:

- Equipa de desenvolvimento de *software* – é a equipa que desenvolve todas as interfaces visuais que serão disponibilizadas aos clientes; estas interfaces disponibilizam conteúdo textual pelo que é necessário que exista traduções desse mesmo conteúdo; no “labels-manager” devem definir um identificador único para cada uma das traduções que necessitam, com o intuito de as utilizar adequadamente;
- Equipa de traduções – tendo em conta as necessidades da equipa de desenvolvimento, a equipa de traduções é responsável por criar as devidas traduções para o produto em várias línguas;
- Equipa de operações – esta equipa é a que lida com os pedidos dos clientes; recebe pedidos por parte destes, para serem criadas traduções personalizadas apenas para eles; assim, a equipa tem o interesse de inserir estas novas traduções no projeto “labels-manager”, gerar os ficheiros finais “*_LOCAL.json”⁷ que possuam as traduções dos clientes e enviar-lhes tais ficheiros.

Cada uma destas equipas é, então, responsável por realizar determinadas tarefas que até então não eram disponibilizadas neste projeto, pelo que as melhorias seguintes devem ser executadas:

1. É necessário que o “labels-manager” possua vários ficheiros “flatLabels.json”, consoante cada uma das versões do produto ALERT® PFH, pois a empresa possui vários

⁷ Nome do ficheiro explicado na secção 4.2 do documento.

- clientes e estes não utilizam todos a mesma versão; é necessário também que seja possível editar cada um destes ficheiros;
2. Disponibilizar uma interface de utilizador para as traduções da própria empresa de forma a que a equipa de traduções consiga realizar as mesmas quando requisitado;
 3. Permitir que sejam criados identificadores únicos para as traduções, estabelecidos pela equipa de desenvolvimento de *software*;
 4. Disponibilizar uma interface de utilizador para que a equipa de operações insira as traduções pedidas pelos clientes; deve ser possível, na mesma interface, consultar as traduções definidas pela empresa como forma de comparação;
 5. Permitir a pesquisa por termos que se encontrem nas traduções já existentes;
 6. Permitir a pesquisa por uma tradução completa nas traduções já existentes;
 7. Permitir a substituição dos termos/tradução completa pesquisados por outros;
 8. Comunicação, através de uma API REST, com uma base de dados específica para armazenar as traduções dos clientes;
 9. Criação de uma interface de utilizador para criação de clientes (o cliente deve ter os seguintes atributos: número identificador de cliente, nome, versão do ALERT® PFH utilizada e traduções);
 10. Disponibilizar a possibilidade de alteração da versão do ALERT® PFH utilizada pelo cliente, sem que sejam perdidas todas as traduções anteriormente criadas;
 11. Eliminação de clientes da base de dados e respetivas traduções⁸;
 12. Existência de um repositório para armazenar as diversas versões do ficheiro “flatLabels.json”;
 13. Possibilidade de criação dos ficheiros finais “*_LOCAL.json”⁹ de traduções para cada uma das interfaces gráficas disponibilizadas no ALERT® PFH;
 14. *Download* da pasta que possui os ficheiros finais “*_LOCAL.json” criados com o intuito de que esta seja enviada aos clientes para utilização;
 15. *Download* do ficheiro “flatLabels.json” com atualizações de tradução para inserção no repositório próprio.

Pelas melhorias propostas, é possível verificar que existem duas formas de armazenamento das traduções (base de dados acedida através de uma API REST para as traduções dos clientes e repositório para as traduções realizadas pela empresa). Isto deve-se ao facto de serem gerados os ficheiros finais “*_LOCAL.json” das interfaces gráficas, com as traduções feitas pela empresa, aquando do processo de produção. Estas traduções são enviadas para todos os clientes da empresa, daí estar inserido neste processo. Recorre-se a uma tarefa Grunt que acede ao repositório com os ficheiros “flatLabels.json” para a geração dos ficheiros finais como se referiu na secção 4.2. Esta tarefa Grunt encontra-se num projeto individual, “grunt-ux-i18n”, de forma a que seja possível ser reutilizada. Também, para geração destes ficheiros finais, o “labels-manager” necessita de ter conhecimento dos ficheiros com os identificadores únicos de cada uma das traduções necessárias para as interfaces gráficas, pelo que deve comunicar com o

⁸ Pode ser necessário no caso de o cliente não querer ter mais traduções por ele personalizadas.

⁹ Nome do ficheiro explicado na secção 4.2 do documento.

ALERT® PFH para obtenção de tal conhecimento (o nome destes ficheiros devem iniciar por “i18n.”, seguido do nome da interface gráfica a que correspondem e com extensão “.json”¹⁰).

Como não se deve ter nenhuma referência a personalizações dos clientes no processo de produção do produto ALERT® PFH, os ficheiros finais, que incluem as traduções dos clientes, não são gerados durante este processo, mas sim através da ferramenta “labels-manager”. Assim, recorre-se a uma base de dados para armazenar estas traduções. É de salientar que como nem todas as traduções são personalizadas pelos clientes, os ficheiros finais gerados devem possuir tanto as traduções realizadas pela empresa como as traduções personalizadas dos clientes.

Para finalizar estas alterações a desenvolver, relacionadas com o projeto “labels-manager”, é apresentado, de seguida, um diagrama de componentes com todas as relações que devem existir entre os diferentes projetos (Figura 33):

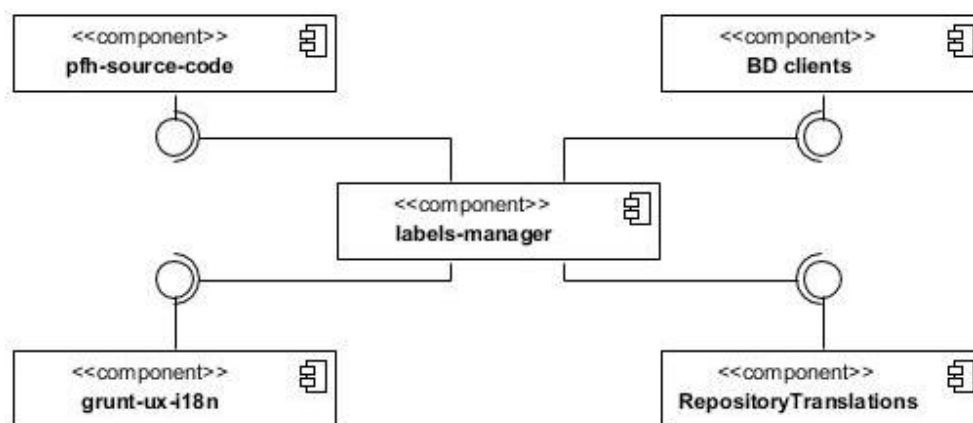


Figura 33 – Diagrama de componentes do “labels-manager” e suas relações

4.3.2 Outras melhorias

Para além do projeto “labels-manager” que gere as traduções do produto ALERT® PFH, existem outras melhorias que devem ser realizadas para que o produto esteja corretamente adaptado a outras culturas.

Assim, relacionado com as fontes de texto disponibilizadas no produto, é necessário se proceder às seguintes alterações:

1. Definir, por defeito, as fontes a utilizar para cada uma das línguas;
2. Permitir que um cliente altere o tipo de fonte de texto que é utilizado no produto, de entre as opções pela empresa disponibilizadas;
3. Armazenar as configurações definidas pelos clientes numa base de dados;

¹⁰ Como referenciado na secção 4.2 do documento.

4. Retirar, do ALERT® PFH, todas as referências a personalizações dos clientes no que a fontes de texto diz respeito;
5. Utilizar as definições dos clientes para disponibilização das fontes (nomeadamente quando é realizado o *login* por parte deste).

Relativamente a estas alterações é importante referir que, devido à fase atual de reconversão de código Flash para HTML5, existem identificadores de fontes de texto específicas para cada uma das tecnologias, mas que disponibilizam o mesmo resultado. Portanto, o cliente ao alterar a fonte para o HTML5, é alterada a fonte para o Flash automaticamente, e vice-versa.

Com o intuito de resolver outros problemas relacionados com o tema deste documento, deve-se recorrer aos ficheiros JavaScript disponibilizados pela ferramenta Angular Locale (secção 2.1.7.11). Foi esta a ferramenta que apresentou melhores resultados na Análise de Valor (secção 3.5.4), daí ser a utilizada para auxiliar o processo de internacionalização e localização.

Estes ficheiros, permitem utilizar diferentes propriedades, como o caso da formatação de datas (dia, mês, dia curto, mês curto, data completa, data longa, data média, data curta, tempo, etc.) e a formatação de números (símbolo monetário, separador decimal, separador dos milhares, etc.).

Cada um destes ficheiros representa um local do globo e possui as propriedades que lá devem ser utilizadas. Por exemplo, o ficheiro “angular-locale-pt-pt.js” (Anexo E) possui as propriedades a utilizar para a língua portuguesa em Portugal. O ficheiro “angular-locale-pt-br.js” já tem as propriedades que devem ser utilizadas quando o produto é disponibilizado em português, mas no Brasil.

Assim, é necessário que o ALERT® PFH utilize as configurações existentes nestes ficheiros e que se adapte consoante a mudança do local onde o produto é disponibilizado. Logo, é preciso realizar as devidas alterações neste produto para que fique compatível com tais ficheiros.

Apesar de a ferramenta Angular Locale disponibilizar o uso destes ficheiros para personalização do produto em questão, é necessário que o ALERT® PFH aceda aos ficheiros consoante a língua/local que o utilizador que faz *login* define para utilização. Para tal, é necessário se recorrer ao uso da biblioteca Angular Dynamic Locale (Angular Dynamic Locale, 2018) para que, em tempo de execução do produto, o local seja alterado.

Para além dos ficheiros provenientes do Angular Locale e a possibilidade de alteração do local, os próprios clientes da empresa podem querer personalizar determinadas propriedades, como o caso do formato da data a disponibilizar. Assim, é necessário armazenar as personalizações do cliente numa base de dados e, posteriormente, aceder a esses dados através de um serviço REST criado para o efeito.

Em termos de personalizações que são permitidas que o cliente configure temos as seguintes propriedades:

- Separador decimal;

- Separador dos milhares;
- Símbolo monetário;
- Formato longo da data;
- Formato médio da data;
- Formato curto da data.

Todas estas personalizações, onde se deve incluir também a personalização das fontes por parte dos clientes, devem ser armazenadas numa tabela da base de dados destinada às configurações realizadas por parte do cliente com os respetivos valores personalizados. Por exemplo, inserir para o cliente com identificador “123”, a configuração “DECIMAL_SYMBOL” correspondente ao separador decimal o valor “,”.

Posteriormente, através de um serviço REST, serão acedidas estas configurações e alterados os valores provenientes dos ficheiros do Angular Locale, bem como as configurações das fontes, para que sejam utilizadas as exigências dos clientes.

Estes valores devem ser alterados quando um cliente faz login na aplicação, como é demonstrado no diagrama de sequência da figura seguinte (Figura 34):

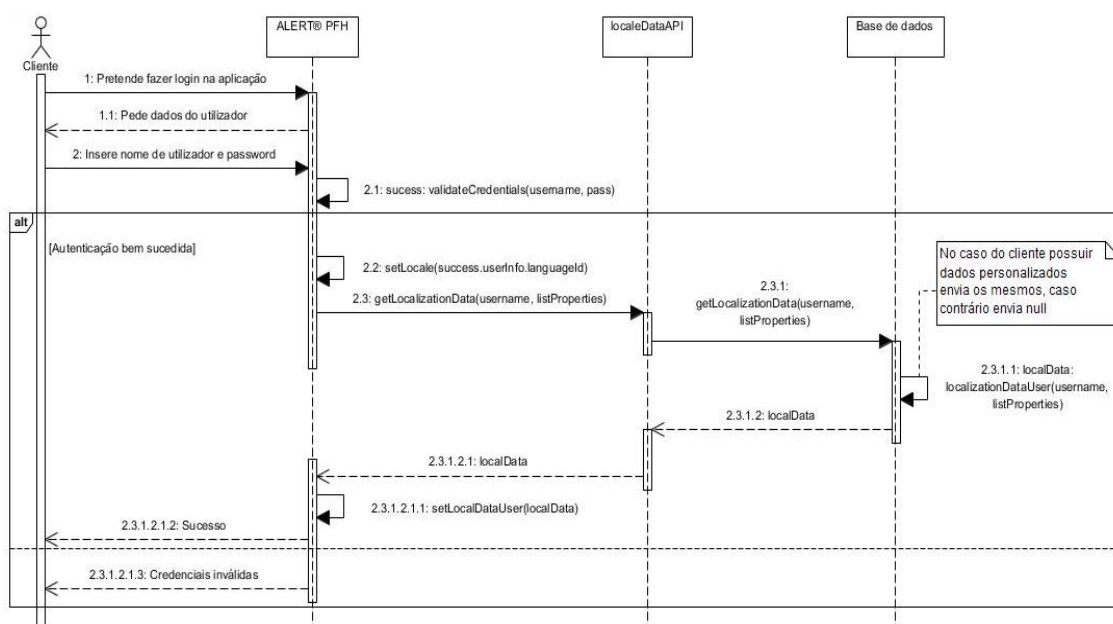


Figura 34 – Diagrama de sequência representativo do login de um cliente

5 Desenvolvimento

Tendo em conta a Análise e Desenho desenvolvidos no capítulo anterior (capítulo 4), este capítulo descreve o processo de desenvolvimento da solução. Inicialmente é abordado o desenvolvimento relacionado com o projeto “labels-manager”, ou seja, relacionado com as traduções do produto. Posteriormente é apresentado o desenvolvimento da solução relacionado com os demais problemas de internacionalização e localização no âmbito do ALERT® PFH.

5.1 Desenvolvimento do projeto “labels-manager”

Como referido na secção 4.3.1 do presente documento, existem melhorias que devem ser realizadas para cumprir com as necessidades de criação de traduções para o produto ALERT® PFH. Portanto, de seguida, são apresentados os desenvolvimentos efetuados ao “labels-manager”, sendo que esta apresentação se encontra dividida consoante as várias páginas de interação com o utilizador, criadas para dar resposta às melhorias identificadas.

Além disso, apresenta-se as alterações feitas ao ALERT® PFH para integração das traduções, bem como um exemplo prático da solução desenvolvida.

5.1.1 Página inicial do “labels-manager”

Inicialmente, utilizador deve escolher a versão do ALERT® PFH à qual pretende editar as traduções (Figura 35). Uma das opções de escolha é a versão de desenvolvimento, a qual corresponde a novas criações de interfaces gráficas¹¹ que ainda não estão disponíveis nos clientes e é, maioritariamente, escolhida quer pelos elementos da equipa de desenvolvimento de *software*, quer pelos elementos da equipa de tradução. Quanto às demais versões, estas são necessárias pois nem todos os clientes utilizam a mesma versão do ALERT® PFH.

¹¹ Interfaces gráficas convertidas para HTML5.

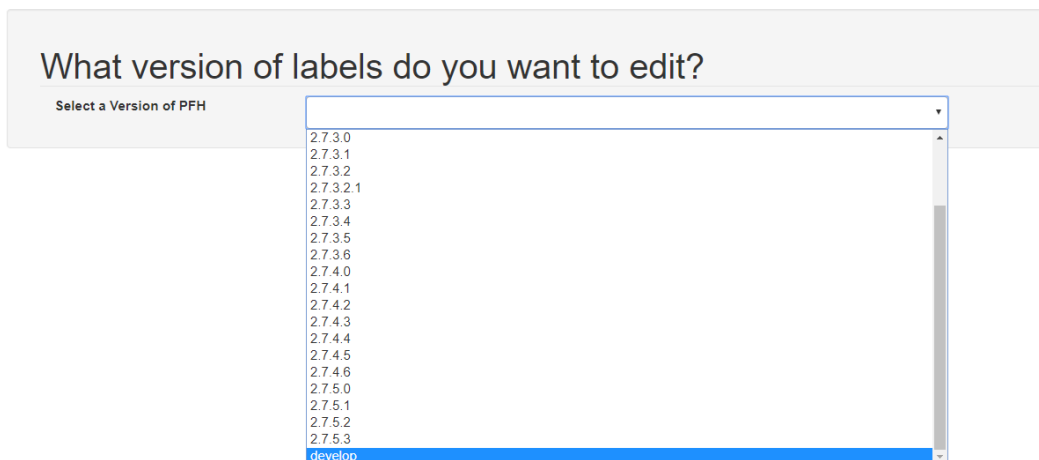


Figura 35 – Escolha da versão do ALERT® PFH

Todas estas versões estão armazenadas num ficheiro JSON que é atualizado sempre que é lançada uma nova versão do ALERT® PFH. Este ficheiro possui a seguinte estrutura:

```
"2.7.5.3":{
  "versionHTML": "4.0.2",
  "versionLabels": "1.0.9"
},
"develop":{
  "versionHTML": "develop",
  "versionLabels": "develop"
}
```

Extrato de código 5 – Exemplo com a relação entre versões

Verificando o Extrato de código 5, para cada uma das versões do ALERT® PFH estão associadas outras duas:

- Versão do código HTML utilizado – “versionHTML” – versão utilizada para aceder aos ficheiros com os códigos únicos para cada uma das traduções necessárias nas interfaces gráficas do produto;
- Versão do ficheiro das traduções – “versionLabels” – ficheiro que possui as traduções criadas pela empresa para a versão do ALERT® PFH correspondente.

É necessário ter conhecimento destas duas versões uma vez que o “labels-manager” precisa de aceder às mesmas de forma a disponibilizar as suas funcionalidades corretamente.

Para se ter acesso aos conteúdos de cada uma das versões recorreu-se a tarefas Grunt¹² para realizar um clone dos ficheiros necessários. No caso, as tarefas Grunt em causa são denominadas de “watch” e “gitclone”. A tarefa “watch” é utilizada para saber qual a versão do ALERT® PFH escolhida pelo utilizador de modo a executar a tarefa “gitclone” com as versões

¹² Exemplos de tarefas Grunt definidas no Anexo C.

corretas do HTML e das traduções. “gitclone” é utilizada para clonar os ficheiros necessários consoante a versão escolhida. Esta clonagem é realizada a dois repositórios. Um com o código HTML e o outro com as várias versões do ficheiro “flatLabels.json”.

Assim, as melhorias números 1 e 12 definidas na secção 4.3.1 encontram-se resolvidas.

5.1.2 Página central do “labels-manager”

Tendo a versão escolhida, e as tarefas Grunt executadas corretamente, o utilizador é redirecionado para a página central do “labels-manager” (Figura 36).

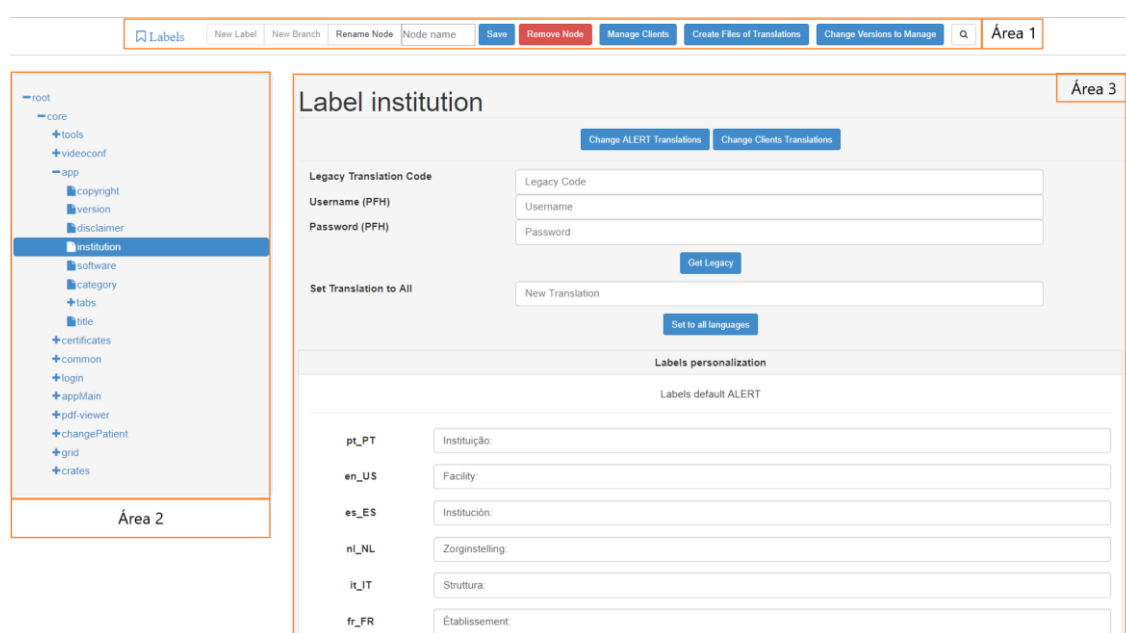


Figura 36 – Página central do “labels-manager”

Esta página pode ser dividida em três grandes áreas:

- Área 1 – cabeçalho da página; onde se encontram alguns dos botões de ações disponibilizados;
- Área 2 – árvore com os identificadores únicos para cada uma das traduções;
- Área 3 – local para inserção das traduções.

A área 1 é constituída por diversos botões. Cada um deles é responsável pela execução de determinada tarefa.



Figura 37 – Botões de criação/alteração de identificadores de traduções

Os botões presentes na Figura 37 são responsáveis pela gestão de todos os identificadores únicos e encontram-se apresentados na lista seguinte:

- “*New Label*” – responsável por criar um identificador único ao qual serão posteriormente associadas as traduções.
- “*New Branch*” – responsável por criar um agregador onde serão colocados vários identificadores únicos das traduções; normalmente, este agregador recebe o nome da interface gráfica do ALERT® PFH onde as traduções serão utilizadas;
- “*Rename Node*” – responsável por alterar o nome de um agregador ou de um identificador único;
- Campo *input* – o campo *input*, com o *placeholder* “*Node name*”, é onde o utilizador irá inserir o nome a ser utilizado quando se clica em cada um dos botões descritos anteriormente;
- “*Save*” – grava todas as alterações efetuadas até ao momento relacionadas com a gestão das traduções;
- “*Remove Node*” – responsável por eliminar agregadores e respetivos identificadores únicos que possui ou então apenas um indetificador único selecionado.

Os botões descritos anteriormente estão associados às alterações que acontecem à Área 2 da Figura 36.



Figura 38 – Restantes botões do cabeçalho da página central do “labels-manager”

Os botões presentes na Figura 38 possuem as seguintes funcionalidades:

- “*Manage Clients*” – redireciona o utilizador para a página de gestão de clientes;
- “*Create Files of Translations*” – redireciona o utilizador para a página de criação de ficheiros JSON com as traduções criadas e posterior *download* dos mesmos;
- “*Change Versions to Manage*” – redireciona o utilizador para a página inicial, onde é escolhida a versão do ALERT® PFH;
- Lupa – utilizado quando o utilizador pretende pesquisar por traduções existentes; depois de efetuar a pesquisa é permitido ao utilizador que troque o termo pesquisado por um outro em todos os resultados obtidos.

A Área 2, representada na Figura 36, apresenta uma árvore onde são colocados todos os identificadores únicos das traduções.

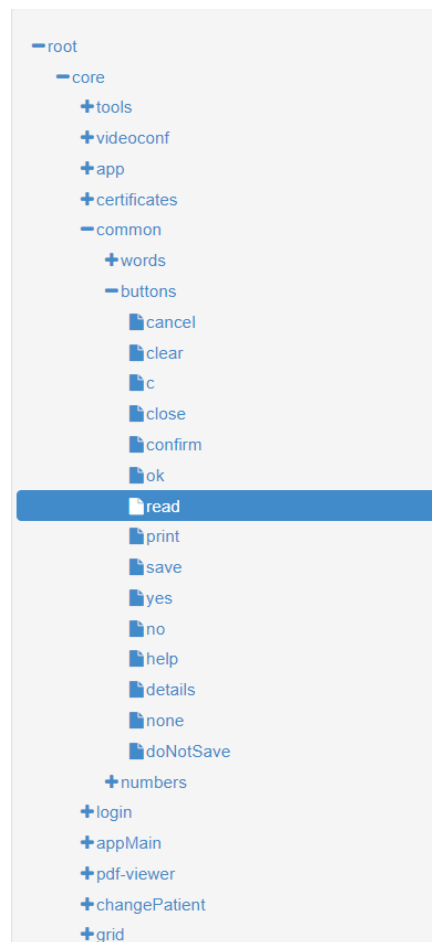


Figura 39 – Árvore com os identificadores únicos para cada uma das traduções

Analisando a Figura 39, é importante destacar que os símbolos “+” e “-” significam agregadores de identificadores únicos de traduções. Quando aparece o símbolo “+” pode-se estender a árvore para verificar o seu conteúdo, sendo que quando aparece o símbolo “-” é exatamente para realizar o oposto. Dentro de cada um destes agregadores podem existir mais agregadores ou então identificadores únicos para as traduções.

Por exemplo, o destacado na Figura 39, “*read*”, representa o identificador único para as traduções a serem colocadas no botão de “*Read*” que pode existir em diversas interfaces gráficas do ALERT® PFH. O seu caminho, ou seja, o seu identificador único é, então, “*root.core.common.buttons.read*”. É este o caminho que os ficheiros “*i18n.*.json*”¹³ presentes no ALERT® PFH possuem para aceder às respetivas traduções (no caso, “*root.*” é omitido por ser a raiz da árvore e comum a todos os identificadores únicos).

Ao ser selecionado um destes identificadores únicos, é disponibilizado ao utilizador do “*labels-manager*” a Área 3 representada na Figura 36.

¹³ Ficheiro apresentado na secção 4.2 do documento.

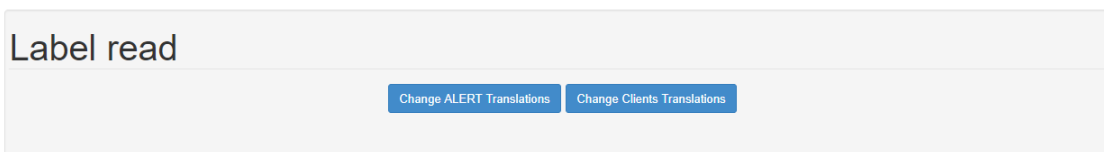


Figura 40 – Seleção para o tipo de traduções a inserir

Inicialmente, o utilizador necessita de escolher que tipo de traduções quer inserir para o identificador único selecionado (*“read”*) (Figura 40). A primeira opção (*“Change ALERT Translations”*) é para criação de traduções feitas pela própria empresa. A segunda opção, *“Change Clients Translations”* é para inserir traduções personalizadas pelos clientes.

Antes da continuação da explicação desta Área 3 é importante esclarecer como estão a ser geridas estas traduções dos clientes. Estas traduções estão a ser armazenadas numa base de dados não relacional *“MongoDB”* (Pinto, 2015). *“MongoDB”* foi a opção escolhida pois permite armazenar grandes quantidades de dados (Pinto, 2015). Para além disso, o facto de possuir alto desempenho e não ser necessário para o desenvolvimento da solução a criação de um esquema de base de dados para armazenar este conteúdo de traduções, influenciou esta escolha (Pinto, 2015). Esta base de dados, para cada cliente, cria um documento com o identificador do cliente, o nome, a versão do ALERT® PFH que utiliza e as respetivas traduções. Para comunicação com esta base de dados foi criada uma API REST que disponibiliza diferentes pedidos:

- Pedidos *get* – obtenção das traduções de um determinado cliente; obtenção da lista de clientes com traduções personalizadas; obtenção da lista de clientes para determinada versão do ALERT® PFH; obtenção da versão do ALERT® PFH para um determinado cliente;
- Pedidos *post* – criação de um novo cliente;
- Pedidos *put* – atualização das traduções para um cliente; atualização da versão do ALERT® PFH utilizada pelo cliente;
- Pedidos *delete* – eliminação de um cliente e suas informações;

De volta à análise da Figura 40, ao clicar no botão *“Change ALERT Translations”* aparece uma área onde se pode inserir as traduções para cerca de 22 línguas (Figura 41).

Legacy Translation Code

Legacy Code

Username (PFH)

Password (PFH)

Set Translation to All

New Translation

Labels personalization

Labels default ALERT

pt_PT	Lido
en_US	Read
es_ES	Leído
nl_NL	Gelezen
it_IT	Ok
fr_FR	Lu
en_GB	Read
ar_KW	Read
de_DE	Translation for German locale for Germanv

Figura 41 – Área de traduções feitas pela empresa

Verificando a Figura 41, constata-se que existe uma área relacionada com as “Legacy Translation Code”. Esta funcionalidade está relacionada com o facto de que antes de reconversão de código para HTML5, muitas das traduções estavam num base de dados. Assim, de forma a facilitar o processo de inserção de traduções, foi criado um serviço que acede à base de dados com as traduções. Este serviço necessita de receber o código identificador da tradução da base de dados, bem como as credenciais de acesso a essa base de dados. A resposta deste serviço permite preencher as traduções para cada uma das línguas (no caso de não existir traduções para alguma língua, esta fica vazia). Também, foi desenvolvida a funcionalidade de, para todas as línguas, preencher os campos das traduções com um mesmo texto (“Set Translation to All” da Figura 41).

Ao clicar no botão “*Change Clients Translations*” da Figura 40, é disponibilizado ao utilizador o seguinte conteúdo (Figura 42):

Label read

Change ALERT Translations

Change Clients Translations

Select a Client

Set Translation to All

New Translation

Set to all languages

Labels personalization

Labels default ALERT

pt_PT

Lido

en_US

Read

es_ES

Leído

nl_NL

Gelezen

it_IT

Ok

fr_FR

Lu

en_GB

Read

ar_KW

Read

Labels for Client

pt_PT

Translation for Portuguese locale for Portugal

en_US

Translation for English locale for United States

es_ES

Translation for Spanish locale for Spain

nl_NL

Translation for Dutch locale for the Netherlands

it_IT

Translation for Italian locale for Italy

fr_FR

Translation for French locale for France

en_GB

Translation for English locale for Britain

ar_KW

Translation for Arabic locale for Kuwait

Figura 42 – Área de traduções para pedidos de clientes

Observando a Figura 42, o utilizador necessita de escolher qual o cliente para o qual pretende adicionar uma tradução. No caso, a lista de clientes que lhe é apresentada apenas corresponde aos clientes que utilizam a versão do ALERT® PFH escolhida no início do uso do “labels-manager”.

Selecionado o cliente é possível personalizar as traduções para cada uma das línguas. Estas traduções inseridas são armazenadas na base de dados do cliente. É de salientar que é mostrado também ao utilizador as traduções definidas pela empresa como termo de comparação.

Tendo estes desenvolvimentos efetuados, as melhorias números 2, 3, 4, 5, 6, 7 e 8 definidas na secção 4.3.1 encontram-se devidamente concluídas.

5.1.3 Página de gestão de clientes

Como outras interfaces gráficas disponibilizadas ao utilizador, temos a que permite a gestão de clientes (Figura 43):

Manage Clients

Create New Client

ID Client

Name Client

PFH Version

ID Client

Name Client

PFH Version

Create Client

Change Client PFH Version

Select a Client

PFH Version

develop

Update App Version

Delete Client

Select a Client

☐

☐

Delete Client

Figura 43 – Página de gestão de clientes

Nesta página de gestão de clientes, é disponibilizado ao utilizador as seguintes operações:

- *“Create New Client”* – o utilizador deve inserir o identificador do cliente, o seu nome e a versão do ALERT® PFH que utiliza; este cliente é criado na base de dados com os dados inseridos;
- *“Change Client PFH Version”* – no caso de um cliente mudar a versão do ALERT® PFH que utiliza este dado deve ser também aqui atualizado;
- *“Delete Client”* – para eliminar um cliente da base de dados e respetivas traduções personalizadas.

Assim, tendo em conta as melhorias definidas na secção 4.3.1, dá-se por concluídas as números 9, 10 e 11.

5.1.4 Página de gestão dos ficheiros JSON com traduções

Por fim, existe uma outra página do “labels-manager” a referir. A que permite a criação e posterior *download* dos ficheiros finais “*_LOCAL.json”¹⁴ com as traduções individuais (Figura 44).

¹⁴ Ficheiro apresentado na secção 4.2 do documento.

Manage File Creation

Create File With Clients Translations

Select Clients

☐ ☐

Generate File

Create File ALERT Translations

Generate & Download File

Download Folder of i18n

Download i18n Folder

© 2019, ALERT Life Sciences Computing, S.A.

Figura 44 – Página de gestão dos ficheiros JSON com traduções

Nesta página, é possível realizar um “merge” entre as traduções definidas pela empresa e as traduções definidas pelos clientes e posteriormente gerar os ficheiros finais com as traduções a serem utilizadas no ALERT® PFH. O resultado deste “merge” pode ser visto no Extrato de código 6:

```
"root.core.common.buttons.read": {
  "idALERT": {
    "pt_PT": "Lido",
    "en_US": "Read",
    "es_ES": "Leído",
    "nl_NL": "Gelezen",
    "it_IT": "Ok",
    "fr_FR": "Lu",
    "en_GB": "Read",
    "pt_BR": "Lido",
    "zh_CN": "已读",
  },
  "idCliente": {
    "pt_PT": "Visto",
    "en_US": "Read!",
    "es_ES": "Leído!",
  },
}
```

Extrato de código 6 – Exemplo de um ficheiro resultante do merge

Verificando o Extrato de código 6, é possível reparar que para um identificador único de traduções (“root.core.common.buttons.read”) existem traduções realizadas pela empresa e outras a pedido do cliente.

Um exemplo de um ficheiro final criado é o seguinte (“core_pt_PT.json”) (Extrato de código 7):

```
{
  "idALERT": {
    "core.app.title": "ALERT® PFH",
```

```

    "core.app.version": "Versão ALERT®",
    "core.common.words.warning": "Aviso",
    "core.common.buttons.close": "Fechar",
    "core.common.buttons.confirm": "Confirmar",
    "core.common.buttons.ok": "ok",
    "core.common.buttons.read": "Lido"
  },
  "idCliente": {
    "core.common.buttons.read": "Visto"
  }
}

```

Extrato de código 7 – Ficheiro “core_pt_PT.json” criado

São criados ficheiros para cada uma das línguas (p. ex. “core_pt_PT.json”, “core_pt_BR.json”, entre outros). No excerto de código anterior (Extrato de código 7) está representado um ficheiro para a língua portuguesa a ser usado quando o ALERT® PFH é utilizado em Portugal. Como existia uma personalização por parte do cliente para uma das traduções esta aparece também no ficheiro.

Relativamente à criação do ficheiro, é necessário ter o conhecimento da versão do código HTML (obtido inicialmente) a que correspondem as traduções. Os ficheiros “i18n*.json”¹⁵ com os identificadores únicos de cada uma das traduções são acedidos e é possível a geração dos ficheiros finais com identificadores únicos e respetivas traduções, através de uma tarefa Grunt (grunt-ux-i18n¹⁶).

Tendo os ficheiros finais gerados é necessário copiar os mesmos para o ALERT® PFH. No caso de serem ficheiros com configurações de clientes esta cópia é-lhes enviada. Assim, só eles têm acesso às suas traduções personalizadas. Se os ficheiros contiverem apenas traduções da empresa recorre-se a uma tarefa Grunt, existente no ALERT® PFH, para cópia dos ficheiros.

Com estes desenvolvimentos, as melhorias números 13, 14 e 15 da secção 4.3.1 são dadas por realizadas.

5.1.5 Alterações ao ALERT® PFH

Em termos de alterações a efetuar no produto ALERT® PFH de forma a aceder às traduções é necessário o seguinte:

- No local onde se pretende inserir a tradução deve-se referir qual o identificador único da mesma (Figura 45);

¹⁵ Ficheiro apresentado na secção 4.2 do documento.

¹⁶ Tarefa apresentada na secção 4.3.1 do documento.

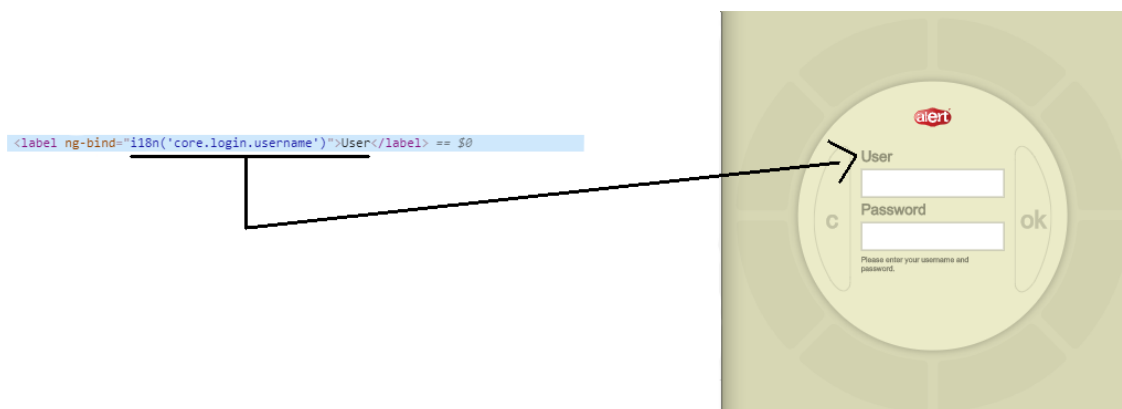


Figura 45 – Uso do identificador único da tradução

- Necessário recorrer a uma *factory*¹⁷ (denominada “i18n”) criada para aceder aos ficheiros das traduções, e, consoante o identificador único da tradução e a língua configurada na aplicação, acede à tradução específica existente nos ficheiros; posteriormente, no controlador da interface gráfica criada é necessário se colocar a propriedade “i18n” a verdadeiro para se aceder às traduções; esta propriedade a verdadeiro permite que os ficheiros com as traduções sejam devidamente carregados;
- É criado um ficheiro “i18n.*.json”¹⁸ onde se colocam os identificadores únicos;
- Por fim, tendo os ficheiros finais gerados com os identificadores únicos e respetivas traduções, a interface gráfica é apresentada corretamente.

É de salientar que no caso de haver traduções definidas pela empresa e traduções definidas por clientes, as traduções de clientes são as que vão ser disponibilizadas. Para tal, quando o cliente faz *login* no produto o seu identificador é verificado e as traduções que ele possui apresentadas. Apesar disso, nem todas as traduções são personalizadas pelos clientes, pelo que as traduções definidas pela empresa são apresentadas nestes casos.

Para apresentar de forma gráfica estas alterações que ocorrem no ALERT® PFH é apresentado, de seguida, um diagrama de sequência (Figura 46):

¹⁷ Gera um objeto ou função que representa um serviço para o resto da aplicação. Este objeto ou função retornada pode ser injetada em qualquer componente da aplicação, como o caso dos controladores (AngularJS, 2019a).

¹⁸ Nome do ficheiro explicado na secção 4.2.

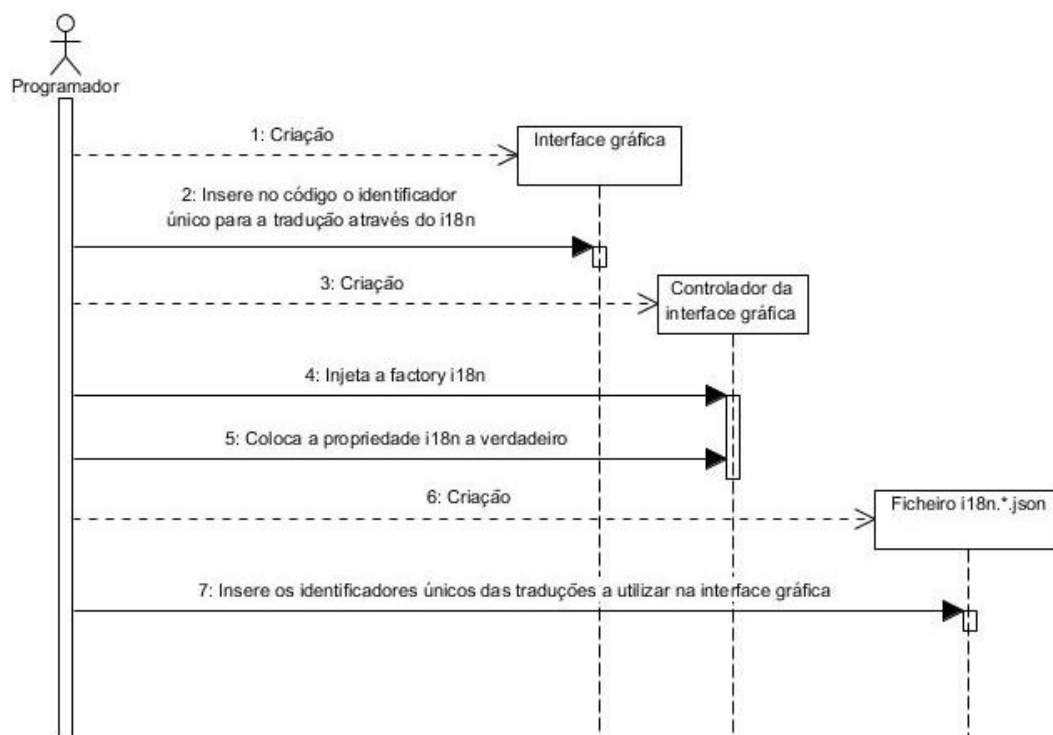


Figura 46 – Alterações ao ALERT® PFH

5.1.6 Exemplo prático – pedido de tradução de cliente

Um cliente, depois de entrar no ALERT® PFH e realizar *login*, depara-se com a seguinte interface gráfica (Figura 47):

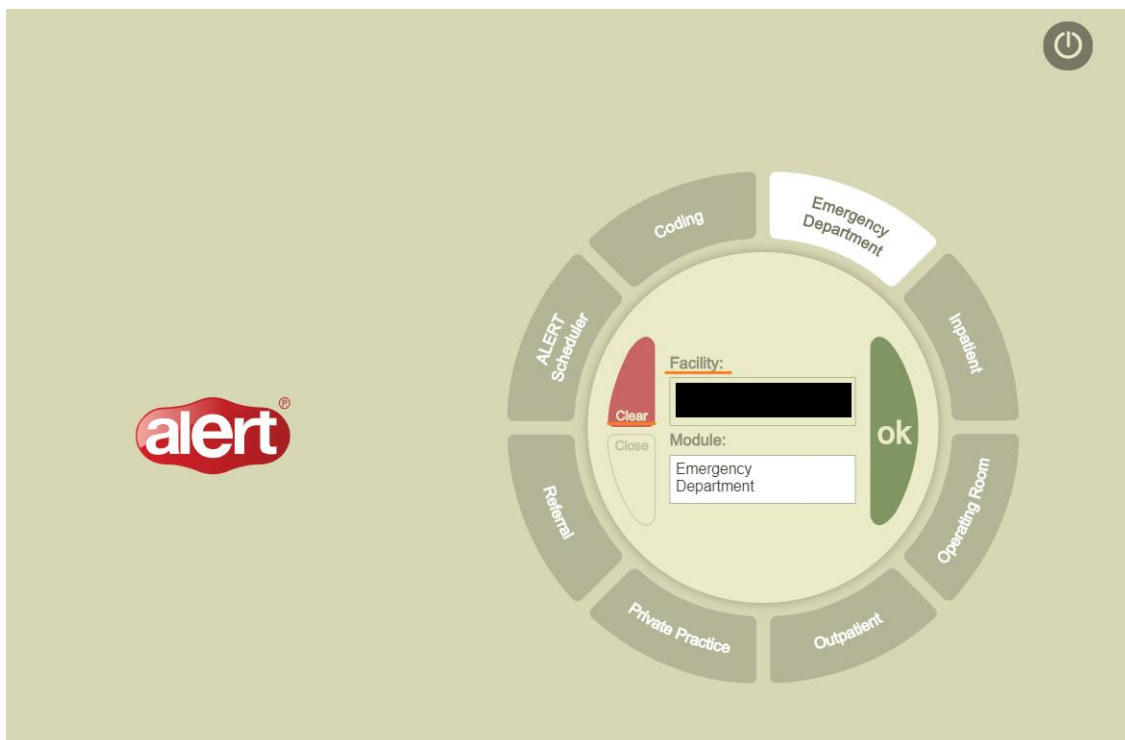


Figura 47 – Interface gráfica de seleção de produto

O cliente pretende alterar a tradução de “Facility:” para “Super Facility” e “Clear” para “CLEAR”. Para tal, é realizado um pedido a um elemento da equipa de operações para que sejam efetuadas tais alterações. O elemento da equipa de operações realiza as seguintes operações (considerando que é a primeira vez que o cliente pede traduções personalizadas) (Figura 48):

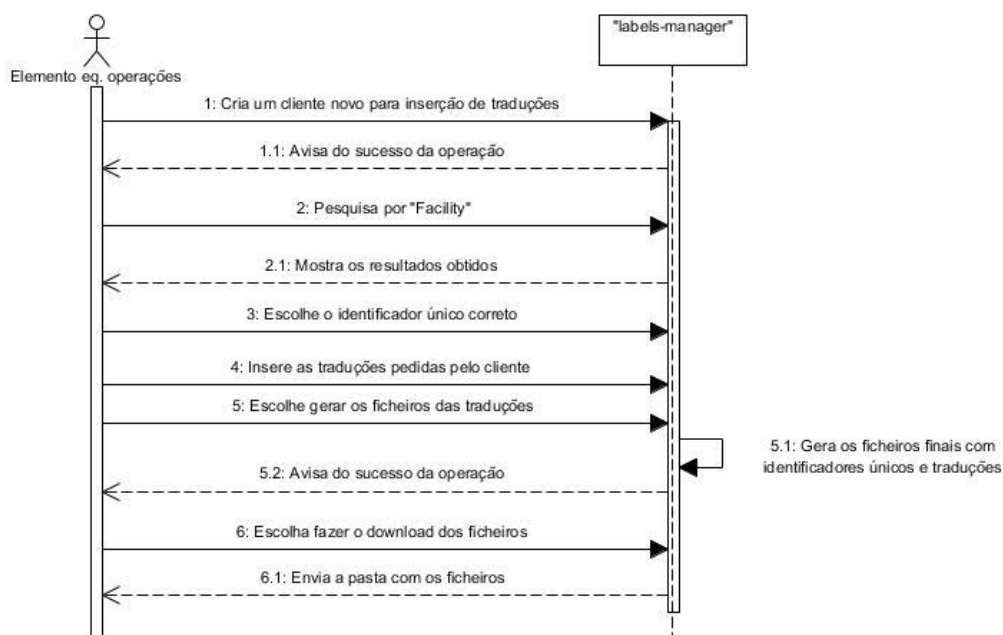


Figura 48 – Diagrama de operações a realizar no “labels-manager”

Obtendo a pasta com todos os ficheiros gerados, o elemento da equipa de traduções coloca a mesma na distribuição do ALERT® PFH do cliente. Tendo esta ação realizada, o cliente apenas necessita de aceder novamente ao produto, realizar o *login* e verificar as traduções pedidas (Figura 49):



Figura 49 – Alterações à interface gráfica de seleção do produto

5.2 Desenvolvimento de outras melhorias

Nesta secção são apresentados os desenvolvimentos efetuados para dar resposta às melhorias definidas na secção 4.3.2.

Inicialmente refere-se a forma como se armazena as características personalizadas pelos clientes. De seguida, expõem-se as alterações realizadas para a definição de fontes de texto no ALERT® PFH. Por fim apresenta-se a integração da ferramenta Angular Locale no produto, pois esta permite dar resposta a vários problemas de internacionalização e localização.

5.2.1 Armazenamento de personalizações de clientes

Relativamente ao desenvolvimento de soluções para outros problemas de internacionalização e localização é importante começar por referir que em alguns casos o cliente pode querer definir características próprias.

Para tal, é necessário que essas características¹⁹ sejam armazenadas numa base de dados para que depois sejam utilizadas. Assim, recorrer-se à base de dados relacional da empresa onde estão algumas das configurações dos clientes para armazenar estes dados relacionados com a localização. Foi, então, necessário criar uma tabela para armazenar todas as diferentes possibilidades de fontes de texto que o ALERT® PFH permite disponibilizar. Estas fontes foram escolhidas tendo em conta que permitem representar todos os caracteres disponibilizados em cada local do globo.

Posteriormente, visto que a tabela onde são armazenadas as configurações dos clientes possuem outras que não as relacionadas com a localização e internacionalização, criou-se uma vista específica para estes novos dados (Extrato de código 8 – alguma lógica de negócio foi omitida):

```
create or replace view v_locale_cfg as
select
    LÓGICA_DE_NEGÓCIO
from TABELA_CONFIGURAÇÕES ct
join TABELA_FONTES vdf on vdf.FONTE = ct.FONTE
where ct.CONFIGURAÇÃO = 'DISPLAY_FONT'
union all
select
    LÓGICA_DE_NEGÓCIO
from TABELA_CONFIGURAÇÕES ct
where ct.CONFIGURAÇÃO in ('DATE_LONG_DATE', 'DATE_SHORT_DATE',
'DATE_MEDIUM_DATE', 'MIL_SYMBOL', 'CURRENCY_SYMBOL')
union all
select
    LÓGICA_DE_NEGÓCIO
from TABELA_CONFIGURAÇÕES_SISTEMA scg where scg.CONFIGURAÇÕES
in( 'DECIMAL_SYMBOL');
```

Extrato de código 8 – Vista da base de dados para armazenar conteúdo do cliente

Com a vista criada, foi necessário produzir uma função que permitisse aceder aos respetivos dados. Esta função deve receber como parâmetros a linguagem que é utilizada no produto, o identificador do cliente e uma lista de configurações às quais queremos obter os dados (Extrato de código 9 – alguma lógica de negócio foi omitida):

```
FUNCTION get_locale_data
(
    CLIENTE          IN professional,
    LINGUAGEM        IN NUMBER,
    CONFIGURAÇÕES   IN table_varchar,
    RESPOSTA         OUT pk_types.cursor_type,
    ERRO             OUT t_error_out
) RETURN BOOLEAN IS
BEGIN
    LÓGICA_DE_NEGÓCIO
EXCEPTION
    WHEN OTHERS THEN
        LÓGICA_DE_NEGÓCIO
```

¹⁹ Características definidas na secção 4.3.2

```
END get_locale_data;
```

Extrato de código 9 – Função para obtenção de dados dos clientes

Tendo a função, foi criado um serviço REST, na linguagem Java, para que o produto ALERT® PFH possa aceder às configurações definidas pelo cliente (Extrato de código 10 – alguma lógica de negócio foi omitida):

```
@ApiOperation(value = "Returns the Localization data of a institution")
@RequestMapping(method = RequestMethod.GET, value = "CAMINHO", produces =
"application/json")
public ResponseEntity<EnterpriseServiceResponseDto<List<LocalizationDTO>>>
getLocalizationData( @RequestHeader(HttpHeaderHelper.ID_CLIENTE)
BigDecimal idProfessionalHeader,
@RequestHeader(HttpHeaderHelper.ID_LINGUAGEM) BigDecimal idLanguageHeader,
@RequestHeader(HttpHeaderHelper.ID_SOFTWARE) BigDecimal idSoftwareHeader,
@RequestHeader(HttpHeaderHelper.ID_INSTITUTION) BigDecimal
idInstitutionHeader, String[] configTable) {
    return ResponseEntityBuilder.buildResponseEntity(
        localizationService.getLocalizationData(idProfessionalHeader,
        idInstitutionHeader, idSoftwareHeader, idLanguageHeader,
        Arrays.asList(configTable)));
}
```

Extrato de código 10 – Função criada no serviço REST

Assim, todas as configurações dos clientes podem ser acedidas através deste serviço. É de salientar ainda que é necessário que o serviço seja chamado sempre que um utilizador realize *login* no ALERT® PFH para que as configurações sejam corretamente configuradas (Extrato de código 11 – alguma lógica de negócio foi omitida):

```
asg$webserviceResources.localeData.getLocalization({configTable: ''},
{configTable:'DISPLAY_FONT, DECIMAL_SYMBOL, MIL_SYMBOL, CURRENCY_SYMBOL,
DATE_LONG_DATE, DATE_MEDIUM_DATE,
DATE_SHORT_DATE'}).then(function(response){
    LÓGICA_DE_NEGÓCIO
})
```

Extrato de código 11 – Chamada ao serviço REST

5.2.2 Implementação da solução relativa às fontes de texto

Relativamente ao problema relacionado com as fontes de texto, foi criado um projeto à parte do ALERT® PFH onde estão definidas num ficheiro, para cada língua, as fontes²⁰ a utilizar. Fontes essas que comportam todos os caracteres utilizados ao redor do globo, tendo em conta a língua do local (Extrato de código 12 – alguma lógica de negócio foi omitida).

```
"1" : {
  "locale" : "pt_PT",
  "font" : {
    "name" : "AlertDefault"
  },
  "legacyFont" : {
    "name" : "FontHolder_Latin"
```

²⁰ As fontes utilizadas foram definidas pela empresa.

```

    }
  },
  "2" : {
    "locale" : "en_US",
    "font" : {
      "name" : "AlertDefault"
    },
    "legacyFont" : {
      "name" : "FontHolder_Latin"
    }
  },
  ...
  "12" : {
    "locale" : "zh_CN",
    "font" : {
      "name" : "WenQuanYiMicroHei"
    },
    "legacyFont" : {
      "name" : "FontHolder_Latin_SCh"
    }
  }, ...

```

Extrato de código 12 – Ficheiro com definição de fontes de texto

Considerando o extrato de código anterior (Extrato de código 12), é possível verificar que para cada *locale* existem dois identificadores de fontes. Um é para ser utilizado no código HTML5 (“*font*”) e o outro no Flash (“*legacyFont*”). Estas configurações são definidas pela empresa, contudo, como o cliente pode querer configurar as fontes com que a aplicação é apresentada foi criado, no mesmo ficheiro, o seguinte (Extrato de código 13 – alguma lógica de negócio foi omitida):

```

"client" : {
  "0" : {
    "locale" : "default",
    "font" : {
      "name" : "AlertDefault"
    },
    "legacyFont" : {
      "name" : "FontHolder_Latin"
    }
  }
}

```

Extrato de código 13 – Configuração de fontes de texto para o cliente

Apesar de estar definido este “*client*” no ficheiro das fontes de texto, este ficheiro é agnóstico a dados reais dos clientes. Ou seja, nenhuma informação configurada pelo cliente é colocada no ficheiro. Com este ficheiro definido, recorre-se a uma tarefa Grunt para clonar o mesmo para o ALERT® PFH. Com o intuito de transformar o ficheiro JSON clonado para um objeto a ser utilizado no produto, recorre-se a uma outra tarefa Grunt criada para o propósito. Como resultado desta tarefa, é criado o seguinte objeto (Extrato de código 14 – alguma lógica de negócio foi omitida):

```

angular.module('odin.font-loader')
.value('FontLoaderConfiguration', Object({ "0": {
  "0": {

```

```

        "locale": "default",
        "font": {
            "name": "AlertDefault",
            "variations": {
                "regular": true,
                "bold": true,
                "italic": true,
                "boldItalic": true
            }
        },
        "legacyFont": {
            "name": "FontHolder_Latin"
        }
    },
    "client": {
        "0": {
            "locale": "default",
            "font": {
                "name": "AlertDefault",
                "variations": {
                    "regular": true,
                    "bold": true,
                    "italic": true,
                    "boldItalic": true
                }
            },
            "legacyFont": {
                "name": "FontHolder_Latin"
            }
        }
    }
});

```

Extrato de código 14 – Objeto com as fontes de texto

Posteriormente, utilizando o serviço REST criado²¹, as configurações do cliente são alteradas para o produto ser apresentado conforme as suas especificações. Isto acontece em tempo real de execução, nomeadamente quando o *login* é realizado pelo cliente. De referir ainda que existem variações disponíveis para o HTML5, nomeadamente a possibilidade de definir texto com características “*bold*” e “*italic*”.

Na Figura 50 são apresentadas as diferenças relativamente à fonte de texto.

²¹ Apresentado na secção 5.2.1



Figura 50 – Diferença de fontes de texto

5.2.3 Integração da ferramenta Angular Locale

Para dar resposta aos restantes problemas recorre-se ao uso do Angular Locale (apresentado na secção 2.1.7.11 do documento). Assim, é necessário aceder ao repositório (AngularJS, 2019b) onde estão disponibilizados cada um dos ficheiros com as características de um local do globo e copiá-los para uma pasta específica do produto ALERT® PFH. Os nomes dos ficheiros devem ser alterados de forma a que o identificador do local do globo corresponda ao identificador da linguagem definido no produto (Extrato de código 15):

```
1 : {locale : "pt_PT", description : "Portugues (Portugal)"}
```

Extrato de código 15 – Identificador para português de Portugal

Por exemplo o ficheiro “angular-locale-pt-pt.js” proveniente do Angular Locale tem de ser renomeado para “angular-locale-pt_PT.js” como se verifica pelo identificador do extrato de código anterior.

Como o Angular Locale não permite que seja alterado, em tempo de execução, o local da aplicação, recorre-se à biblioteca Angular Dynamic Locale para que tal seja possível (Angular Dynamic Locale, 2018). Assim, existe a necessidade de incluir a biblioteca no módulo raiz do projeto (Extrato de código 16):

```
angular.module('root', ['LocalStorageModule', 'tmh.dynamicLocale'])
```

Extrato de código 16 – Inserção da biblioteca Angular Dynamic Locale

Depois, é preciso mudar o caminho para os ficheiros do Angular Locale que a biblioteca Angular Dynamic Locale está à espera (Extrato de código 17):

```
.config(function(tmhDynamicLocaleProvider){
    tmhDynamicLocaleProvider.localeLocationPattern(
        'CAMINHO/i18n/locale/angular-locale_{{locale}}.js');
});
```

```
})
```

Extrato de código 17 – Alteração do caminho para os ficheiros do Angular Locale

Assim, sempre que seja necessário alterar o local do ALERT® PFH, basta, na classe de código suposta recorrer à biblioteca, como por exemplo quando é feito o *login* do utilizador (Extrato de código 18):

```
tmhDynamicLocale.set(asg$locale.getLocale(locale));
```

Extrato de código 18 – Alteração do local através da biblioteca Angular Dynamic Locale

Portanto, já é possível utilizar todas as características definidas pelos ficheiros do Angular Locale. Existem duas formas para utilizar as mesmas: ou através da utilização direta das características (\$locale) ou através de um filtro de formatação (p. ex. para formatar as datas - \$filter). Para tal, é necessário se injetar \$locale e \$filter nas nossas classes (Extrato de código 19 – alguma lógica de negócio foi omitida).

```
angular.module('NOME_MÓDULO')
  .factory('NOME_FACTORY', ['$locale', '$filter',
    function (
      $locale, $filter) { LÓGICA_DE_NEGÓCIO }
```

Extrato de código 19 – Inserção das variáveis \$locale e \$filter

Para se utilizar o \$filter basta usar o seguinte código (Extrato de código 20):

```
$filter('date')(cert.info.EXPIRES_DATE, 'short');
```

Extrato de código 20 – Exemplo de uso de \$filter

Já para utilizar o \$locale é necessário se realizar o seguinte (Extrato de código 21):

```
$locale.NUMBER_FORMATS.DECIMAL_SEP
```

Extrato de código 21 – Exemplo de uso de \$locale

Com isto, é possível então criar código adaptável ao local do globo onde o ALERT® PFH vai ser utilizado.

Por exemplo, antes desta solução, no componente “keypad-numeric” que permite a inserção de números, incluindo decimais, o símbolo decimal era inserido manualmente. Ou seja, não estava a ter em consideração as mudanças a nível do local do globo (Extrato de código 22 – alguma lógica de negócio foi omitida):

```
$scope.keypadNumeric = odn$keypad.keypadNumeric({
  data: {
    fields: [{
      LÓGICA_DE_NEGÓCIO
      separator: ',',
      LÓGICA_DE_NEGÓCIO})
```

Extrato de código 22 – Componente “keypad-numeric” sem \$locale

Usando o Angular Locale, o código anterior já fica corretamente adaptado ao local (Extrato de código 23 – alguma lógica de negócio foi omitida):

```
$scope.keypadNumeric = odn$keypad.keypadNumeric({
  data: {
    fields: [{
      LÓGICA_DE_NEGÓCIO
      separator: $locale.NUMBER_FORMATS.DECIMAL_SEP,
      LÓGICA_DE_NEGÓCIO})
  }
});
```

Extrato de código 23 – Componente “keypad-numeric” com \$locale

No caso de configurações por parte dos clientes recorre-se ao serviço REST descrito na secção 5.2.1 para alterar os dados provenientes do Angular Locale (exemplo para o símbolo decimal) (Extrato de código 24):

```
if(localizationDataClient.data[i].configTable == 'DECIMAL_SYMBOL'){
  $locale.NUMBER_FORMATS.DECIMAL_SEP=localizationDataClient.data[i].cfgValue;
}
```

Extrato de código 24 – Personalização do símbolo decimal pelo cliente

Relativamente a outros problemas que não os resolvidos pelo Angular Locale atualmente temos o caso da direção de texto e a posição do texto relacionado com campos de inserção de dados. Contudo, tanto um como o outro dependem também das decisões da equipa de *Design* da empresa, pelo que as decisões não podem ser completamente tomadas tendo em conta os *standards* de internacionalização e localização.

Apesar disso, no caso da direção de texto, tal como descrito na secção 2.1.4.3 do presente documento foi utilizada a direção de texto automática. Assim, consoante o texto disponibilizado, o sentido de orientação da leitura muda automaticamente e o texto ajusta-se a tal direção (Figura 51).

Summary	الهوية
History and Physical	الصورة
Progress notes	شفرة التعرف

Figura 51 – Diferenças no sentido da direção de texto

Para o caso da posição do texto relativamente a campos de inserção de dados, foi definido que o texto era colocado por cima do campo de inserção, assim como foi explicado na secção 2.1.4.4 deste documento. Contudo, em algumas interfaces tal não acontece devido a exigências por parte da equipa de *Design*.

5.2.4 Exemplo prático – alteração do separador decimal a pedido do cliente

Com as alterações efetuadas ao componente “keypad-numeric” para que utilize as características definidas pelo Angular Locale, o cliente tem acesso ao seguinte (cliente com local definido para inglês no Reino Unido) (Figura 52):



Figura 52 – Componente “keypad-numeric” com “.” como separador decimal

Verificando a Figura 52, o separador decimal que é apresentado ao cliente é um “.”²². Contudo, o cliente pretende antes ver disponibilizada uma “,”. Assim, avisa um elemento da equipa de operações de tal facto, sendo que de seguida ele insere o separador decimal na base de dados da empresa.

Posteriormente, quando o cliente for realizar *login* na aplicação, o serviço acede à base de dados e altera o separador decimal para a “,” pedida pelo cliente como se verifica na Figura 53.

²² “.” é o definido pelo Angular Locale para separador decimal em inglês no Reino Unido.



Figura 53 – Componente “keypad-numeric” com “,” como separador decimal

5.3 Conclusões

Tendo em conta toda a análise efetuada no capítulo 4 deste documento, sobretudo as melhorias definidas na secção 4.3, e verificando o desenvolvimento realizado, é de destacar que todas as melhorias identificadas foram concluídas com sucesso.

Também, é de salientar que à medida que se foi desenvolvendo as alterações ao código do ALERT® PFH, foram desenvolvidos testes unitários. Tal como o nome indica, testes unitários permitem testar pequenas unidades de código e verificar o seu correto funcionamento. Estes testes devem ser realizados durante todo o processo de desenvolvimento de *software* (Nidhra & Dondeti, 2019). Os testes foram desenvolvidos na plataforma proprietária da empresa, Ranch (Anexo D), utilizando a ferramenta *Jasmine*²³.

Estes testes foram realizados para que se verificasse que o uso das soluções desenvolvidas para o processo de internacionalização e localização não impactassem nas funcionalidades do produto. Tal pode ser verificado na Figura 54, onde se apresentam os testes ao componente “keypad-numeric”. Todos os testes são concluídos com sucesso o que indica que o uso das soluções não comprometeu o anterior funcionamento do componente. Verifica-se, também, que os testes realizados ao uso do símbolo decimal utilizado nas definições do Angular Locale passam com sucesso.

²³ *Framework* para implementação de testes unitários em JavaScript (Jasmine, 2019).

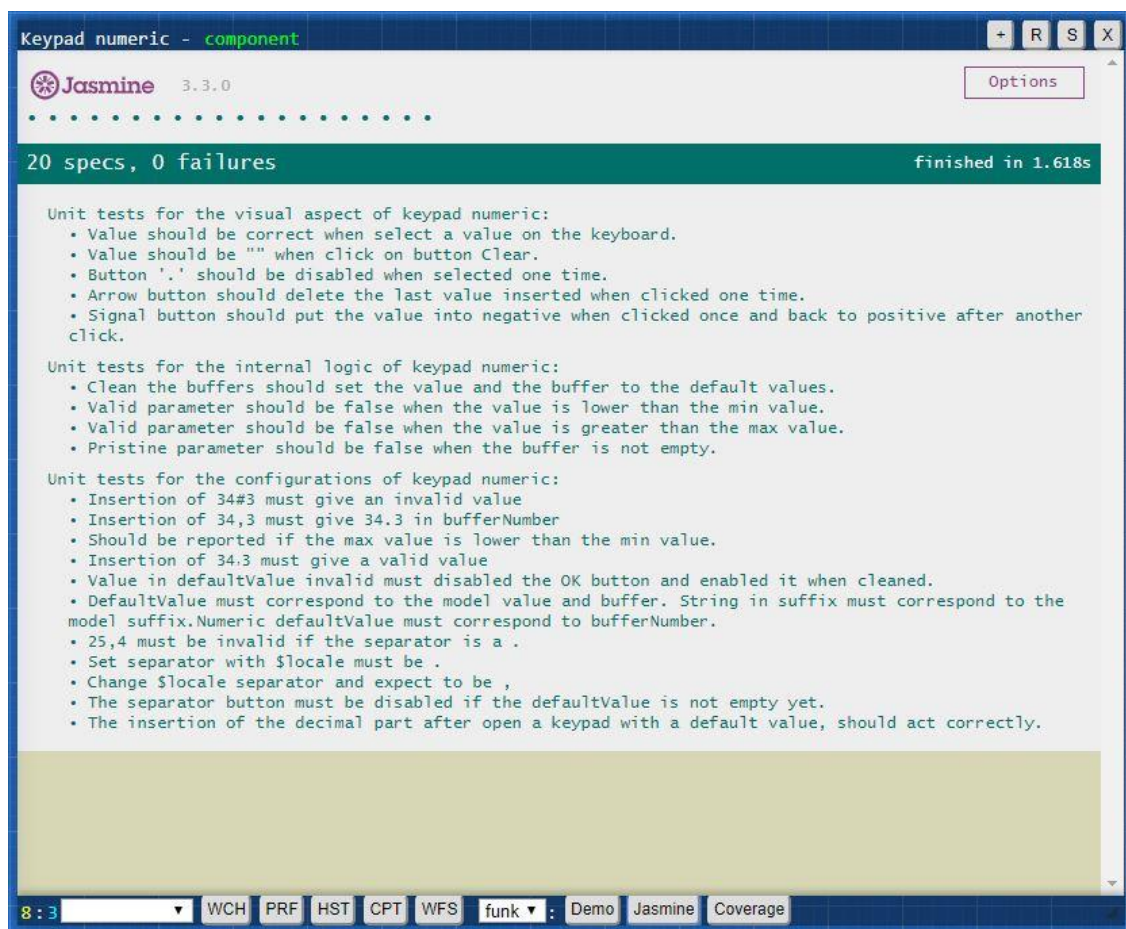


Figura 54 – Testes unitários realizados ao “keypad-numeric”

Também, a par do desenvolvimento das soluções, foram criados cursos/documentação na plataforma ALERT® Code Academy²⁴, tendo sido criado um curso sobre o projeto “labels-manager” (Figura 55) e outro relacionado com o uso do Angular Locale (Figura 56).

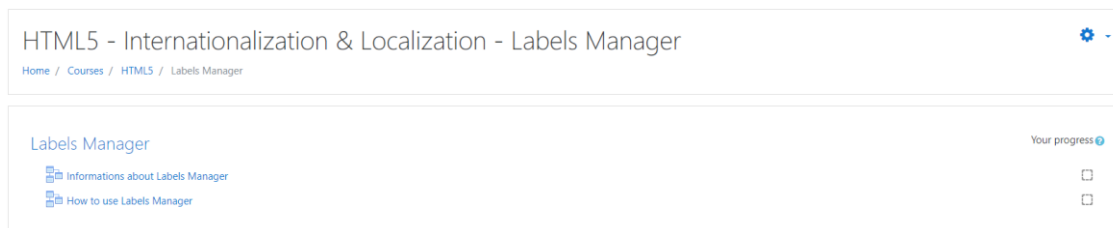


Figura 55 – Curso/documentação sobre o uso do projeto “labels-manager”

²⁴ Plataforma moodle proprietária da empresa, onde são inseridos cursos/documentação sobre as soluções desenvolvidas.

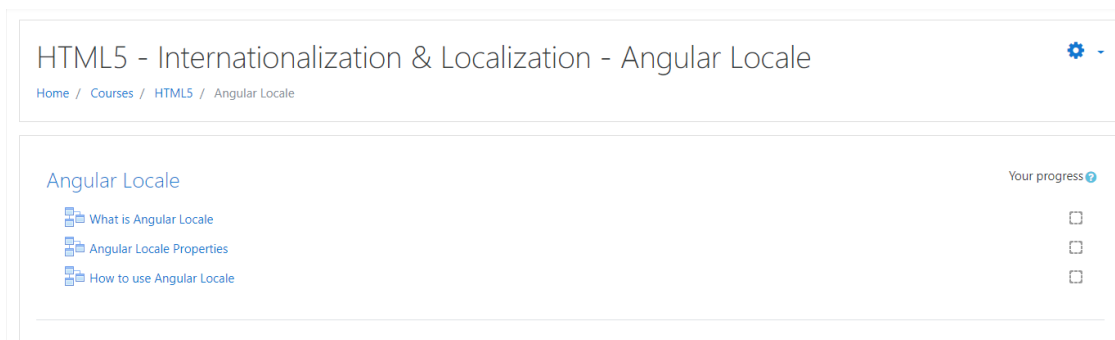


Figura 56 – Curso/documentação sobre o uso do Angular Locale

O curso/documentação sobre o projeto “labels-manager”, é dividido em duas lições, na primeira é explicado o seu intuito, quem deve utilizá-lo e como escolher os identificadores únicos das traduções. Na segunda lição é apresentado o funcionamento do projeto através da explicação de cada uma das funcionalidades disponibilizadas.

Quanto ao curso sobre o Angular Locale, nele é explicado o intuito do uso desta ferramenta, as suas propriedades e como deve ser utilizada nos produtos da empresa.

Estes cursos/documentação estão disponíveis a todos os elementos da empresa e é possível retirar estatísticas sobre a conclusão/visualização dos mesmos.

6 Experimentação e Avaliação

A solução desenvolvida e apresentada nos capítulos anteriores (capítulos 4 e 5) é aqui avaliada para averiguar se, de facto, esta é eficaz e produz os resultados desejados. É necessário também envolver nesta avaliação as partes interessadas no seu desenvolvimento, principalmente a equipa de Arquitetura da empresa.

Com o objetivo de apurar a eficácia do processo de internacionalização e localização desenvolvido, descreve-se neste capítulo os resultados da avaliação efetuada tendo em conta a hipótese definida na secção 1.3 do presente documento.

Em geral, pretende-se verificar se a solução de internacionalização e localização desenvolvida no âmbito deste projeto gera ou não benefícios para o cliente. Deseja-se aferir qual o grau de satisfação dos clientes relativamente à solução e se esta corresponde às suas necessidades.

Em particular, pretende-se avaliar se a hipótese definida se corrobora, ou seja, se a solução desenvolvida (processo e ferramentas) permite implementar um processo de internacionalização e localização dos produtos de *software* da ALERT, tornando a adaptação a um local do globo mais simples, abrangente e eficaz que a que era utilizada anteriormente.

Os indicadores explorados para avaliar esta hipótese derivam das respostas a inquéritos de avaliação e de um conjunto de outras estatísticas disponíveis na empresa, nomeadamente o número de acessos à documentação criada sobre as soluções desenvolvidas no ALERT® *Code Academy*. Os inquéritos de avaliação foram desenhados para avaliar o grau de satisfação do cliente tendo em conta a forma como o processo vai ao encontro das suas necessidades.

6.1 Inquérito

Nesta secção é apresentado, inicialmente, o guião realizado para que os inquiridos consigam responder adequadamente aos inquéritos. Posteriormente, caracteriza-se a população que

responde ao inquérito, bem como é feita a apresentação do mesmo. No fim avaliam-se os resultados obtidos pelas respostas dos inquiridos e retiram-se as devidas conclusões.

6.1.1 Guião

Antes de dar resposta aos inquéritos desenvolvidos, é apresentado ao inquirido um guião com o intuito de o auxiliar nos testes de usabilidade da solução encontrada, para que assim este consiga responder da melhor forma ao questionário desenvolvido. Este guião pretende fazer com que os utilizadores interajam com as partes mais relevantes da solução desenvolvida e assim tenham um contacto direto com a mesma para que verifiquem a sua usabilidade. O guião tem em conta que os inquiridos não necessitam de ter conhecimento prévio da solução, apenas devem seguir o indicado no mesmo para a utilizar.

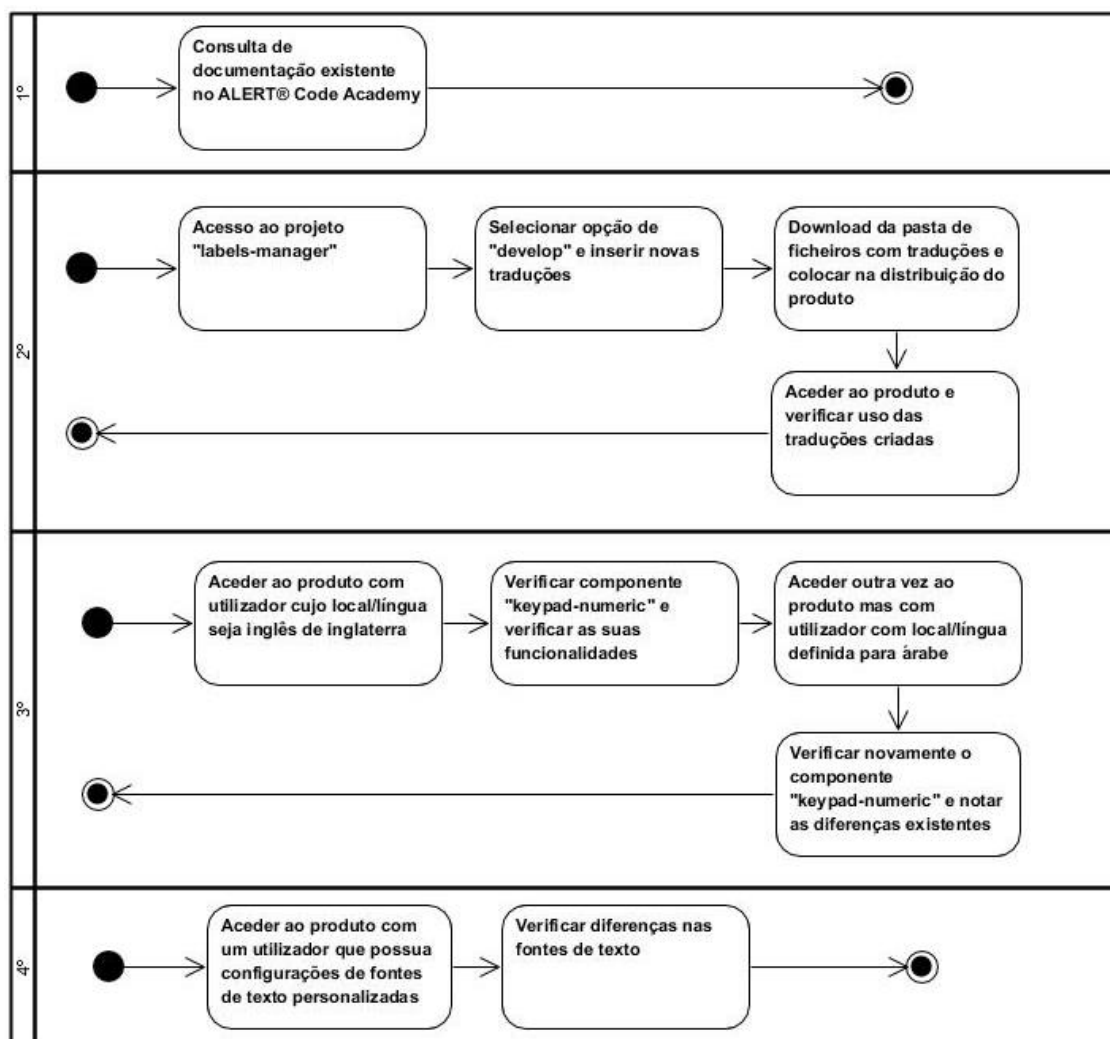


Figura 57 – Diagrama de atividades com as várias fases do guião

De uma forma resumida, o guião pede para se realizar um conjunto de tarefas como se pode verificar no diagrama de atividades da Figura 57: (1º) pede para aceder ao ALERT® *Code Academy* de forma a consultar toda a documentação sobre as soluções desenvolvidas; (2º) é pedido que aceda ao projeto “labels-manager” e para a versão “*develop*” adicione traduções novas para uma interface gráfica; é requerido que selecione um cliente de teste e que lhe acrescente traduções personalizadas; tendo efetuado as alterações anteriores, é pedido para que o inquirido faça *download* da pasta com os ficheiros finais das traduções e que a coloque na distribuição local que possui do produto ALERT® PFH; é pedido para aceder ao produto e verificar que as novas traduções estão a ser utilizadas; (3º) é solicitado ao inquirido que aceda ao produto com o *login* de um utilizador cujo local/língua definida seja inglês de Inglaterra; aceda, de seguida, ao componente “keypad-numeric” e verifique as suas funcionalidades bem como o separador decimal mostrado; é pedido ainda que, depois, realize o mesmo processo, mas com um utilizador que tenha como local/língua definida o árabe; é necessário que o inquirido note as diferenças existentes no componente, pois todas as alterações ocorrem automaticamente quando o *login* é realizado; (4º) por fim, é solicitado o acesso ao produto mais uma vez com um utilizador que possua configurações para as fontes de texto e que verifique que a fonte de texto disponibilizada, depois de *login* realizado, é diferente daquela que foi apresentada anteriormente.

6.1.2 Caracterização da população

Os inquiridos são todos programadores que fazem parte da equipa de desenvolvimento da empresa ALERT, mais precisamente da equipa de Arquitetura. Assim, estão todos familiarizados com as funcionalidades do produto, permitindo uma melhor utilização do guião, e ao mesmo tempo elevando a exigência associada às funcionalidades disponibilizadas e requeridas.

Para além disso, como o projeto foi desenvolvido para ser utilizado com a nova tecnologia em uso, estes são os elementos mais indicados para se perceber a usabilidade da solução, para além de que são estes que lidam diretamente com os problemas de internacionalização e localização do produto, tendo detetado problemas nas soluções que usavam anteriormente.

Todos os inquiridos, com idades compreendidas entre os 24 e os 41 anos, são de nacionalidade portuguesa, sendo o português a sua língua materna, pelo que o inquérito será realizado em português de Portugal.

Foram inquiridos 8 programadores.

6.1.3 Apresentação do Inquérito

Com o intuito de se entender a satisfação dos clientes relativamente aos requisitos definidos anteriormente na secção 3.3 e as melhorias identificadas na secção 4.3, é disponibilizado um inquérito que segue determinadas convenções (Dawson, 2007):

- Composto por questões quer de resposta fechada (o utilizador selecciona uma ou mais respostas apresentadas, não lhe sendo possível definir novas) como de resposta aberta (o utilizador tem a liberdade de responder à questão com as suas próprias palavras);
- Em algumas questões de resposta aberta, é expectável que o inquirido responda com “Sim” ou “Não”, e, se achar necessário, poderá justificar a resposta;
- Inicialmente são apresentadas as questões de resposta fechada e posteriormente as questões de resposta aberta;
- As questões de resposta fechada e aberta, e as opções de resposta fechada, são curtas, objetivas e simples, não dando azo a ambiguidades de interpretação;
- As questões não forçam os utilizadores a responderem de uma determinada forma dando origem a respostas falsas, nem utilizam termos negativos;
- No caso de determinação de satisfação de um requisito numa escala, a escala utilizada é compreendida entre os valores 1 a 5, sendo que 1 significa nada satisfeito e 5 completamente satisfeito;
- O questionário não associa respostas a determinado inquirido para que não sejam condicionadas as mesmas.

Tendo estas considerações em mente, importa ainda referir o tempo que o inquirido deve despendar para realizar o questionário. Prevê-se que o questionário demore, no máximo, 45 minutos a ser respondido, uma vez que é necessário que os utilizadores realizem algumas ações para testar a usabilidade da solução desenvolvida, nomeadamente sigam o guião desenvolvido (secção 6.1.1), o que lhes tomará algum tempo.

O questionário é composto por 17 questões, incluindo 9 questões de resposta fechada e 8 questões de resposta aberta. Todos os inquiridos devem responder a todas as questões do questionário visto estas serem obrigatórias.

6.1.4 Resultado dos Inquéritos

Tendo em consideração as respostas dadas pelos 8 inquiridos aos inquéritos disponibilizados, efetuou-se uma análise a cada uma das respostas com o objetivo de retirar conclusões sobre a usabilidade do processo de internacionalização e localização desenvolvido.

Para cada uma das questões de resposta fechada do inquérito foram elaborados gráficos de forma a melhor se perceber os resultados obtidos. No caso das respostas abertas são apresentadas todas as respostas, analisando as mesmas e retirando conclusões que possibilitam a definição de trabalho futuro e limitações das soluções existentes.

Qual o grau de satisfação relativamente à forma como é apresentada a lista de traduções no "labels-manager"?

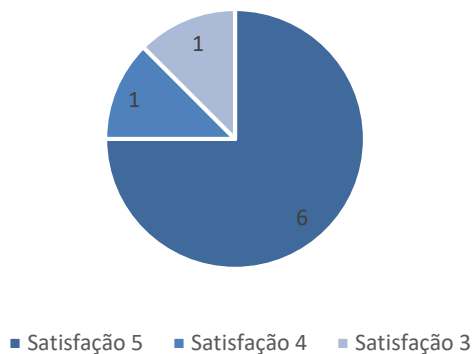


Figura 58 – Resposta à questão 1 do inquérito

Qual o grau de satisfação relativamente à forma como é realizada a pesquisa e substituição de traduções no "labels-manager"?

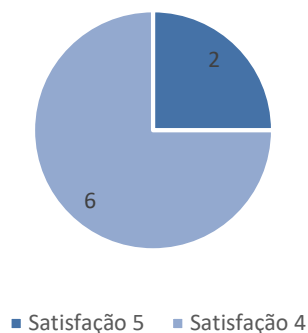


Figura 59 – Resposta à questão 2 do inquérito

Qual o grau de satisfação relativamente à facilidade de criação de novas traduções?

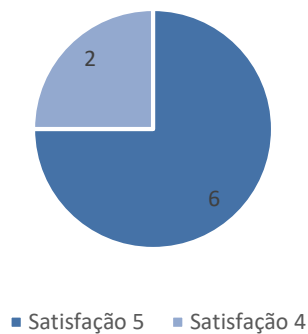


Figura 60 – Resposta à questão 3 do inquérito

Qual o grau de satisfação relativamente à personalização de traduções por parte do cliente?

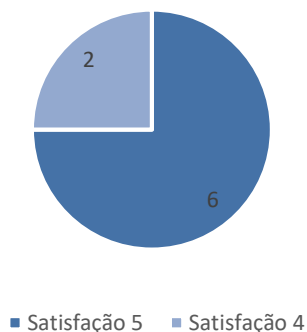


Figura 61 – Resposta à questão 4 do inquérito

Qual o grau de satisfação relativamente à alteração das traduções no produto ALERT® PFH?

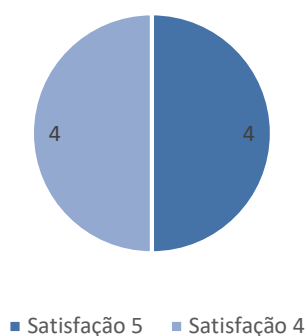


Figura 62 – Resposta à questão 5 do inquérito

Qual o grau de satisfação relativamente ao uso do Angular Locale (formatos numéricos, de datas, ...)?

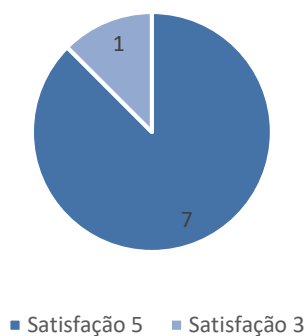


Figura 63 – Resposta à questão 6 do inquérito

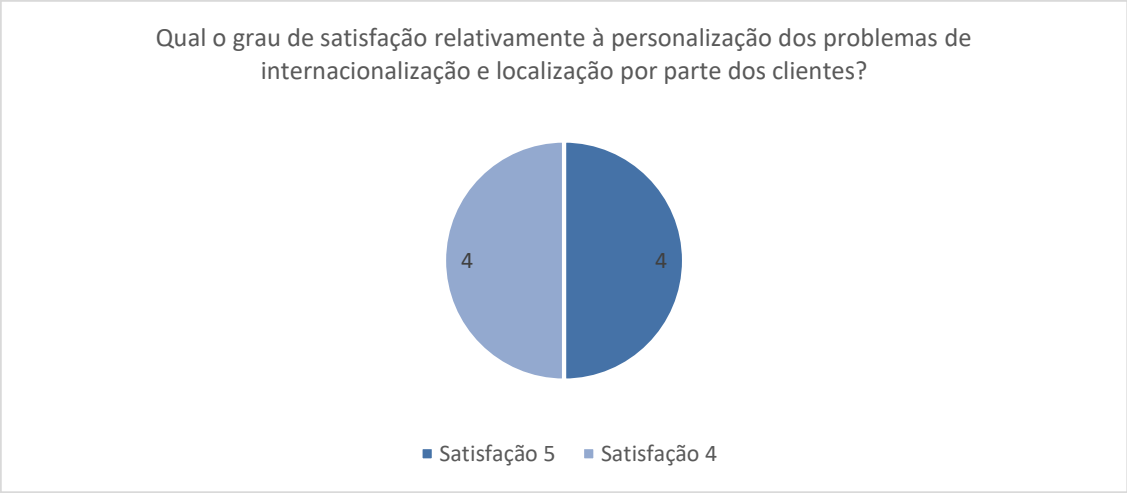


Figura 64 – Resposta à questão 7 do inquérito

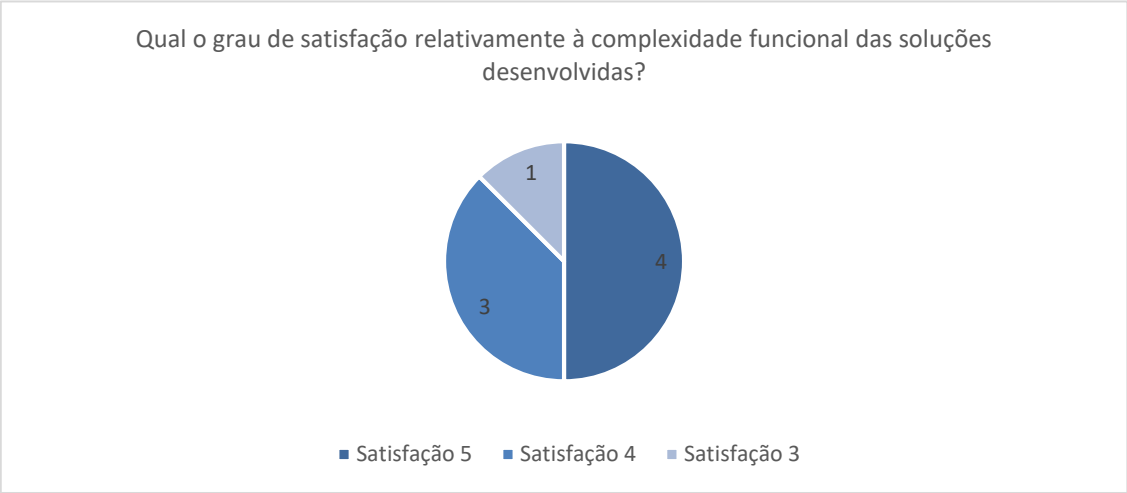


Figura 65 – Resposta à questão 8 do inquérito

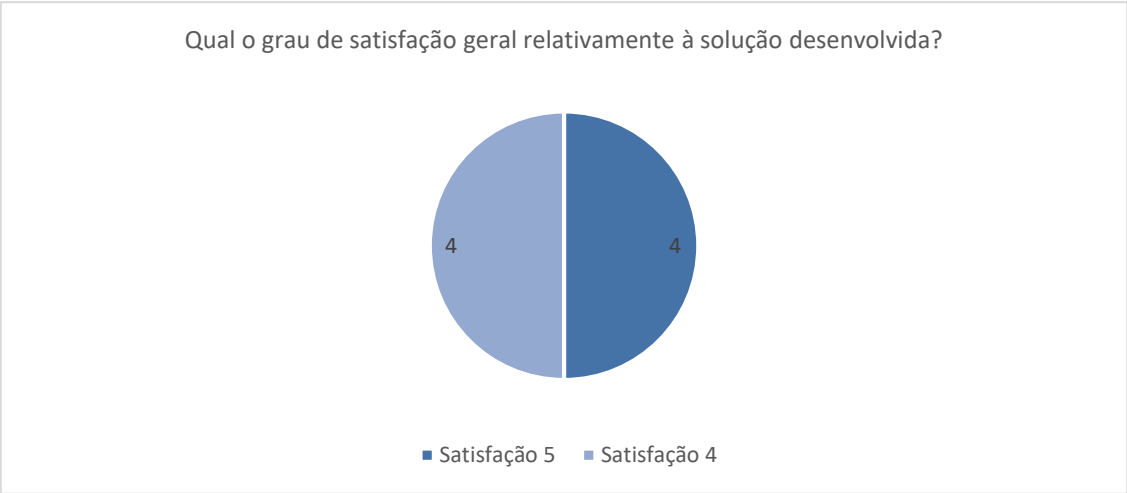


Figura 66 – Resposta à questão 9 do inquérito

À pergunta número 10, “Que problemas encontrou ao analisar as alterações efetuadas pelo processo de internacionalização e localização desenvolvido?” obteve-se as seguintes respostas:

- Nada a acrescentar;
- Algumas pequenas melhorias que podem ser efetuadas ao nível da interface;
- Não encontrei problemas. As alterações ocorreram com sucesso;
- Mecanismo de *search and replace* poderá trazer alguns problemas, quando algum utilizador fizer *replace* inadvertidamente. Método de pesquisa deverá ser mais robusto;
- Não foram encontrados problemas;
- Não;
- Nenhum;
- O principal problema é a atualização das traduções implicar substituir os ficheiros de *deploy*. Problema de *cache*.

Quanto à pergunta número 11, “Que melhorias gostava que acontecessem ao nível do processo de internacionalização e localização?”, os inquiridos responderam da seguinte forma:

- Nada a acrescentar;
- Por exemplo, a pesquisa podia ter um *scroll* automático para as traduções alteradas. Filtros na pesquisa e permitir alterar traduções apenas para uma língua;
- Temos agora uma melhor resposta para os problemas de que o PFH padecia antes desta solução, por isso não vislumbro nenhuma de momento;
- Permitir escolher qual as línguas a substituir durante o processo de *search and replace*, por exemplo;
- Melhorar a pesquisa/alteração das *labels* introduzindo mais critérios;
- Nenhuma alteração. Há sempre pontos a melhorar, mas no que diz respeito ao âmbito do projeto foram cumpridos todos os pontos;
- Nenhuma;
- O projeto “labels-manager” tivesse perfis de utilização.

À questão número 12, “Acha que é capaz de utilizar as soluções desenvolvidas frequentemente? Porquê?” obteve-se as seguintes respostas:

- Sim, devido à facilidade de utilização;
- Sim porque é uma solução *user e developer friendly*;
- Sim, agilizam o processo e salvaguardam de que os dados são alterados de forma independente;
- Sim. A solução é bastante intuitiva e de fácil utilização;
- Sim, porque corresponde perfeitamente aos requisitos necessários;
- Sim. Porque a solução é intuitiva e está bem documentada;
- Sim;
- Sim. Porque é um processo simples e eficaz.

Para a questão número 13, “Acha as soluções fáceis de utilizar? Porquê?” os inquiridos responderam da seguinte forma:

- Sim, devido a serem intuitivas;

- Sim, porque a interface é muito intuitiva e o *locale* tem muita documentação;
- Sim. Permitem de forma rápida e segura efetuar as alterações necessárias;
- Sim. A solução é bastante simples (no sentido de ser de fácil utilização). Permite efetuar as tarefas desejadas de forma rápida e intuitiva;
- Sim, é bastante intuitiva e de fácil utilização;
- Sim. Para além de bem documentada, com os vários exemplos apresentados torna-se simples perceber a dinâmica;
- Sim;
- Sim, solução intuitiva.

Na questão número 14, “A documentação sobre as soluções disponibilizadas é esclarecedora?”, as respostas obtidas foram:

- Sim – resposta obtida 6 vezes;
- Sim é;
- Sim. A documentação é bastante esclarecedora e explora a fundo algumas temáticas que permite compreender a solução ainda melhor.

Quanto à questão número 15, “Acha que necessita de suporte técnico para utilizar as soluções desenvolvidas (para além da documentação fornecida)?”, as respostas foram:

- Não – resposta obtida 7 vezes;
- Não, a documentação e o material de apoio são suficientes na minha opinião.

À pergunta número 16, “Tem alguma observação a relatar?”, os inquiridos responderam da seguinte forma:

- Não – resposta obtida 5 vezes;
- Nada a acrescentar;
- Talvez no futuro aplicar mais formas de *find/replace* das *labels*, como por exemplo substituir todas as ocorrências de uma *label* mas apenas para uma língua;
- Tirando aquelas pequenas notas apresentadas nas respostas anteriores, não tenho nada a relatar.

Por fim, as respostas obtidas à questão número 17, “Qual a sua opinião geral sobre as soluções desenvolvidas?”, foram:

- Bastante satisfeito, nenhum ponto negativo a relatar;
- Muito favorável;
- Temos agora uma melhor resposta para os problemas de que o PFH padecia antes desta solução, por isso não vislumbro nenhuma de momento. Agilizam o processo e salvaguardam de que os dados são alterados de forma independente. Permitem de forma rápida e segura efetuar as alterações necessárias;
- Penso que as soluções são bastante adequadas, permitem resolver a 100% as necessidades da empresa e são ótimas na perspetiva tanto do programador, como do utilizador (poderá não ser programador). Estas soluções agradam-me bastante;
- Muito bom;

- Na minha opinião a solução é simples, está bem documentada e resolve os problemas que existiam no início do estágio. Há sempre espaço para melhorias por exemplo (uso de métricas, melhorias de pesquisa, etc.) mas os pontos que foram definidos no início do estágio foram resolvidos;
- Respondem ao problema apresentado;
- As mesmas responderam com sucesso aos problemas identificados.

Tendo-se obtido todas estas respostas, importa agora analisar as mesmas e auferir a eficácia das soluções desenvolvidas.

Nas questões de resposta fechada os valores de satisfação correspondem aos seguintes: 1 – nada satisfeito, 2 – pouco satisfeito, 3 – satisfeito, 4 – muito satisfeito²⁵ e 5 – completamente satisfeito.

Para se avaliar estatisticamente os resultados, agrupou-se as questões nas seguintes categorias:

- Satisfação relativamente à resolução de problemas relacionados com a internacionalização e localização de *software*;
- Satisfação relativamente à possibilidade de personalização de conteúdos por parte dos clientes;
- Satisfação do geral do processo desenvolvido e possíveis melhorias.

6.1.4.1 Satisfação relativamente à resolução de problemas relacionados com a internacionalização e localização de *software*

Esta é a categoria mais importante a ser analisada pois permite determinar qual a satisfação das soluções desenvolvidas e se estas conseguem responder aos problemas propostos. Assim, para avaliar esta categoria analisou-se a respostas às perguntas 1, 2, 3, 5, 6, 10, 12, 13, 14 e 15.

Destas questões, as perguntas 1, 2 e 3 estão relacionadas diretamente com o projeto “labels-manager”. À questão 1, que pretendia saber se o inquirido estava satisfeito com a forma como a lista de traduções era apresentada, 75% (6 inquiridos) responderam que estavam completamente satisfeitos, 12,5% (1 inquirido) como muito satisfeito e 12,5% (1 inquirido) como apenas satisfeito. Para a pergunta 2, em que se auferia a satisfação quanto à pesquisa e substituição de traduções, 25% (2 inquiridos) dos inquiridos encontram-se completamente satisfeitos e 75% (6 inquiridos) muito satisfeitos. Na pergunta 3, quanto à facilidade de criação de traduções, 75% (6 inquiridos) responderam estar completamente satisfeitos e 25% (2 inquiridos) responderam estar muito satisfeitos. Assim, constata-se que para o projeto “labels-manager” a grande maioria da população se encontra completamente satisfeita com o resultado obtido. É de ressaltar que, tendo em conta os resultados, a pesquisa e substituição de traduções é o tópico que necessita de um maior aperfeiçoamento pois a maioria dos inquiridos respondeu estar muito satisfeito, contudo existem sempre melhorias a efetuar.

²⁵ Corresponde ao desenvolvimento de pequenas melhorias que tornariam a solução completamente satisfeita, mas que não implicam as funcionalidades até então disponibilizadas.

À questão 5, relativamente à satisfação quanto à alteração das traduções no produto ALERT® PFH, 50% da população (4 inquiridos) responderam estar completamente satisfeitos e outros 50% (4 inquiridos) muito satisfeitos. Com estes resultados, conclui-se que tendo em conta a criação das traduções e disponibilização das mesmas no ALERT® PFH, os inquiridos encontram-se muito satisfeitos com a solução.

Na questão 6, onde se questionava sobre a satisfação quanto ao uso do Angular Locale, 87,5% dos inquiridos (7 inquiridos) respondeu estar completamente satisfeito com a solução desenvolvida, enquanto que 12,5% (1 inquirido) mostrou estar satisfeito. Estes resultados comprovam que a ferramenta Angular Locale utilizada para tratar de alguns dos problemas de internacionalização e localização foi uma mais valia.

Passando agora às questões de resposta aberta, quando questionado sobre se tinha encontrado problemas ao analisar as alterações efetuadas pelo processo de internacionalização e localização, 62,5% (5 inquiridos) responderam não ter encontrado problemas. Outros 2 inquiridos não relatam problemas, mas sim melhorias que podem ser realizadas (mecanismo de pesquisa e substituição de traduções por exemplo). Por fim, 1 inquirido relata ter tido problemas ao atualizar os ficheiros com as novas traduções na distribuição do produto pois necessitou de limpar a *cache* do mesmo para tais terem efeito. Isto deve-se ao facto do ALERT® PFH armazenar dados em *cache* para facilitar o carregamento das interfaces gráficas, daí a necessidade de se ter de limpar estes dados.

Relativamente à questão 12 onde era questionado se o inquirido era capaz de utilizar as soluções desenvolvidas frequentemente, todos eles, ou seja 100% da população, respondeu que sim. É referido que as soluções são intuitivas e de fácil utilização, tendo respondido a todos os requisitos necessários. Estes dados permitem constatar que em termos de usabilidade, as soluções desenvolvidas permitem que os utilizadores recorram às mesmas para tratar das alterações do produto ao nível da internacionalização e localização, como era o objetivo do projeto.

Na questão 13, se considerava as soluções fáceis de utilizar, mais uma vez, 100% da população respondeu que sim. Assim, constata-se que a complexidade da solução facilita o desenvolvimento do produto e não acarreta problemas para o programador.

As questões 14 e 15 referem-se à documentação desenvolvida sobre as soluções. Numa primeira fase (questão 14) foi questionado se a documentação era esclarecedora, tendo-se obtido 100% de resultados positivos. Posteriormente, questionou-se sobre se seria necessário suporte técnico, para além da documentação existente, para que as soluções fossem utilizadas. Nesta questão, 100% da população respondeu que achava que não necessitava. Estes resultados indicam que a documentação realizada é suficiente e esclarecedora para a utilização das soluções.

Assim, com todos estes resultados, conclui-se que os inquiridos se encontram satisfeitos com as soluções desenvolvidas, havendo sempre espaço para melhorias que podem ser desenvolvidas.

6.1.4.2 Satisfação relativamente à possibilidade de personalização de conteúdos por parte dos clientes

Para analisar esta categoria, recorre-se às respostas dadas às questões números 4, 5 e 7, todas elas de resposta fechada.

Quando questionados sobre a satisfação relativamente à personalização de traduções por parte do cliente, 6 inquiridos (75% da população) responderam estar completamente satisfeitos e 2 inquiridos (25% da população) demonstraram estar muito satisfeitos. À questão 5, relativamente à satisfação quanto à alteração das traduções no produto ALERT® PFH, 50% da população (4 inquiridos) responderam estar completamente satisfeitos e outros 50% (4 inquiridos) muito satisfeitos. Com a resposta a estas duas questões, conclui-se que grande parte da população se encontra completamente satisfeita quanto à forma como o cliente pode configurar as suas próprias traduções.

Na pergunta 7, sobre o grau de satisfação relativamente à personalização de outros problemas de internacionalização e localização por parte dos clientes, 50% da população (4 inquiridos) respondeu estar completamente satisfeita e outros 50% (4 inquiridos) respondeu estar muito satisfeita.

Conclui-se que a satisfação dos inquiridos quanto à personalização por parte dos clientes é positiva, uma vez que as soluções dão ao cliente a possibilidade de configurar determinados conteúdos do produto de forma a responderem às suas necessidades.

6.1.4.3 Satisfação geral do processo desenvolvido e possíveis melhorias

Para averiguar a satisfação por parte dos inquiridos quanto ao processo realizado verificou-se as respostas às questões 8, 9, 11, 16 e 17 do inquérito.

Na questão 8, onde se inquiria sobre a satisfação relativamente à complexidade funcional das soluções desenvolvidas, 50% dos inquiridos (4 inquiridos) responderam estar completamente satisfeitos. 37,5% (3 inquiridos) responderam estar muito satisfeitos e 12,5% (1 inquirido) respondeu estar satisfeito. À questão 9, sobre o grau de satisfação geral da solução desenvolvida, 50% da população (4 inquiridos) encontra-se completamente satisfeita e os restantes 50% (4 inquiridos) responderam estar muito satisfeitos. Com os resultados a estas questões constata-se que apesar da grande maioria estar bastante satisfeita com o processo, existe sempre oportunidade de melhoria e de tornar o processo mais simples.

Na questão 11, de resposta aberta, em que é questionada a população sobre que outras alterações gostavam que acontecessem ao nível do processo de internacionalização e localização realizado, 3 inquiridos (37,5% da população) responderam que podiam ser adicionadas mais funcionalidade à pesquisa de traduções, nomeadamente a criação de filtros/critérios de pesquisa. 4 inquiridos (50% da população) demonstraram estar satisfeitos com as alterações disponibilizadas, não encontrando mais nenhuma a realizar. Por fim 1 inquirido (12,5% da população) afirma que o projeto “labels-manager” poderia ter perfis de utilização. Esta é uma melhoria futura a ter em conta, pois permitirá mais facilmente que o utilizador saiba quais as tarefas que tem disponível para realizar.

À questão número 16, em que se perguntava se havia alguma observação a relatar sobre o uso das soluções, 6 inquiridos (75% da população) responderam não ter nenhuma observação. 2 inquiridos (25% da população) disseram que no futuro o funcionamento da pesquisa e alteração de traduções poderia ser melhorado.

Por fim, na questão 17, sobre a opinião geral das soluções desenvolvidas, 100% da população (8 inquiridos) demonstrou estar bastante satisfeito. Nesta questão foi referido que não existem pontos negativos a relatar, sendo que existe, agora, uma solução adequada às necessidades dos produtos da empresa. É também referido que a solução é simples e intuitiva, de fácil utilização e com uma documentação adequada.

Assim, conclui-se que os inquiridos se encontram muito satisfeitos com todo o projeto desenvolvido, existindo sempre a possibilidade de desenvolvimento de melhorias e novos funcionamentos. O processo desenvolvido é uma mais valia para a empresa e permite dar resposta aos problemas identificados de internacionalização e localização dos produtos.

A hipótese (secção 1.3) verifica-se com valor acrescentado para a empresa.

6.2 Outros indicadores

Como forma de obtenção de outros indicadores, recorreu-se à ferramenta ALERT® *Code Academy*, para se extrair dados de consulta aos cursos/documentação criada sobre a solução desenvolvida (até à data de 18 de junho de 2019).

Os resultados obtidos encontram-se apresentados na Figura 67 e Figura 68.

Activity	Views	Related blog entries	Last access
Labels Manager			
 Informations about Labels Manager	178 views by 15 users	-	Tuesday, 18 June 2019, 2:31 PM (20 hours 2 mins)
 How to use Labels Manager	211 views by 15 users	-	Tuesday, 18 June 2019, 2:34 PM (20 hours 5 mins)

Figura 67 – Consulta da documentação/corso sobre o projeto “labels-manager”

Activity	Views	Related blog entries	Last access
Angular Locale			
 What is Angular Locale	273 views by 13 users	-	Tuesday, 18 June 2019, 2:23 PM (19 hours 55 mins)
 Angular Locale Properties	204 views by 11 users	-	Tuesday, 18 June 2019, 2:24 PM (19 hours 56 mins)
 How to use Angular Locale	256 views by 11 users	-	Tuesday, 18 June 2019, 2:26 PM (19 hours 58 mins)

Figura 68 – Consulta da documentação/curso sobre o Angular Locale

Observando as figuras anteriores (Figura 67 e Figura 68) constata-se que as duas lições sobre o projeto “labels-manager” foram consultadas por 15 utilizadores diferentes, sendo que a lição referente a como utilizar o projeto apresentou um maior número de visualizações por parte destes utilizadores. Quanto à documentação sobre o Angular Locale, a primeira lição, a que define o intuito desta ferramenta, foi consultada por 13 utilizadores diferentes e as outras duas lições por apenas 11. Em termos de visualizações, como era de esperar, a primeira lição é a que apresenta um maior número de visualizações, sendo seguida da lição sobre como usar as funcionalidades do Angular Locale no produto.

É de salientar, que dentre todos os utilizadores que consultaram esta documentação/curso, toda a população alvo do inquérito referido na secção 6.1 faz parte destes utilizadores, sendo um indicativo que seguiram adequadamente o guião disponibilizado.

Outro dado estatístico que é possível ser obtido, é o número de traduções feitas pela empresa inseridas através do projeto “labels-manager”, com as novas funcionalidades²⁶. Até ao dia 18 de junho de 2019, tinham sido inseridos 186 identificadores únicos, com traduções para 20 línguas diferentes, perfazendo, um total, de 3 720 traduções inseridas. Todas estas traduções encontram-se já a ser utilizadas no ALERT® PFH.

Por fim, importa ainda referir quando foram disponibilizadas as soluções em questão. O projeto “labels-manager”, com todas as atuais funcionalidades, encontra-se em utilização desde o dia 27 de maio de 2019. Quanto às soluções provenientes do Angular Locale, estas já se encontram disponíveis desde 3 de maio de 2019, sendo que a personalização das características por parte dos clientes apenas ficou disponível a 10 de maio de 2019.

Ainda sobre as soluções desenvolvidas, importa referir, que as características disponibilizadas pelo Angular Locale são utilizadas por diversos programadores da empresa, não só pela equipa de Arquitetura. Contudo, não foi possível obter dados concretos sobre o número de programadores a usar tal solução. Quanto às personalizações dos clientes em termos de traduções, estas podem ser enviadas para o cliente assim que estejam criadas, não sendo necessário que o cliente espere pelo lançamento de uma nova versão do produto, para ter acesso a estas alterações.

²⁶ Esta informação é obtida através da consulta do número de identificadores únicos de traduções na versão mais recente do ficheiro “flatLabels.json” – ficheiro apresentado na secção 4.2 do documento.

6.3 Conclusões

Através da análise dos inquéritos e respetivas conclusões, para além das outras estatísticas obtidas, é possível concluir que as soluções desenvolvidas para dar resposta aos problemas de internacionalização e localização foram bem-sucedidas. Das questões de resposta fechada (9), respondidas por 8 inquiridos obteve-se os seguintes resultados totais:

Tabela 30 – Dados finais de resposta ao inquérito

	Grau de satisfação				
	1	2	3	4	5
Número de respostas	0	0	3	26	43

Considerando os dados da Tabela 30, numa totalidade de 72 respostas distribuídas por 9 questões, 3 representam a resposta “satisfeito” (4,17%), 26 “muito satisfeito”²⁷ (36,11%) e 43 “completamente satisfeito” (59,72%). Somando os dois níveis superiores do grau de satisfação (grau de satisfação 4 e 5 – com satisfação superior ou igual a “muito satisfeito”) obtém-se 95,83% de satisfação positiva. Assim, tendo em conta estes dados e a hipótese formulada (secção 1.3), conclui-se que as soluções apresentam um elevado grau de satisfação e que dão resposta às necessidades do produto.

É ainda importante referir que melhorias podem vir a ser desenvolvidas de forma a tornar o projeto mais bem conseguido e aumentar a satisfação de todos os inquiridos para “completamente satisfeito”.

²⁷ Corresponde ao desenvolvimento de pequenas melhorias que tornariam a solução completamente satisfeita, mas que não implicam as funcionalidades até então disponibilizadas.

7 Conclusão

Neste último capítulo são apresentadas as conclusões retiradas relativamente ao projeto desenvolvido. Refere-se, também, os objetivos que foram alcançados, bem como as principais limitações que foram surgindo durante o desenvolvimento desta solução. Menciona-se ainda o que deve ser realizado futuramente para aprimorar a solução.

7.1 Objetivos Alcançados

O grande objetivo deste projeto passava pela criação de um processo de internacionalização e localização que facilitasse a adaptação do produto ALERT® PFH ao mundo. Para tal, foi necessário que o produto fosse estudado e que se criassem as melhores soluções para atingir o objetivo proposto.

De acordo com o problema definido na secção 3.1, ficaram assentes quatro principais necessidades do produto. Relativamente à “Gestão de conteúdos relativos a locais especificados pelo cliente”, com o desenvolvimento deste projeto, permite-se que o cliente defina as diferentes características, como o caso dos formatos de datas e os separadores numéricos. Todos os seus dados estão a ser armazenados numa base de dados, para que quando forem lançadas novas versões dos produtos estes dados não se percam. Quanto ao ponto “Gestão do mecanismo de carregamento de fontes” refere-se que tal foi atingido com sucesso, na medida em que foram definidas as fontes a usar para cada uma das linguagens do produto bem como é possível que o cliente defina as fontes de texto que pretende. Estas fontes definidas pelos clientes estão também armazenadas na base de dados para que assim o produto não possua, no seu código fonte, dados dos clientes. Relativamente ao ponto “Gestão de diferenças existentes entre os diversos locais do globo”, recorreu-se à ferramenta Angular Locale para realização desta necessidade do produto. Esta ferramenta permite definir um conjunto de características que variam de local para local do globo. Usando as suas características, o produto adapta-se posteriormente ao local onde é disponibilizado, não tendo de ser realizada nenhuma alteração por parte do desenvolvedor de *software*. Já quanto a “Fazer

com que o projeto das *labels* seja independente de versões”, desenvolveu-se um projeto que permite que a empresa defina as suas próprias traduções, bem como possibilita que os clientes personalizem as mesmas. Também, o cliente não necessita de esperar pelo lançamento de uma nova versão para ter acesso às suas traduções, bastando apenas que lhe seja enviada a pasta que contém os ficheiros finais com os identificadores únicos e as próprias traduções.

Como se pode verificar, todas estas necessidades foram solucionadas com o desenvolvimento do projeto.

Ainda, tendo em conta a tabela presente no Anexo F sobre o estado do desenvolvimento dos requisitos funcionais definidos na secção 3.3.1 deste documento, observa-se que todos os requisitos definidos pelos clientes foram concluídos. Para além disso, as melhorias identificadas na secção 4.3 foram todas atingidas com sucesso, existindo sempre a possibilidade de melhorar algumas funcionalidades.

Por fim, referir que a hipótese definida na secção 1.3 e avaliada no capítulo 6 do presente documento foi corroborada, pelo que a solução acrescenta valor à empresa.

7.2 Limitações e Trabalho Futuro

Como limitações que foram surgindo ao longo do desenvolvimento do projeto, existe o facto do ALERT® PFH ser já um produto bastante maturado e que disponibiliza um conjunto elevado de características e funcionalidades. Com o intuito de dar resposta às necessidades do produto, foi necessário se realizar uma análise profunda ao projeto bem como a todos os componentes a ele associados. Foi, então, essencial despendir uma elevada quantidade de tempo nesta análise, para que as soluções desenvolvidas não afetassem as funcionalidades do produto e dessem resposta aos requisitos destes.

Relativamente a trabalho futuro, e apesar das soluções desenvolvidas responderem às necessidades do ALERT® PFH, existe sempre espaço para a realização de melhorias e novas implementações.

Algum deste trabalho futuro surgiu através da análise dos inquéritos efetuados aos membros da equipa de Arquitetura da empresa (secção 6.1). Temos o caso da criação de novos filtros de pesquisa no projeto “labels-manager”, a disponibilização de métricas relativamente às traduções efetuadas e a criação de perfis de utilização também nesse projeto. Para além disso, durante o desenvolvimento das soluções foram identificados novos desenvolvimentos que fazem sentido que a solução possua. Nomeadamente, a possibilidade de associar uma imagem da interface gráfica onde vai ser utilizada a tradução ao identificador único da mesma, com o intuito de dar maior contexto à equipa de traduções.

Também, de forma a que a equipa de desenvolvimento de *software* possa verificar as alterações que ocorrem usando cada uma das características provenientes da solução, propõe-se o

desenvolvimento de um simulador. O simulador pode ter por base a plataforma Ranch²⁸ já existente e desenvolvida pela empresa (Anexo D). Assim, poderão ser desenvolvidos novos componentes nesta plataforma para que se consiga verificar com atenção quais as diferenças, ao nível das soluções criadas, que acontecem e como estas afetam a interface disponibilizada ao utilizador.

7.3 Apreciação Final

Fazendo um balanço sobre todo o trabalho desenvolvido ao longo deste projeto, considero que o conhecimento sobre a adaptação de um produto ao mundo aumentou em todos os níveis. Quando se pensa num projeto disponível em vários locais do globo, pensa-se logo de imediato nas traduções que são necessárias se realizar para que o produto seja utilizado por diferentes utilizadores. Contudo, e apesar das traduções serem na verdade um fator importante nesta disponibilização, existem muitas outras características que devem ser tidas em conta. Foram estas características e a diversidade delas que fizeram que o meu conhecimento aumentasse. Não basta apenas querer disponibilizar o produto noutros locais, é necessário que este se adapte para que seja utilizado da melhor forma possível aos clientes.

O contacto com diferentes tecnologias e ferramentas existentes no mercado que ajudam a solucionar a internacionalização e localização do *software* permitiu aumentar o conhecimento sobre a área, visto que estas são utilizadas por várias empresas e dão resposta a diversos problemas. Assim, ao serem utilizadas por um grande número de empresas, significa que estas estão cada vez mais a disponibilizar os seus produtos em diversos locais do globo, pelo que a adaptação dos mesmos é algo que as preocupa.

Também, é de salientar que durante todo o desenvolvimento do projeto, o facto de estar integrado numa equipa e de seguir a metodologia Scrum permitiu discutir e encontrar melhores soluções para os problemas a resolver. Toda a integração, o ambiente, a motivação e a disponibilidade dos elementos, facilitou o desenvolvimento do projeto. O trabalho junto destes profissionais continuará depois do fim deste projeto de mestrado, devido ao facto de ter sido renovada a relação com a empresa.

Em termos da solução criada e de como esta deveria ser desenvolvida para corresponder às necessidades do ALERT® PFH, permitiu contactar com um produto maduro, complexo e com diversas funcionalidades, que se encontra numa área bastante exigente. Assim, o saber sobre a área, no caso área da saúde, aumentou.

Em suma, considero que o desenvolvimento deste projeto foi uma mais valia tanto para conhecimento pessoal como para experiência profissional.

²⁸ Plataforma proprietária da empresa para realização de testes a componentes.

Referências

- Adobe. (2018). *What is Flash Player?* Retrieved dezembro 19, 2018, from Adobe: <https://www.adobe.com/pt/products/flashplayer.html>
- Adobe Corporate Communications. (2017, julho 07). *Flash & The Future of Interactive Content*. Retrieved dezembro 19, 2018, from Adobe Blog: <https://theblog.adobe.com/adobe-flash-update/>
- Airbnb. (2018). *Polyglot.js*. Retrieved dezembro 3, 2018, from GitHub: <https://github.com/airbnb/polyglot.js>
- Alchemy Software Development. (2018a). *Alchemy CATALYST 2019*. Retrieved maio 14, 2019, from Alchemy Software Development: https://www.alchemysoftware.com/products/alchemy_catalyst.html
- Alchemy Software Development. (2018b). *Software Applications*. Retrieved maio 14, 2019, from Alchemy Software Development: https://www.alchemysoftware.com/solutions/software_applications.html
- ALERT® Life Sciences Computing. (1999). *Missão e Valores*. Retrieved novembro 22, 2018, from ALERT ONLINE: <http://org-www.alert-online.com/pt/mission-values>
- ALERT® Life Sciences Computing. (2017). ALERT NEWS AND CATALOGUE. *ALERT NEWS*, pp. 56-91.
- ALERT® Life Sciences Computing. (2018a). *ALERT® - Linhas de Produto*. Retrieved dezembro 20, 2018, from ALERT® ONLINE: <http://www.alert-online.com/pt/product-lineage>
- ALERT® Life Sciences Computing. (2018b). *ALERT® PAPER FREE HOSPITAL (PFH)*. Retrieved dezembro 21, 2018, from ALERT® ONLINE: <http://www.alert-online.com/pfh>
- Ambler, S. (2018). *User Stories: An Agile Introduction*. Retrieved janeiro 14, 2019, from Agile Modeling: <http://www.agilemodeling.com/artifacts/userStory.htm>
- Angular. (2018). *Angular*. Retrieved dezembro 3, 2018, from GitHub: <https://github.com/angular/angular/tree/master/packages/common/locales>
- Angular Dynamic Locale. (2018). *Angular Dynamic Locale*. Retrieved junho 11, 2019, from angular-dynamic-locale: <https://www.npmjs.com/package/angular-dynamic-locale>
- Angular Translate. (2018a). *Angular Translate*. Retrieved dezembro 3, 2018, from GitHub: <https://github.com/angular-translate/angular-translate>
- Angular Translate. (2018b, maio 24). *Angular translate n working with IE 10*. Retrieved dezembro 3, 2018, from GitHub: <https://github.com/angular-translate/angular-translate/issues/1473>

- Angular Translate. (2018c). *i18n for your Angular apps, made easy*. Retrieved dezembro 3, 2018, from Angular Translate: <https://angular-translate.github.io/docs/#/guide>
- AngularJS. (2018). *i18n and l10n*. Retrieved dezembro 2018, 3, from AngularJS: <https://docs.angularjs.org/guide/i18n>
- AngularJS. (2019a). *AngularJS*. Retrieved junho 7, 2019, from Services: <https://docs.angularjs.org/guide/services#creating-services>
- AngularJS. (2019b). *Index of 1.3.5/i18n/*. Retrieved junho 14, 2019, from AngularJS: <https://code.angularjs.org/1.3.5/i18n/>
- Apache OpenOffice. (2018). *Globalisation in Software Design*. Retrieved novembro 8, 2018, from Apache OpenOffice: http://www.openoffice.org/specs/collaterals/guides/l18n_in_Software.html
- Araújo, M. (2016, junho 3). *Internacionalização de Softwares e Aplicações*. Retrieved janeiro 11, 2019, from scriptcase blog: <https://www.scriptcaseblog.com.br/internacionalizacao-de-softwares-e-aplicacoes/>
- Atlassian. (2019). *Jira Software*. Retrieved maio 20, 2019, from Atlassian: <https://www.atlassian.com/software/jira>
- Bhatia, M., Tomar, V., & Sharma, A. (2013, agosto). A survey of Software Localization work. *Journal of Global Research in Computer Science*, 4(8), 44-52.
- Carneiro, J., & DIB, L. (2007). Avaliação comparativa do escopo descritivo e explanatório dos principais modelos de internacionalização de empresas. *INTERNEXT - Revista Eletrônica de Negócios Internacionais da ESPM*, 2, 1-25.
- Collins, R. (2002, março/abril). Software localization for Internet software, issues and methods. *IEEE Software*, 19(2), 74-80.
- Dal-Sato, F., Alves, J., Bulé, A., & Amarante, C. (2015). O Processo de Internacionalização da Empresa de Software TOTVS sob a Ótica da Abordagem Comportamental. *REGE Revista de Gestão*, 22, 493-508.
- Dawson, C. (2007). How to Construct Questionnaires. In C. Dawson, *A Practical Guide to Research Methods - A user-friendly manual for mastering research techniques and projects* (3rd ed., pp. 91-104). Oxford, United Kingdom: How To Books. Retrieved janeiro 10, 2019
- DevToolsGuy. (2017, janeiro 25). *L10N and l18N Best Practices*. Retrieved novembro 8, 2018, from Infragistics: <https://www.infragistics.com/community/blogs/b/devtoolsguy/posts/l10n-and-i18n-best-practices>

- DN/Lusa. (2017, dezembro 13). *Metade da população mundial sem acesso a cuidados essenciais de saúde*. Retrieved janeiro 11, 2019, from Diário de Notícias: <https://www.dn.pt/mundo/interior/metade-da-populacao-mundial-sem-acesso-a-cuidados-essenciais-de-saude---oms-e-bm-8982814.html>
- ECMAScript®. (2018a). *Compatibility Table*. Retrieved dezembro 3, 2018, from ECMAScript: <http://kangax.github.io/compat-table/esintl/>
- ECMAScript®. (2018b, junho). *ECMAScript® 2018 Internationalization API Specification - Standard ECMA-402*. Retrieved dezembro 3, 2018, from ECMA International: <https://www.ecma-international.org/publications/files/ECMA-ST/ECMA-402.pdf>
- Eeles, P. (2005). *Non-Functional Requirements*. IBM Software Group. Retrieved janeiro 16, 2019
- Enders, J. (2014, maio 14). *The Definitive Guide to Form Label Positioning*. Retrieved novembro 7, 2018, from SitePoint Pty. Ltd.: <https://www.sitepoint.com/definitive-guide-form-label-positioning/>
- Ferreira, R. (2015, outubro 19). *A sua empresa está pronta para se internacionalizar?* Retrieved janeiro 11, 2019, from Saldo Positivo Empresas: <http://saldopositivo.cgd.pt/empresas/sua-empresa-esta-pronta-para-se-internacionalizar-descubra-em-5-questoes/>
- Figuredo, C., Canhadas, P., & Rojas, A. (2004). Internacionalização de Software - Requisitos Básicos. *Cadernos do IME - Série Informática*, 25-45.
- Git. (2019). *git --distributed-is-the-new-centralized*. Retrieved maio 20, 2019, from git: <https://git-scm.com/>
- Glezos, D. (2018). *Design a Localization-Friendly User Interface: Managing Locale Specific Content (Part 2)*. Retrieved novembro 9, 2018, from Transifex: <https://www.transifex.com/blog/2015/localization-friendly-ui-design-part-2/>
- Globalization and Localization Association | GALA. (2019). *Machine Translation*. Retrieved maio 14, 2019, from GALA | Globalization & Localization Association: <https://www.gala-global.org/what-machine-translation>
- Globalize. (2018). *Globalize*. Retrieved dezembro 3, 2018, from GitHub: <https://github.com/globalizejs/globalize>
- GNU. (2016, julho 10). *Gettext*. Retrieved dezembro 3, 2018, from Gnu Operating System: <https://www.gnu.org/software/gettext/>
- Gross, S. (2006). *Internationalization and Localization of Software*. Michigan, USA: University Eastern Michigan University. Retrieved janeiro 24, 2019

- GruntJS. (2019). *GRUNT The JavaScript Task Runner*. Retrieved janeiro 29, 2019, from GRUNT: <https://gruntjs.com/>
- i18next. (2018). *i18next*. Retrieved dezembro 3, 2018, from GitHub: <https://github.com/i18next/i18next>
- I18nGuy. (2005, outubro 23). *User Interfaces For Right-To-Left Languages*. Retrieved novembro 6, 2018, from Internationalization (I18n), Localization (L10n), Standards, and Amusements: <http://www.i18nguy.com/MiddleEastUI.html>
- IANA. (2018, outubro 26). *Time Zone Database*. Retrieved dezembro 3, 2018, from IANA - Internet Assigned Numbers Authority: <https://www.iana.org/time-zones>
- iLib. (2018). *iLib*. Retrieved dezembro 3, 2018, from SourceForge: <https://sourceforge.net/projects/i18nlib/>
- Ishida, R., & Miller, S. (2018, novembro 02). *Localização vs. Internacionalização*. Retrieved novembro 15, 2018, from World Wide Web Consortium (W3C): <https://www.w3.org/International/questions/qa-i18n.en>
- ISO. (2018). *Graphical symbols*. Retrieved novembro 9, 2018, from ISO Online Browsing Platform (OBP): <https://www.iso.org/obp/ui/#search/grs/>
- Jasmine. (2019). *Behavior-Driven JavaScript*. Retrieved junho 13, 2019, from Jasmine: <https://jasmine.github.io/>
- Koen, P., Ajamian, G., Boyce, S., Clamen, A., Fisher, E., Fountoulakis, S., . . . Seibert, R. (2002). Fuzzy Front End: Effective Methods, Tools, and Techniques. In P. Belliveau, P. Griffin, & S. Somermeyer, *The PDMA ToolBook for new product development* (pp. 5-35). New York: John Wiley & Sons.
- Koen, P., Ajamian, G., Burkart, R., Clamen, A., Davidson, J., D'Amore, R., . . . Wagner, K. (2001, março/abril). Providing clarity and a common language to the "Fuzzy Front End". *Research-Technology Management*, 46-55.
- Laforge, A. (2017, julho 25). *Sayin goodbye to Flash in Chrome*. Retrieved dezembro 19, 2018, from Google Chrome: <https://www.blog.google/products/chrome/saying-goodbye-flash-chrome/>
- Lingoport. (2019). *Software Globalization that's Agile and Easy*. Retrieved maio 14, 2019, from Lingoport: <https://lingoport.com/software-internationalization-products/>
- Lokalise. (2019). *A better way to localize*. Retrieved maio 14, 2019, from Lokalise: <https://localise.co/>
- Lopes, M. (2018, maio 22). *Temos mesmo o melhor Serviço Nacional de Saúde d mundo? Nem por isso*. Retrieved janeiro 11, 2019, from Observador:

<https://observador.pt/especiais/temos-mesmo-o-melhor-servico-nacional-de-saude-do-mundo-nem-por-isso/>

Microsoft. (2007, dezembro). *Web-Application Internationalization*. Retrieved dezembro 13, 2018, from Microsoft: <https://msdn.microsoft.com/en-us/library/cc168605.aspx>

Microsoft. (2016a, novembro 22). *Font Technology*. Retrieved novembro 6, 2018, from Microsoft: <https://docs.microsoft.com/en-us/globalization/input/font-technology>

Microsoft. (2016b, novembro 15). *Internationalization*. Retrieved dezembro 13, 2018, from Microsoft: <https://docs.microsoft.com/pt-pt/globalization/design/internationalization>

Microsoft. (2016c, março 16). *Time Formatting*. Retrieved dezembro 7, 2018, from Microsoft: <https://docs.microsoft.com/en-us/globalization/locale/time-formatting>

Microsoft. (2017a, janeiro 24). *Date Formatting*. Retrieved dezembro 7, 2018, from Microsoft: <https://docs.microsoft.com/en-us/globalization/locale/date-formatting>

Microsoft. (2017b, janeiro 26). *Number Formatting*. Retrieved dezembro 7, 2018, from Microsoft: <https://docs.microsoft.com/en-us/globalization/locale/number-formatting>

Microsoft. (2017c, julho 14). *Software Internationalization*. Retrieved janeiro 11, 2019, from Microsoft: <https://docs.microsoft.com/en-us/globalization/software-internationalization>

Microsoft. (2017d, agosto 08). *Using Images and Icons*. Retrieved novembro 8, 2018, from Microsoft: <https://docs.microsoft.com/en-us/globalization/localizability/images-and-icons>

Microsoft. (2018). *Designing a World-Ready Program*. Retrieved dezembro 13, 2018, from Microsoft: <https://msdn.microsoft.com/en-us/library/bb978440.aspx>

Moen, Ø., Gavlen, M., & Endresen, I. (2004). Internationalization of Small, Computer Software Firms: Entry Forms and Market Selection. *European Journal of Marketing*, 38(9/10), 1236-1251.

Mozilla and individual contributors. (2018, dezembro 10). *<label>*. Retrieved dezembro 12, 2018, from MDN web docs: <https://developer.mozilla.org/en-US/docs/Web/HTML/Element/label>

Nicola, S. (2018a). *Análise_Vvalor_Aula_4_20NOV_2018_1hora_AHP*.

Nicola, S. (2018b). *Análise_Vvalor_Aula2_14NOV_2018_1hora*.

Nidhra, S., & Dondeti, J. (2019). Black box and white box testing techniques - a literature review. *International Journal of Embedded Systems and Applications (IJESA)*, 29-50. Retrieved junho 13, 2019

- OASIS. (2008). *XLIFF*. Retrieved maio 14, 2019, from OASIS: <http://docs.oasis-open.org/xliff/v1.2/os/xliff-core.html>
- Ortiz-Ospina, E., & Roser, M. (2019). *Global Health*. Retrieved janeiro 31, 2019, from Our World in Data: <https://ourworldindata.org/health-meta>
- Owen, S. (2018, março 13). *Moving From Flash To HTML, CSS, And JavaScript*. Retrieved janeiro 28, 2019, from Smashing Magazine: <https://www.smashingmagazine.com/2018/03/from-flash-html-css-javascript/>
- Penzo, M. (2006, julho 12). *Label Placement in Forms*. Retrieved novembro 7, 2018, from UXmatters: <https://www.uxmatters.com/mt/archives/2006/07/label-placement-in-forms.php>
- Pereira, N. (2018). *escrita_tecnico_cientifica*.
- PhraseApp. (2019a). *Gestão de Tradução Feita para Programadores*. Retrieved maio 14, 2019, from PhraseApp: <https://phraseapp.com/pt/features/developers>
- PhraseApp. (2019b). *PhraseApp API Reference*. Retrieved maio 14, 2019, from PhraseApp: <https://developers.phraseapp.com/api/>
- PhraseApp. (2019c). *Recurso para Gestores*. Retrieved maio 14, 2019, from PhraseApp: <https://phraseapp.com/pt/features/managers>
- Pinochet, L. (2011). Tendências de Tecnologia de Informação na Gestão da Saúde. *O Mundo da Saúde*, 382-394.
- Pinto, P. (2015, dezembro 04). *MongoDB - A base de dados que não é relacional*. Retrieved junho 12, 2019, from pplware: <https://pplware.sapo.pt/software/mongodb-a-base-de-dados-que-nao-e-relacional/>
- PORDATA. (2019). *Esperança de vida à nascença: total e por sexo*. Retrieved janeiro 11, 2019, from PORDATA: <https://www.pordata.pt/Europa/Esperan%C3%A7a+de+vida+%C3%A0+nascen%C3%A7a+total+e+por+sexo-1260>
- Pudełek, J. (2017, julho 25). *Developer News - Migrating Games from Flash to Open Web Standards on Facebook*. Retrieved dezembro 19, 2018, from Facebook for developers: <https://developers.facebook.com/blog/post/2017/07/25/Games-Migration-to-Open-Web-Standards/>
- Resig, J. (2018). *Node.js: i18n-2*. Retrieved dezembro 3, 2018, from GitHub: <https://github.com/jeresig/i18n-node-2>
- Rich, N., & Holweg, M. (2000). *Value Analysis Value Engineering*. United Kingdom: Lean Enterprise Research Centre.

- Rocha, M., Andrade, R., & Leitão, M. (2012). *A internacionalização como estratégia de crescimento para as empresas de software*. Simpósio de Excelência em Gestão e Tecnologia.
- Saaty, T. (1991). *Método de Análise Hierárquica*. São Paulo: Makron Books.
- Saito, J. (2017, janeiro 03). *Design for internationalization*. Retrieved novembro 8, 2018, from A Medium Corporation: <https://medium.com/dropbox-design/design-for-internationalization-24c12ea6b38f>
- Scrum.org. (2019). *What is Scrum?* Retrieved maio 20, 2019, from Scrum.org: https://www.scrum.org/resources/what-is-scrum?gclid=EAlaIqobChMI-Yitq5iq4glVxBbTCh1-FgPZEAAYASAAEgJREvD_BwE
- SDL. (2019). *SDL Passolo*. Retrieved maio 14, 2019, from SDL: <https://www.sdl.com/software-and-services/translation-software/software-localization/sdl-passolo/>
- Serviço Nacional de Saúde. (2018). *Retrato da Saúde em Portugal*. Retrieved fevereiro 10, 2019, from Serviço Nacional de Saúde: <https://www.sns.gov.pt/retrato-da-saude-2018/>
- Smartling. (2019a). *Translation engineered for the future*. Retrieved maio 14, 2019, from Smartling: <https://www.smartling.com/products/automate/>
- Smartling. (2019b). *We speak your language, too*. Retrieved maio 14, 2019, from Smartling: <https://www.smartling.com/>
- Smartling. (2019c). *Your end-user experience matters*. Retrieved maio 14, 2019, from Smartling: <https://www.smartling.com/why-smartling/>
- Smedberg, B. (2017, julho 25). *Firefox Roadmap for Flash End-of-Life*. Retrieved dezembro 19, 2018, from Mozilla: <https://blog.mozilla.org/futurereleases/2017/07/25/firefox-roadmap-flash-end-life/>
- Transifex. (2019a). *A localization platform that moves as fast as you do*. Retrieved maio 14, 2019, from Transifex: <https://www.transifex.com/>
- Transifex. (2019b). *Easily localize all your digital content from start to finish*. Retrieved maio 14, 2019, from Transifex: <https://www.transifex.com/how-it-works/>
- Transifex. (2019c). *Features Overview*. Retrieved maio 14, 2019, from Transifex: <https://www.transifex.com/features/>
- Unicode. (2018, outubro 15). *Unicode CLDR Project*. Retrieved dezembro 3, 2018, from CLDR - Unicode Common Locale Data Repository: <http://cldr.unicode.org/>
- Unicode, Inc. (2017, julho 24). *What is Unicode?* Retrieved novembro 6, 2018, from The Unicode Consortium: <http://www.unicode.org/standard/WhatIsUnicode.html>

- Wang, S. (2016, março 09). *Designing for a Truly World Wide Web*. Retrieved novembro 8, 2018, from A Medium Corporation: <https://medium.com/salesforce-ux/designing-for-a-truly-world-wide-web-b47eb080f412>
- Warwick Manufacturing Group. (2007). Quality Function Deployment. In *Product Excellence using Six Sigma* (p. Section 6). Coventry: University of Warwick.
- Wikimedia. (2018). *jQuery.i18n*. Retrieved dezembro 3, 2018, from GitHub: <https://github.com/wikimedia/jquery.i18n>
- Wood, A. (2015-2018). *<label> HTML Tag*. Retrieved novembro 7, 2018, from HTML.com: <https://html.com/tags/label/>
- Woodall, T. (2003). Conceptualising Value for the Customer: an Attributional, Structural And Dispositional Analysis. *Academy of Marketing Science Review* nº12, 1-42.
- World Wide Web Consortium. (2017, dezembro 14). *HTML 5.2*. Retrieved dezembro 19, 2018, from World Wide Web Consortium: <https://www.w3.org/TR/html5/introduction.html>
- Yahoo! Inc. (2014). *Format.js - Internationalize your web apps on the client & server*. Retrieved dezembro 3, 2018, from Format.js: <https://formatjs.io/>

Anexo A Suíte de produtos ALERT

A empresa ALERT é detentora de uma vasta gama de soluções de *software* relacionadas com a área da saúde (ALERT® Life Sciences Computing, 2018a).



Figura 69 – Suíte de produtos ALERT® (ALERT® Life Sciences Computing, 2017)

Verificando a Figura 69 constata-se que os produtos ALERT® se dividem em 8 grandes áreas: EMR, HIE, PHR, BI, PLANNING SYSTEM, ONLINE STORE, eLEARNING e STUDENT.

O ALERT® EMR (ALERT® *ELECTRONIC MEDICAL RECORD*) engloba um conjunto de soluções de *software* clínico que permitem informatizar as ações que ocorrem num hospital ou centro de saúde e assim tornar estes locais livres de papel (“*paper-free*”) (ALERT® Life Sciences Computing, 2017).

O ALERT® HIE (ALERT® *HEALTH INFORMATION EXCHANGE*) é uma infraestrutura de integração que disponibiliza o acesso aos diferentes dados clínicos de um paciente, sendo que estes dados são originários de diferentes soluções de *software*. Pode-se assim dizer que o ALERT® HIE é uma ferramenta de interligação de dados entre diversos *softwares* clínicos (ALERT® Life Sciences Computing, 2017).

O ALERT® PHR (ALERT® *PERSONAL HEALTH RECORD*) é uma aplicação que permite que um paciente consulte os seus registos médicos para além de poder inserir dados na aplicação com o intuito de completar da melhor forma a sua ficha clínica. O ALERT® PHR é também conhecido por MyALERT® (ALERT® Life Sciences Computing, 2017).

O ALERT® BI (ALERT® *BUSINESS INTELLIGENCE*), anteriormente conhecido como ALERT® *DATA WAREHOUSE* (ALERT® ADW), pode ser descrito como uma ferramenta de análise de dados que suporta a gestão dos serviços de saúde (ALERT® Life Sciences Computing, 2017).

Relativamente ao ALERT® *PLANNING SYSTEM*, este *software* é uma aplicação para múltiplas funções, desde a gestão tática e estratégica até ao planeamento de operações nos serviços de saúde do local onde se encontra a ser utilizado (ALERT® Life Sciences Computing, 2017).

A ALERT® *ONLINE STORE* permite aos utilizadores pesquisarem e comprarem produtos da empresa, sendo que para tal podem consultar demonstrações dos produtos que pretendem adquirir antes de o fazer, para além de que podem consultar detalhes das funcionalidades principais e experimentar uma versão grátis dos mesmos, entre muitas outras operações (ALERT® Life Sciences Computing, 2017).

O ALERT® *eLEARNING* é uma plataforma que permite aos utilizadores da mesma fazerem formações ajustadas às suas necessidades e experimentar os produtos da empresa, com o intuito de aprender ou aperfeiçoar as suas competências (ALERT® Life Sciences Computing, 2017).

Quanto ao ALERT® *STUDENT*, este produto é uma base de conhecimento e aprendizagem desenvolvida com o intuito de ajudar tanto os estudantes como os professores da área da saúde a realizarem os seus estudos e investigações (ALERT® Life Sciences Computing, 2017).

ALERT® EMR:

O âmbito deste projeto encontra-se principalmente focado numa das soluções disponibilizadas pelo produto ALERT® EMR, o ALERT® PFH. Apesar deste foco, todo o ALERT® EMR é importante, pelo que se apresentará em maior pormenor as soluções que este produto possui, para além de que se referencia os locais do globo onde estas soluções se encontram disponíveis e em operação:

- ALERT® *PAPER FREE HOSPITAL* (ALERT® PFH) – é uma solução de *software* que tem o intuito de informatizar os processos que ocorrem num hospital, tornando possível documentar, rever e integrar todas as informações e estado clínico dos pacientes do hospital, em tempo real. Esta solução encontra-se dividida em diferentes especificidades, relacionadas com os diferentes ambientes de um hospital, como o ALERT® *TRIAGE* (*software* de triagem para o serviço de urgência), ALERT® *EMERGENCY DEPARTMENT INFORMATION SYSTEM* (ALERT® EDIS – solução completa para os serviços de urgência), ALERT® *OBSERVATION ROOM* (ALERT® OBS – para registar informação relativa a um episódio de internamento ou de observação num serviço de urgência), ALERT® *PATIENT DATA MANAGEMENT SYSTEM* (ALERT® PDMS – permite recolher dados de dispositivos de monitorização clínica e associa-los a um paciente), ALERT® *WAITING ROOM* (aplicação para gestão das filas de espera de um hospital como por exemplo as do serviço de urgências), ALERT® *OUTPATIENT* (*software* clínico e de gestão de unidades de consulta externa), ALERT® *INPATIENT* (*software* clínico e de gestão de unidades de internamento), ALERT® *OPERATING ROOM INFORMATION SYSTEM* (ALERT® ORIS – solução utilizada em blocos operatórios), ALERT® *PHARMACY* (permite a prescrição, administração, validação, aviamento e gestão de fármacos) e ALERT® *GP PORTAL* (solução que permite a comunicação entre médicos de clínica geral

e outros profissionais de saúde de diversas instituições) (ALERT® Life Sciences Computing, 2017). Esta solução encontra-se representada em mercados como o Brasil, Chile, França, Itália, Kuwait, México, Portugal, Suíça, Reino Unido e Estados Unidos da América (ALERT® Life Sciences Computing, 2017).

- ALERT® *CARE* é uma solução desenvolvida para os centros de saúde com o intuito de disponibilizar informações relativas aos pacientes, informações essas que podem ter origem noutros locais que não o próprio centro de saúde. Permite, principalmente, registar a documentação do paciente, atualizar o seu estado clínico e revê-lo. Está disponível no mercado brasileiro e no mercado português (ALERT® Life Sciences Computing, 2017).
- ALERT® *PRIVATE PRACTICE* é um sistema de informação médica para clínicas privadas e pequenas instituições de saúde que possui determinadas características que variam consoante as necessidades destes locais onde estão disponíveis. A principal funcionalidade do sistema é gerir e documentar o processo clínico dos seus pacientes. Atualmente é utilizado apenas em Portugal (ALERT® Life Sciences Computing, 2017).
- ALERT® *REFERRAL*, também conhecido como ALERT® P1, é um sistema disponibilizado para realizar a triagem e a transferência de pacientes entre diferentes instituições de saúde (por exemplo centros de saúde e hospitais). No momento é utilizado em Portugal e no Chile (ALERT® Life Sciences Computing, 2017).
- ALERT® *HOME VISIT* é uma solução desenvolvida para dispositivos móveis, com o principal objetivo de ajudar diferentes profissionais de saúde a realizarem o registo clínico *off-line* aquando da realização de visitas ao domicílio. É utilizado por profissionais de saúde no mercado brasileiro (ALERT® Life Sciences Computing, 2017).

Anexo B Consistência dos Critérios

- Consistência do critério de Suporte por parte da comunidade:

$$Matriz\ prioridades \times Vetor\ próprio \cong \lambda_{max} \times Vetor\ próprio$$

$$\Leftrightarrow \begin{bmatrix} 1 & \frac{1}{3} & \frac{1}{4} & 5 & 6 & \frac{1}{3} \\ 3 & 1 & \frac{1}{3} & 6 & 7 & 2 \\ 4 & 3 & 1 & 7 & 8 & 3 \\ \frac{1}{5} & \frac{1}{6} & \frac{1}{7} & 1 & 2 & \frac{1}{6} \\ \frac{1}{6} & \frac{1}{7} & \frac{1}{8} & \frac{1}{2} & 1 & \frac{1}{7} \\ 3 & \frac{1}{2} & \frac{1}{3} & 6 & 7 & 1 \end{bmatrix} \times \begin{bmatrix} 0,12 \\ 0,23 \\ 0,39 \\ 0,04 \\ 0,03 \\ 0,19 \end{bmatrix} \cong \lambda_{max} \times \begin{bmatrix} 0,12 \\ 0,23 \\ 0,39 \\ 0,04 \\ 0,03 \\ 0,19 \end{bmatrix}$$

$$\Leftrightarrow \begin{bmatrix} 0,73 \\ 1,55 \\ 2,65 \\ 0,25 \\ 0,18 \\ 1,25 \end{bmatrix} \cong \lambda_{max} \times \begin{bmatrix} 0,12 \\ 0,23 \\ 0,39 \\ 0,04 \\ 0,03 \\ 0,19 \end{bmatrix}$$

$$\Leftrightarrow \lambda_{max} \cong 6,40$$

Tendo o valor do λ_{max} calcula-se agora o valor do IC:

$$IC = \frac{\lambda_{max} - n}{n - 1} = \frac{6,40 - 6}{6 - 1} = 0,08$$

Por fim, calcula-se o RC onde o valor de IR, segundo a tabela definida pelo Laboratório Nacional de Oak Ridge, é de 1,24 devido à avaliação de 6 critérios:

$$RC = \frac{IC}{1,24} = \frac{0,08}{1,24} \cong 0,06$$

Como se verifica, o valor da Razão de Consistência (0,006) é inferior a 0,1 pelo que os valores das prioridades são consistentes.

- Consistência do critério de Alcance de diversos tratamentos de problemas de i18n e l10n:

$$Matriz\ prioridades \times Vetor\ próprio \cong \lambda_{max} \times Vetor\ próprio$$

$$\Leftrightarrow \begin{bmatrix} 1 & \frac{1}{4} & \frac{1}{3} & \frac{1}{4} & \frac{1}{2} & \frac{1}{4} \\ 4 & 1 & \frac{1}{2} & \frac{1}{2} & 2 & \frac{1}{2} \\ 3 & 2 & 1 & \frac{1}{3} & 2 & \frac{1}{2} \\ 4 & 2 & 3 & 1 & 3 & 2 \\ 2 & \frac{1}{2} & \frac{1}{2} & \frac{1}{3} & 1 & \frac{1}{2} \\ 4 & 2 & 2 & \frac{1}{2} & 2 & 1 \end{bmatrix} \times \begin{bmatrix} 0,06 \\ 0,15 \\ 0,16 \\ 0,32 \\ 0,09 \\ 0,22 \end{bmatrix} \cong \lambda_{max} \times \begin{bmatrix} 0,06 \\ 0,15 \\ 0,16 \\ 0,32 \\ 0,09 \\ 0,22 \end{bmatrix}$$

$$\Leftrightarrow \begin{bmatrix} 0,33 \\ 0,92 \\ 1,04 \\ 2,05 \\ 0,58 \\ 1,42 \end{bmatrix} \cong \lambda_{max} \times \begin{bmatrix} 0,06 \\ 0,15 \\ 0,16 \\ 0,32 \\ 0,09 \\ 0,22 \end{bmatrix}$$

$$\Leftrightarrow \lambda_{max} \cong 6,24$$

Tendo o valor do λ_{max} calcula-se agora o valor do IC:

$$IC = \frac{\lambda_{max} - n}{n - 1} = \frac{6,24 - 6}{6 - 1} = 0,048$$

Por fim, calcula-se o RC onde o valor de IR, segundo a tabela definida pelo Laboratório Nacional de Oak Ridge, é de 1,24 devido à avaliação de 6 critérios:

$$RC = \frac{IC}{1,24} = \frac{0,048}{1,24} \cong 0,04$$

Como se verifica, o valor da Razão de Consistência (0,04) é inferior a 0,1 pelo que os valores das prioridades são consistentes.

- Consistência do critério de Adoção de *standards* internacionais:

$$Matriz\ prioridades \times Vetor\ próprio \cong \lambda_{max} \times Vetor\ próprio$$

$$\Leftrightarrow \begin{bmatrix} 1 & \frac{1}{3} & 3 & 2 & \frac{1}{2} & 2 \\ 3 & 1 & 4 & 3 & 2 & 3 \\ 1 & \frac{1}{3} & 1 & \frac{1}{2} & 1 & 1 \\ \frac{1}{3} & \frac{1}{4} & 1 & 2 & \frac{1}{3} & 2 \\ \frac{1}{2} & \frac{1}{3} & 2 & 1 & \frac{1}{3} & 1 \\ 2 & \frac{1}{2} & 3 & 3 & 1 & 3 \\ \frac{1}{2} & \frac{1}{3} & 2 & 1 & \frac{1}{3} & 1 \end{bmatrix} \times \begin{bmatrix} 0,16 \\ 0,34 \\ 0,06 \\ 0,10 \\ 0,24 \\ 0,10 \end{bmatrix} \cong \lambda_{max} \times \begin{bmatrix} 0,16 \\ 0,34 \\ 0,06 \\ 0,10 \\ 0,24 \\ 0,10 \end{bmatrix}$$

$$\Leftrightarrow \begin{bmatrix} 0,97 \\ 2,14 \\ 0,38 \\ 0,59 \\ 1,51 \\ 0,59 \end{bmatrix} \cong \lambda_{max} \times \begin{bmatrix} 0,16 \\ 0,34 \\ 0,06 \\ 0,10 \\ 0,24 \\ 0,10 \end{bmatrix}$$

$$\Leftrightarrow \lambda_{max} \cong 6,13$$

Tendo o valor do λ_{max} calcula-se agora o valor do IC:

$$IC = \frac{\lambda_{max} - n}{n - 1} = \frac{6,13 - 6}{6 - 1} = 0,026$$

Por fim, calcula-se o RC onde o valor de IR, segundo a tabela definida pelo Laboratório Nacional de Oak Ridge, é de 1,24 devido à avaliação de 6 critérios:

$$RC = \frac{IC}{1,24} = \frac{0,026}{1,24} \cong 0,02$$

Como se verifica, o valor da Razão de Consistência (0,02) é inferior a 0,1 pelo que os valores das prioridades são consistentes.

- Consistência do critério de Compatibilidade com diversos navegadores:

$$Matriz\ prioridades \times Vetor\ próprio \cong \lambda_{max} \times Vetor\ próprio$$

$$\Leftrightarrow \begin{bmatrix} 1 & \frac{1}{2} & 1 & 1 & \frac{1}{2} & \frac{1}{2} \\ 2 & 1 & 2 & 2 & \frac{1}{2} & \frac{1}{2} \\ 1 & \frac{1}{2} & 1 & 1 & \frac{1}{2} & \frac{1}{2} \\ 1 & \frac{1}{2} & 1 & 1 & \frac{1}{2} & \frac{1}{2} \\ 2 & 2 & 2 & 2 & 1 & 1 \\ 2 & 2 & 2 & 2 & 1 & 1 \end{bmatrix} \times \begin{bmatrix} 0,11 \\ 0,17 \\ 0,11 \\ 0,11 \\ 0,25 \\ 0,25 \end{bmatrix} \cong \lambda_{max} \times \begin{bmatrix} 0,11 \\ 0,17 \\ 0,11 \\ 0,11 \\ 0,25 \\ 0,25 \end{bmatrix}$$

$$\Leftrightarrow \begin{bmatrix} 0,67 \\ 1,08 \\ 0,67 \\ 0,67 \\ 1,5 \\ 1,5 \end{bmatrix} \cong \lambda_{max} \times \begin{bmatrix} 0,11 \\ 0,17 \\ 0,11 \\ 0,11 \\ 0,25 \\ 0,25 \end{bmatrix}$$

$$\Leftrightarrow \lambda_{max} \cong 6,10$$

Tendo o valor do λ_{max} calcula-se agora o valor do IC:

$$IC = \frac{\lambda_{max} - n}{n - 1} = \frac{6,10 - 6}{6 - 1} = 0,02$$

Por fim, calcula-se o RC onde o valor de IR, segundo a tabela definida pelo Laboratório Nacional de Oak Ridge, é de 1,24 devido à avaliação de 6 critérios:

$$RC = \frac{IC}{1,24} = \frac{0,02}{1,24} \cong 0,02$$

Como se verifica, o valor da Razão de Consistência (0,02) é inferior a 0,1 pelo que os valores das prioridades são consistentes.

Anexo C Grunt

O Grunt é um executor de tarefas JavaScript (GruntJS, 2019). Foi desenvolvido com o intuito de automatizar tarefas repetitivas de um projeto, como por exemplo compilação e testes unitários. A sua configuração é realizada através de um ficheiro denominado “Gruntfile”. Este ficheiro deve ser criado dentro de um projeto. Nele são definidas diferentes tarefas e *plugins* que ajudam na automatização de código (GruntJS, 2019). No ficheiro “Gruntfile” do Extrato de código 25, existem duas tarefas criadas, “*jshint*” e “*watch*”, bem como a definição de dois *plugins*, “*grunt-contrib-jshint*” e “*grunt-contrib-watch*”.

```
module.exports = function(grunt) {

  grunt.initConfig({
    jshint: {
      files: ['Gruntfile.js', 'src/**/*.js', 'test/**/*.js'],
      options: {
        globals: {
          jQuery: true
        }
      }
    },
    watch: {
      files: ['<%= jshint.files %>'],
      tasks: ['jshint']
    }
  });

  grunt.loadNpmTasks('grunt-contrib-jshint');
  grunt.loadNpmTasks('grunt-contrib-watch');

  grunt.registerTask('default', ['jshint']);

};
```

Extrato de código 25 – Exemplo de um “Gruntfile”

Anexo D Ranch

O Ranch, é uma aplicação que permite aceder e testar componentes gráficos existentes na interface do utilizador dos produtos ALERT® de forma independente e assim verificar os seus comportamentos.

O Ranch permite, visualmente, verificar os conteúdos desenvolvidos e averiguar qual o comportamento associado aos mesmos. Assim, consegue-se testar as vistas do utilizador e os componentes gráficos, sendo que são utilizados dados simulados para verificação dos mesmos.

Cada um dos componentes gráficos representados no Ranch apresentam uma parte da interface gráfica mostrada ao utilizador, desde os botões utilizados e associados a ações despoletadas no produto, assim como a visualização de dados textuais e de como estes se encontram adaptados à interface

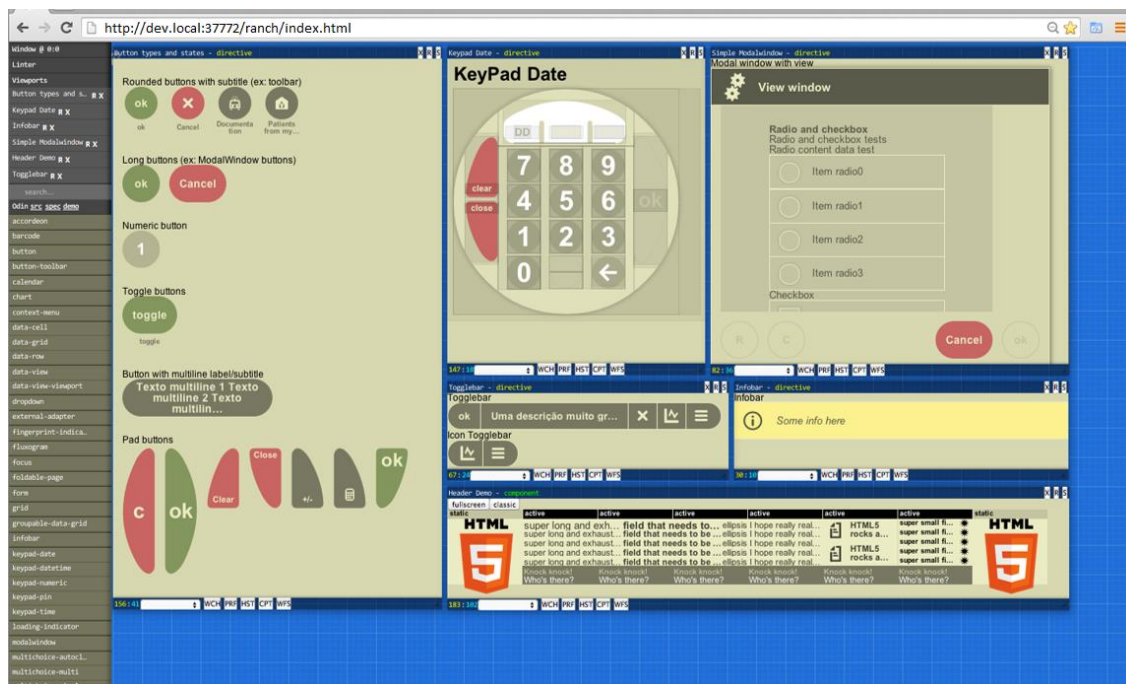


Figura 70 – Plataforma Ranch

Anexo E Ficheiro “angular-locale-pt-pt”

```
'use strict';
angular.module("ngLocale", [], ["$provide", function($provide) {
var PLURAL_CATEGORY = {ZERO: "zero", ONE: "one", TWO: "two", FEW: "few",
MANY: "many", OTHER: "other"};
$provide.value("$locale", {
  "DATETIME_FORMATS": {
    "AMPMS": [
      "da manh\u00e3",
      "da tarde"
    ],
    "DAY": [
      "domingo",
      "segunda-feira",
      "ter\u00e7a-feira",
      "quarta-feira",
      "quinta-feira",
      "sexta-feira",
      "s\u00e1bado"
    ],
    "MONTH": [
      "janeiro",
      "fevereiro",
      "mar\u00e7o",
      "abril",
      "maio",
      "junho",
      "julho",
      "agosto",
      "setembro",
      "outubro",
      "novembro",
      "dezembro"
    ],
    "SHORTDAY": [
      "dom",
      "seg",
      "ter",
      "qua",
      "qui",
      "sex",
      "s\u00e1b"
    ],
    "SHORTMONTH": [
      "jan",
      "fev",
      "mar",
      "abr",
      "mai",
      "jun",
      "jul",
      "ago",
      "set",
      "out",
      "nov",
      "dez"
    ],
  },

```

```

    "fullDate": "EEEE, d 'de' MMMM 'de' y",
    "longDate": "d 'de' MMMM 'de' y",
    "medium": "dd/MM/y HH:mm:ss",
    "mediumDate": "dd/MM/y",
    "mediumTime": "HH:mm:ss",
    "short": "dd/MM/yy HH:mm",
    "shortDate": "dd/MM/yy",
    "shortTime": "HH:mm"
  },
  "NUMBER_FORMATS": {
    "CURRENCY_SYM": "\u20ac",
    "DECIMAL_SEP": ",",
    "GROUP_SEP": "\u00a0",
    "PATTERNS": [
      {
        "gSize": 3,
        "lgSize": 3,
        "maxFrac": 3,
        "minFrac": 0,
        "minInt": 1,
        "negPre": "-",
        "negSuf": "",
        "posPre": "",
        "posSuf": ""
      },
      {
        "gSize": 3,
        "lgSize": 3,
        "maxFrac": 2,
        "minFrac": 2,
        "minInt": 1,
        "negPre": "-",
        "negSuf": "\u00a0\u00a4",
        "posPre": "",
        "posSuf": "\u00a0\u00a4"
      }
    ]
  },
  "id": "pt-pt",
  "pluralCat": function(n, opt_precision) { if (n >= 0 && n <= 2 && n !=
2) { return PLURAL_CATEGORY.ONE; } return PLURAL_CATEGORY.OTHER;}
});
}]);

```

Extrato de código 26 – Ficheiro “angular-locale-pt-pt”

Anexo F Estado de desenvolvimento de requisitos

Na Tabela 31 apresenta-se o estado de desenvolvimento dos requisitos funcionais definidos pelas partes interessadas.

Tabela 31 – Estado de desenvolvimento de requisitos

Requisito	Estado de desenvolvimento
Alterar facilmente o local em vigor nos produtos para poder disponibilizá-los segundo os requisitos desse novo local	Realizado
Verificar que todos os caracteres que se introduz nos produtos são bem representados para que a informação introduzida seja disponibilizada adequadamente	Realizado
Todos os textos da interface do utilizador sejam disponibilizados com os caracteres adequados para que o utilizador final utilize os produtos corretamente	Realizado
A fonte de texto disponibilizada seja adequada ao local onde os produtos são utilizados para que o utilizador final consulte adequadamente o texto	Realizado
A direção de texto apresentada nos produtos seja a mais indicada, consoante o local/país onde são disponibilizados, para que os utilizadores finais consigam os utilizar da forma correta	Realizado
As <i>labels</i> apresentadas estejam corretamente posicionadas sobre o campo de <i>input</i> para que o utilizador final da aplicação considere o <i>layout</i> dos produtos o mais apelativo possível	Realizado
Consoante o local/país onde são disponibilizados os produtos, o texto apresentado esteja corretamente traduzido para que o utilizador final o consiga perceber e assim utilizar adequadamente os produtos	Realizado
A data e hora apresentadas nos produtos esteja de acordo com o formato utilizado no local/país onde estes são disponibilizados para que o cliente final consulte estes dados consoante aquilo que espera verificar	Realizado
Os números apresentados nos produtos sejam mostrados num formato numérico utilizado no local/país onde os produtos são disponibilizados para que não cause confusão e má interpretação na sua leitura aos utilizadores	Realizado
O separador decimal e dos milhares dos dados numéricos sejam mostrados de acordo com o que é utilizado no local/país onde os produtos são disponibilizados para que não cause má interpretação aos utilizadores	Realizado

Requisito	Estado de desenvolvimento
O símbolo monetário apresentado seja adequado ao local/país onde o produto é consultado para que o utilizador final consiga entender os dados disponibilizados	Realizado
Ter um processo que auxilie a internacionalização e a localização dos produtos para que o tempo necessário para realizar tal atividade diminua	Realizado
Permitir que os clientes personalizem características relacionadas com a internacionalização e localização como o caso das traduções para que estes se encontrem o mais satisfeitos possível com o produto disponibilizado	Realizado
As personalizações exigidas pelos clientes da empresa não sejam eliminadas quando é lançada uma nova versão dos produtos para não se perder a consistência dos mesmos	Realizado
Alterar facilmente a camada de apresentação dos produtos para que seja disponibilizada ao cliente a versão mais correta destes	Realizado