



GesSAD - Gestão de serviços de apoio domiciliário

JOSÉ MIGUEL RIBEIRO FERREIRA

Outubro de 2022

GesSAD - Gestão de serviços de apoio domiciliário

José Miguel Ribeiro Ferreira

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

Orientador: Ana Maria Madureira

Porto, Outubro de 2022

“O sucesso não está apenas na conquista,
mas sim em todo o percurso.”

Mahatma Gandhi

Resumo

Considerando a aplicabilidade, diversidade e constante evolução na área da Informática, é fundamental desenvolver estratégias inovadoras e economicamente viáveis, usando as mais recentes ferramentas disponíveis, de forma prática e universal.

O atual contexto de evolução das Tecnologias de Informação requer que, cada vez mais, os diferentes processos de negócio de uma empresa sejam “digitais”. Esta evolução tem sido uma mais-valia para a área da saúde, desde a telemedicina, monitorização do estado de saúde dos utentes, registos informáticos, etc. contribuindo assim para a melhoria da qualidade dos cuidados prestados.

Atualmente, a empresa tem um processo moroso, com os registos e no acesso aos dados devido à sua dispersão, necessitando de mais recursos para a supervisão, o que implica custos acrescidos e mais sujeito a falhas. Como não possui ferramentas de apoio, faz a gestão dos utentes, funcionários, horários e planos de cuidados com registo em papel e documentos Excel¹.

Este projeto permitirá a informatização dos processos utilizados na empresa **Vida e Conforto**, o que levará a uma redução de custos, de material utilizado (nomeadamente papel) e desperdício de tempos na pesquisa de informação.

Com esta dissertação pretende-se descrever o processo de desenvolvimento do trabalho realizado no âmbito da unidade curricular Tese/Dissertação/Estágio do Mestrado em Engenharia Informática, na área de especialização em Engenharia de *Software* que tem como propósito desenvolver uma aplicação *web* progressiva de apoio à gestão interna. O sistema desenvolvido permitirá à **Vida e Conforto** melhorar os registos dos funcionários, dos utentes, dos planos de cuidados, dos serviços prestados e melhorar a gestão de horários. Esta aplicação vai possibilitar o acesso aos dados em tempo real, trazendo ganhos para os utentes e para a empresa.

Palavras-chave: Aplicação *web* progressiva, Saúde, Apoio Domiciliário, Idosos, Utes

¹ Os registos recolhidos em papel são depois tratados em folhas de cálculo

Abstract

Considering the applicability, diversity and constant evolution in the area of Information Technology, it is essential to develop innovative and economically viable strategies, using the latest tools available, in a practical and universal way.

The current context of Information Technology evolution requires that, increasingly, the different business processes of a company must be "digital". This evolution has been an asset to the health area, from telemedicine, monitoring the health status of users, computerized records, etc. thus contributing to the improvement of the quality of care.

Currently, the company has a slow process with the records and in accessing the data due to its dispersion, requiring more resources for supervision, which implies increased costs and more subject to failures. As it does not have support tools, it manages users, employees, schedules and care plans with paper records and Excel documents.

This project will allow the computerization of the processes used in the company **Vida e Conforto**, which will lead to a reduction of costs, of material used (namely paper) and waste of time in the search for information.

This dissertation aims to describe the development process of the work done in the scope of the course unit Thesis/Dissertation/Internship of the Master in Computer Engineering, in the specialization area of Software Engineering, with the purpose of developing a progressive web application to support internal management. The developed system will allow **Vida e Conforto** to improve the records of employees, users, care plans, services provided, and to improve the management of schedules. This application will allow access to data in real time, bringing gains to the users and to the company.

Keywords: Progressive Web application, Health, Home Support, Seniors, Patients.

Agradecimentos

A concretização deste Projeto não teria sido possível sem a colaboração de um conjunto de pessoas que, ao longo da minha vida, me ajudaram a chegar até esta fase final.

À minha orientadora desta Instituição, Doutora Ana Maria Madureira que aceitou orientar-me e me guiou nas várias etapas, sendo o seu tributo crucial na concretização deste projeto.

À Vida e Conforto e aos seus colaboradores pelo excelente ambiente de trabalho proporcionado e pela disponibilidade em ajudar nos mais diversos problemas e dificuldades que foram surgindo ao longo de todo o projeto.

Aos meus pais e à minha família, por todo o apoio demonstrado ao longo do meu percurso académico e tudo o que fizeram para que isto fosse possível.

À minha avó e avô que sempre me incentivaram a completar e, sem a força deles nada seria igual, guardo na memória a força que sempre me proporcionaram.

À minha namorada Lurdes por toda a paciência, amor e amizade, sem ela tudo teria sido ainda mais difícil, obrigada por acreditares em mim, por seres a minha melhor amiga, pelo abraço que tantas vezes me reconfortou.

Aos colegas de turma, pelo apoio que sempre demonstraram, sem o qual seria muito mais difícil.

Aos meus amigos, que sempre me incentivaram a continuar.

Deixo também a minha gratidão a todos os professores e funcionários do ISEP que ao fazerem parte da minha formação académica, contribuíram para melhorar o meu futuro.

Índice

Capítulo 1 - Introdução	1
1.1 Enquadramento	2
1.2 Apresentação da empresa	2
1.3 Problema	2
1.4 Motivação	4
1.5 Objetivos	5
1.6 Planeamento	5
1.7 Resultados Esperados	6
1.8 Estrutura do documento.....	6
Capítulo 2 - Contexto e Estado da Arte	9
2.1 Contexto e Estado da Arte	10
2.2 Trabalhos relacionados	11
MySenior	12
Ankira	12
AsisGes – Software de Gestão de serviços de Apoio Domiciliário	13
GSAD – Gestão de Serviços de Apoio Domiciliário	14
e-SAD – Serviço de Apoio Domiciliário	15
Help2care	16
Comparação de soluções	16
2.3 Tecnologias.....	19
2.3.1 Ecossistema de soluções tecnológicas	20
Aplicações <i>desktop</i>	20
Aplicações <i>web</i>	20
Aplicações móveis	21
Progressive Web Applications (PWA)	21
2.3.2 Custo de desenvolvimento.....	23
2.3.3 Acesso à internet.....	23
2.3.4 Desafios no desenvolvimento de <i>software</i> para diferentes plataformas.....	24
2.4 Progressive Web Applications.....	25
2.4.1 Evolução das PWA.....	25
2.4.2 Compatibilidade de Browsers versus Sistemas Operativos nas PWA.....	26

2.4.3 Fatores de sucesso em PWA	30
2.5 Sistematização de tecnologias	30
2.6 Questionários	31
2.6.1 Desenvolvimento de um gestor de questionários de raiz	31
2.6.2 Integrar uma solução já existente.....	31
2.7 Tecnologias de relevância no projeto	32
2.8 Sumário	33
Capítulo 3 - Análise de Valor do Negócio	35
3.1 Modelo de Desenvolvimento de Novos Conceitos	36
3.2 New Concept Development	36
3.2.1 Identificação de Oportunidades.....	37
3.2.2 Análise de oportunidades	39
3.3 Valor da solução.....	40
3.3.1 Proposta de valor	40
3.3.2 Modelo de Negócio de Canvas.....	41
3.4 Quality Function Deployment	43
3.5 Sumário	46
Capítulo 4 - Análise e Design da solução	47
4.1 Engenharia de requisitos.....	48
4.1.1 Stakeholders.....	48
4.1.2 Atores do sistema	48
4.1.3 Requisitos não funcionais	49
4.1.4 Requisitos funcionais	50
4.2 Casos de uso.....	51
4.3 Modelo de domínio.....	54
4.4 Design da solução.....	56
4.4.1 Arquitetura de uma PWA	56
4.4.2 Escolha de Arquitetura da solução	58
4.4.2.1 Arquitetura Monolítica.....	58
4.4.2.2 Arquitetura de Microsserviços.....	59
4.4.2.3 Sistematização das Arquiteturas Alternativas	60
4.4.3 Diagrama de Implantação da solução	61
4.5 Modelo de Dados	62
4.6 Sumário	63

Capítulo 5 – Desenvolvimento da Solução	65
5.1 Metodologia de Trabalho.....	66
5.2 GesSAD Web Api	66
5.2.1 Considerações iniciais	66
5.2.2 Diagrama de classes	68
5.2.3 Autenticação e Autorização	70
5.2.4 Funcionalidades	73
5.2.4.1 Criar plano de cuidados.....	73
5.2.4.2 Listar planos de cuidados	75
5.3 GesSAD PWA	76
5.3.1 Considerações iniciais	77
5.3.2 Diagrama de classes	78
5.3.3 Instalação	79
5.3.4 Autenticação e Autorização	80
5.3.5 Modo offline.....	82
5.3.6 Funcionalidades	83
5.3.6.1 Gestão de Utilizadores e Colaboradores.....	84
5.3.6.2 Gestão de Planos de Cuidados	85
5.4 Horário	87
5.5 Sumário	90
Capítulo 6 – Teste e Validação da Solução	91
6.1 Indicadores de avaliação	92
6.2 Metodologia de Avaliação.....	92
6.2.1 Testes de software	92
6.2.2 Inquéritos de satisfação	93
6.3 Quantitative Evaluation Framework	93
6.4 Testes Funcionais	94
6.5 Análise da Usabilidade	96
6.6 Sumário	97
Capítulo 7 - Conclusão e Trabalho Futuro	99
7.1 Conclusão e Perspetivas de Trabalho Futuro.....	100
Referências.....	103
Anexo A - Ficha de Contactos.....	111
Anexo B - Ficha de Controlo da Medicação	112

Anexo C - Ficha de Registo de Horas dos colaboradores	113
Anexo D - Ficha de Registo de Ocorrências.....	114
Anexo E - Ficha de Registo de Serviços (A).....	115
Anexo F - Ficha de Registo de Serviços (B).....	116
Anexo G - Ficha de Registo de Serviços (C)	117
Anexo H - Ficha de Inscrição do Utente	118
Anexo I - Questionário de qualidade.....	120
Anexo J - Casos de Uso	121
Anexo K - Quantitative Evaluation Framework (QEF)	127
Anexo L – Inquérito de Satisfação do Utilizador	128

Lista de Figuras

Figura 1 - Diagrama de Gantt	6
Figura 2 – MySenior (MaisEficaz, 2018)	12
Figura 3 – Asisges (Asisges, 2022)	14
Figura 4 - Help2Care (AppDevice, 2019)	16
Figura 5 - Evolução das PWA ((Divante, 2019)).....	26
Figura 6 - Compatibilidade para Geolocalização (Santoni, 2021)	28
Figura 7 - Compatibilidade para Home Screen (Santoni, 2021).....	28
Figura 8 - Compatibilidade para Push Notifications (Santoni, 2021)	29
Figura 9 - Compatibilidade para Experiência de utilização (Santoni, 2021)	29
Figura 10 - Processo de Inovação (Koen, et al., 2002)	36
Figura 11 - New Concept Development (Koen et al, 2002).....	37
Figura 12 - Acesso a páginas PWA (Rykov, 2021).....	39
Figura 13 - Proposta de Valor.....	40
Figura 14 - Proposta de valor	42
Figura 15 - Modelo QFD (Silva, 2011)	44
Figura 16 - Matriz Correlação dos requisitos	45
Figura 17 - Requisitos técnicos vs cliente.....	46
Figura 18 - Gestão de Planos de Cuidados.....	51
Figura 19 - Gestão horários.....	52
Figura 20 - Gestão de Atividade Diária.....	53
Figura 21 - Gestão de Utilizadores	54
Figura 22 - Modelo de domínio.....	55
Figura 23 - Arquitetura de uma PWA.....	57
Figura 24 - Arquitetura Client-Side PWA.....	58
Figura 25 - Diagrama componentes arquitetura monolítica.....	59
Figura 26 - Diagrama componentes arquitetura micro serviços.....	60
Figura 27 - Diagrama de implantação	61
Figura 28 – Modelo de Dados	62
Figura 29 - Api Rest	67
Figura 30 – Exemplo do Swagger da GesSAD Web API.....	68
Figura 31 – Diagrama de Classes do Plano de Cuidados	69
Figura 32 - Exemplo de JWT gerado.....	70
Figura 33 - Mecanismo de autenticação/autorização	71
Figura 34 - Excerto código autenticação	72
Figura 35 – Diagrama de Sequência para criar Plano de Cuidados.....	73
Figura 36 – Excerto da classe CareplanAddDto.....	74
Figura 37 - Exemplo método Add (POST) de um CarePlan	74
Figura 38 – Diagrama de sequência para listar Plano de Cuidados	75
Figura 39 – Classes para filtragem, ordenação e paginação do Plano de Cuidados	75
Figura 40 – Excerto de código de ordenação/filtragem de Planos de Cuidados	76
Figura 41 – Ngrok.....	77
Figura 42 - Diagrama Classes.....	78
Figura 43 - Instalar.....	79
Figura 44 - Aplicação instalada.....	79

Figura 45 – Ficheiro Manifest.json	80
Figura 46 - Página de Entrada	80
Figura 47 - Exemplo LocalStorage	81
Figura 48 - Exemplo configuração.....	82
Figura 49 – Página Principal	84
Figura 50 - Página de Utilizadores.....	84
Figura 51 - Página de Colaboradores	85
Figura 52 - Página de Planos de Cuidados.....	86
Figura 53 - Criar Plano de Cuidados	86
Figura 54 - Editar Plano Cuidados	87
Figura 55 - Detalhes Plano Cuidados.....	87
Figura 56 - Horário	88
Figura 57 - Exemplo Horário	89
Figura 58 – Usabilidade da Aplicação.....	96
Figura 59 - Funcionalidade da Aplicação.....	96
Figura 60 - Adaptabilidade da Aplicação.....	97
Figura 61 - Ficha de Contactos	111
Figura 62 - Ficha de Controlo da Medicação.....	112
Figura 63 - Ficha de registo de horas	113
Figura 64 - Ficha de registo de ocorrências	114
Figura 65 - Ficha de registo de serviços (A).....	115
Figura 66 - Ficha de registo de serviços (B).....	116
Figura 67 - Ficha de registo de serviços (C).....	117
Figura 68 - Ficha de inscrição do utente	118
Figura 69 - Ficha de inscrição do utente	119
Figura 70 - Questionário de Qualidade	120
Figura 71 - Gerir Utilizadores	121
Figura 72 - Gestão de plano de cuidados.....	121
Figura 73 - Gestão Atividade Diária.....	122
Figura 74 - Gestão de Serviços	122
Figura 75 - Gestão de Colaboradores.....	123
Figura 76 - Gestão de Controlo de Medicação.....	123
Figura 77 - Gestão de Medicamentos	124
Figura 78 - Gestão de Prescrição de Medicamentos	124
Figura 79 - Gestão de Paciente	125
Figura 80 - Gestão de Pessoas Relacionadas	125
Figura 81 - Gestão de Horário	126
Figura 82 - Gestão de Serviços	126
Figura 83 - Inquérito de Satisfação	128

Lista de Tabelas

Tabela 1 - Comparação por critério das soluções	18
Tabela 2 - Comparação de soluções (AltexSoft, 2018).....	30
Tabela 3 - Diferentes opções para os questionários.....	32
Tabela 4 - Matriz de Corelação	45
Tabela 5 - Matriz de Relação.....	45
Tabela 6 - Verificar se a autenticação funciona corretamente.....	94
Tabela 7 - Verificar se os registos são corretamente gravados na base de dados	94
Tabela 8 - Verificar se é possível editar um registo	95
Tabela 9 - Verificar a seleção de um elemento de uma combobox no Registo de Serviços.....	95
Tabela 101 - Verificar os menus de acesso	95
Tabela 112 - Quantitative Evaluation Framework	127

Acrónimos

AES	Advanced Encryption Standard
API	Application Programming Interfaces
BYOD	Bring Your Own Device
CRUD	Create, Read, Update and Delete
CSS	Cascading Style Sheets
EF	Entity Framework
ER	Engenharia de requisitos
FEE	Fuzzy Front End
GesSAD	Gestão de serviços de apoio domiciliário
GPS	Global Positioning System
GSAD	Gestão de Serviços de Apoio Domiciliário
HTTP	Hypertext Transfer Protocol
ID	Injeção de Dependência
IDE	Ambiente de desenvolvimento integrado
INE	Instituto Nacional Estatística
IPSS	Instituição Particular de Solidariedade Social
ISEP	Instituto Superior de Engenharia do Porto
JWT	JSON Web Token
NCD	New Concept Development
NPD	New Product Development
ORM	Object-Relational mapping
PDF	Portable Document Format
PWA	Aplicação web progressiva
QFD	Quality Function Deployment
REST	Representational State Transfer
RGPD	Regulamento Europeu para a Proteção de Dados
SAD	Serviço de apoio domiciliário
SQL	Structured Query Language
UI	User Interface
UML	Unified Modeling Language

WWW

World Wide Web

Glossário

Face ID	Possibilidade de autenticação no telemóvel através do reconhecimento facial
GPS	Sistema de navegação por satélite que indica no aparelho recetor móvel a posição atual em que se encontra
HTTP	Protocolo de comunicação utilizado para sistemas de informação de hipermédia, distribuídos e colaborativos, sendo a base para a comunicação de dados da World Wide Web (WWW)
IDE	Ambiente de desenvolvimento integrado para desenvolvimento de <i>software</i>
Push notifications	Notificação que o utilizador recebe em numa aplicação de telemóvel, <i>tablet</i> ou num <i>browser</i> sem requisitar a mesma
Software Open Source	Temo também conhecido como “ <i>Software livre</i> ” refere-se genericamente a <i>software</i> que respeita as quatro liberdades definidas pela <i>Free Software Foundation</i>
Siri	Assistente virtual que atualmente incorpora todo o ecossistema desenvolvido pela Apple
SQL	Linguagem de consulta estruturada, isto é, uma linguagem de pesquisa declarativa para a base de dados.
Touch ID	Possibilidade de autenticação no telemóvel através de uma impressão digital
Tweet	A ideia do <i>tweet</i> na rede social é uma comparação com a sequência de pequenas publicações que os usuários do Twitter fazem, já que só é possível apenas a postagem de textos com até 140 caracteres
UML	Linguagem de modelação que permite que os programadores de <i>software</i> visualizem os produtos dos seus projetos em diagramas padronizados

Capítulo 1 - Introdução

Neste capítulo é feito um breve enquadramento do projeto no âmbito em que esta dissertação se enquadra, sendo descrito o problema, a motivação para a sua realização, os objetivos identificados e os resultados esperados do desenvolvimento deste projeto.

1.1 Enquadramento

Este projeto foi desenvolvido no âmbito da Tese, do 2º ano do Mestrado em Engenharia de Software do Instituto Superior de Engenharia do Porto (ISEP) com a orientação da Professora Doutora Ana Madureira.

Este surge da necessidade de dar resposta a uma empresa externa ao Instituto Superior de Engenharia do Porto, a Vida e Conforto, situada no Porto. Tem como principal objetivo o desenvolvimento de uma aplicação *web* progressiva (PWA) que será a ferramenta de trabalho dos cuidadores e dos profissionais de saúde no terreno, terá também um papel de apoio à gestão e monitorização dos utentes e funcionários. Esta dissertação irá focar-se no desenvolvimento da aplicação *web* progressiva.

1.2 Apresentação da empresa

A **Vida e Conforto** (Conforto, 2009), é uma empresa que, desde janeiro de 2009, presta serviços de apoio domiciliário e serviços domésticos profissionais e pretende ser uma referência nessa área. A missão da empresa é fornecer cuidados a idosos, dependentes, doentes, serviços domésticos, teleassistência, *babysitting*, ajudas técnicas, etc. no âmbito da assistência domiciliária e proporcionar aos clientes um serviço de excelência.

Na base dos serviços fornecidos pela **Vida e Conforto** está uma equipa de profissionais especializados, selecionados em função das necessidades dos clientes.

1.3 Problema

Nos últimos anos tem havido um acréscimo do envelhecimento da população a um ritmo bastante acentuado (Público, 2018). Como resposta, surge um grupo profissional, os auxiliares de cuidados domiciliários ou cuidadores, responsáveis por prestar cuidados de uma forma digna e qualificada ao idoso que se encontra na sua casa, nomeados serviços de apoio domiciliário (Bonfim & Veiga, 1996).

A **Vida e Conforto** (Conforto, 2009), é uma empresa que, desde janeiro de 2009, presta serviços de apoio domiciliário e serviços domésticos profissionais. Para isso oferece uma opção

inovadora de assistência, disponível 24 horas por dia, 365 dias por ano, no ambiente que lhe é familiar, em sua casa.

Atualmente a empresa faz a gestão a nível de utentes (pessoas doentes ou dependentes), dos serviços prestados e do horário dos colaboradores em ficheiros Excel. Para cada utente é elaborado um plano de cuidados, baseado nas necessidades identificadas pela empresa e a família. Este plano é composto pelas seguintes informações: dados pessoais do utente, contacto das pessoas responsáveis, problemas, cuidados a prestar e a sua periodicidade e data de início. Para cada plano é atribuído um ou mais colaboradores.

Os registos dos serviços prestados aos utentes são efetuados pelos colaboradores e são relevantes, no sentido em que se registam as mudanças e intervenções realizadas ao utente ao longo do tempo pelo qual foram contratados. A análise destes registos permite fazer uma avaliação e reformulação do plano de cuidados elaborado anteriormente, caso não se esteja a obter os resultados esperados. Para este registo, são utilizadas grelhas e folhas de observações, onde o colaborador faz uma síntese do serviço realizado ao utente.

A empresa é constituída por uma assistente social, um diretor técnico e os cuidadores (funcionários) que fazem os serviços de apoio domiciliário. Todos estes intervenientes têm de preencher os documentos visíveis na Figura 61 (Anexo A) até a Figura 70 (Anexo I). O registo destes processos e o planeamento do horário é todo realizado sem o apoio de ferramentas específicas e enquadradas com o setor.

Esta gestão, sem o apoio de ferramentas específicas, consome um grande número de horas de trabalho diário. Os colaboradores preenchem toda a documentação com registos escritos em papel (registo dos serviços prestados, registo de horas gastas, observações, etc), que mais tarde são passados para uma página em Excel, de modo a serem processadas as horas. É um processo moroso também para a consulta de dados, o que o torna sujeito a falhas e dificulta a análise dos dados.

A empresa disponibiliza aos utentes/pessoas responsáveis questionários em papel no final dos serviços, de modo a ter uma melhor perceção do desempenho dos colaboradores, e da satisfação do cliente.

1.4 Motivação

O progresso e a competitividade levam a alterações constantes nas organizações, exigem maior produtividade, menores custos, melhor atendimento aos clientes e criam a necessidade de novas formas de gestão (INE, 2017).

As tecnologias da informação têm vindo a desempenhar um papel cada vez mais relevante em toda a sociedade ao longo dos últimos anos. O *software*, como meio de dispersão da informação, deixou de ser um diferenciador de mercado para se tornar a base de sustentação de diversas organizações. No entanto, à medida que ocorre este crescimento na importância da utilização do *software*, também aumenta a necessidade e a dependência pelo desenvolvimento de ferramentas com maior capacidade de resposta.

Atualmente e, de acordo com (Totvs, 2020), uma boa gestão de documentos e organização dos ficheiros de uma empresa, sejam eles físicos ou virtuais, pode contribuir para o seu sucesso, especialmente num mundo, onde cada vez mais, as organizações atuam em ambientes sujeitos a constantes mudanças, causadas essencialmente pelo forte avanço das Tecnologias e Sistemas de Informação. Como consequência, o fluxo de informação e documentação nas empresas torna-se mais complexo, o que vai dificultar a sua gestão.

A gestão de documentos e processos é fundamental para o bom desempenho de uma empresa, sendo um reflexo da sua organização interna. A existência de grandes volumes de informação pode levar à perda de documentos, dificultar o acesso e perturbar o controlo dos fluxos de informação.

A implementação de um sistema de gestão documental e processos é um fator decisivo para a empresa, na medida em que permite gerir a informação não estruturada, tendo por base conceitos como: digitalização dos documentos em formato papel, originando documentos digitais que são classificados e disponibilizados segundo um determinado critério; classificação dos documentos de forma desmaterializada (podendo garantir a gestão integrada do ficheiro físico e digital); definição das várias ações e estados pelas quais os documentos passam; possibilidade de pesquisar documentos pelo seu conteúdo, permitindo o acesso imediato em qualquer parte; melhoria da eficiência na gestão de processos, e redução dos custos com cópias e com o espaço de ficheiro (FileDoc, 2019).

1.5 Objetivos

O principal objetivo deste trabalho é o desenvolvimento de um sistema de apoio à prestação de cuidados no domicílio. Este sistema será uma ferramenta de registo para os cuidadores e profissionais de saúde, permitindo apoiar e gerir os cuidados que são realizados à pessoa no domicílio. Pretende-se ainda que este sistema substitua os registos em papel (visíveis da Figura 61 a Figura 70) efetuados por estes profissionais, podendo desta forma reduzir o tempo e o custo gasto nesta atividade, que, de acordo com a empresa PowerDMS a gestão de processos sem papel beneficia tanto o meio ambiente como a organização. A empresa indica ainda que, segundo estatísticas, cerca de 3% dos documentos da empresa são arquivados incorretamente e 7% desaparecem (PowerDMS, 2020). Pretende-se ainda informatizar a gestão dos horários dos colaboradores, o que resultará numa otimização e agilização dos processos existentes na empresa.

1.6 Planeamento

No planeamento deste projeto foram estipuladas seis etapas: O levantamento do estado da arte, a análise de valor, a experimentação e avaliação, o desenho da solução e a implementação e testes.

No levantamento do estado da arte, são identificadas possíveis soluções, sendo também identificadas as técnicas e tecnologias a utilizar. Na análise de valor, são identificadas e analisadas as oportunidades e, apresentada a técnica *Quality Function Deployment* (QFD). No capítulo de Testes e validação da solução são descritos os testes e validações realizados à solução desenvolvida. No desenho da solução, são definidos os requisitos necessários e avaliadas as diferentes soluções para aplicar no projeto. Na implementação, são implementadas e descritas as decisões tomadas na fase anterior. Nos testes, são realizados testes de usabilidade, de forma a compreender se este projeto dá resposta ao problema levantado. É nesta fase que se pretende perceber se foram ou não cumpridos todos os objetivos propostos inicialmente. Na Figura 1, podemos analisar as diferentes etapas da dissertação e o cronograma com a previsão da sua concretização com início em setembro de 2021, até setembro de 2022.

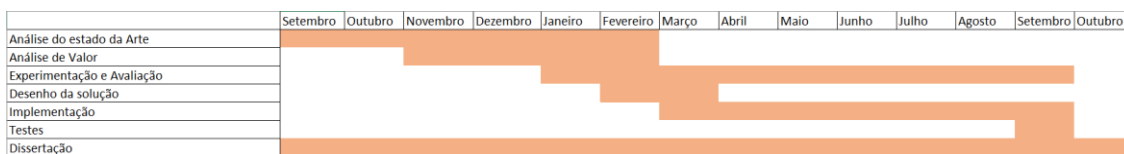


Figura 1 - Diagrama de *Gantt*

1.7 Resultados Esperados

Com o desenvolvimento da aplicação *web* progressiva pretende-se que esta se torne numa ferramenta de trabalho que vai permitir melhorar os registos e monitorizar a informação dos serviços prestados aos clientes, que facilite a comunicação e a transmissão de informação entre os diversos intervenientes na **Vida e Conforto**, ao centralizar a informação vai melhorar o fluxo destes processos.

A utilização da aplicação *web* progressiva, acessível por qualquer *smartphone*, em qualquer altura do dia, desde que tenha acesso à *internet*, irá permitir à empresa um melhor controlo e monitorização do trabalho desenvolvido, aumentando a eficiência da gestão e reduzindo os custos para a empresa e para os clientes, pretende-se ainda que contribua para a geração de indicadores de qualidade dos cuidados prestados, e esta aplicação permitirá abandonar a documentação que é preenchida à mão pelos colaboradores da **Vida e Conforto**.

1.8 Estrutura do documento

Esta dissertação está organizada em sete capítulos, cada um com um objetivo específico por forma a descrever as várias etapas do desenvolvimento da aplicação *web* progressiva.

O primeiro capítulo, Introdução, apresenta um enquadramento do projeto, a definição do problema, a motivação para o desenvolvimento da solução e os objetivos, o planeamento do trabalho, e os resultados esperados para este projeto.

No segundo capítulo, Contexto e Estado de Arte, é realizada a pesquisa bibliográfica sobre o tema com investigação e estudo das soluções existentes no mercado, identificando as tecnologias mais relevantes para o desenvolvimento do projeto.

O terceiro capítulo, Análise de Valor do Negócio, pretende fazer a análise e estruturação do valor da solução, com a elaboração da proposta de valor e modelo de negócio do projeto.

No quarto capítulo, Análise da Solução, é detalhado o desenho da solução, desde o processo de engenharia de requisitos com a identificação dos requisitos do projeto, modelo de domínio e vistas relevantes que permitam a obtenção de uma visão e compreensão da solução. Para finalizar é realizada a análise de alternativas à solução.

No quinto capítulo, Implementação, é documentado o trabalho elaborado para o desenvolvimento e implementação do protótipo, apresentadas as ferramentas utilizadas, detalhes de código relevantes e opções tomadas.

O sexto capítulo, Testes e Validação da Solução, descreve os procedimentos de testes a serem aplicados ao protótipo e é descrita a validação e avaliação realizada à solução desenvolvida.

No sétimo capítulo, Conclusão, é apresentada uma síntese do projeto, com análise e síntese do trabalho desenvolvido.

Capítulo 2 - Contexto e Estado da Arte

Este capítulo faz uma revisão da literatura, onde se foca o contexto, apoiado com artigos e estudos da área de cuidados domiciliares. É apresentada a organização e o estado da arte, com base no estudo do mercado relativamente às soluções existentes e uma análise comparativa das soluções identificadas. Por fim, é efetuado um estudo sobre as tecnologias e abordagens existentes neste âmbito.

2.1 Contexto e Estado da Arte

Atualmente temos vivenciado uma rápida evolução a nível económico, social e tecnológico que levou a uma melhoria da qualidade de vida da população. De acordo com dados do (INE, 2015), observa-se um aumento da população idosa em Portugal e conseqüentemente, um aumento do número de pessoas dependentes.

O envelhecimento populacional advém do aumento da esperança média de vida, da baixa taxa de natalidade e da diminuição da taxa de mortalidade. Neste processo, a melhoria das condições de vida, o desenvolvimento das políticas sociais e o maior acesso aos cuidados de saúde constituem fatores importantes à compreensão do fenómeno. (Lage, 2005) defende que as políticas sociais e de saúde potenciam cada vez mais a viabilidade da manutenção dos idosos no domicílio, tendo em conta a importância do apoio prestado pelos cuidadores informais.

Segundo (Oliveira, 2010), é necessário adotar medidas que favoreçam o envelhecimento com mais qualidade.

O envelhecimento deverá ser encarado como uma problemática mundial, pois é e será uma das principais características demográficas, quer do mundo desenvolvido, quer das regiões em desenvolvimento. Estas tendências terão importantes conseqüências na composição da população ativa e nas relações de dependência económica das pessoas (Grevi & Gnesotto, 2008).

A sociedade tem que se organizar de forma a permitir dar respostas adequadas às necessidades dos vários grupos etários da população, como nos chama à atenção Maria Rosa (Rosa, 2000) ao referir que *“em lugar de se pensarem soluções orientadas para manipular a evolução da população ou para se travar o envelhecimento demográfico, é tempo de se aceitar o curso dos factos”*. Todos estes factos levam à necessidade crescente de uma resposta social.

Uma das possíveis respostas sociais encontra-se nos serviços de proximidade. Estes serviços têm tido um grande crescimento, fruto da dificuldade das famílias em cuidarem dos seus e também dos que não têm família. Em Portugal, estes serviços de proximidade, designadamente os serviços de apoio social, são prestados por instituições públicas ou por instituições de solidariedade (Centros Regionais de Segurança Social, Instituições Particulares de Solidariedade

Social, Misericórdias, Cooperativas, Centros Sociais e Paroquiais, etc.), bem como por entidades particulares, com ou sem fins lucrativos.

O Serviço de Apoio Domiciliário segundo a Portaria nº 38/2013 de 30 de janeiro (Republica, 2013) é uma resposta social que visa prestar cuidados e serviços a famílias e pessoas que se encontrem no seu domicílio, em situação de dependência física e ou psíquica e que não possam assegurar a satisfação das suas necessidades básicas e ou a realização das atividades da vida diária, nem disponham de apoio familiar para o efeito.

São objetivos do Apoio Domiciliário: contribuir para a melhoria da qualidade de vida dos utentes e famílias; facilitar a conciliação da vida familiar e profissional do agregado familiar e ajudar as pessoas a permanecer nas suas casas, adiando ou evitando o recurso a estruturas residenciais.

A Portaria n.º 38/2013, de 30 de janeiro (Republica, 2013) prevê que os cuidados prestados pelo serviço de Apoio Domiciliário devem, preferencialmente, ser disponibilizados todos os dias da semana e sempre que possível, garantir-se o serviço aos fins de semana (sábados e domingos) e feriados. A empresa deve estar licenciada pela Segurança Social para poder prestar os serviços. Esta resposta deve ser baseada numa avaliação das necessidades dos utentes e pode prestar cuidados e serviços de: higiene e conforto pessoal; higiene habitacional; apoio nas refeições; tratamento de roupas; aquisição de bens e géneros alimentícios; serviço de teleassistência; confeção de alimentos no domicílio, serviços de enfermagem, etc.

Segundo o (Ine, 2020), o envelhecimento demográfico continuou a acentuar-se. O índice de envelhecimento atingiu o valor de 167,0 idosos por cada 100 jovens (163,2 em 2019).

Num serviço de apoio domiciliário, um funcionário de uma empresa responsável e com qualificações visita a casa de um cliente e averigua os variados serviços que o potencial cliente precisará, tais como cuidados de enfermagem, apoio, etc, é avaliada também as condições do cliente, mentais e físicas e faz-se um plano de cuidados (Itabashi et al, 2005).

2.2 Trabalhos relacionados

Nesta secção, são abordadas algumas soluções disponibilizadas no mercado na área do apoio domiciliário. No que diz respeito ao levantamento de soluções já existentes no mercado, algumas foram mencionadas pela empresa e outras foram obtidas através de uma pesquisa

realizada aos repositórios de aplicações Google Play (Google, 2022) e App Store (Apple, 2022). É feita também uma comparação entre as diversas soluções encontradas.

MySenior

A aplicação *web* MySenior foi desenvolvida pela empresa MaisEficaz. Esta é constituída por uma aplicação *web* mais direcionada à gestão e por uma aplicação móvel mais focada para efetuar os registos diários (MaisEficaz, 2018).

Esta aplicação está direcionada para três diferentes situações: centros de dia, apoio domiciliário e estrutura residencial. A aplicação *web* inclui uma área de *backend* onde a direção das empresas podem fazer a gestão de utentes, funcionários, gestão de planos individuais por utentes, consultar as mensalidades de cada utente, inquéritos. A aplicação móvel é disponibilizada para os funcionários fazerem os registos de forma mais rápida. Na Figura 2 podemos observar esta aplicação.

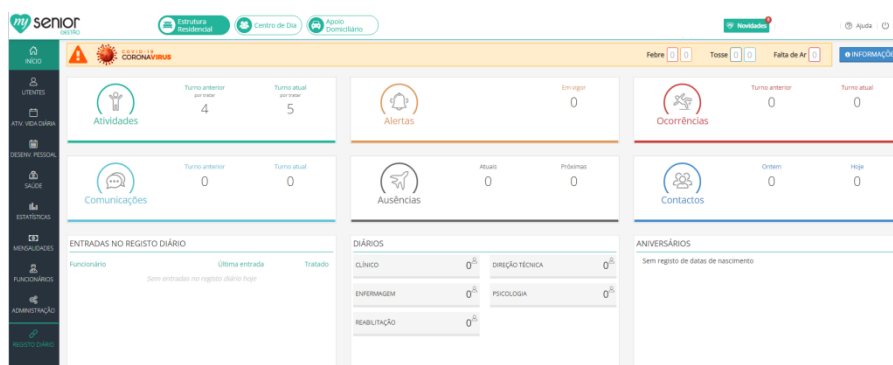


Figura 2 – MySenior (MaisEficaz, 2018)

O preço para a utilização desta plataforma varia de acordo com o número de utentes registados na organização, tendo por base um custo fixo de 12€/mês por cada utente.

Ankira

A Ankira é uma plataforma *web* que tem como principal objetivo “ajudar na gestão de equipas em lares de idosos, casas de repouso, centros de dia e serviços de apoio domiciliário a gerir e prestar cuidados aos utentes.” (Metatheke, 2022).

Esta plataforma tem como base um conjunto de módulos que estão integrados entre si, para facilitar a transmissão de informação, melhorar o processo de tomada de decisão e a monitorização dos cuidados prestados. O módulo base corresponde às funcionalidades mínimas que podemos adquirir neste produto:

- *Dashboard* – Para aceder a informação em tempo real sobre a organização (Ocorrências, Eventos, Inscrições, Indicadores, Consultas e exames médicos);
- Calendário – Para consultar a agenda dos utentes e saber quais os eventos planeados (Aniversários, Tratamentos, Consultas);
- Candidatos – Registrar os pedidos de inscrição, e inscrever de acordo com os critérios de admissão da organização;
- Uteses – Acesso aos processos individuais dos utentes;
- Plano Individual – Estabelecer um plano individual para cada utente, com base na avaliação efetuada pela equipa;
- Apoio nas atividades de vida diária – Cuidados pessoais (Alimentação, Banho, Higiene);
- Apoio Clínico – Cuidados de saúde (Registos médicos, Consultas e exames);
- Apoio Psicossocial - Cuidados psicossociais (Registo de diligências);
- Atividades – Planear as atividades, e registar as presenças;
- Controlo – Monitorizar a despesa e os cuidados aos idosos;
- Planeamento – Consultar e exportar para PDF estatísticas;
- Estatísticas e indicadores – Visualizar estatísticas e indicadores por área.

O módulo de Registo não vem com a versão base, no entanto este permite o registo rápido e dos cuidados prestados:

- Atividades de vida diária;
- Atividades instrumentais de vida diária;
- Tratamentos de enfermagem e fisioterapia;
- Medicação administrada;
- Sinais vitais;
- Ocorrências.

O preço para a utilização desta plataforma varia de acordo com o número de utentes registados na organização e os tipos de serviços pretendidos, podendo ir dos 26€/mês aos 40€/mês (Metatheke, 2022).

AsisGes – Software de Gestão de serviços de Apoio Domiciliário

O Asisges é uma aplicação que foi desenvolvida para permitir a gestão completa de um Serviço de Apoio Domiciliário, podendo integrar os dados dos clientes, dos serviços e faturação, dos

colaboradores internos e externos, assim como o registo de todas as atividades, procedimentos e incidências de cada serviço. (Asisges, 2022)

É uma solução integrada, constituída por um *software online* com armazenamento na *cloud*, que permite o acesso direto e imediato a todos os dados e informações da atividade, registo e planeamento de tarefas, em qualquer momento e em qualquer lugar. Esta solução tem como principais funcionalidades:

- Registo diário e integral de todas as atividades, ocorrências, serviços prestados;
- Gestão dos serviços prestados;
- Faturação mensal simples (Controlo de pagamentos);
- Ficha de utente;
- Gestão de utilizadores.

Na Figura 3 é possível observar uma página do ASISGES.

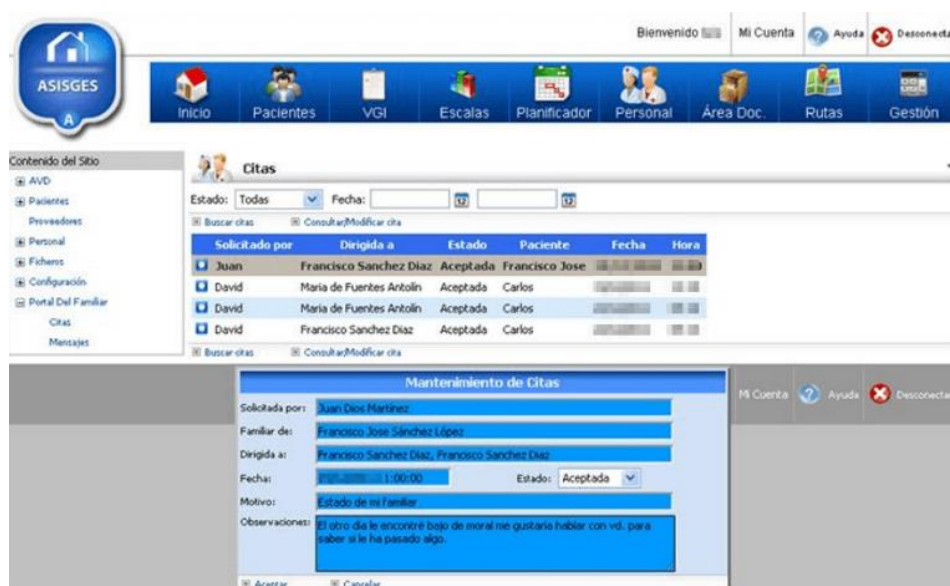


Figura 3 – Asisges (Asisges, 2022)

A aquisição desta solução requer o pagamento de uma licença, estando o seu valor dependente do número de clientes da empresa. Até 10 clientes o plano é gratuito (Asisges, 2022).

GSAD – Gestão de Serviços de Apoio Domiciliário

O GSAD “foi desenhado com o intuito de otimizar toda a gestão da equipa de apoio domiciliário e garantir maior comodidade no dia a dia na prestação de cuidados” (InoveSaude, 2022).

O GSAD é uma aplicação que está disponível para Android (Android, 2022) apenas, no entanto não está disponível para *download* ao público. Esta aplicação tem como principais funcionalidades:

- Plano de domicílios por turno;
- Serviços a efetuar em cada cliente;
- Possibilidade de efetuar serviços extra;
- Agendamentos (Consultas, reuniões, etc);
- Tratamento de roupas dos clientes;
- Geolocalização dos clientes;
- Registo de ocorrências e comunicações.

O preço para a utilização desta aplicação depende do número de utentes da empresa, tendo um custo inicial entre os 15€ a 20€/mês.

e-SAD – Serviço de Apoio Domiciliário

O e-SAD (Serviço de Apoio Domiciliário) (F3M, 2020) assegura a gestão de serviços de apoio domiciliário, gestão de equipas, registo de serviços de apoio domiciliário e análises dos serviços realizados. Destina-se ao responsável pelo planeamento dos serviços de Apoio Domiciliário e às equipas, constituindo não só uma verdadeira ferramenta de gestão do *backoffice*, bem como das equipas no terreno. O e-SAD permite a manutenção de um vasto conjunto de informações relativas ao processo de Apoio Domiciliário, nomeadamente:

- Planeamento de serviços de Apoio Domiciliário por equipa ou com base na análise automática entre as necessidades dos utentes e as disponibilidades das equipas;
- Gestão de Serviços realizados pelas equipas de Apoio Domiciliário;
- Controlo da Lavandaria;
- Integração com a solução Utenes: possibilidade de faturação de consumos adicionais, não abrangidos pelo contrato de serviços;
- Análises Estatísticas de serviços.

Embora o e-SAD seja um sistema bastante completo e bem estruturado, é um sistema com custos, o seu preço varia de acordo com o número de utentes e a empresa não está disponível para fazer o investimento neste *software*.

Help2care

A plataforma Help2care foi desenvolvida pelo Centro de Inovação em Tecnologias e Cuidados de Saúde do Instituto Politécnico de Leiria. Esta plataforma é constituída por uma plataforma *web* e uma aplicação móvel (Help2care, 2019).

A plataforma *web* inclui uma área de *backoffice* onde os profissionais de saúde fazem a gestão das necessidades e da formação dos cuidadores. Na aplicação móvel serão disponibilizados os conteúdos selecionados pelo profissional de saúde que estarão ajustados às necessidades do cuidador.

Na Figura 4 podemos observar esta aplicação.

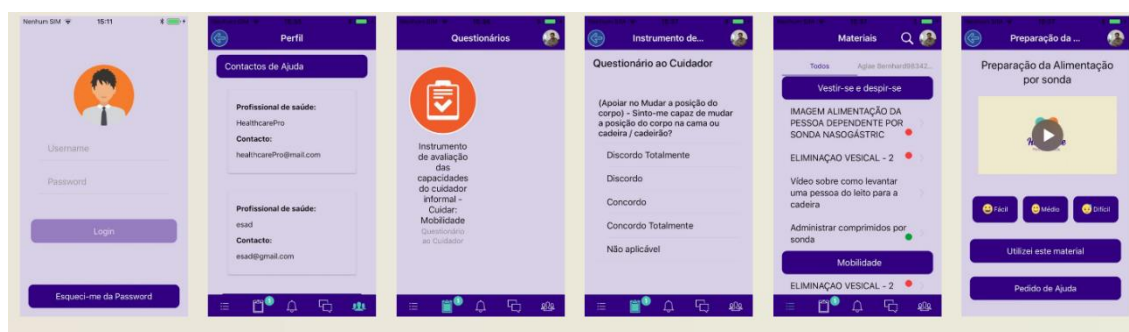


Figura 4 - Help2Care (AppDevice, 2019)

Comparação de soluções

Nesta secção pretende-se fazer uma comparação e uma análise crítica das soluções descritas na secção 2.3, tendo por base os requisitos identificados para esta dissertação. Os critérios identificados para a solução a desenvolver resultaram de reuniões realizadas com o diretor técnico da empresa **Vida e Conforto**, em conjunto com os seus colaboradores. Esta análise irá auxiliar no planeamento, e na definição das funcionalidades a serem implementadas no desenvolvimento deste projeto. Do Anexo A ao Anexo I encontram-se os documentos atuais que a empresa utiliza de modo a garantir uma boa coordenação de trabalho e que são a base para os requisitos requeridos pelo cliente.

Nesse sentido, os critérios identificados foram:

1. **Proteção e segurança de dados:** Preocupação da empresa relativa à proteção e à segurança dos dados. São dados pessoais e sensíveis dos utentes. Preocupação também em fazer cumprir o Regulamento Europeu para a Proteção de Dados;

2. **Autenticação:** Sendo uma aplicação *web* progressiva para registo dos serviços por parte dos cuidadores, e também para consulta no âmbito da gestão interna da empresa é necessário haver uma autenticação para diferentes níveis de utilizadores;
3. **Gestão de utilizadores e permissões:** A permissão de acesso deve ser dada de acordo com o perfil de utilizador para as funções que desempenha;
4. **Gestão de utentes:** Segundo o Anexo H - Ficha de Inscrição do Utente (Figura 68 e Figura 69), permite fazer a colheita de dados para elaborar o plano de cuidados e planear a gestão do serviço;
5. **Acesso a informação de utentes:** A empresa deve definir o perfil de utilizador para permitir apenas o acesso. no âmbito das funções que desempenhe;
6. **Gestão e controlo de horários:** De acordo com o Anexo C - Ficha de Registo de Horas dos colaboradores (Figura 63), o sistema deverá contabilizar as horas de serviço de cada colaborador por utente;
7. **Gestão de plano de cuidados:** Para proceder a registos e alteração aos planos de cuidados sempre que se justifique;
8. **Acesso à informação dos planos de cuidados:** Permitir o acesso dos cuidadores à informação para verificar os registos e as alterações ao plano de cuidados;
9. **Sistema em Português:** Por ser uma empresa portuguesa, deverá a linguagem utilizada nas definições ser o português;
10. **Registo de ocorrências:** Segundo o Anexo D - Ficha de Registo de Ocorrências (Figura 64), o sistema deverá incluir um espaço para registo de ocorrências/notas dos serviços prestados aos utentes;
11. **Questionários:** Segundo o Anexo I - Questionário de qualidade (Figura 70), no final da prestação do serviço a empresa envia um questionário ao utente ou ao seu responsável para avaliação do serviço prestado. Este questionário serve para avaliar a satisfação do utente e melhorar a qualidade;
12. **Gratuito:** O sistema deverá estar isento de custos, pois a empresa não tem disponibilidade económica para fazer investimento neste *software*.
13. **Registo de serviços efetuados:** Segundos os Anexo E - Ficha de Registo de Serviços (A), Anexo F - Ficha de Registo de Serviços (B) e Anexo G - Ficha de Registo de Serviços (C) (Figura 65, Figura 66 e Figura 67), verificamos que o sistema deverá permitir aos cuidadores registar, em cada utente, a informação relativa ao serviço prestado;

14. **Controlo de medicação do utente:** Segundo o Anexo B - Ficha de Controlo da Medicação (Figura 62), o sistema deverá permitir registar este controlo sobre a toma da medicação pelo utente;
15. **Pessoas responsáveis:** Segundo o Anexo A - Ficha de Contactos (Figura 61), o sistema deverá possibilitar o registo dos contactos dos responsáveis pelo utente.

Para melhor perceção, a Tabela 1 apresenta a comparação das soluções analisadas tendo como critérios de avaliação os requisitos identificados no projeto GesSAD. Caso o critério esteja na solução foi catalogado com “S” e caso não esteja com “N”. Os critérios encontram-se numerados de 1 a 15 como indicado acima. Os casos em que não foi possível avaliar, o critério está apresentado como “-”.

Tabela 1 - Comparação por critério das soluções

Solução/Critério	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
MySenior	S	S	S	S	S	-	S	S	S	S	N	N	S	S	-
Ankira	S	S	-	-	S	S	S	S	S	-	N	N	-	-	-
Asisges	-	S	S	S	S	-	S	S	S	S	-	N	S	-	-
GSAD	-	S	-	-	S	S	S	S	S	-	N	-	S	-	-
e-SAD	-	S	-	S	S	S	S	S	S	S	N	N	S	-	-
Help2care	-	S	-	N	N	N	N	N	S	N	N	S	N	N	N

A aplicação MySenior (MaisEficaz, 2018), é uma solução bastante completa e com várias vertentes como analisado, que satisfazem a maioria dos critérios avaliados. Tem funcionalidades além do esperado e solicitado, no entanto requer a aquisição de licença mediante pagamento, algo que neste momento a empresa não pode suportar.

A aplicação Ankira (Metatheke, 2022), vem com um módulo base composto por diversas funcionalidades que poderiam ajudar na resolução do problema. Não preenche os requisitos porque é necessário pagamento de licença de acordo com o número de utentes que a empresa tenha, o que neste caso trazia custos.

O Asisges (Asisges, 2022) é uma aplicação com armazenamento na *cloud*, desenvolvida de raiz para permitir a gestão de um serviço de apoio domiciliário, além de poder integrar os dados existentes na empresa, faturação, etc. Esta é uma solução gratuita para os primeiros dez

clientes, é necessário adquirir licença acima desse número, situação que a empresa não pode suportar.

O G-SAD e o e-SAD (InoveSaude, 2022), (F3M, 2020) , são aplicações direcionadas ao serviço de apoio domiciliário, sendo as duas muito idênticas a nível de funcionalidades. Não apresentam evidência de funcionalidades importantes como controlo de medicação, gestão de utilizadores, segurança dos dados pessoais. Além de não ter sido possível verificar algumas funcionalidades que são requisitos, os mesmos carecem de licença para a sua utilização.

A aplicação Help2Care (Help2care, 2019) apesar de não se enquadrar numa parte dos critérios avaliados, é referida pelo facto de ser uma aplicação portuguesa desenvolvida por uma equipa de estudantes portugueses, que estabelece uma melhoria dos canais de comunicação entre os profissionais de saúde e os cuidadores.

Após análise e discussão dos resultados da pesquisa concluiu-se que não havia um sistema que reunisse os requisitos estipulados. Encontramos sistemas com funcionalidades que poderiam satisfazer as necessidades, outros sem funcionalidades requeridas, alguns até que iriam exigir uma curva de aprendizagem bastante superior ao que se pretende. Ao mesmo tempo alguns apresentam um conjunto de funcionalidades que não se justifica, porque a maioria não vai ser utilizada. Relativamente à aquisição do licenciamento mediante os utentes registados, no caso da empresa tem cerca de 60 utentes, custo que a empresa não pode suportar dado o nº de registos necessários.

Em suma, como não encontramos no mercado uma ferramenta com os requisitos necessários, optou-se por desenvolver uma solução de raiz.

2.3 Tecnologias

Atualmente, a empresa tem um processo moroso no registo e no acesso à informação relativa a utentes, a cuidadores e a serviços prestados. O registo em papel e, posteriormente, em Excel torna-se insustentável com o crescimento da empresa, havendo necessidade de mais recursos para a supervisão, o que implica custos acrescidos e mais sujeito a falhas. Nesse sentido, torna-se importante o desenvolvimento de um sistema que permita informatizar, uniformizar e centralizar toda a informação, que agora é tratada de forma manual, implicando muitas horas de trabalho e aumento de custos.

2.3.1 Ecosistema de soluções tecnológicas

Os diferentes públicos-alvo deste sistema cuidadores, utentes, assistente social e gestores da empresa têm necessidades de soluções diferentes. A administração/gestão da empresa faz uma parte do processo de gestão de utentes/cuidadores em escritório. Neste caso, será mais prático aceder ao sistema através de um computador portátil ou um *desktop*. Os cuidadores, como prestam cuidados ao utente no domicílio, o uso de um *smartphone* ou *tablet* seria mais vantajoso. Em relação ao utente/pessoa responsável dependerá especificamente, de caso a caso, algo que não foi analisado no âmbito desta dissertação. Em qualquer um dos casos, o acesso à *internet* é um fator importante, de modo a permitir que todos os utilizadores do *software* tenham acesso a dados o mais atualizados possível. Outro ponto a ter em atenção na escolha da solução é o custo de desenvolvimento, algo que também é tido em consideração na decisão final da solução a adotar.

Nesse sentido, podem ser avaliadas algumas possíveis soluções, sendo elas: uma aplicação *desktop*, uma aplicação *web*, uma aplicação *web* progressiva ou ainda uma aplicação móvel. Apresenta-se de seguida uma breve referência a cada uma das possíveis soluções.

Aplicações *desktop*

Uma aplicação *desktop* é um *software* desenvolvido especificamente para ser instalada em computadores com sistema operativo Windows (Windows, s.d.), MAC (Apple, s.d.) ou Linux (Linux, s.d.). Os cuidadores têm de preencher a informação relativa aos cuidados prestados aos utentes em cada serviço no domicílio. O desenvolvimento de uma aplicação *desktop* para ser instalada individualmente num computador não é uma solução viável, visto que só serviria as necessidades da administração/gestão. Poderia servir as necessidades dos cuidadores também, mas obrigá-los-ia a ter acesso a um computador portátil e acesso à *internet*. Seria necessário também que cada novo cuidador que colaborasse com a empresa, tivesse o *software* no computador portátil. Dada a elevada rotatividade de pessoas neste tipo de trabalhos, não seria algo viável. Nesse sentido, esta solução é, desde logo, descartada.

Aplicações *web*

As aplicações *web* são executados em várias plataformas, independentemente do sistema operativo ou do dispositivo, desde que o *browser* seja compatível (Gibb, 2016). Sendo acessível a partir de qualquer dispositivo (computador/*smartphone/tablet*) com acesso à *internet* é uma

solução viável que responde às necessidades em ambas as situações. Neste caso, seria necessário que o *layout* fosse responsivo, isto é, adaptar-se e tornar-se legível e de fácil acesso para ser acedido em qualquer tipo de dispositivo ou plataforma.

Aplicações móveis

Uma aplicação móvel é um *software* desenhado para ser executado num dispositivo móvel (*smartphone* ou *tablet*). Ao ser instalado e acessível apenas através de um dispositivo móvel torna-se essencialmente vantajoso para os cuidadores, e não tanto a nível administrativo e de gestão, que também é fulcral para a empresa. Abordar o tema das aplicações móveis implica a análise e tomada de decisão do desenvolvimento de aplicações móveis nativas ou híbridas e do, ou dos sistemas operativos para os quais desenvolver. Considerando que, em Portugal, os sistemas operativos mais usados são o Android e o IOS (Statcounter, 2022), optar por esta solução implica o desenvolvimento ou de duas aplicações móveis nativas ou de uma híbrida. Optar pelo desenvolvimento de uma aplicação híbrida seria um processo mais rápido. No entanto, poder-se-ia perder um pouco da experiência de utilização que as aplicações nativas trazem (Brothers, 2021). De qualquer das formas, esta opção é desde já descartada, pelo facto das camadas de gestão da empresa não terem de efetuar o seu trabalho através de um *tablet/smartphone*.

Progressive Web Applications (PWA)

Uma aplicação *web* progressiva é executada na *web* através do *browser*, o que implica que possa ser executada em várias plataformas, independentemente do sistema operativo. Tenta combinar o melhor de uma aplicação *web* e de uma aplicação móvel (Sam Richard, 2020). É uma solução viável porque satisfaz as necessidades dos cuidadores, e dos administradores/gestores da empresa.

Comparando com a tradicional aplicação *web* e aplicações móveis, existem algumas diferenças e vantagens que se salientam (CssChopper, 2021) (Dziuba, 2022):

- **Compatibilidade e facilidade de acesso:** Uma PWA pode ser instalada no computador, no *smartphone* ou *tablet*, ao contrário da aplicação *web*. É de fácil acesso, bastando partilhar o *link* da “aplicação *web*” para conseguir aceder e instalar;
- **Envolvimento e *look and feel* semelhante às aplicações móveis:** Proporciona uma experiência semelhante à de uma aplicação móvel e consegue usar muitas das

funcionalidades que só se conseguiria com o desenvolvimento de uma aplicação móvel, como por exemplo as *push notifications*². Não se pode esperar que a interface de utilizador (UI) seja como uma aplicação móvel nativa. Ainda que não seja uma característica importante para o desenvolvimento deste projeto, dado o público-alvo ser muito específico, o facto de poder recorrer ao uso de *push notifications* permite criar um maior envolvimento com a aplicação em si;

- **Auto-atualizações e funcionamento em modo *offline*:** Uma PWA é atualizada sem que o utilizador necessite de clicar em nenhum botão, ou seja, todos os novos recursos estão disponíveis automaticamente. Por outro lado, uma das grandes vantagens das PWA é que funcionam mesmo quando o utilizador não tem ligação à *internet*;
- **Pouca memória:** Como não necessita ser instalado como uma aplicação móvel “normal”, uma PWA ocupa muito pouca memória nos dispositivos móveis, ao contrário de outras aplicações móveis.

Também existem desvantagens que convém salientar (Dziuba, 2022) :

- **Acesso limitado ao hardware e funcionalidades limitadas:** uma PWA tem acesso limitado a determinado *hardware* presente nos *smartphones* como o *Global Positioning System*³(GPS), a câmara, o bluetooth, a lista de contactos, entre outros. Por outro lado, para dispositivos iOS, o PWA ainda não funciona com o *Face ID/Touch ID*⁴, a Siri⁵ entre outros recursos principais da Apple. No entanto, para o problema que se pretender resolver, nenhum deste *hardware* e funcionalidades são necessárias;
- **Desempenho e consumo de bateria:** comparativamente com uma aplicação móvel nativa, em princípio conseguirá um melhor desempenho, por ser desenvolvida na linguagem nativa do dispositivo móvel. Por outro lado, o facto da PWA precisar de acesso à *internet*, a bateria do *smartphone* é sempre consumida mais rapidamente.

² Push notifications é uma notificação que o utilizador recebe numa aplicação de telemóvel, tablet ou num browser sem a requisitar

³ GPS – é um sistema de navegação por satélite que indica no aparelho recetor móvel a posição atual em que se encontra

⁴ Face ID – Possibilidade de autenticação no telemóvel através do reconhecimento facial

Touch ID – Possibilidade de autenticação no telemóvel através de uma impressão digital

⁵ Siri - A Siri é um assistente virtual que atualmente incorpora todo o ecossistema desenvolvido pela Apple

2.3.2 Custo de desenvolvimento

O custo é um fator importante na escolha da solução a adotar. Mesmo que o desenvolvimento da aplicação móvel já tenha sido colocado de parte, por apenas servir melhor uma das partes interessadas, o seu custo de desenvolvimento também seria mais elevado. Uma empresa de desenvolvimento de *software* a ClockWise apresenta um pequeno estudo que compara os custos de desenvolvimento de uma aplicação web versus os custos de desenvolvimento de aplicações móveis nativas e híbridas, ambas para IOS e Android (Clockwise, 2021). No estudo, é salientado que o custo do desenvolvimento de uma aplicação *web* pode ser substancialmente mais reduzido que as outras opções em análise. Por outro lado, é importante referir que caso se optasse pelo desenvolvimento de aplicações móveis, nativas ou híbridas, teria de adicionar o custo das App Stores onde a aplicação móvel fosse publicada. Por exemplo, se o desenvolvimento fosse para Android e IOS, o custo da Google Play Store são de 25\$ (Google, s.d.) e da Apple App Store 99\$ anuais (Store, s.d.). No caso do desenvolvimento de uma aplicação *web*, seja ela progressiva ou não, a mesma seria instalada nos servidores já existentes da empresa, o que tornaria o seu desenvolvimento menos oneroso. Por outro lado, já existe conhecimento consolidado na empresa no desenvolvimento e instalação de aplicações *web*, o que é sempre uma vantagem face ao desenvolvimento de aplicações móveis.

2.3.3 Acesso à internet

Em qualquer das opções mencionadas é necessário o acesso à *internet*, o que não aparenta ser um problema para o público-alvo. Estatísticas do INE, indicam que em 2016, cerca de 74% das famílias portuguesas tinham *internet* em casa, que os equipamentos mais utilizados para aceder à *internet* eram o *smartphone* (78%) e o computador portátil (73%) e que cerca de 72% dos utilizadores já acediam em mobilidade à *internet* (INE, 2016). Além disso, atualmente, todos os cuidadores quando preenchem os documentos necessários aquando da realização de um serviço já têm de enviar os mesmos à administração da empresa ou entregar pessoalmente. Atualmente os cuidadores não se deslocam às instalações da empresa para entregar a documentação, assumindo assim que todos dispõem de acesso à *internet*. Relativamente ao tipo de dispositivo para aceder à *internet*, um estudo da TGI da Marktest, de julho de 2020, indica que cerca de 7 milhões de portugueses já usam *smartphone* (Marktest, 2020), o que implica que o desenvolvimento de uma solução para dispositivos móveis possa ser uma opção.

2.3.4 Desafios no desenvolvimento de *software* para diferentes plataformas

Já tendo sido descartadas as opções da aplicação *desktop* e da aplicação móvel (nativa ou híbrida), resta apenas a aplicação *web* e a aplicação *web* progressiva. Tanto uma solução como a outra implicam o desenvolvimento de um *software* para diferentes plataformas. Ou seja, é expectável que, qualquer que seja a escolha, possa ser acessível em *smartphones* ou *tablets* para o sistema operativo Android e IOS (preferencialmente Android) e em computadores pelo menos com o sistema operativo Windows. No entanto, o universo de sistemas operativos é mais amplo, podendo ainda incluir os populares MacOs e Linux. Além disso, sendo ambas as hipóteses aplicações *web*, têm de ser sempre acessíveis através de um *browser*. Neste caso, são inúmeras as opções que podem existir, desde o Google Chrome, o Safari, o Microsoft Edge (Microsoft, 2022), o Firefox (Mozilla, 2022), o Opera (Opera, 2022), entre outros.

Desenvolver para cada uma dessas plataformas requer diferentes tipos de conhecimento de programação, o que implica um maior custo no seu desenvolvimento. O desenvolvimento multiplataforma é vantajoso porque é desenvolvida apenas uma aplicação, numa determinada linguagem que pode ser compilada para várias plataformas. Nesse sentido, optar por uma solução baseada na *web* permite mitigar este problema do desenvolvimento para multiplataformas (Ruparel & Vidhya, 2018).

Ao desenvolver em multiplataforma teremos um maior alcance do público pois a aplicação poderá ser utilizada em diversas plataformas, seja ele Windows, IOS, Linux, MacOs, Android, entre outros. Como já referido, numa aplicação multiplataforma, o código desenvolvido é somente um, visto que o mesmo pode ser executado nos variados dispositivos. Estas também têm uma maior longevidade, isto é, conforme os sistemas operativos sofrem atualizações com novas versões, a aplicação continua a funcionar, necessitando eventualmente de pequenas modificações, no entanto, as atualizações são sincronizadas simultaneamente em todas as plataformas.

Uma outra vantagem que se enquadra bastante com o caso uso desta dissertação é o movimento "*Bring Your Own Device*" (BYOD) (Forcepoint), que se refere ao incentivo, por parte das empresas, para que os seus funcionários tragam os seus próprios dispositivos móveis para o trabalho. Tal como já foi referido, os cuidadores trabalham essencialmente no terreno e vão depender muito de um dispositivo móvel (*smartphone/tablet*) para preencher todos os dados necessários a cada serviço prestado. Também foi referido que outra opção pode ser o uso de

um computador portátil, mas não lhes trará a mesma facilidade no preenchimento de toda a informação necessária. Por outro lado, dada a elevada rotatividade deste tipo de trabalhos, não se consegue ter uma perceção do tipo de dispositivo móvel que cada um use. Nesse sentido, o desenvolvimento de uma aplicação multiplataforma torna-se ainda mais relevante.

Um dos desafios relaciona-se com a interface do utilizador, havendo a necessidade de que esta seja simples e funcional para os variados dispositivos. Os ícones usados devem ser de fácil perceção, pois, se um utilizador comum não entender o funcionamento da aplicação, é bem provável que procure outra opção, portanto este ponto é necessário a ter em conta no decorrer do desenvolvimento, pois existem computadores, *smartphones* e *tablets* de diferentes tamanhos, resoluções e desempenho (Madeinweb). Outro desafio está relacionado com as funcionalidades da aplicação, alguma alteração futura que seja disponibilizada após o término do desenvolvimento pode trazer dificuldades na implementação, visto que pode não haver um suporte instantâneo a essas novas funcionalidades.

O desenvolvimento em multiplataforma está cada dia mais evoluído, a ponto de já apresentar mais vantagens que desafios. O desenvolvimento em multiplataforma não pode simplesmente ser implementado para toda e qualquer aplicação, é necessário compreender a verdadeira necessidade, para assim se escolher a melhor abordagem. Não quer dizer que esta tecnologia seja mais vantajosa que a nativa, pois cada uma tem particularidades distintas, no entanto, olhando para o cenário atual do avanço da tecnologia é importante garantir que as aplicações possam ser utilizadas nas diversas plataformas existentes, para que os utilizadores tenham limitações no seu uso.

2.4 Progressive Web Applications

Nesta secção é exposta a evolução das PWA ao longo do tempo e, também a compatibilidade entre *browsers* vs sistemas operativos.

2.4.1 Evolução das PWA

Apesar de existirem algumas referências que indicam que, em 2007, o conceito de PWA foi introduzido por Steve Jobs durante a introdução do iPhone, o termo oficial surge anos mais tarde, em julho de 2015, por parte de dois Engenheiros da Google, Frances Berriman e Alex Russell.

Frances Berriman e Alex Russell sugerem que as ideias de aplicações nativas sempre foram de regressão, de segurança duvidosa e muitas atualizações. Por outro lado, eles observaram que, já em 2015, começavam a surgir um conjunto de *sites* que ofereciam uma experiência para o utilizador diferenciada das tradicionais aplicações *web*, as quais chamaram de aplicações *web* progressivas. Um ano mais tarde, numa conferencia da Google, este conceito foi apresentado como sendo um novo padrão de desenvolvimento *web*. Desde então que as PWA têm sido vistas como o futuro. Logo em 2017, a Microsoft deu uma resposta positiva e proporcionou os primeiros sinais de suporte às PWA para a sua *stack* tecnológica Windows. Os primeiros sinais de suporte às PWA por parte da Apple surgem apenas em 2018 e de forma bastante limitativa.

À medida que as gigantes tecnológicas Google, Microsoft e Apple iam anunciando novas evoluções nos seus suportes, muitas empresas não tardaram em desenvolver ou converter a sua atual aplicação para uma PWA. Logo em 2016, após o anúncio da Google deste novo paradigma de desenvolvimento, a FlipKart (Flipkart, s.d.) (uma plataforma indiana de e-commerce) e o AliExpress (Aliexpress, s.d.) converteram para PWA e obtiveram bons resultados. Desde então, inúmeras empresas têm seguido os mesmos passos (ScandiPWA, 2019).

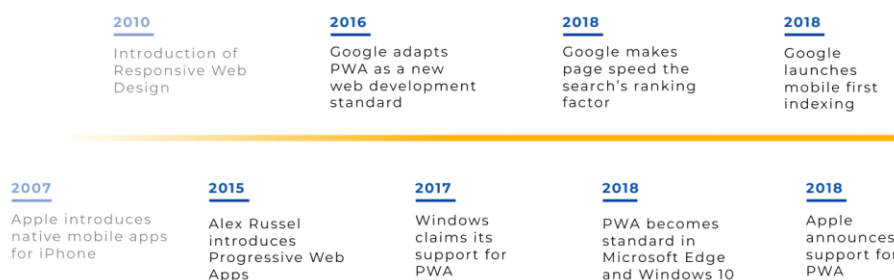


Figura 5 - Evolução das PWA ((Divante, 2019))

Como é possível observar na Figura 5, é clara a evolução e o suporte cada vez maior neste tipo de aplicações *web*. Ainda assim, a Apple continua a ser a mais restritiva. A Figura 5 pretende sintetizar os principais marcos no aparecimento e evolução das PWA.

2.4.2 Compatibilidade de Browsers versus Sistemas Operativos nas PWA

Um fator importante a ter em atenção no desenvolvimento de uma PWA é a sua compatibilidade com os diversos *browsers* e sistemas operativos, quer seja num dispositivo

móvel ou num computador. A importância desta compatibilidade e dos seus desafios em termos gerais já foi abordada na secção 2.4.4, no entanto, é fundamental referir o estado atual no caso do desenvolvimento para uma aplicação *web* progressiva.

Das vantagens que as aplicações *web* progressivas trazem, para este projeto em específico, é importante analisar em termos de compatibilidade as seguintes funcionalidades: *Geolocalização*, *Home Screen*, *Push Notifications*, Experiência de utilização como uma aplicação móvel e Funcionamento em modo *offline*. Para cada uma destas funcionalidades é apresentada uma tabela indicativa do seu funcionamento em termos de *browsers* (Google Chrome, Safari, Edge, Firefox e Opera) e em termos de sistema operativo (Android, IOS, Windows, MacOS e Linux).

Em termos de dispositivos móveis, como já foi mencionado, é expectável que possa suportar Android e IOS, no entanto, o Android será sempre preferencial. Em termos de *desktop*, a empresa espera que, pelo menos seja suportado para Windows, sendo este o requisito mínimo. Em termos de dispositivos móveis, como já foi mencionado, é expectável que possa suportar Android e IOS, no entanto, o Android é sempre preferencial. Em termos de *desktop*, a empresa espera que, pelo menos seja suportado para Windows, sendo este o requisito mínimo. Considera-se importante:

- **Geolocalização:** não é, uma funcionalidade requisitada atualmente, mas a empresa considera que, em funcionalidades futuras, poderá vir a ser necessária. Observando a Figura 6 , destaca-se uma total disponibilidade em todos os sistemas operativos para o Google Chrome. Considerando que este *browser* é o mais usado, por estatísticas já referidas, considera-se que não existirão problemas de compatibilidade. Além disso, com exceção do Safari, todos os restantes *browsers* têm um elevado grau de compatibilidade;











			CHROME	SAFARI	EDGE	FIREFOX	OPERA
		ANDROID	✓	--	✓	✓	✓
		WINDOWS	✓	--	✓	✓	✓
		IOS	✓	✓	✓	✓	✓
		macOS	✓	✓	✓	✓	✓
		LINUX	✓	--	--	✓	✓

Figura 6 - Compatibilidade para Geolocalização (Santoni, 2021)

- Home screen:** o facto da aplicação poder ser acedida através de um *click*, ao invés de aceder pelo *browser*, dá a sensação de que a mesma se encontra instalada no dispositivo (seja ele móvel ou um computador), tornando assim mais simples o seu acesso. Pela Figura 7, observa-se que esta funcionalidade ainda tem uma baixa compatibilidade em termos de *browsers* por sistema operativo. Ainda assim, o Google Chrome é o que tem maior compatibilidade, não sendo apenas suportado, à atualidade, no IOS. Nesse sentido, os utilizadores de IOS teriam sempre de aceder pelo *browser*, como se de uma simples aplicação *web* se tratasse. Ainda assim, podem usufruir das restantes potencialidades de aplicações *web* progressivas;











			CHROME	SAFARI	EDGE	FIREFOX	OPERA
		ANDROID	✓	--	✓	✓	✓
		WINDOWS	✓	--	✓	✗	✗
		IOS	✗	Home screen Launch screen ✓ ✗	✗	✗	✗
		macOS	Home screen Launch screen ✓ ✓ --	--	✓	✗	✗
		LINUX	Home screen Launch screen ✓ ✓ --	--	--	✗	✗

Figura 7 - Compatibilidade para Home Screen (Santoni, 2021)

- **Push notifications:** serão usadas para enviar notificações relevantes aos diversos utilizadores. Analisando a Figura 8 , o sistema operativo IOS é o mais limitado em termos de compatibilidade, mesmo usando o Google Chrome;











			CHROME	SAFARI	EDGE	FIREFOX	OPERA
		ANDROID	✓	-	✓	✓	✓
		WINDOWS	✓	-	✓	✓	✓
		IOS	✗	✗	✗	✓	✗
		macOS	✓	✓	✓	✓	✓
		LINUX	✓	-	-	✓	✓

Figura 8 - Compatibilidade para Push Notifications (Santoni, 2021)

- **Experiência de utilização como uma aplicação móvel e funcionamento em modo *offline*:** o *service worker* é uma característica usada nas aplicações *web* progressivas que disponibiliza um conjunto de funcionalidades que habilitam o funcionamento em modo *offline* e uma experiência de utilização mais avançada. Mais detalhes sobre este conceito são referidos na secção 4.4.1. Observando a Figura 9 , o sistema operativo IOS é o mais limitado em termos de compatibilidade, mesmo usando o Google Chrome. No entanto, para utilizadores de IOS, estes podem usar o Safari, pois já existe compatibilidade.





			CHROME	SAFARI	EDGE	FIREFOX	OPERA
		ANDROID	✓	-	✓	✓	✓
		WINDOWS	✓	-	✓	✓	✓
		IOS	✗	✓	✗	✗	✗
		macOS	✓	✓	✓	✓	✓
		LINUX	✓	-	-	✓	✓

Figura 9 - Compatibilidade para Experiência de utilização (Santoni, 2021)

2.4.3 Fatores de sucesso em PWA

De uma forma geral, as aplicações *web* progressivas tem apresentado bons resultados. Por exemplo, o Twitter (Twitter, s.d.) teve um aumento de 65% nas páginas por sessão, 75% mais de Tweets⁶ e uma redução de 20% na taxa de rejeição, não esquecendo que o tamanho da aplicação foi reduzido em mais de 97% (Sam Richard, 2020). Outro bom exemplo é a Pinterest (Pinterest, s.d.), que após fazer a transição para uma PWA, aumentou em mais de 60% o envolvimento e mais de 50% o número de cliques em publicidade (YML, 2021).

2.5 Sistematização de tecnologias

A Tabela 2 mostra o estudo comparativo entre as duas principais soluções em análise (Aplicação *Web* e Aplicação *Web* Progressiva) tendo em conta fatores como *Push notifications*, Instalação, Dependência de conectividade e Velocidade de carregamento.

De acordo com as necessidades do público-alvo, o custo de desenvolvimento/manutenção, o conhecimento já presente na empresa e numa perspetiva de desenvolver uma solução que possa evoluir e adaptar-se mais facilmente à evolução natural da tecnologia, a escolha do desenvolvimento de uma aplicação *web* progressiva é a mais assertiva. Por outro lado, o facto de já existirem inúmeros casos de sucesso comprovados de empresas que transitaram para PWA e obtiveram bons resultados, só ajuda a reforçar a escolha (YML, 2021).

Tabela 2 - Comparação de soluções (AltexSoft, 2018)

	Push Notifications	Instalação	Dependência de conectividade	Velocidade de carregamento
Aplicação web	O envio de notificações requer uma prévia aceitação por parte do utilizador	Não é instalada no dispositivo móvel ou computador. Acessível pelo <i>browser</i> .	Não funciona sem acesso à <i>internet</i> .	Não dispõe do mesmo mecanismo que as PWA
Aplicação web progressiva	Permite o envio de notificações como se se tratasse de uma aplicação móvel	Não é instalada no dispositivo móvel ou computador. Basta apenas aceder a ela pelo <i>browser</i> . Pode ser adicionada à <i>home screen</i> no caso de um dispositivo móvel ou ainda como aplicação no caso do sistema operativo Windows.	Consegue funcionar em modo <i>offline</i> , caso não tenha acesso à <i>internet</i>	O modo como as PWA foram desenhadas permite o carregamento de páginas em <i>cache</i> , tornando assim o seu carregamento mais rápido

⁶ A ideia do tweet na rede social é uma comparação com a sequência de pequenas publicações que os usuários do Twitter fazem, já que só é possível apenas a postagem de textos com até 140 caracteres

Mesmo que ainda existam algumas incompatibilidades por parte de alguns *browsers* nos diversos sistemas operativos, essa tendência é para desaparecer. Mesmo o sistema operativo IOS, o que atualmente ainda tem alguma incompatibilidade, nas atualizações que tem lançado, tem adicionado cada vez mais funcionalidades que habilitam o desenvolvimento de uma PWA (Chris, 2021). Nesse sentido, nota-se uma preocupação e uma percepção de que o desenvolvimento deste tipo de aplicações pode mesmo ser o futuro.

2.6 Questionários

Os questionários são uma das funcionalidades que deve ser implementada na solução adotada. Tal como já foi referido na secção 2.4 os utentes e/ou pessoas relacionadas são convidados a responder a um breve questionário, atualmente em papel, no fim da prestação de serviços.

Independente da solução tecnológica escolhida na secção 2.5, podem ser analisadas duas principais abordagens: desenvolver um gestor de questionários de raiz ou usar uma solução já existente no mercado. O custo é um fator importante na escolha a seguir. Por outro lado, seria importante que estes questionários pudessem ser atualizados à medida das necessidades. É ainda importante existirem estatísticas dos questionários realizados, de modo a que os órgãos de gestão da empresa possam analisar, tirar conclusões e agir, em prol da qualidade dos serviços prestados. Nesse sentido, analisa-se de seguida cada uma das abordagens referidas.

2.6.1 Desenvolvimento de um gestor de questionários de raiz

Uma solução de raiz provavelmente levará a um acréscimo de tempo no desenvolvimento do projeto, o que poderia ser mais dispendioso, apesar de o questionário atualmente ser simples segundo o Anexo I, pode ser ajustado no futuro.

2.6.2 Integrar uma solução já existente

Em termos de soluções já existentes, é objetivo que a mesma possa ser facilmente integrada na PWA. Nesse sentido e, após pesquisa, foram encontrados diversos gestores de questionários online, sendo que muitos deles disponibilizam *Application Programming Interfaces* (APIs), para que possam facilmente ser integrados neste projeto.

Na Tabela 3 é possível observar as diversas opções analisadas face aos critérios definidos. Todas as opções analisadas dispõem de uma API que permita aceder aos questionários criados e às suas estatísticas, o custo torna-se assim um fator eliminatório.

Apenas o *QuestionPro* (QuestionPro, 2022) tem um plano gratuito e ilimitado do número de respostas e questionários a criar.

Tabela 3 - Diferentes opções para os questionários

Crítérios	Preço	Acesso a estatísticas	Disponibiliza API	Comentários
Survey Monkey (SurveyMonkey, 2022)	Plano grátis apenas permite 10 questões por questionário; Planos pagos a partir de 16€/mês, até 39€/mês.	Sim	Sim, com acesso aos questionários e estatísticas	
Google Forms (Google, 2022)	Plano grátis para utilizadores individuais e com planos pagos a começar nos 4.68€/mês/utilizador. Todos eles com limitações a nível de espaço	Sim	Sim, com acesso aos questionários e estatísticas	A API ainda se encontra na versão beta, tendo saído apenas no final do ano passado
QuestionPro (QuestionPro, 2022)	Plano grátis	Sim	Sim, com acesso aos questionários e estatísticas	
SurveyLegend (Surveylegend, 2022)	Plano grátis com limitações a nível de questionários (apenas 3). De seguida o custo passa para 15€/mês permitindo apenas 20 questionários e apenas por 25€/mês se consegue sem restrições no número de questionários	Sim	Sim, com acesso aos questionários e estatísticas	

A única limitação que deve ser salientada, mas que não foi identificada como um problema para a empresa, é o facto de apenas permitir 1000 respostas por dia. O número de questionários aplicados é reduzido e espaçados no tempo, não se tornando por isso um problema. Conclui-se que das opções analisadas é a que melhor se enquadra, sendo por isso a selecionada.

2.7 Tecnologias de relevância no projeto

Concluída a análise de abordagens tecnológicas possíveis para o desenvolvimento deste trabalho, na secção 2.4 e, tendo sido concluído o desenvolvimento de uma PWA, torna-se agora importante referir as restantes tecnologias usadas. Sendo um projeto desenvolvido para uma empresa, por vezes existem condicionantes tecnológicas que, logo à partida, ditam em quais tecnologias vai ser desenvolvido.

Neste sentido e, por uma questão de algum conhecimento e experiência na empresa, a PWA é desenvolvida numa *stack* tecnológica *open-source*⁷. Apesar das PWA já poderem ser

⁷ O termo "open source" é bastante usado para se referir a programas que não têm custos para utilização.

desenvolvidas em diversas linguagens de programação, esta é desenvolvida em C#, usando a *framework* ASP.NET Core 6.0 Blazor. De uma forma bastante resumida e introdutória, o Blazor é uma *framework web* baseado em tecnologias web já existentes como o HTML e o CSS, que usa C# e Razor ao invés de JavaScript (Microsoft, 2022).

Em termos de base de dados, também por decisão da empresa, é usada o MySQL (MySQL, 2022). Como ambiente de desenvolvimento é usado o Visual Studio 2022 Community (Microsoft, 2022). Usará como gestor de versões o Git (Git, 2022) e o código fonte do projeto estará na plataforma de hospedagem Bitbucket (Bitbucket, 2022).

2.8 Sumário

Neste capítulo, foram abordados os principais conceitos subjacentes no âmbito desta dissertação. Contextualizou-se o setor no apoio domiciliário, trabalhos de interesse relacionados com o tema. Por fim, foram analisadas as diferentes formas de resolver o problema, assim como feita uma sistematização das tecnologias que podem ser usadas para o seu desenvolvimento.

Capítulo 3 - Análise de Valor do Negócio

Neste capítulo será apresentada a análise de valor dos resultados do trabalho desenvolvido no âmbito desta tese/dissertação. É feita uma identificação e análise das oportunidades existentes.

A análise de valor é uma metodologia utilizada para encontrar o melhor valor de um produto, bem ou serviço com o objetivo de reduzir custos desnecessários na sua produção e, em simultâneo, maximizar o seu valor.

3.1 Modelo de Desenvolvimento de Novos Conceitos

Segundo (Koen et al, 2002) o processo de inovação pode ser dividido em três fases: *Fuzzy Front End* (FFE), o processo *New Product Development* (NPD) e comercialização. A Figura 10 apresenta o processo de inovação.

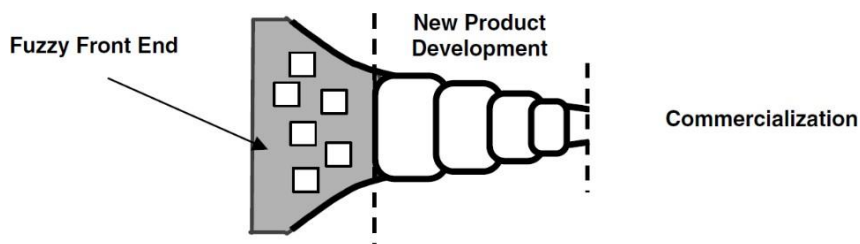


Figura 10 - Processo de Inovação (Koen, et al., 2002)

De acordo com (Koen et al, 2002), as fases podem ser descritas como:

- **Fuzzy Front End (FFE):** A primeira fase, o *Fuzzy Front End*, corresponde ao momento da identificação do problema ou à captação de oportunidades/ideias para o projeto. Esta fase é caracterizada por processos e decisões não estruturadas que têm como objetivo a identificação das necessidades dos clientes;
- **New Product Development (NPD):** A segunda fase, o *New Product Development*, centra-se por ser uma fase onde as ideias já estão estruturadas, os objetivos já estão definidos. As datas são definidas e existem equipas a tratar do desenvolvimento e teste dos produtos;
- **Comercialização:** A terceira fase, a Comercialização, corresponde à fase em que o produto é introduzido no mercado, ou seja, corresponde à comercialização do produto desenvolvido.

3.2 New Concept Development

O *New Concept Development Model (NCD)* é um modelo não linear de inovação, criado por Peter Koen (Koen et al, 2002) e trata-se de um modelo de representação das principais atividades do FFE. Na Figura 11 é possível analisar que este modelo tem três componentes:

- **Motor:** O motor encontra-se relacionado com a liderança, o plano estratégico e a cultura existentes numa empresa. Este é responsável por suportar os cinco elementos do processo de inovação;

- **Fatores de influência:** Os fatores de influência consistem nas variáveis de natureza que não são controláveis e que afetam todo o processo de inovação, podendo ter origem interna ou externa à organização. Estes fatores podem ser de natureza política, tecnológica, económica, etc;
- **Cinco Elementos:** Os cinco elementos, representam as cinco atividades do FFE: identificação de oportunidades, análise de oportunidades, geração e aperfeiçoamento de ideias, seleção da ideia e definição de conceitos.

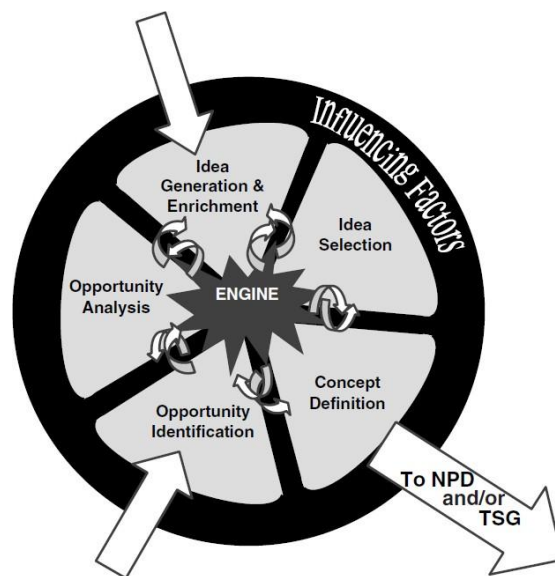


Figura 11 - New Concept Development (Koen et al, 2002)

O modelo tem duas entradas e uma única saída. Este pode iniciar-se pela identificação de uma oportunidade ou pela geração de uma ideia.

3.2.1 Identificação de Oportunidades

A identificação de oportunidades é um dos pontos de partida do modelo NCD. Este consiste na fase onde são identificadas as diferentes oportunidades existentes, ou seja, tem como objetivo identificar as oportunidades que se pretende analisar. Estas oportunidades podem estar relacionadas com o contexto de dar resposta à concorrência, através de melhorias nos processos, alterações a serviços atuais ou até a invenção de novos produtos e serviços (Koen et al, 2002). Em 2020 segundo o estudo “*Impacto da pandemia de Covid19 nas IPSS e seus utentes em Portugal*” elaborado pela Área Transversal de Economia Social (ATES) com a Confederação

Nacional das Instituições de Solidariedade (CNIS), o apoio domiciliário aumento 67.6%. (ATES & CNIS, 2021)

Atualmente os sistemas de *software* são um contributo para a melhoria da qualidade e eficiência dos processos na área da saúde. Estes sistemas, mostraram-se uma ferramenta de trabalho fundamental no sentido da disponibilização da informação de uma forma rápida e estruturada, o que simplifica o trabalho e a tomada de decisão das organizações (Solanas et al, 2017). O aumento da procura por este tipo de serviços de apoio domiciliário, pode levar a um aumento em 15% neste mercado, constituindo-se uma oportunidade de negócio para o sector privado. (Yoopies, 2021).

É neste mercado de apoio domiciliário que se insere a empresa **Vida e Conforto**. Com a empresa identificou-se a necessidade de informatização e otimização dos processos internos derivado ao elevado volume de trabalho para melhorar os fluxos de trabalho da empresa e a comunicação entre os diferentes profissionais. Atualmente a empresa faz a gestão dos utentes dos serviços prestados e do horário dos colaboradores em ficheiros Excel. Esta gestão, sem o apoio de ferramentas específicas consome um grande número de horas de trabalho diário. É um processo moroso também para a consulta de dados o que o torna sujeito a falhas e dificulta a análise dos dados.

O projeto GesSAD tem como objetivo o desenvolvimento de um sistema de apoio á prestação de cuidados no domicílio. Este sistema será uma ferramenta de trabalho/registo para os cuidadores e profissionais de saúde, permitindo apoiar e gerir os serviços de apoio domiciliário e ajudar na gestão interna da empresa.

De acordo com a Figura 12, podemos observar o aumento das páginas em PWA entre 2016 e 2021.

É ainda importante salientar que as PWA são indicadas como sendo o futuro da web para dispositivos móveis (YML, 2021), o que reforça a certeza da escolha. Comparativamente com as soluções analisadas na secção 2.3, nenhuma se encontra desenvolvida nesta tecnologia, o que faz com que esta solução esteja um passo à frente nesse sentido.

As soluções analisadas ou são aplicações móveis nativas para Android/IOS ou aplicações *web*.



Figura 12 - Acesso a páginas PWA (Rykov, 2021)

Esta abordagem é abrangente o suficiente caso a empresa decida evoluir a aplicação e, por exemplo, passar a disponibilizar acesso a determinadas funcionalidades aos seus utentes e/ou pessoas de referência.

3.2.2 Análise de oportunidades

A análise de oportunidades consiste na fase segundo (Koen et al, 2002) onde é ou não confirmado se as oportunidades identificadas previamente são a abordagem correta a seguir, ou seja, consiste na validação da viabilidade da oportunidade identificada anteriormente. Para a oportunidade identificada, aplicaram-se os seguintes métodos: *Customer assessment*⁸ e *Competitor analysis*⁹. Foram realizadas reuniões com os gestores da empresa **Vida e Conforto** com o objetivo de identificar as principais necessidades, identificar os problemas, compreender o fluxo do dia-a-dia e as necessidades existentes. Estas necessidades encontram-se descritas na secção 2.4.

Foi realizado um estudo de mercado de forma a identificar e analisar as soluções já existentes nesta área de prestação de serviços ao domicílio. Este estudo encontra-se descrito na secção 2.3.

⁸ Customer assessment - Avaliação das necessidades e expectativas do cliente

⁹ *Competitor analysis* - Avaliação dos pontos fortes e fracos dos concorrentes atuais e potenciais futuros.

3.3 Valor da solução

Após a identificação e análise de oportunidades da solução a ser executada é necessário perceber o valor que esta tem para os utilizadores finais, no caso deste projeto, as empresas e organizações que estejam no setor dos serviços de apoio domiciliário. Desta forma serão utilizados o conceito de proposta de valor e o modelo de negócio *CANVAS*.

3.3.1 Proposta de valor

A proposta de valor tem um papel essencial na forma como se desenvolve um produto de modo a apresentar os seus benefícios para os seus clientes. Visa apresentar de forma clara os benefícios do produto, quais os problemas que pretende resolver, descrever como este se diferencia da concorrência, e de que modo o cliente deverá optar por este produto para colmatar as suas dificuldades. Existe uma ferramenta utilizada normalmente em empresas para organizar as ideias e entender os problemas existentes de modo a desenvolver um produto sólido e que vá de acordo com a expectativa, a proposta de valor *Canvas* (Strategyzer, 2022). Este é um modelo dividido em duas partes, uma parte sobre os utilizadores na qual são abordados os problemas, e a proposta de valor na qual se esclarece como se pretende valorizar o utilizador (Alexander Osterwalder, 2014).

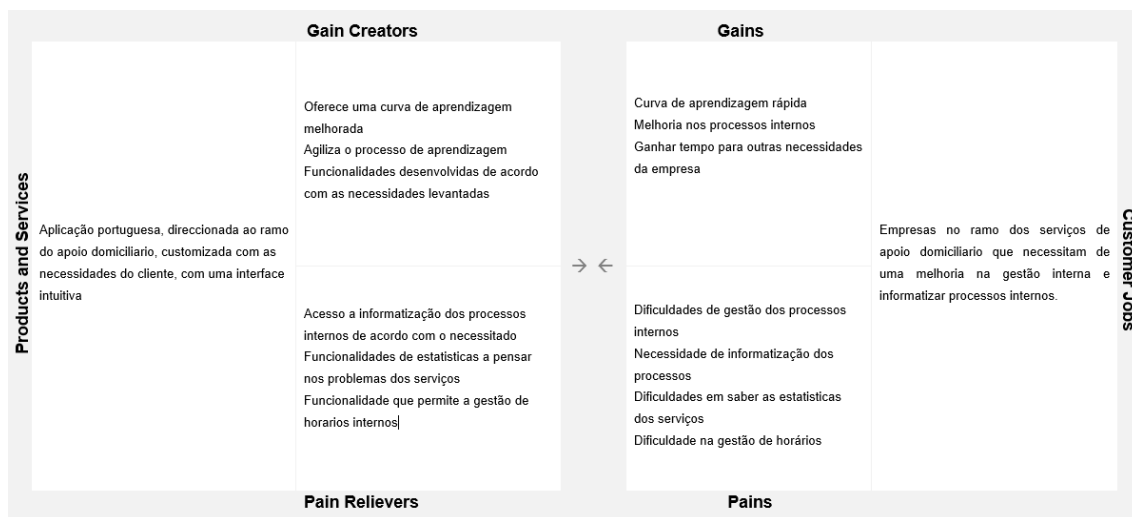


Figura 13 - Proposta de Valor

A proposta de valor deste projeto consta no desenvolvimento de uma aplicação *web* progressiva que permita melhorar a gestão e o fluxo de trabalho da empresa, bem como a

informatização dos processos internos e facilitar a interação entre os diversos utilizadores da aplicação.

De acordo com a análise da Figura 13 Figura 13 - Proposta de Valor destaca-se sobre as empresas:

- **Gains:** As empresas procuram melhorias nos processos internos, uma curva de aprendizagem rápida e pretendem ganhar tempo para outras necessidades da empresa;
- **Customer Jobs:** Os trabalhos procurados pelas empresas no setor de serviço de apoio domiciliário encontram-se associados a necessidade de uma melhoria na gestão interna da empresa e a informatização dos processos utilizados internamente;
- **Pains:** Os principais problemas encontram-se na dificuldade de gestão dos processos internos, a necessidade de informatização dos processos utilizados, a dificuldade em saber as estatísticas dos serviços e a dificuldade na gestão de horários.

Sobre a proposta de valor:

- **Gain Creators:** A proposta pretende criar valor a estas empresas, através do foco no desenvolvimento de funcionalidades desenvolvidas de acordo com as necessidades destas empresas, oferecer uma curva de aprendizagem melhorada e agilizar esse mesmo processo;
- **Pain Relievers:** Os problemas das empresas são colmatados através da criação de funcionalidades que permitem a gestão dos horários internos, a criação de funcionalidades para verificar as estatísticas, e a informatização dos processos de acordo com as necessidades;
- **Products and Services:** O produto oferecido é uma aplicação portuguesa, direcionada a este ramo específico e customizada de acordo com as necessidades levantadas, com uma interface intuitiva.

3.3.2 Modelo de Negócio de Canvas

O modelo de negócio de *Canvas*, proposto inicialmente por Alexander Osterwalder em 2004 (Osterwalder, 2004), é uma ferramenta de gestão estratégica que permite a estruturação do negócio e das suas estratégias de uma forma simples e prática. Este modelo pode descrever as características de uma organização como as diferentes partes se organizam para dar valor aos consumidores, como estas se unem dentro da organização e como a organização gera valor

através destas ligações. (Paquin & Joyce, 2016). A implementação deste modelo possibilita uma melhor visualização do negócio.

Parceiros-chave	Atividades -Chave	Proposta de Valor	Relacionamentos	Segmentos de Clientes
Empresas que prestam serviço de apoio domiciliário Organizações de apoio domiciliário	Levantamento de requisitos Desenvolvimento da aplicação web Preparação da infraestrutura e instalação do software Testes à aplicação Recursos-chave Dispositivos móveis/ computador Cuidadores e Profissionais de saúde Programador Infraestrutura para instalar o software	Informatização dos registos internos Registo em tempo real das tarefas e acções realizadas pelos cuidadores/profissionais de saúde Monitorização do trabalho em tempo real Melhor organização do fluxo de trabalho Análise de métricas consideradas importantes	Sistema de inqueritos de satisfação Redes sociais Acompanhamento ao utilizador Formação Canais Internet Comunicação Social Jornais Redes Sociais Publicidade	Empresas que prestem serviços de apoio domiciliário Cuidadores de idosos com elevado numero de utentes Clínicas com serviços de apoio domiciliário Lares
Estrutura de Custos Desenvolvimento da aplicação web Internet Manutenção da aplicação Programador Manutenção da infraestrutura onde o software fica alojado (Servidor, Dominio)		Fontes de Receita Subsidios governamentais e não governamentais		

Figura 14 - Proposta de valor

Da análise da Figura 14 apresenta-se um possível modelo de negócio ligado ao projeto com nove itens:

- **Parceiros Chave:** Neste segmento encontram-se representadas todas as entidades externas à organização que contribuem para a concretização do projeto. As empresas que prestam serviços de apoio domiciliário e as organizações de apoio domiciliário serão os parceiros chave;
- **Atividades Chave:** As atividades chaves designam-se como as ações principais que uma organização deve efetuar para que o seu modelo de negócio tenha sucesso. Neste projeto, o levantamento de requisitos é a base de todas as ações que levará ao melhor desenvolvimento da aplicação *web* progressiva. Por fim a preparação da infraestrutura, instalação do *software* e testes à aplicação serão também necessários;
- **Recursos Chave:** Os recursos chave simbolizam todo o tipo de recursos sem os quais se torna difícil a concretização do modelo de negócio. Deste modo é relevante um programador na vertente de implementação da aplicação *web* progressiva, os cuidadores e profissionais de saúde e ainda ter um dispositivo móvel e/ou computador;
- **Proposta de Valor:** No segmento descreve-se o conjunto de produtos que concebem valor. A proposta de valor exhibe os fatores diferenciadores, pretendendo ser a razão pela qual um cliente opta por uma organização em vez de outra. Neste projeto na secção 3.2 foi

elaborada a proposta de valor deste projeto, no entanto, o fator diferenciador da proposta de valor consiste na informatização dos registos internos da organização;

- **Relação com Clientes:** No segmento de relação com clientes, indica principalmente o tipo de relação que a organização pretende estabelecer com o cliente. A forma como se pretende avaliar a relação com os clientes será através de inquéritos de satisfação, pretende-se também dar formação sobre a aplicação e o devido acompanhamento ao utilizador;
- **Canais:** Neste segmento descreve-se a forma como a organização pretende chegar ao cliente, ou seja, que canais poderão ser usados para dar a conhecer o produto ou serviço. Neste projeto existe um foco na *internet*, ou seja, na publicidade *online* e nas redes sociais, sendo este o meio que cada vez mais as pessoas utilizam (Ine, 2019). As parcerias com a comunicação social e jornais serão também abordagens a adotar na divulgação do produto. Em relação ao canal de distribuição e venda, pretende-se optar por uma distribuição *online*;
- **Segmentos de Clientes:** Neste segmento descreve-se os diferentes segmentos de clientes que se espera que obtenham o produto. Neste projeto, os potenciais compradores desta solução abrangem todo o tipo de organismos que prestem serviços de apoio domiciliários, desde lares, cuidadores de idosos com elevado número de utentes, a clínicas com serviços de apoio domiciliário e empresas que prestem estes serviços;
- **Estrutura de Custos:** Os custos visam descrever todos os gastos com a implementação do modelo de negócio. Neste projeto os custos estão associados ao desenvolvimento da aplicação *web* progressiva, seja com o programador, a manutenção da aplicação e a manutenção da infraestrutura (servidor e domínio) como também a despesa da *internet*;
- **Fontes de Receita:** Neste segmento descreve-se as fontes de retorno resultantes do produto. As fontes de receita basear-se-ão através de subsídios governamentais e não governamentais (Portugal2020, s.d.) ou de âmbito privado.

3.4 Quality Function Deployment

Segundo (Hauser & Gerry Katz, 2010), Quality Function Deployment (QFD), em português designada de técnica de desdobramento da função qualidade é “*uma metodologia de desenvolvimento de produto cujo objetivo é implementar a voz do cliente ao longo de todo o*

processo de desenvolvimento do produto.” Segundo (Silva, 2011), esta técnica teve origem em 1960 pelos professores Shigeru Mizuno e Yoji Akao.



Figura 15 - Modelo QFD (Silva, 2011)

Este método começou a ser desenvolvido na Mitsubishi em 1972 com a aplicação do QFD para o desenho de um navio petroleiro na Estaleiros Kobe. A introdução do QFD para a América e Europa começou em 1983, quando a *American Society for Quality Control* publicou o trabalho de Akao, "*Progresso Qualidade e Pesquisa*". Na Figura 15 é possível verificar as diferentes variáveis deste modelo proposto pelos professores Shigeru Mizuno e Yoji Akao. Para um melhor entendimento deste método, o primeiro passo consiste na definição das necessidades da organização. Nas reuniões realizadas com a empresa foram descritas as necessidades da organização para o projeto. De modo a quantificar o peso de cada uma destas necessidades foi utilizada a escala de Likert, com uma escala de 1 (menos importante) até 5 (mais importante) (Cunha, 2007). Calculou-se o peso absoluto, o qual assinala o grau de importância dos requisitos. Com os requisitos do cliente, realizou-se a conversão para requisitos técnicos. Da relação entre os requisitos do cliente com os requisitos técnicos obtivemos o grau de importância dos requisitos do projeto.

A Tabela 4 permite representar as relações entre as necessidades dos consumidores e as características de engenharia

Tabela 4 - Matriz de Corelação

Matriz de Corelação:		
+	.	-
Positiva	Sem corelação	Negativa

Na Tabela 5 ilustra o sistema utilizado para corresponder a um grau de corelação: 9 corresponde a uma corelação forte, 5 a uma corelação moderada e 1 a uma corelação fraca.

Tabela 5 - Matriz de Relação

Matriz de Relação:			
9	3	1	
Relação Forte	Relação Moderada	Relação Fraca	Sem Relação

Analisando a Figura 16 e a Figura 17 verificamos que os quatro requisitos com maior importância foram: utilização de algoritmos para encriptação dos dados, possibilidade de fazer login na aplicação, a informação da aplicação e o texto sejam adequados e funcionais, e a necessidade de gerir as pessoas com acesso à aplicação.

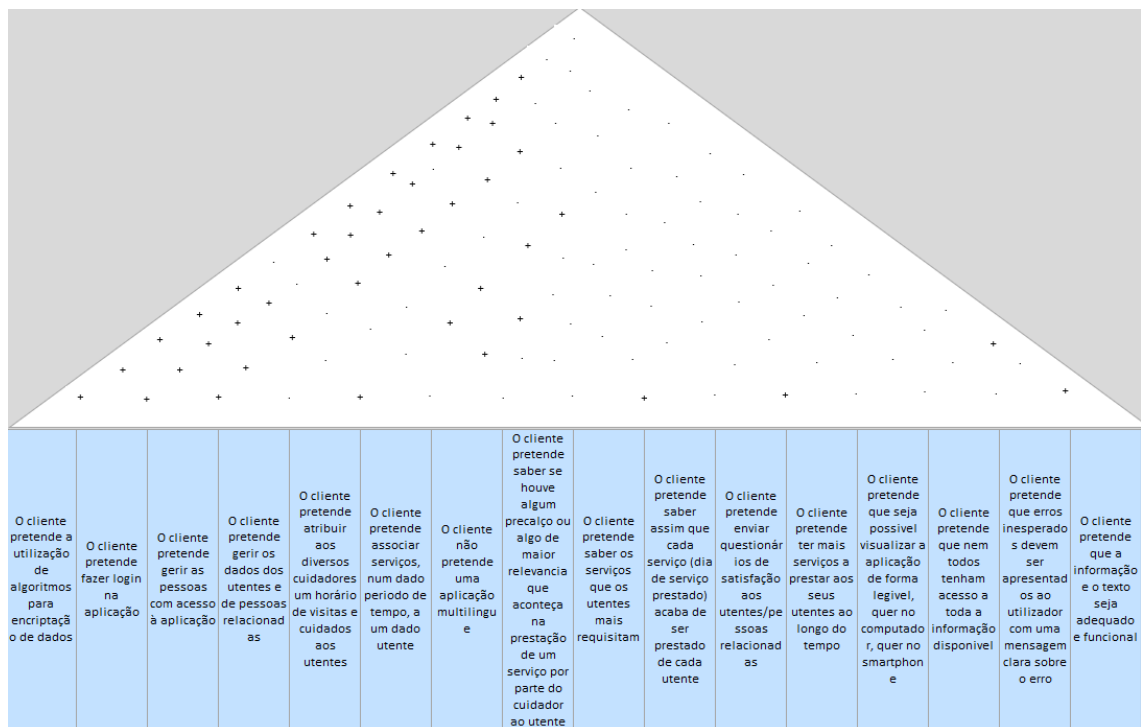


Figura 16 - Matriz Corelação dos requisitos

Cada um destes requisitos apresenta uma importância relativa de 10%, 10%, 12% e 9% respetivamente. A análise do relacionamento entre os requisitos técnicos, permite identificar as relações de compromisso entre eles e verificar quais serão as implicações quando algum atributo do produto for alterado.

Importância para o consumidor	Requisitos Técnicos	O cliente pretende a utilização de algoritmos para encriptação de dados	O cliente pretende fazer login na aplicação	O cliente pretende gerir as pessoas com acesso à aplicação	O cliente pretende gerir os dados dos utentes e de pessoas relacionadas	O cliente pretende atribuir aos diversos cuidadores um horário de visitas e cuidados aos utentes	O cliente pretende associar serviços, num dado período de tempo, a um dado utente	O cliente não pretende uma aplicação multilingue	O cliente pretende saber se houve algum precalço ou algo de maior relevância que aconteça na prestação de um serviço por parte do cuidador ao utente	O cliente pretende saber os serviços que os utentes mais requisitam	O cliente pretende saber assim que cada serviço (dia de serviço prestado) acaba de ser prestado de cada utente	O cliente pretende enviar questionários de satisfação aos utentes/pe	O cliente pretende ter mais serviços a prestar aos seus utentes ao longo do tempo	O cliente pretende que seja possível visualizar a aplicação de forma legível, quer no computador, quer no smartphone	O cliente pretende que nem todos tenham acesso a toda a informação disponível	O cliente pretende que erros inesperados devem ser apresentados ao utilizador com uma mensagem clara sobre o erro	O cliente pretende que a informação e o texto seja adequado e funcional	
	Requisitos do cliente																	
5	Protecção e segurança dos dados	9	9	3	9	3	3					9	9			9		
5	Autenticação	3	9	3	9	3	3					3				9		
3	Gestão de utilizadores	9	3	9											9			3
4	Gestão de utentes	9	3	9	9		9					3						3
4	Gestão e controlo de horários		3			9												9
3	Gestão de planos de cuidados		3				9											9
5	Sistema em português							9	3	3	3	3	3				9	9
3	Registo de ocorrências	3	3						9									9
3	Dashboard Estatístico									9								3
4	Notificações			3					3		9							
2	Realização de questionários	9	3	9								9						
3	Gestão de serviços		3				9						9					
4	Layout responsivo													9				9
5	Permissões de acessos	9	9	9								9	9			9		
2	Erros inesperados devem ser tratados										3			3	3	9		
3	Aplicação com informação e textos coesos				3	3	3			3	3	3	3	9		9	9	9
	Importancia	195	201	168	135	75	129	45	54	51	66	159	141	69	168	90	238	
	Porcentagem de importância	10%	10%	9%	7%	4%	7%	2%	3%	3%	3%	8%	7%	3%	9%	5%	12%	

Figura 17 - Requisitos técnicos vs cliente

3.5 Sumário

Ao longo deste capítulo foi abordado a análise de valor que este módulo tem para os clientes-alvo. Por fim, foi descrita a análise QEF, onde é possível verificar quais os requisitos com maior e menor importância neste trabalho.

Capítulo 4 - Análise e Design da solução

Neste capítulo, pretende-se identificar os *stakeholders* do projeto e os atores do sistema, contextualizar o desenvolvimento da solução, identificando os requisitos funcionais e não funcionais. Será também apresentada a arquitetura na qual será desenvolvida a solução e o modelo de domínio identificado.

4.1 Engenharia de requisitos

A engenharia de requisitos, de acordo com (Sommerville, 2011) , tem como principal objetivo definir o que o sistema deve fazer, quais as suas necessidades e identificar quais as restrições que existem para que o *software* seja desenvolvido. É neste processo que ocorre a comunicação com o cliente para a especificação dos requisitos. Quando essa comunicação não é bem-sucedida pode causar impacto no custo no prazo de entrega. A engenharia de requisitos age como uma ponte entre as necessidades reais dos clientes, e as capacidades oferecidas pelas tecnologias de *software*. É um processo que engloba fazer o levantamento de requisitos para um serviço ou produto, de forma a ser estudado e documentado. Este processo tem um papel fulcral ao verificar se os objetivos estão de acordo com as necessidades levantadas.

Com o desenvolvimento da aplicação *web* progressiva GesSAD, pretende-se auxiliar a gestão dos processos internos da empresa, informatizando-os, bem como facilitar a comunicação dos diversos intervenientes no processo, os cuidadores e a administração da empresa. Considerando o perfil de utilizadores que irão utilizar a aplicação *web* progressiva, esta foi criada para ser de fácil utilização e com uma interface intuitivo.

4.1.1 Stakeholders

Os *stakeholders*, são pessoas ou grupos que dependem de uma organização para alcançarem os seus objetivos e de quem, por consequência, a organização depende. Desta forma, no contexto do desenvolvimento deste projeto identificaram-se os seguintes *stakeholders*:

- **Instituições que prestam serviços de apoio domiciliário:** Instituição Particular de Solidariedade Social (IPSS), empresas do setor privado, hospitais, clínicas;
- **Cuidador:** Responsável pelos cuidados/acompanhamento dos utentes;
- **Utente:** Pessoa alvo de cuidados por parte dos cuidadores;
- **Coordenadores:** Gestores de organizações responsáveis pela gestão do trabalho dos cuidadores.

4.1.2 Atores do sistema

Os atores do sistema correspondem a pessoas, organizações que interagem com o sistema, fazendo uso das suas funcionalidades. Um ator do sistema poderá ser considerado um

stakeholder, uma vez que interage com o sistema, no entanto o oposto pode não se verificar, ou seja, um *stakeholder* pode não ser um ator do sistema.

A aplicação *web* progressiva integrará uma ferramenta de trabalho a ser utilizada nas visitas domiciliárias aos utentes, bem como na gestão do trabalho da administração da organização. Os atores do sistema corresponderão às pessoas que realizam as visitas domiciliárias aos utentes, sendo eles, os cuidadores, a administração da empresa e a assistente social que corresponderão às pessoas que realizam toda a gestão da empresa.

4.1.3 Requisitos não funcionais

Nesta secção são especificados todos os requisitos não funcionais, atributos e qualidades que condicionam o comportamento da aplicação. De modo a representar estes requisitos recorreu-se à utilização do modelo FURPS+, este é um modelo proposto por *Robert Grady and Hewlett-Packar (Khosravi e Guéhé- neuc 2004)* que possibilita identificar detalhadamente os requisitos não funcionais. Cada uma das letras deste acrónimo representam uma escala de análise, dividindo-se da seguinte forma: Funcionalidade, Usabilidade, Confiabilidade "*Reliability*", Performance e Suportabilidade. O "+" refere-se a restrições extra de desenho, implementação ou de interface. (Larman, 2001)

É necessário ter uma aplicação que seja o mais intuitiva possível, dada a elevada rotatividade de funcionários, para não ser necessário dar muita formação. É também necessário ter em atenção a proteção de dados para ir em conformidade com o que já é atualmente utilizado. Para este trabalho, os requisitos não funcionais são:

- **Usabilidade:** Pretende-se um *software* de uso simples com um design responsivo, de forma a que se adapte às diferentes dimensões dos ecrãs dos utilizadores, deverá também permitir o registo de forma rápida, evitando a introdução manual de textos;
- **Desempenho:** Deve consumir poucos recursos de modo a ser o mais eficiente possível e tornar mais fácil a escalabilidade do sistema no futuro;
- **Segurança:** O software deve garantir a segurança dos dados, bem como as permissões de acesso às suas funcionalidades. Como se tratam de dados sensíveis a aplicação deverá fazer cumprir o Regulamento Europeu para a Proteção de Dados (RGPD);

- **Responsividade:** A aplicação deverá apresentar um *design* adequado para cada dispositivo, podendo ser acedida por via *browser* num computador, num telemóvel ou *tablet*;
- **Multiplataforma:** O sistema deve poder ser acedido em mais do que uma plataforma, independentemente do seu sistema operativo;
- **Sistema em Português:** O sistema deve ser totalmente em português;
- **“Bring Your Own Device”:** O colaborador deve ter o seu próprio telemóvel ou tablet;
- **Custo:** Pretende-se um sistema que tenha um custo suportado pela empresa, neste momento só os servidores.

4.1.4 Requisitos funcionais

Os requisitos funcionais vão de acordo com as necessidades levantadas pelo cliente, indicando as funcionalidades que o sistema deverá apresentar. Como mencionado na secção 1.3, a documentação que é preenchida na empresa **Vida e Conforto** para gerir o negócio foi o principal meio de identificação dos requisitos. Nesse sentido, enumeram-se os requisitos funcionais identificados:

- Acesso ao sistema por parte dos gestores e dos cuidadores com acesso restrito a determinadas funcionalidades aos cuidadores;
- Gestão dos utentes e toda a informação a eles inerente (identificado pelo Anexo H - Ficha de Inscrição do Utente e Anexo A - Ficha de Contactos);
- Gestão dos serviços praticados pela empresa;
- Gestão dos planos de cuidados dos utentes:
 - Registo do controlo da medicação do utente pelo cuidador (identificado pelo Anexo B - Ficha de Controlo da Medicação);
 - Registo de ocorrências relevantes do utente pelo cuidador (identificado pelo Anexo D - Ficha de Registo de Ocorrências);
 - Registo dos cuidados/serviços efetuados ao utente (identificados pelos Anexo E - Ficha de Registo de Serviços (A), Anexo F - Ficha de Registo de Serviços (B) e Anexo G - Ficha de Registo de Serviços (C))
- Gestão de questionários e resultados (identificado pelo Anexo I - Questionário de qualidade);

4.2 Casos de uso

Os requisitos funcionais de um sistema vão de encontro às necessidades reportadas pelo cliente, referindo-se às funcionalidades que o sistema deverá apresentar. Recorreu-se a diagramas de casos de uso para a representação das ações realizadas pelos atores do sistema.

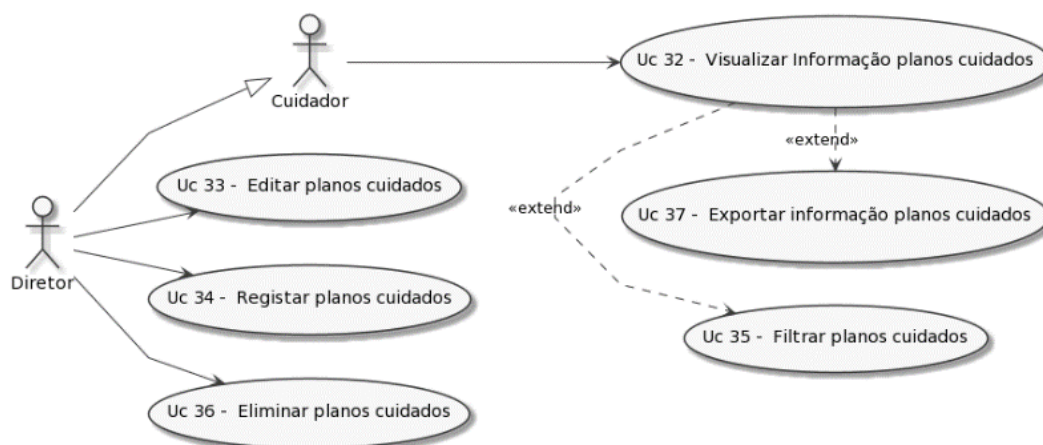


Figura 18 - Gestão de Planos de Cuidados

Da Figura 18 até a Figura 21 descrevem-se os casos de uso identificados para este trabalho, os restantes encontram-se disponíveis no Anexo J.

Gestão de planos de cuidados (UC 32 a UC 37):

- Requisitos
 - Autenticação efetuada
 - Plano criado previamente
- Descrição do Caso de Uso

Ao abrir a aplicação, o utilizador deverá inserir o *username* e *password* de modo a ser validada a sua autenticação. Se o utilizador autenticado apresentar o perfil de cuidador é carregada uma página com o horário de trabalho do cuidador para o dia com as tarefas que este terá de realizar num determinado utente. Ao selecionar as tarefas poderá verificar o plano de cuidados com detalhe, poderá exportar o mesmo para um PDF, e poderá filtrar consoante o utente que deseje verificar.

Caso o perfil seja de diretor, além destas opções, poderá editar e eliminar o plano, e poderá criar novos planos ajustados as novas situações.

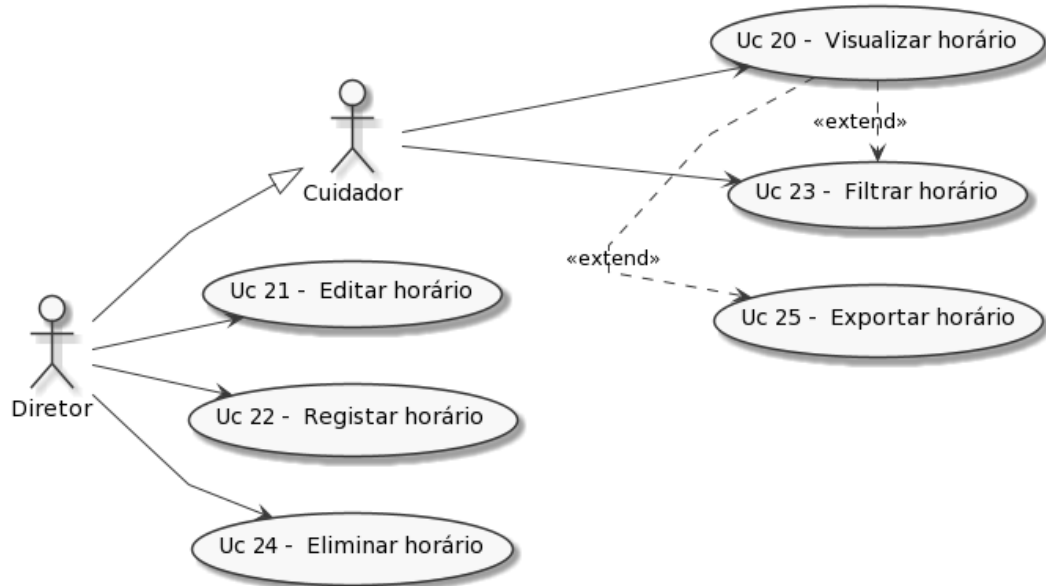


Figura 19 - Gestão horários

Gestão de horários (UC 20 a UC 25)

- Requisitos
 - Autenticação efetuada
 - Horário criado previamente
- Descrição do Caso de Uso

Ao abrir a aplicação, o utilizador deverá inserir o *username* e *password* de modo a ser validada a sua autenticação. Se o utilizador autenticado apresentar o perfil de cuidador é carregada uma página com o horário de trabalho do cuidador, no qual poderá visualizar, filtrar e exportar para PDF conforme necessário. Caso o perfil seja de diretor, além destas opções, poderá editar e eliminar o horário, e poderá criar novos horários ajustados às novas situações.

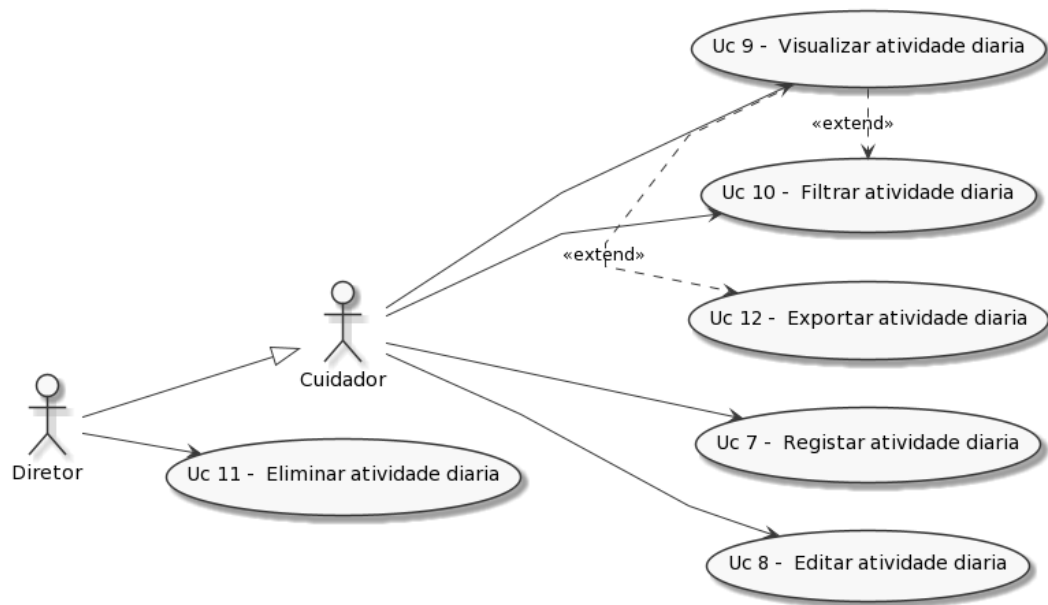


Figura 20 - Gestão de Atividade Diária

Gestão de Atividade Diárias (UC 7 a UC 12)

- Requisitos
 - Autenticação efetuada
- Descrição do Caso de Uso

Ao abrir a aplicação, o utilizador deverá inserir o *username* e *password* de modo a ser validada a sua autenticação. Se o utilizador autenticado apresentar o perfil de cuidador é carregada uma página para efetuar o seu registo de atividades nos utentes. Poderá verificar depois o seu registo, e editar o mesmo se for necessário, poderá exportar o mesmo para um PDF, e poderá filtrar consoante o utente que deseje verificar. Caso o perfil seja de diretor, além destas opções, poderá eliminar este registo.

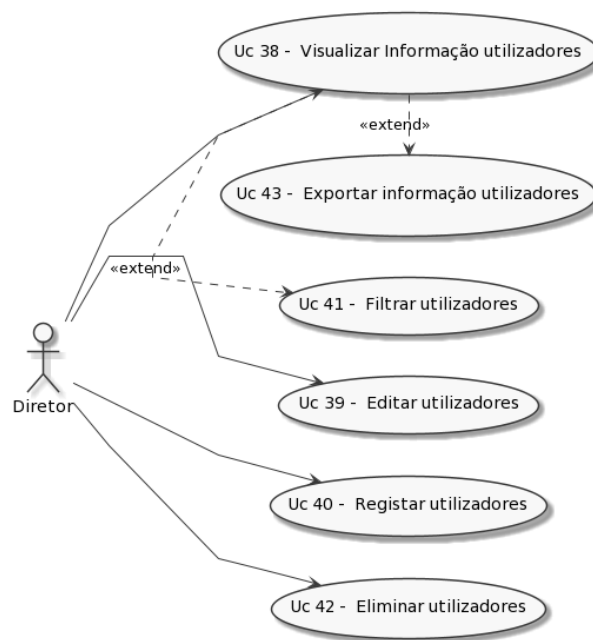


Figura 21 - Gestão de Utilizadores

Gestão de Utilizadores (UC 38 a UC 43)

- Requisitos
 - Autenticação efetuada
- Descrição do Caso de Uso

Ao abrir a aplicação, o utilizador deverá inserir o *username* e *password* de modo a ser validada a sua autenticação. Se o utilizador autenticado apresentar o perfil de cuidador não terá esta opção visível para si, no entanto, se tiver a usar um perfil de diretor, é carregada uma página para efetuar toda a gestão de utilizadores desta aplicação.

4.3 Modelo de domínio

O modelo de domínio é uma representação visual estruturada simples das entidades do mundo real para uma dada situação ou problema, permitindo identificar os principais conceitos, os seus atributos e as associações entre eles, que contêm interesse para o negócio (Larman, 2001).

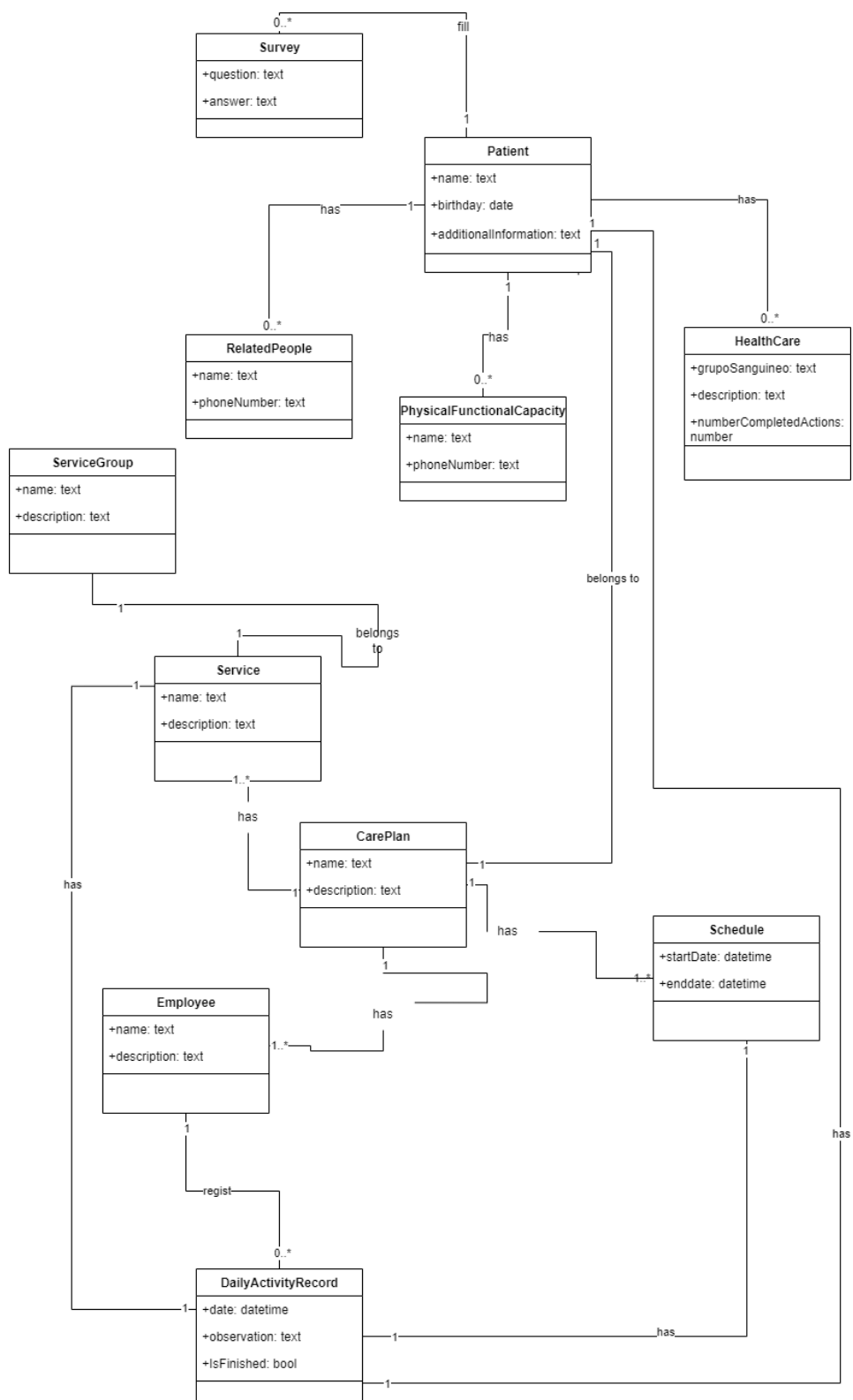


Figura 22 - Modelo de domínio

Na Figura 22 observa-se o modelo de domínio definido para o problema apresentado. A entidade fulcral é o *Patient*, sendo nela que se baseiam as restantes entidades de domínio

identificadas. O *Patient* representa o utente, que pode ter, ou não, pessoas associadas, caso sinta necessidade de ter. O utente (*Patient*) tem um conjunto de serviços associados (*Service*), representados por um plano de cuidados (*CarePlan*) e que são efetuados mediante um dado horário (*Schedule*) por um, ou vários cuidadores (*Employee*). O utente, pode necessitar de medicação, estando esse conceito representado pela entidade *MedicationReceipt*. Em relação aos cuidadores (*Employee*), estes devem fazer o registo de todo o serviço prestado ao utente. O controlo da medicação, em caso de existir, encontra-se representado pela entidade *MedicationControl*. O conceito *DailyServiceRecord* representa os serviços efetuados pelo cuidador, ao utente, naquele dia e hora. Qualquer ocorrência relevante deve ser registada e encontra-se identificada pelo conceito *DailyActivityRecord*. Por último o conceito de *Survey* representa os questionários que o utente (*Patient*) ou familiar ou pessoa responsável (*RelatedPeople*) preencha no final da prestação do serviço.

Importa salientar que o registo de utilizadores (gestores e cuidadores) não se encontra definido neste modelo de domínio por não ser considerado o foco principal do âmbito desta dissertação.

4.4 Design da solução

Nesta secção, apresenta-se a arquitetura de uma PWA, e uma possível vista de implementação da solução.

4.4.1 Arquitetura de uma PWA

A Figura 23, apresenta de forma simplificada o funcionamento de uma aplicação *web* normal e aquilo que foi adicionado e melhorado no conceito de progressiva. O componente Aplicação *Web*, que representa a interface gráfica, faz pedidos diretamente ao componente Servidor *Web*. Este, por sua vez, acede a uma base de dados e retorna a informação necessária novamente para o componente Aplicação *Web*.

No caso de uma PWA são introduzidos novos componentes, nomeadamente o *Service Worker* e a *Cache API*. O *Service Worker* funciona como um *proxy* entre o *browser* e o acesso à *internet* e a sua principal funcionalidade é armazenar dados em *cache* quando existe acesso à *internet* e disponibilizar esses dados quando o dispositivo do utilizador estiver *offline*. O *Service Worker* fornece uma das principais características das PWA que é o funcionamento *offline*. Desse modo, fica mais simples entender que a *Cache API* é um sistema que permite armazenar e recuperar

pedidos de rede e as suas respetivas respostas, permitindo fornecer respostas rápidas independentemente da disponibilidade da rede (LePage, 2020).

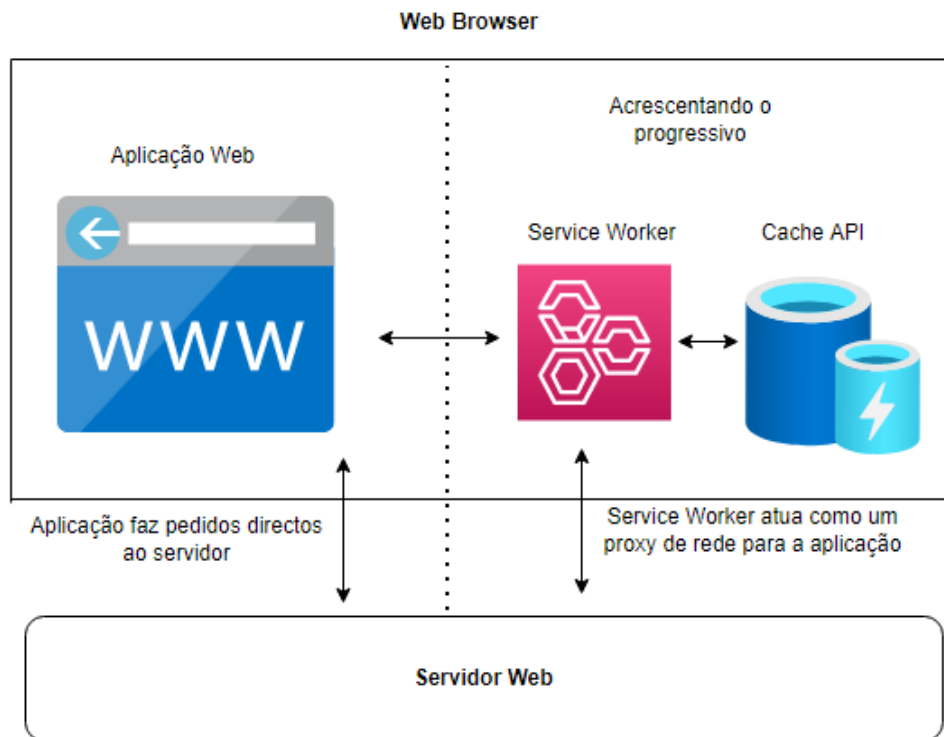


Figura 23 - Arquitetura de uma PWA

A Figura 24 aprofunda os conceitos da Figura 23 e permite explicar a arquitetura adotada em termos de desenvolvimento de aplicações *web*.

Quando se fala em arquiteturas para desenvolvimento de aplicações *web*, refere-se a uma arquitetura *client-side* ou *server-side*, ou uma solução híbrida. Neste caso, o uso de ASP.NET Core Blazor WebAssembly para desenvolvimento desta solução impõe desde logo uma abordagem *client-side*, por ser este o funcionamento base desta *framework* (Microsoft D. , 2022). Uma arquitetura *client-side* implica que seja feito um pedido ao servidor e que, a partir daí, tudo seja carregado. Esta abordagem torna a navegação mais fluída, mas o esforço no carregamento inicial da informação será mais longo do que uma arquitetura *server-side* (Ward, 2021).

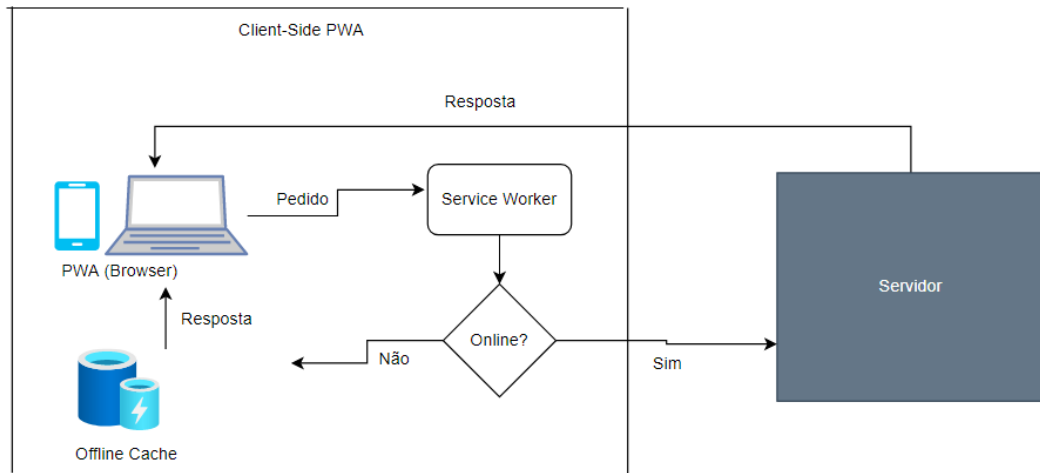


Figura 24 - Arquitetura Client-Side PWA

Tal como é possível observar na Figura 24 , a PWA, acessível através de um *browser* (ou já instalada) num dispositivo como tablet, computador ou smartphone, efetua um pedido. Esse pedido vai diretamente para o *Service Worker* que verifica a ligação à *internet*. Caso exista, faz um pedido ao Servidor para lhe devolver a informação requisitada. Caso contrário, acede à *Offline Cache*, para que esta devolva também a informação requisitada.

4.4.2 Escolha de Arquitetura da solução

De seguida apresentam-se duas soluções possíveis para a arquitetura do problema identificado nesta dissertação. Ambas as soluções serão apresentadas usando diagramas de componentes. No fim, após análise é apresentada a arquitetura selecionada.

4.4.2.1 Arquitetura Monolítica

Uma arquitetura monolítica é uma aplicação de *software* de uma só camada na qual integra a camada de interface do utilizador, lógica do negócio e o acesso aos dados. Estes são combinados num único programa, numa única plataforma (Nova, 2018). A principal razão para a utilização de uma aplicação monolítica é a simplicidade.

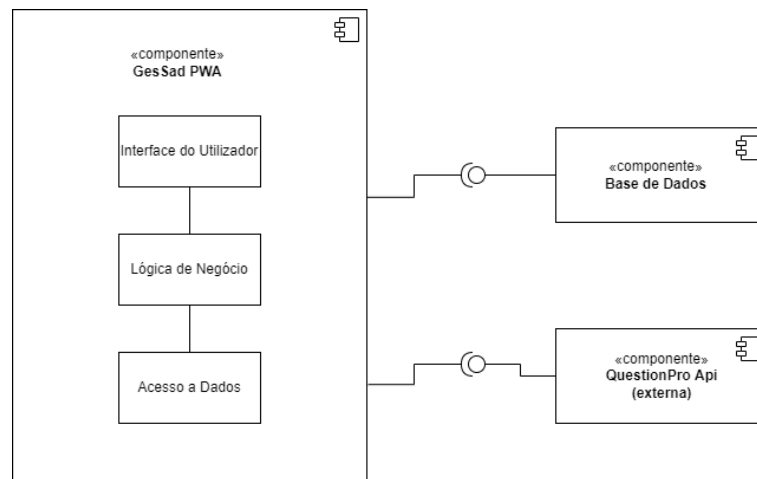


Figura 25 - Diagrama componentes arquitetura monolítica

Na Figura 25, é possível observar o diagrama de componentes para uma solução monolítica. Usando uma arquitetura destas apenas existiriam dois componentes. A PWA GesSAD seria uma única solução que integraria as três principais camadas: de Interface do Utilizador, Lógica de Negócio e Acesso à camada de dados. Por sua vez, este componente comunicaria com a base de dados, de modo a obter todos os dados necessários. Além disso, o componente GesSAD PWA comunica ainda com outro componente, neste caso externo, a QuestionPro API. A QuestionPro foi a API escolhida na secção 2.7.2 para gerir os questionários feitos aos utentes no final de cada serviço ou período de serviços prestados.

4.4.2.2 Arquitetura de Microserviços

Microserviços é um termo geralmente utilizado para descrever uma determinada forma arquitetónica de organizar uma aplicação ou software específico em serviços, de acordo com as suas funcionalidades, subdomínio ou capacidades (Lewis & Fowler, 2014).

Uma aplicação de microserviços divide-se em serviços de acordo com a sua funcionalidade, replicando-os quando necessário, ou seja, cada serviço pode ser executado independentemente de outros. Tem como principal vantagem a sua capacidade de manutenção, por exemplo, se uma funcionalidade falhar apenas essa funcionalidade específica torna-se indisponível, todas as restantes funcionalidades permanecem funcionais.

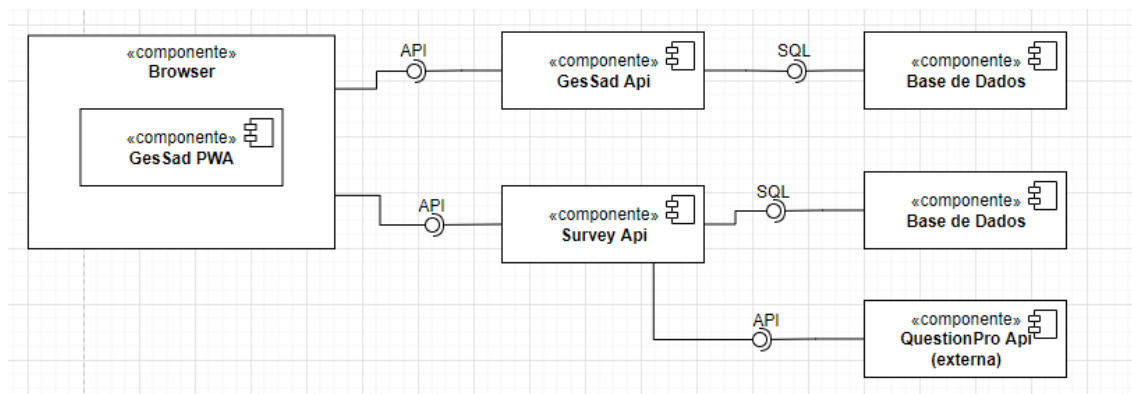


Figura 26 - Diagrama componentes arquitetura micro serviços

A arquitetura de microsserviços (Figura 26) já seria composta por diversos componentes diferentes. O componente GesSAD PWA representaria a aplicação *web* progressiva que, neste caso, consuma dados de dois outros componentes, a GesSAD API e a *Survey* API. Tal como o nome indica seriam duas APIs em que a primeira seria responsável por toda a gestão de utentes e planos de cuidados prestados. Por sua vez, o componente GesSAD API comunica com a respetiva base de dados, que alojaria todos os dados necessários ao componente.

Como a gestão de questionários é algo que não depende intrinsecamente do componente acima referido, optou-se por se colocar num componente à parte e ter a sua própria API. Além disso, tal como já foi referido, estes questionários podem futuramente abranger outras áreas dentro da empresa, daí também a importância de estarem separados do principal componente GesSAD API. Além disso, a *Survey* API comunica ainda com outro componente, neste caso externo, a *QuestionPro* API, que, por sua vez comunicará também com a sua respetiva base de dados.

4.4.2.3 Sistematização das Arquiteturas Alternativas

Implementar um sistema monolítico pode ser mais rápido em termos de desenvolvimento, no entanto, existem outros fatores que não o tornam uma escolha tão assertiva. Com a arquitetura orientada a microsserviços não existe um único ponto de falha, ou seja, se ocorrer algum erro na *Survey* API, toda a restante aplicação continua a funcionar. Por exemplo, os cuidadores conseguem continuar a preencher os dados sobre um serviço. Além disso, é possível atualizar as versões de cada um dos componentes sem ser necessário parar toda a aplicação. Por outro lado, tal como já foi referido, é possível na arquitetura orientada a microsserviços isolar a componente de questionários, que também é um ponto importante para a empresa.

Por outro lado, a empresa tem como objetivo refazer o *website* empresarial (Conforto, 2009) e consumir dados que estão na GesSAD API, nomeadamente a gestão de serviços. Futuramente e, não sendo âmbito desta tese, a GesSAD API deverá conseguir persistir informação de testemunhos dos seus clientes, dados esses que também serão apresentados no *website* da empresa. Nesse sentido, a arquitetura orientada a microsserviços é a adotada para esta solução.

4.4.3 Diagrama de Implantação da solução

É possível observar, na Figura 27, a vista de implantação para a solução adotada.

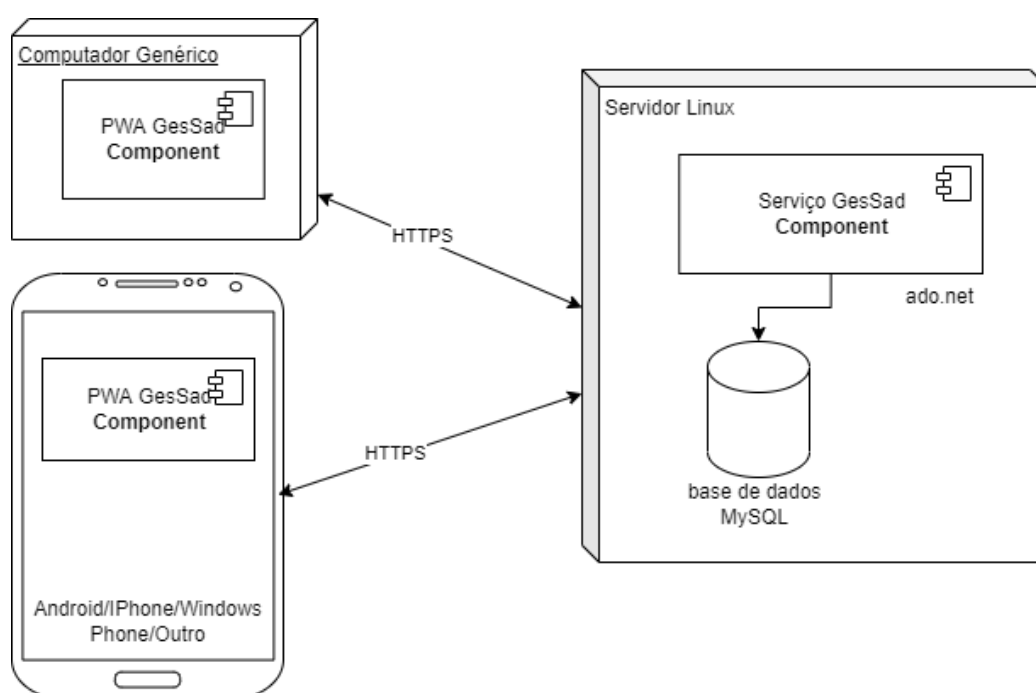


Figura 27 - Diagrama de implantação

O *backend* da solução (Serviço GesSAD), ou seja, os serviços que fornecem toda a informação necessária à PWA encontram-se instalados num servidor Linux e acede via *ado.net* a uma base de dados MySQL, instalada no mesmo servidor Linux.

A PWA alojada num servidor Linux, está acessível no *browser* de qualquer computador, genérico ou dispositivo móvel, independentemente do sistema operativo. Essa ligação é sempre feita via HTTPS, uma das características das PWA (Mozilla, 2021).

4.5 Modelo de Dados

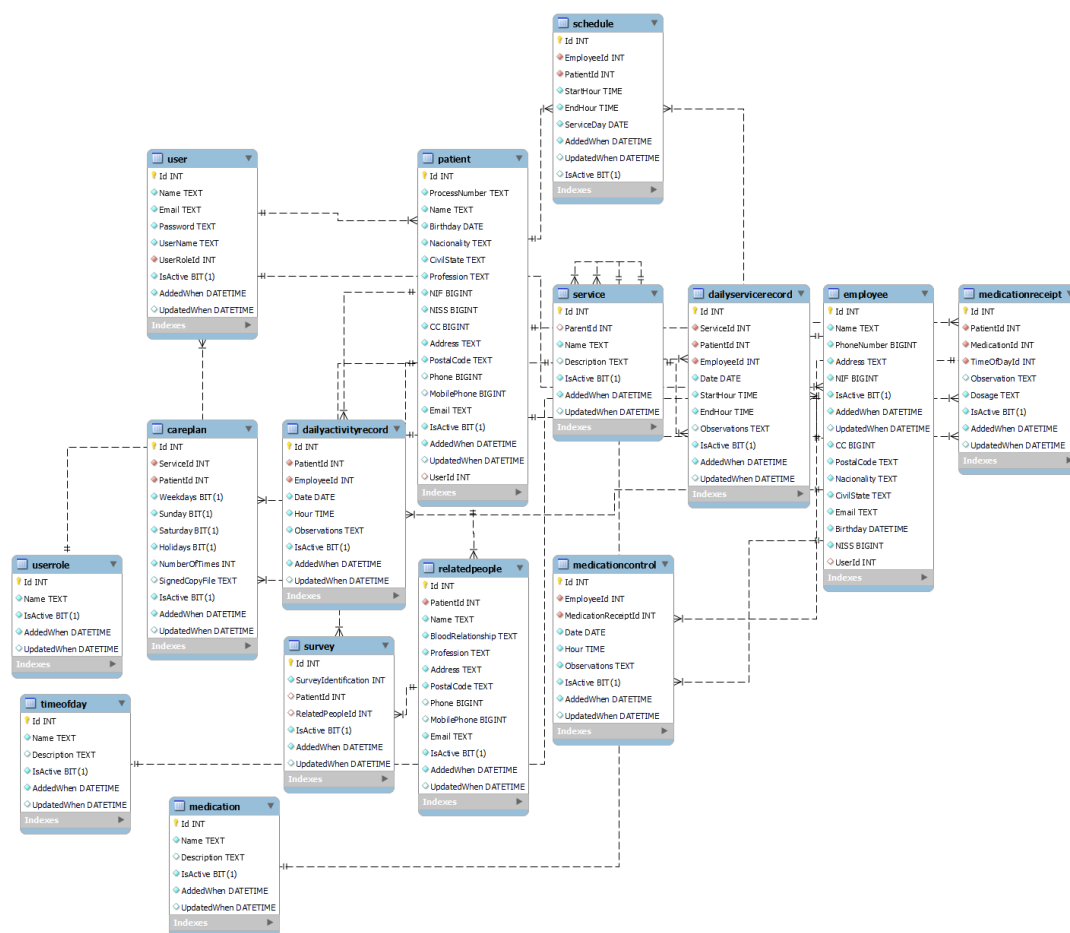


Figura 28 – Modelo de Dados

Na Figura 28 é possível observar o modelo de dados implementado. A gestão de utilizadores é feita com recurso às tabelas *user* e *userrole*, sendo que a *userrole* define o tipo de utilizador e, com isso, as permissões de acesso que tem no sistema a implementar. A tabela *employee* diz respeito aos dados dos cuidadores. Além disso, um *employee* também é considerado um utilizador da aplicação, daí ser observável a relação entre ambas as tabelas (*user/employee*) pela chave estrangeira *UserId* na tabela *employee*. As tabelas *patient* e *relatedpeople* são usadas para persistir informação de todos os pacientes, bem como das pessoas associadas a cada uma delas, mais concretamente familiares e/ou pessoas mais chegadas. De notar que a tabela *patient* também tem uma relação com a tabela *user*, pela chave estrangeira *UserId*. Neste caso, aplica-se a mesma lógica no caso dos cuidadores, mas o acesso por parte dos pacientes à aplicação não é âmbito desta dissertação. Ainda assim, e por ser algo a desenvolver futuramente, a relação já foi adicionada.

A tabela *careplan* permite definir o plano de cuidados para um determinado paciente, ou seja, que serviços vão ser contratados e em que dias da semana/fins de semana/feriados devem ser prestados. Este plano de cuidados pode ser complementado, no caso de haver necessidade, com recurso à tabela *medicationreceipt*, que é detalhada mais abaixo. Outra tabela que permite complementar os serviços prestados a um dado cliente é a tabela *schedule*. Mesmo sendo uma tabela mais administrativa, permite definir o escalonamento de horários entre pacientes e cuidadores.

Algumas das tabelas persistem configurações importantes para o sistema. É o caso da tabela *services*, usada para guardar os dados relativos aos serviços prestados pela empresa. De notar que a tabela tem, uma relação si própria, através da chave estrangeira *ParentId*, o que permite a flexibilidade de definir uma árvore de serviços, algo pretendido para o sistema. Outra tabela de configuração é a *medication*. Esta permite ter um catálogo de medicamentos disponível para se usar em duas principais operações no sistema: a primeira é na atribuição de medicação diária e necessária que um paciente deva tomar. Esse plano de medicação é definido desde início e é guardado na tabela *medicationreceipt*, onde indicamos o medicamento em questão (*medication*), a dosagem e a altura do dia (tabela *timeofday*) em que o mesmo deve ser consumido/administrado (manhã, noite, etc). A outra operação está relacionada o consumo/administração do medicamento em si, persistida na tabela *medicationcontrol*. Neste caso, o cuidador em questão deve sempre deixar registado que administrou o medicamento A ao paciente, num dado dia/hora.

As restantes tabelas, digam-se *dailyactivityrecord*, *dailyservicerecord* e a já mencionada, *medicationreceipt* permitem descrever todas as atividades, serviços e medicação administrada/consumida entre os pacientes e os cuidadores. No fundo funcionam como um diário, preenchido pelos cuidadores que descrevem as principais interações com o paciente em questão.

4.6 Sumário

Neste capítulo, foi descrito todo o processo de análise e modelação da engenharia de requisitos, onde foram identificados os requisitos que o sistema deverá ter. Perante isso foi desenhado o diagrama de casos de uso, o modelo de dados e foi feito o levantamento de conceitos importantes para este problema no diagrama de domínio. Foi ainda feito o design da solução, modelo de dados, bem como a escolha da arquitetura a ser usada. O trabalho será desenvolvido

na *framework* ASP.NET, na linguagem C# e com acesso à base de dados MySQL.

Capítulo 5 – Desenvolvimento da Solução

Neste capítulo, é descrito o trabalho elaborado durante todo o processo de desenvolvimento deste trabalho, e descritas todas as decisões.

5.1 Metodologia de Trabalho

O desenvolvimento do trabalho, teve como base a metodologia *Scrum*. Utilizar *Scrum* num projeto pressupõe a aplicação de ideias de trabalho com a utilização de casos de uso, reuniões diárias de acompanhamento numa equipa, trabalho dividido em sprints de uma ou duas semanas, reuniões de planeamento, reuniões de retrospectiva de sprints e desenvolvimento organizado.

No processo de desenho e desenvolvimento do projeto foram realizadas reuniões quinzenais com os envolvidos, nomeadamente o Engenheiro António José e colaboradores da **Vida e Conforto**. Estas reuniões tiveram como objetivo o planeamento das tarefas semanais, esclarecimentos e tomada de decisões a nível de design da aplicação e estratégias a seguir.

Na fase inicial do projeto houve necessidade de disponibilizar tempo para a pesquisa das tecnologias a adotar para o desenvolvimento da aplicação.

Como suporte ao desenvolvimento da aplicação *web* progressiva e repositório do código fonte recorreu-se ao *GitHub* e *Bitbucket*.

5.2 GesSAD Web Api

Nesta secção descreve-se, a implementação da *framework* Swagger. É apresentado um diagrama de classes com os planos de cuidados, as restantes classes apresentam uma estrutura semelhante. A autorização, autenticação e gestão de planos de cuidados, encontram-se explicados com maior detalhe, sendo apresentados alguns excertos de código mais relevantes, exemplificando o modo de implementação. As restantes funcionalidades encontram-se implementadas com a mesma estrutura.

5.2.1 Considerações iniciais

A GesSAD Web API foi desenvolvida em ASP.NET core 6.0, com recurso a Entity Framework (EF) *core* para interação com a base de dados MySQL. O EF é um *Object-Relational* mapping (ORM) que permite o acesso e manipulação da base de dados através de objetos .NET, abstraindo todas as operações típicas de acesso a uma base de dados (como por exemplo, executar uma instrução *select*, ou *update*) (Microsoft, 2022).

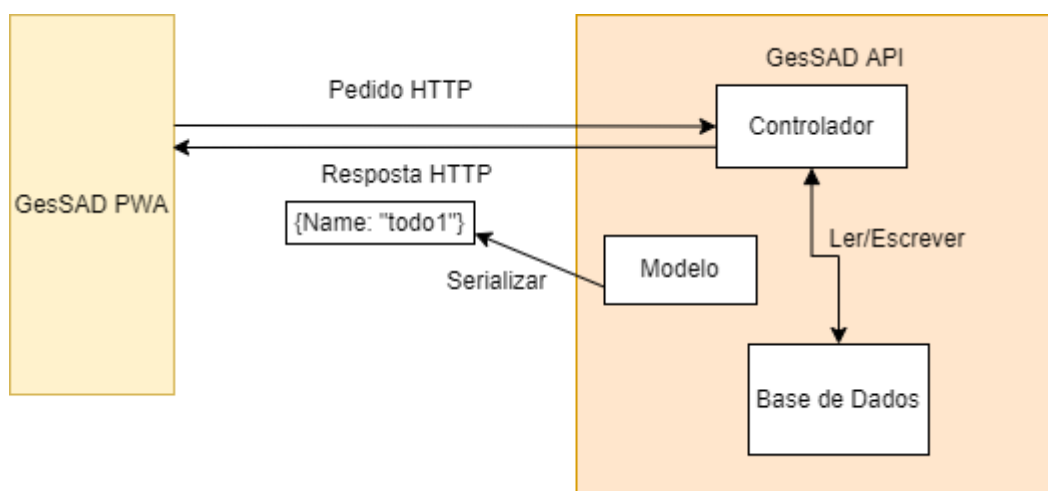


Figura 29 - Api Rest

A API *Representational State Transfer* (REST) desenvolvida segue o modelo referido na Figura 29. Expõe um conjunto de *endpoints*, que podem ser invocados via HTTP pelo cliente que a use. Feito o pedido, o mesmo chega ao respetivo Controlador que, por sua vez comunica com a camada de negócio (Serviço) e que, interage com a base de dados MySQL para aceder e/ou modificar dados por via dos modelos de entidade previamente definidos (Microsoft, 2022).

Encontra-se exposta e documentada através do *Swagger*. O *Swagger* é uma *framework* que auxilia na descrição, consumo e visualização de serviços de uma API REST. (Swagger, 2022) Na Figura 30, é possível observar um exemplo de como a GesSAD API se encontra exposta com recurso a esta *framework*.

A API considera exceções e erros que possam acontecer durante a sua execução, sendo os mesmos, bem como outras informações relevantes, gravados em ficheiros de log com recurso à biblioteca *Serilog*, (Serilog, 2022).

Para desenvolver a API foi usado o padrão de *Injeção de Dependência* (ID). Este padrão de desenvolvimento de *software* permite a criação de objetos dependentes fora de uma classe e fornece esses objetos para uma classe de diferentes formas.

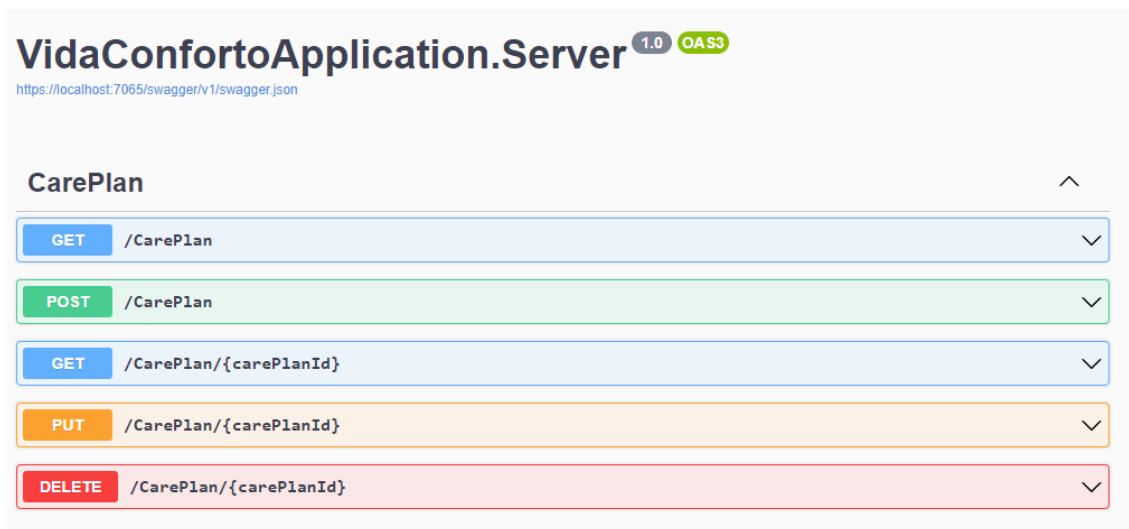


Figura 30 – Exemplo do Swagger da GesSAD Web API

Ao usar ID, move-se a criação e vinculação dos objetos dependentes para fora da classe que depende deles. (TutorialsTeacher, 2022) Neste caso, a injeção de dependências deste projeto é feita usando os recursos já existentes da *framework* .NET, neste caso o *IServiceProvider*, sendo que todas as dependências são registradas na classe *Program.cs*.

5.2.2 Diagrama de classes

Tal como é possível verificar e, dado o modelo de dados apresentado na Figura 28, a Figura 31 não representa o diagrama de classes total do microsserviço GesSAD Web Api.

O principal objetivo é explicar a estrutura genérica de classes que são usadas e, nesse sentido, optou-se por representar o diagrama para uma entidade, a *CarePlan*, sendo que para as restantes entidades é muito semelhante.

A classe *CarePlanController* representa todas as ações que estão disponíveis para o exterior, desde a listagem de todos os planos de cuidados, à listagem de detalhes de um específico, passando pela inserção, atualização e remoção de um plano de cuidados em particular. De uma forma mais simples, cada *Controller* é responsável por permitir executar as operações de *Create, Read, Update and Delete* (CRUD) para cada entidade deste projeto. Para cada uma destas ações em específico, a classe *CarePlanController* comunica com a classe *CarePlanService* (*ICarePlanService*) que é onde se encontra toda a lógica de negócio. Esta classe é responsável

for fazer a ponte entre a informação que chega do exterior, através de *DataTransferObjects*¹⁰ (DTOs), e as entidades de domínio.

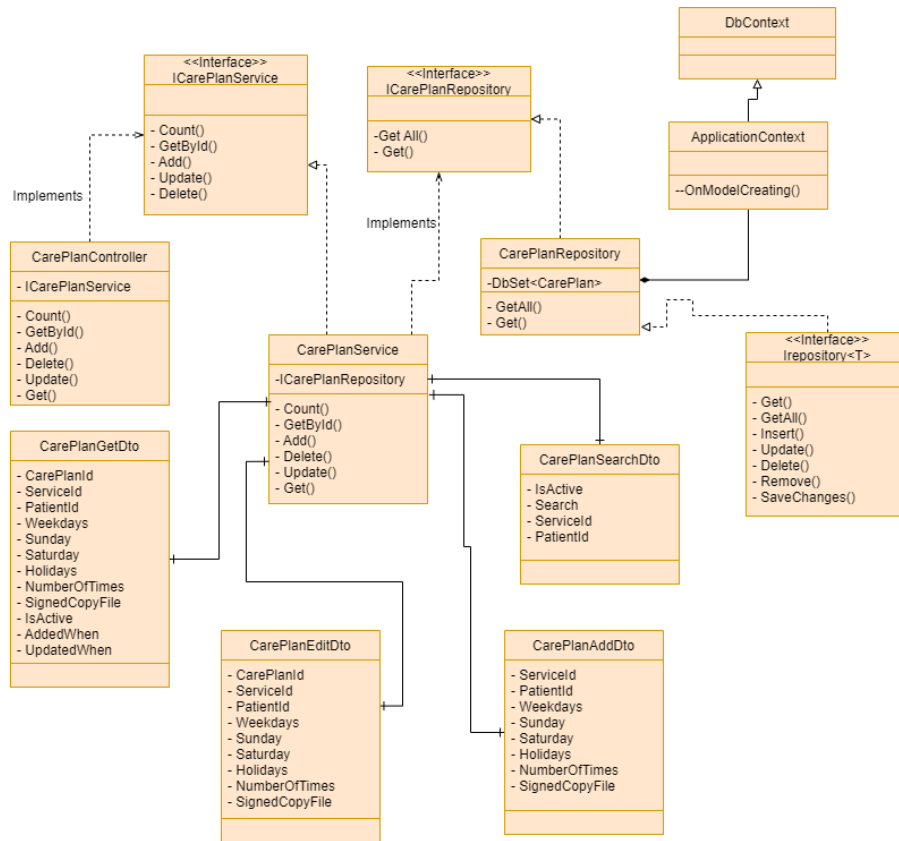


Figura 31 – Diagrama de Classes do Plano de Cuidados

Além disso, é também responsável por comunicar com a classe *CarePlanRepository* (*IRepository*) que, com ajuda da classe *ApplicationContext* comunica com a base de dados para efetuar as operações de CRUD para a entidade *CarePlan*.

De assinalar que a classe *ApplicationContext* é uma extensão da classe nativa do EF *core DbContext*. Esta classe tem algum relevo, pois é nela que definimos as relações entre as nossas classes de domínio, a definição das chaves primárias e estrangeiras em termos de base de dados, a definição das propriedades obrigatórias para cada entidade, entre outros pequenos detalhes. Como já referido, esta classe disponibiliza o acesso ou ponte, entre as nossas entidades domínio e as tabelas de base de dados, permitindo uma fácil interação e gestão dos dados, abstraindo toda a habitual lógica de gestão da base de dados e escrita de

¹⁰ Padrão usado para o transporte de dados entre os diferentes componentes de um sistema

código *Structured Query Language* (SQL) para efetuar as operações CRUD ou outras mais complexas que possam ser necessárias.

5.2.3 Autenticação e Autorização

De modo que a API não seja pública, ou seja, que não seja possível aceder à informação que fornece, fazendo um *GET /patients* (apenas um exemplo), implementou-se um mecanismo autenticação/autorização por *JSON Web Token* (JWT). Um JWT fornece uma forma segura de transmitir dados entre duas partes, neste caso, um cliente e um servidor.

```
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJodHRwOi8vc2NoZW1hcy54bWxzb2FwLm9yZy93cy8yMDA1LzA1L2lkZW50aXR5L2NsYWltcy9uYW1lIjoiYW1lbnRlcyIsImh0dHA6Ly9zY2hlbWVzLm1pY3Jvc29mdC5jb20vd3MvMjAwOC8wNi9pZGVudGl0eS9jbGFpbXMvcm9sZSI6IkhYXWx0aFRlY2huaWNpYW4iLCJleHAiOiJlY2NjMwMDc5NzMsImVzcyI6IkhvZGVNYXplQVBJIiwiaXVkljoiaHR0cHM6Ly9sb2NhbGhvc3Q6NzA2NSJ9.C8F9b0OV3FztAN8Z-Zdv6_G_W8IULBzqW3eQVixDDfQ
```

Figura 32 - Exemplo de JWT gerado

No excerto de código visível na Figura 32 é possível observar um exemplo de um JWT, que se divide em três distintas partes. A primeira parte, o Cabeçalho (assinalado a azul), contém um objeto *JSON* codificado em base64¹¹, onde é identificado o *token* (JWT) e o algoritmo usado (HS256). A segunda parte, o *Payload* (assinalado a verde), contém um objeto *JavaScript* também codificado em base64 que inclui alguma informação relacionada com o utilizador autenticado. Nesta implementação específica contém o *Username* do utilizador autenticado e a posição (*Role*) que lhe está atribuída. Como os JWT não são encriptados, e podem facilmente ser descobertos aplicando um decodificador de base64, não deve ser usada informação sensível, como o Id de base de dados ou a *Password*. A última parte, a Assinatura (assinalado a laranja), que é usada pela API para validar se o JWT recebido é válida. Este mecanismo de segurança é usado, caso tentem manipular o JWT a enviar para a API, aí têm de tentar descobrir a chave-secreta que está internamente definida.

Na Figura 33 , é possível observar o mecanismo de funcionamento de autenticação/autorização por JWT entre o cliente (GesSAD PWA), e o servidor (a GesSAD API). De forma bastante sucinta, ao invocar o *endpoint user/authenticate* (1), o servidor devolve o *token* de acesso (2), que é

¹¹ Base64 é um método para codificação de dados para transferência na Internet.

gravado do lado do cliente para ser enviado em todos os restantes pedidos que sejam feitos à API. Antes de ser efetuado um pedido à API Servidor, é pedido o *token* de acesso previamente guardado do lado do Cliente (3), sendo devolvido e usado para efetuar os pedidos (4). Este processo acontece até que o *token* expire (5). Nesse momento é necessário repetir o processo (1) e (2), e assim sucessivamente.

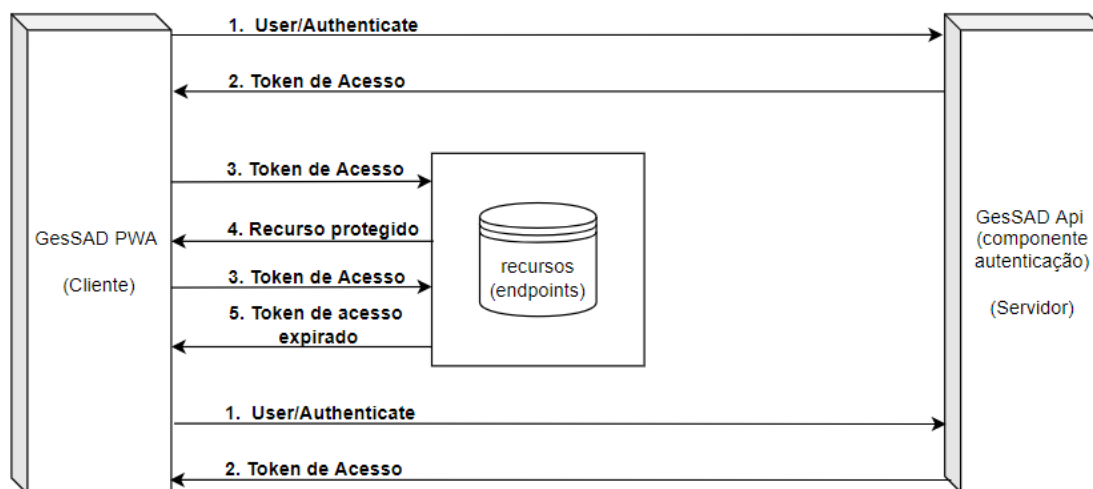


Figura 33 - Mecanismo de autenticação/autorização

Apesar de não estar detalhado na imagem, nem ser detalhado em pormenor a sua implementação, este mecanismo de autenticação por *token* é mais complexo, pois o *token* tem de expirar em algum momento. Se o *token* por utilizador fosse sempre o mesmo, acabaria por ser decifrado. Para tal, além do *token* de acesso usado para aceder a cada um dos *endpoints*, é também usado um *token* de atualização. Quando o *token* de acesso expirar, o servidor enviará uma resposta de *token* expirado. O cliente, ao receber a resposta, envia o *token* de acesso expirado e o *token* de atualização para obter um novo *token* de acesso e um novo *token* de atualização.

O excerto de código apresentado na Figura 34 apresenta o método `AuthenticateAsync(UserAuthenticateDto userAuthenticateDto)` da classe `UserService`, classe essa responsável pela gestão de utilizadores (listar, apagar, criar/registar, editar) e pela autenticação.

```

public async Task<UserGetDto?> AuthenticateAsync(UserAuthenticatedDto
userAuthenticatedDto)
{
    var user = _userRepository.GetByUsername(userAuthenticatedDto.UserName);
    if (user == null) { return null; }

    string hashedPassword =
HasherPassword.Encrypt(userAuthenticatedDto.Password);

    if (hashedPassword == user.Password)
    {
        var signingCredentials = _tokenService.GetSigningCredentials();
        var claims = await _tokenService.GetClaims(user);
        var tokenOptions =
_tokenService.GenerateTokenOptions(signingCredentials, claims);
        var token = new JwtSecurityTokenHandler().WriteToken(tokenOptions);

        user.RefreshToken = _tokenService.GenerateRefreshToken();
        user.RefreshTokenExpiryTime = DateTime.UtcNow.AddDays(7);
        _userRepository.Update(user);

        return new UserGetDto
        {
            Id = user.Id,
            AddedWhen = user.AddedWhen,
            Name = user.Name,
            Username = user.UserName,
            UpdatedWhen = user.UpdatedWhen,
            Token = token,
            RefreshToken = user.RefreshToken
        };
    }
    return null;
}

```

Figura 34 - Excerto código autenticação

Como é possível observar, a autenticação é feita através do *Username* e da *Password* do utilizador. Neste caso, assume-se que o *Username* deve ser único, sendo essa validação feita aquando do registo/criação do mesmo. Em relação à *Password*, a mesma é encriptada pelo algoritmo do tipo Advanced Encryption Standar (AES) e gravada na base de dados já encriptada. Caso o utilizador seja válido e a *Password* coincida, é então gerado o *token* de acesso (*Token*) que é enviado na resposta ao pedido de autenticação, bem como o *token* de atualização (*RefreshToken*), essencial quando o *token* de acesso expirar. A propriedade *RefreshTokenExpiryTime* guarda a validade do *Token*, neste caso, durante sete dias. De notar que mesmo na resposta completa ao pedido de autenticação não são enviados dados sensíveis.

No que concerne à Autorização, esta só é aplicada na *API* caso os controladores usem o atributo *[Authorize]*. Se forem efetuados pedidos sem o envio do *token* a *endpoints* que não usem o *[Authorize]* é retornado o erro “HTTP 401 Não Autorizado”.

As posições (*Roles*) definidas são bastante simples. Um utilizador apenas tem uma posição atribuída. Pode ser administrador, técnico de saúde ou paciente (sendo que esta ainda não tem

nenhuma aplicação prática). Como foi possível verificar, a posição é enviada no *token* de acesso, sendo posteriormente usada no Cliente (GesSAD PWA). Na API, optou-se por não se aplicar regras rígidas de acesso aos *endpoints*, ou seja, não aplicar o atributo *[Authorize]* por uma posição em específico. Este nível de segurança será trabalho futuro a desenvolver.

5.2.4 Funcionalidades

Tal como referido na secção 5.2.2 não existe a necessidade de especificar em detalhe tudo o que foi desenvolvido. A API é extensa, mas as funcionalidades implementadas são semelhantes entre as várias entidades. Seguindo a abordagem do diagrama de classes é apresentado de seguida, em detalhe, com recurso a diagramas de sequência, as funcionalidades de criar um plano de cuidados e de listar um plano de cuidados.

5.2.4.1 Criar plano de cuidados

Na Figura 35, é possível observar o diagrama de sequência para a criação de um plano de cuidados. A operação inicia-se com um *POST* onde são passados os dados relativos ao plano de cuidados a inserir. O método *Add* da classe *CarePlanController* recebe um *CarePlanAddDto* que é validado antes de ser enviado para a classe *CarePlanService*. No caso de ser nulo, é enviado um *BadRequest* para o utilizador, indicando que o objeto de entrada não pode ser nulo. Outras validações internas são feitas com recurso às potencialidades nativas da *framework* .NET. Ou seja, a classe *DataAnnotation* que nos permitem validar os dados de forma mais eficaz.

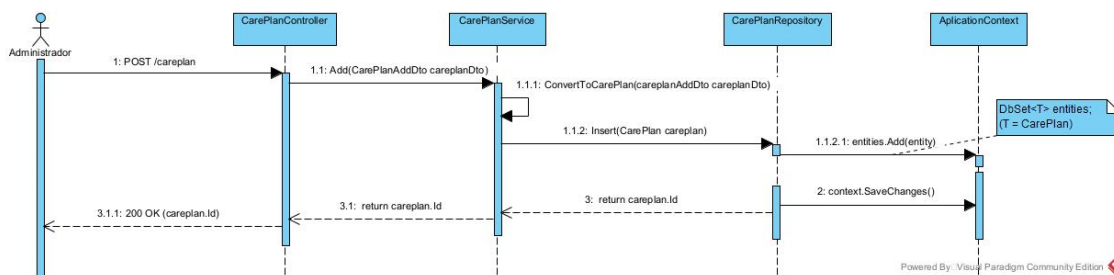


Figura 35 – Diagrama de Sequência para criar Plano de Cuidados

O excerto de código da Figura 36 mostra a utilização do atributo *Required* para indicar que aquelas duas propriedades são obrigatórias, poupando assim o programador de fazer a validação manualmente. Caso as propriedades *PatientId* e *ServiceId* não venham preenchidas um *BadRequest* é enviado de imediato.

```

public class CarePlanAddDto
{
    [Required]
    public long? PatientId { get;
set; }
    [Required]
    public long? ServiceId { get;
set; }
    (...)
}

```

Figura 36 – Excerto da classe CareplanAddDto

Esta informação pode ser complementada com recurso ao excerto de código da Figura 37 , onde é possível observar o método *Add()* da classe *CarePlanController*. Além do que foi referido, se ocorrer algum erro durante o processo de adição de um novo plano de cuidados, será intercetado nesta fase, será gravado em ficheiro de *logs* e será enviado *IntervalServerError* para o utilizador, informando-o que ocorreu um erro. Não é objetivo informá-lo do detalhe do erro, para não expor dados e informação que o mesmo não deva ter acesso. Esses detalhes são gravados em *log*, acessíveis apenas pela equipa que gere a API.

Ao ler o diagrama de sequência, após validação do *CarePlanAddDto()* com sucesso, este é enviado para a camada de negócio, a *CarePlanService*. O método *Add()* da classe *CarePlanService* é responsável por converter o *Dto* na entidade de domínio, isto é, de *CarePlanAddDto()* para *CarePlan()* e, de seguida, invocar a classe *CarePlanRepository*, nomeadamente o método *Insert()* que trata da sua inserção na base de dados.

```

[HttpPost()]
public ActionResult<int> Add(CarePlanAddDto careplanDto)
{
    try
    {
        if (careplanDto == null)
            return BadRequest($"{nameof(careplanDto)} cannot be null");
        return Ok(_careplanService.Add(careplanDto));
    }
    catch (Exception ex)
    {
        var errorMessage = "An error occurred while trying to add a new careplan";
        _logger.LogError($"{errorMessage} {ex.StackTrace}");
        return StatusCode(StatusCodes.Status500InternalServerError, errorMessage);
    }
}

```

Figura 37 - Exemplo método Add (POST) de um CarePlan

O *CarePlanRepository* invoca assim a classe *ApplicationContext* que detém a capacidade de adicionar o novo *CarePlan* à lista de *CarePlans* já existentes (*DbSet<CarePlan>*) e de efetivar a operação na base de dados, ao invocar o método *SaveChanges()*. Após ser invocado este método o objeto *CarePlan*, que é passado, fica automaticamente com o *Id* preenchido (propriedade identificadora única deste registo no modelo de dados e chave primária na tabela

de base de dados *CarePlan*). Por fim, o *Id* do novo plano de cuidados é enviado, chegando à classe *CarePlanController*, sendo enviado ao utilizador como OK, indicativo de que a operação foi concluída com sucesso.

5.2.4.2 Listar planos de cuidados

Na Figura 38 é possível observar o diagrama de sequência para a listagem de planos de cuidados. A operação inicia-se com um GET onde são dados passados os dados relativos à listagem, por *query string*.

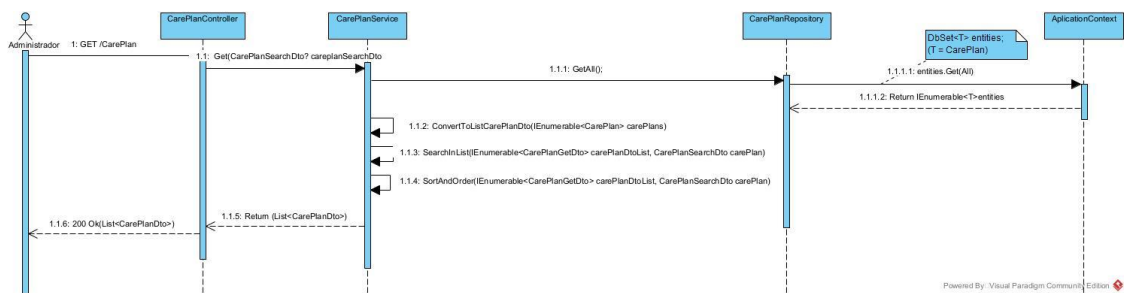


Figura 38 – Diagrama de sequência para listar Plano de Cuidados

Esta listagem inclui filtragem, ordenação e paginação, pelo que a classe *CarePlanSearchDto* pode ser observada na Figura 39 .

<pre> public class CarePlanSearchDto : PaginationSortingDto { public bool? IsActive { get; set; } = null; public string? Search { get; set; } = string.Empty; public long? PatientId { get; set; } public long? ServiceId { get; set; } } </pre>	<pre> public class PaginationSortingDto { public int? CurrentPage { get; set; } public int? PageSize { get; set; } public string SortBy { get; set; } = "Id"; public bool Ascending { get; set; } = false; } </pre>
--	---

Figura 39 – Classes para filtragem, ordenação e paginação do Plano de Cuidados

A validação de dados nesta operação é quase inexistente, dado que praticamente nenhum campo é obrigatório. Ainda assim, caso o objeto *CarePlanSearchDto* seja nulo é lançado um *BadRequest* indicativo para o utilizador, indicando que o objeto de entrada não pode ser nulo. Tal como na conceção de um plano de cuidados, é usada a mesma lógica para tratamento de exceções.

Após validação dos dados, o método *Get* da classe *CarePlanService* é invocado. A paginação, ordenação e filtragem não é feita na base de dados, daí se conseguir observar que o método

GetAll() da classe *CarePlanRepository* é invocado primeiro. Seguidamente, a lista de *CarePlan* recebida é convertida numa lista de *CarePlanDto*, apenas com as propriedades que ficarão visíveis para o exterior. Após convertida a lista é efetuado a pesquisa, sendo para isso invocado o método *SearchInList*. Se não forem aplicados filtros são devolvidos todos os dados. Após a pesquisa é aplicada a paginação e a ordenação, ao chamar o método *OrderAndPaging*, observável no excerto de código da Figura 40.

```
var sortClause = carePlanSearchDto.Ascending ? $"{carePlanSearchDto.SortBy}
Ascending" : $"{carePlanSearchDto.SortBy} Descending";
    if (carePlanSearchDto.CurrentPage == null &&
carePlanSearchDto.PageSize == null)
        return
carePlanDtoList.AsQueryable().OrderBy(sortClause).ToList();

    return
carePlanDtoList.AsQueryable().OrderBy(sortClause).Skip((carePlanSearchDto.Cur
rentPage.Value - 1) *
carePlanSearchDto.PageSize.Value).Take(carePlanSearchDto.PageSize.Value).ToLi
st();
```

Figura 40 – Excerto de código de ordenação/filtragem de Planos de Cuidados

Para fazer a ordenação de forma dinâmica é usada uma extensão do Linq¹² chamada *System.Linq.Dynamic.Core* (Microsoft, 2022). Terminada esta ação, a lista final de *CarePlanDto* é enviada até à classe *CarePlanController*, sendo devolvido ao utilizador um OK com a respetiva lista, indicativo de que a operação foi concluída com sucesso.

5.3 GesSAD PWA

Nesta secção é apresentado um diagrama de classes para os planos de cuidados, as restantes classes apresentam uma estrutura semelhante. É apresentado o modo de instalação da PWA no computador. A autorização, autenticação e gestão de planos de cuidados, encontram-se explicados mais detalhadamente, sendo apresentados excertos de código mais relevantes, exemplificando o modo de implementação. As restantes funcionalidades estão implementadas com a mesma estrutura. É referido também o modo offline da PWA, em que se baseia e como funciona.

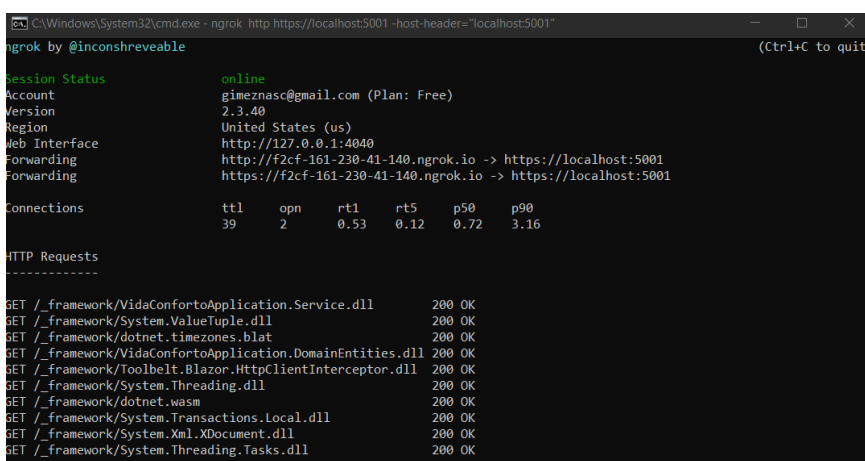
¹² LINQ é um componente do Microsoft .NET que adiciona funcionalidades de consulta em algumas linguagens de programação .NET.

5.3.1 Considerações iniciais

A GesSAD PWA foi desenvolvida em Blazor WebAssembly, para NET.Core 6.0. O Blazor, que advém da junção das palavras Browser + Blazor, é uma *framework* Web baseada em CSS e HTML que usa a linguagem C# e Razor e que é executado via WebAssembly. O WebAssembly (ou Wasm) (WebAssembly, 2022) é um novo web standard, portátil, leve e com tempo de carregamento eficiente, adequado para compilação na Web (Microsoft, 2022).

Tal como a GesSAD Web API, a PWA considera possíveis exceções e erros que possam acontecer durante a sua execução. Estas e outras informações relevantes são gravados em ficheiros de *log* com recurso à biblioteca Serilog (Serilog, 2022). Foi igualmente desenvolvida usando o padrão de Injeção de Dependência (ID) que, como na GesSAD Web API, todas as dependências são registadas na classe *Program.cs*, usando a classe *IServidorProvider*, recurso nativo da *framework* .NET.

Como não havia disponibilidade de grande número de *smartphones* para testar a aplicação em ambiente *mobile*, foi necessário simular a publicação da aplicação, disponibilizando um URL que conseguisse ser acessível através dos simuladores de Android do Visual Studio. Tal como foi referido na secção 2.3.4, seria dado prioridade aos utilizadores de Android, já que os funcionários têm um *smartphone* com este sistema operativo. Nesse sentido, foi usada a ferramenta *Ngrok* (Ngrok, 2022).



```
C:\Windows\System32\cmd.exe - ngrok http https://localhost:5001 -host-header="localhost:5001"
ngrok by @inconshreveable (Ctrl+C to quit)

Session Status      online
Account             gimeznasc@gmail.com (Plan: Free)
Version             2.3.40
Region              United States (us)
Web Interface       http://127.0.0.1:4040
Forwarding           http://f2cf-161-230-41-140.ngrok.io -> https://localhost:5001
Forwarding           https://f2cf-161-230-41-140.ngrok.io -> https://localhost:5001

Connections        ttl    opn    rt1    rt5    p50    p90
                   39     2      0.53  0.12  0.72   3.16

HTTP Requests
-----
GET /_framework/VidaConfortoApplication.Service.dll 200 OK
GET /_framework/System.ValueTuple.dll 200 OK
GET /_framework/dotnet.timezones.blat 200 OK
GET /_framework/VidaConfortoApplication.DomainEntities.dll 200 OK
GET /_framework/Toolbelt.Blazor.HttpClientInterceptor.dll 200 OK
GET /_framework/System.Threading.dll 200 OK
GET /_framework/dotnet.wasm 200 OK
GET /_framework/System.Transactions.Local.dll 200 OK
GET /_framework/System.Xml.XDocument.dll 200 OK
GET /_framework/System.Threading.Tasks.dll 200 OK
```

Figura 41 – Ngrok

O *Ngrok* é uma ferramenta CLI (Comand Line Interface) que permite criar um túnel seguro, que expõem serviços locais para a Internet, de forma fácil e segura, ou seja, expõe serviços que

estão a ser executados localmente (no computador) para a Internet (Ngrok, 2022). Após a publicação da aplicação e, através da execução do comando: `ngrok http https://localhost:5001 -host-header="localhost:5001"`, um URL público é disponibilizado, tal como é possível observar na Figura 41. Esse URL pode assim ser usado nos simuladores de Android.

5.3.2 Diagrama de classes

Seguindo a mesma abordagem aplicada ao diagrama de classes da GesSAD Web API (secção 5.2.2), é apresentada na Figura 42 o diagrama de classes para a gestão dos planos de cuidados, aplicado à GesSAD PWA. Neste diagrama encontra-se excluída a gestão de autenticação/autorização que é comum a todas as funcionalidades.

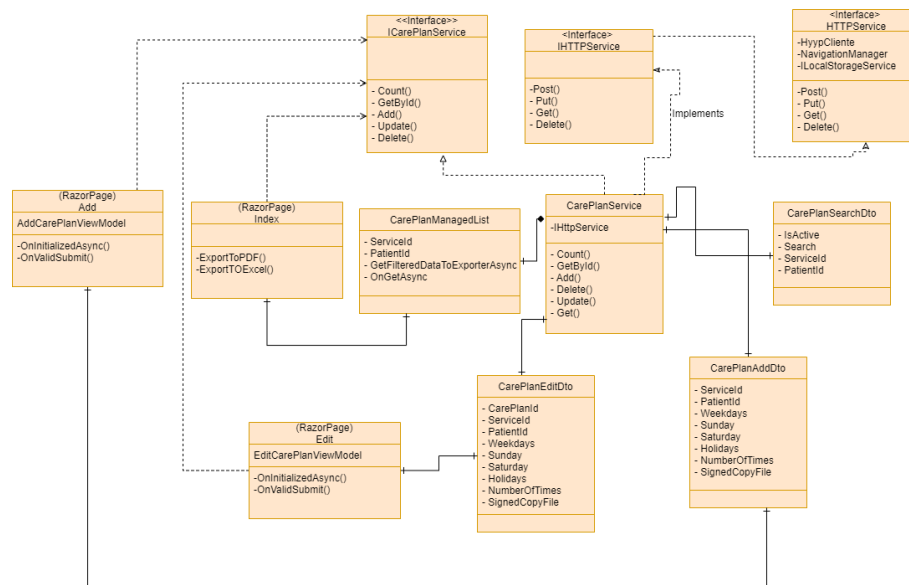


Figura 42 - Diagrama Classes

Sendo a GesSAD PWA um microsserviço e, tendo de funcionar de forma independente da GesSAD Web API, os Dtos necessitam ser “duplicados”, existindo também as classes *CarePlanGetDto*, *CarePlanAddDto* e *CarePlanEditDto*.

É necessária a existência de uma camada de acesso de API, isto é, um local onde sejam efetuados os pedidos HTTP para o CRUD da gestão de planos de cuidados. A classe *CarePlanService* (interface *ICarePlanService*) detém assim todos os métodos que permitem o acesso às funções CRUD. Esta classe comunica com a classe *HttpService* que é uma classe genérica que permite invocar métodos *HTTP Post()*, *Put()*, *Get()* e *Delete()*.

As classes da camada de apresentação, Add (razor), Edit (razor), Index (razor) e View (razor) permitem, respetivamente, adicionar, editar, listar (pesquisar/apagar/exportar dados) e visualizar os planos de cuidados. Estas classes interagem com o mecanismo de Autorização, que não se encontra aqui representado, e com a camada de acesso à API (*CarePlanService*). Existe, no entanto, uma abordagem diferente para a classe Index (razor). Como esta classe é responsável por inúmeras tarefas como listar (com paginação e ordenação), pesquisar, apagar e exportar planos de cuidados, a classe *CarePlanManagedList* atua como um gestor de todas estas funcionalidades.

5.3.3 Instalação

Tal como foi referido na secção 2.3.1 uma das características de uma PWA é poder ser instalada. Ao aceder ao endereço da GesSAD PWA de um browser, é possível escolher a opção instalar, como se observa na Figura 43. Após instalação concluída, consegue-se aceder como se de uma aplicação se tratasse, sem nunca mais haver necessidade de abrir o browser e escolher o endereço. A Figura 44 demonstra a aplicação já aberta, demonstrando as opções que existem de afixar na barra de tarefas ou criar um atalho no ambiente de trabalho, entre outras.

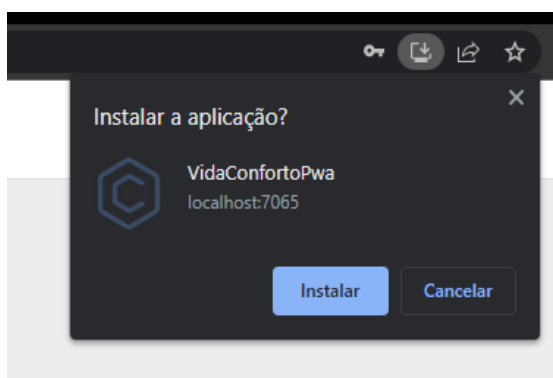


Figura 43 - Instalar

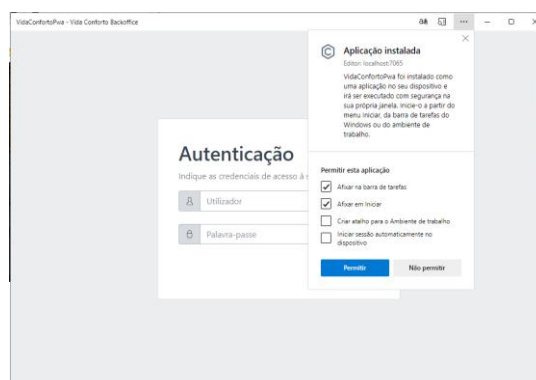


Figura 44 - Aplicação instalada

Esta informação de instalação é configurada no ficheiro *manifest.json*, visível no excerto de código da Figura 45. Nele é definido o nome da aplicação e o ícone, que poderão ser observados na Figura 43 e Figura 44.

```

{
  "name": "GesSAD Backoffice",
  "short_name": "GesSAD",
  "start_url": "./",
  "display": "standalone",
  "background_color": "#ffffff",
  "theme_color": "#03173d",
  "prefer_related_applications": false,
  "icons": [
    {
      "src": "icon-512.png",
      "type": "image/png",
      "sizes": "512x512"
    },
    {
      "src": "icon-192.png",
      "type": "image/png",
      "sizes": "192x192"
    }
  ]
}

```

Figura 45 – Ficheiro Manifest.json

5.3.4 Autenticação e Autorização

Como já referido na secção 5.2.3 existem três posições definidas, o Administrador, o Técnico de Saúde e o Paciente, sendo que para o paciente ainda não existe um exemplo prático no âmbito desta dissertação. Na GesSAD Web API é definida a lógica de gestão de *tokens* e autorização por envio de um *token*. Na GesSAD PWA é efetuada a Autenticação e, após isso, todos os conteúdos apresentados vão de encontro à posição que o utilizador autenticado tem.

Autenticação

Indique as credenciais de acesso à sua conta

Utilizador é de preenchimento obrigatório

Palavra-passe é de preenchimento obrigatório

Figura 46 - Página de Entrada

Como se pode observar na Figura 46, a autenticação requer um utilizador e uma palavra-passe, de preenchimento obrigatório e, caso estejam errados é apresentada uma mensagem indicativa. Se estiverem corretos, o utilizador é redirecionado para a página principal. Quando

a autenticação é efetuada com sucesso é atribuído um *token* de acesso (e respetivo *token* de refrescamento) que são usados para permitir fazer os pedidos à GesSAD Web API de forma mais segura.

O *token* de acesso, o *token* de refrescamento e alguma informação adicional não sensível é guardada na *LocalStorage*, assim que é efetuado o login pela primeira vez. O *LocalStorage* é um objeto *Javascript* que permite guardar informação no formato chave-valor. Esses dados são mantidos até que a própria aplicação os apague ou até que o utilizador limpe a cache do browser em que estiver a aceder à GesSAD Web API. Para uma melhor gestão do estado da autenticação e autorização é usado o serviço *AuthenticationStateProvider* (próprio da framework .NET Core (Microsoft, 2022)). No excerto de código da Figura 47 é possível observar o método que acede à *LocalStorage* para ir buscar os dados gravados, neste caso, o objeto *User* e associa ao cliente HTTP (*_httpClient*) o mecanismo de autenticação implementado, para que todos os pedidos efetuados à API recebam o *token* de acesso. É também observada a configuração do objeto *Authentication*, próprio do serviço *AuthenticationStateProvider* e que será usado em toda a aplicação para verificar se o utilizador está autenticado ou se tem permissões de acesso a determinados recursos.

```
public override async Task<AuthenticationState>
GetAuthenticationStateAsync()
{
    var user = await _localStorage.GetItem<User>("user");
    if (user == null)
        return _anonymous;

    if (string.IsNullOrEmpty(user.Token))
        return _anonymous;

    _httpClient.DefaultRequestHeaders.Authorization = new
AuthenticationHeaderValue("bearer", user.Token);

    return new AuthenticationState(new ClaimsPrincipal(new
ClaimsIdentity(JsonParser.ParseClaimsFromJwt(user.Token), "jwtAuthType")));
}
```

Figura 47 - Exemplo LocalStorage

No que concerne à visualização de conteúdos, o uso do *AuthenticationStateProvider* é bastante útil pela sua simplicidade de implementação. Permite definir de forma global que ações tomar, ou mensagens apresentar em casos como:

- Utilizador tenta aceder a uma página que não existe
- Utilizador autenticado, tenta aceder a uma página que não tem permissões

- Utilizador não autenticado, tenta aceder a uma página que existe. Neste caso é redirecionado para a página de *Login*.

Estas configurações podem ser observadas no excerto de código da Figura 48 . Aqui são tratadas ações globais a toda a aplicação. Para que estas configurações funcionem é necessário que, em cada página, seja definido o acesso que a mesma permite. Por exemplo:

@attribute [Authorize(Roles = "Administrator")]

```

<CascadingAuthenticationState>
  <Router AppAssembly="@typeof(Program).Assembly">
    <Found Context="routeData">
      <AuthorizeRouteView                               RouteData="@routeData"
DefaultLayout="@typeof(MainLayout)">
        <NotAuthorized>
          @if (!_accountService.LoggedIn is false)
          {
            <Login />
          }
          else
          {
            <h1 class="fw-bolder">Não possui acesso a esta página.</h1>
          }
        </NotAuthorized>
      </AuthorizeRouteView>
    </Found>
    <NotFound>
      @{
        <h1 class="fw-bolder">Página não encontrada.</h1>
        <a href="">Retornar à página principal</a>
      }
    </NotFound>
  </Router>
</CascadingAuthenticationState>

```

Figura 48 - Exemplo configuração

Apesar de não estar a ser detalhado, é possível definir acessos mais específicos individualmente numa página.

5.3.5 Modo offline

Como já referido na secção 4.4.1, o *Service Worker* funciona como um *proxy* entre o *browser* e o acesso à *internet* e a sua principal funcionalidade é armazenar dados em *cache* quando existe. É ele que permite implementar o modo *offline* nas aplicações Web PWA. No caso do Blazor, um ficheiro *service-worker.published.js* é criado automaticamente aquando da criação da aplicação, isto é, o programador por defeito não precisa implementar este comportamento.

Em termos de funcionamento *offline* apenas o *Service Worker* seria suficiente se a aplicação desenvolvida fosse bastante simples, apenas com recurso a páginas HTML/CSS/Javascript. No entanto, quando se trata de desenvolvimentos mais complexos que impliquem o uso de bases

de dados ou interações com APIs, é necessário recorrer a outros mecanismos para se conseguir implementar na integridade o funcionamento *offline*.

Não existem muitas soluções já predefinidas ou previamente implementadas para colmatar este problema em Blazor. Foi escolhida a biblioteca DnetIndexedDb (Blazor, 2022) que permite a criação de uma base de dados IndexedDB (Mozilla, 2022) local no lado do Cliente. O objetivo é que todas as alterações que vão sendo feitas em modo *offline* se preservem nessa base de dados e, assim que o utilizador se ligar de novo à *internet*, essas alterações persistam na base de dados oficial.

Mesmo não sendo requisito que a aplicação funcione em modo *offline*, é uma funcionalidade interessante para os técnicos de saúde. No entanto, esta ainda se encontra em testes dada a complexidade de dados que têm de se gravar. Outra questão levantada é a falta confiança que uma base de dados do lado do cliente, neste caso a IndexedDB, possa causar. Como estão a ser gravados dados pessoais dos clientes e mesmo dos colaboradores, a segurança deste sistema de base de dados está a ser posto em causa (Corella, 2022). Ainda assim, esta funcionalidade deve ser repensada e melhorada em trabalho futuro.

5.3.6 Funcionalidades

Nesta secção, são apresentadas as funcionalidades mais relevantes desenvolvidas. Sendo objetivo que a aplicação seja usada a nível da gestão da empresa em computadores portáteis, acedendo ao browser e, no caso dos profissionais de saúde, através do seu próprio smartphone e/ou computador portátil. As figuras apresentadas relativas às funcionalidades desenvolvidas representam ambas as situações.

Na Figura 49 é possível observar a página principal da aplicação, após autenticação efetuada. Do lado esquerdo estão os menus a que o utilizador autenticado tem acesso. Neste caso específico, é um utilizador com o cargo de Administrador. Estes menus, servem de base para o registo dos Técnicos de Saúde pois estes, preenchem os menus de registos que são alimentados com as informações previamente inseridas pelos administradores no sistema. O utilizador com cargo de Técnico de Saúde, apenas tem acesso ao horário e aos menus de registo.

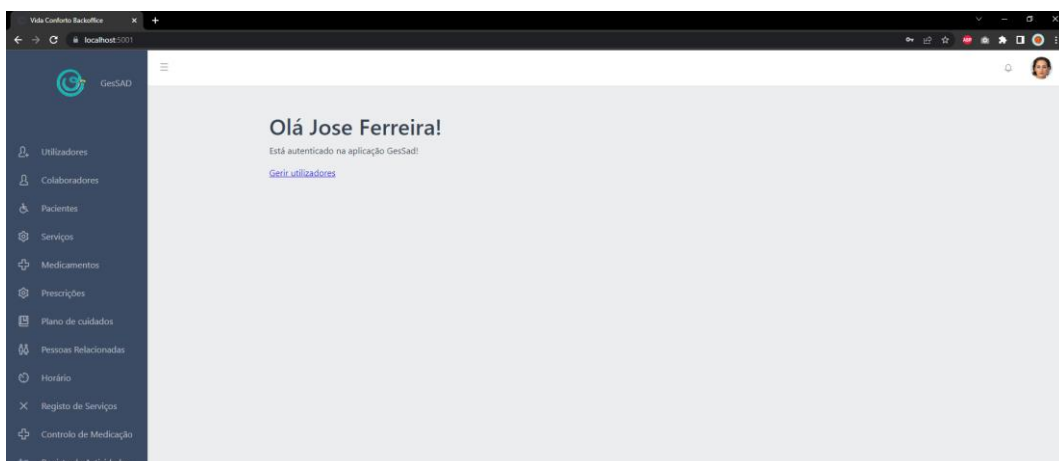


Figura 49 – Página Principal

Do lado direito, a foto que se consegue observar é meramente exemplificativa. Clicando nela, tem-se acesso ao *Logout* da aplicação e a uma secção onde é possível visualizar os dados do utilizador autenticado.

5.3.6.1 Gestão de Utilizadores e Colaboradores

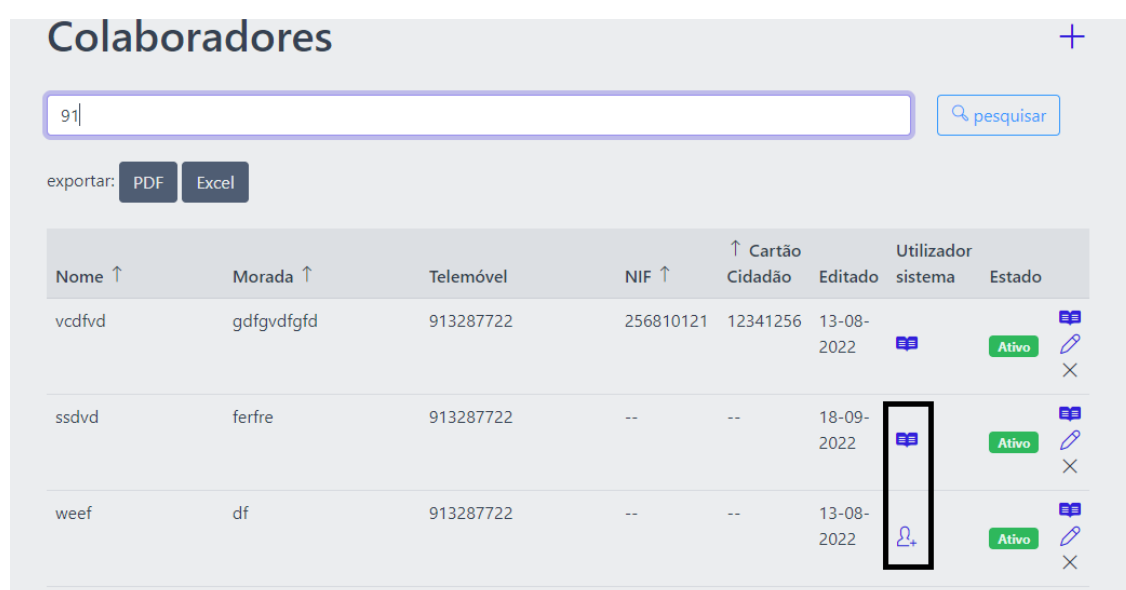
O acesso dos utilizadores ao sistema é de gestão exclusiva do administrador, ou seja, a funcionalidade de registo está apenas disponível para utilização de quem tenha atribuída a posição de Administrador. Isto permite à empresa ter um maior controlo sobre os utilizadores do seu sistema. O Administrador pode, ou adicionar utilizadores livremente ou, a partir de um Colaborador já registado, criar o Utilizador de acesso ao sistema correspondente a esse colaborador. Neste contexto, um Colaborador corresponde a um Cuidador.

A Figura 50 ilustra a primeira situação referida, em que o Administrador tem acesso à lista de utilizadores e consegue editá-los, visualizá-los, apagá-los e adicioná-los.

Utilizadores						+
Nome	Utilizador	Posição	Email	Editado	Estado	
Jose Ferreira	miguel	Administrator	jose@pt.com	27-09-2022	Ativo	
Medico	medico	HealthTechnician	Medico@pt.com	13-09-2022	Ativo	
Pedro	pedro	HealthTechnician	pedro@pt.com		Ativo	
Joao	joao	HealthTechnician	joao@pt.com	15-09-2022	Ativo	
Teste	teste	HealthTechnician	pedro@pedro.com	15-09-2022	Ativo	

Figura 50 - Página de Utilizadores

A Figura 51 representa a lista de colaboradores. O Administrador consegue adicionar, editar, eliminar, pesquisar, ordenar e visualizar a informação relativa a cada colaborador. A criação de um utilizador para aceder ao sistema para um colaborador é feita neste ecrã, através da secção assinalada no retângulo preto. Se não existir utilizador associado é possível adicionar, caso contrário, é possível visualizar os dados do mesmo. Ao clicar em adicionar um novo utilizador, o sistema redireciona para a página de adição de novo utilizador, cria o utilizador associado ao colaborador anteriormente selecionado e escolhe o perfil de utilizador que pretende entre os três já referidos.



The screenshot shows the 'Colaboradores' page with a search bar containing '91', a search button, and export buttons for PDF and Excel. Below is a table with columns for Name, Address, Mobile, NIF, Cardholder, Edited, System User, and Status. The 'System User' column for the second row is highlighted with a black box.










Nome ↑	Morada ↑	Telemóvel	NIF ↑	↑ Cartão Cidadão	Editado	Utilizador sistema	Estado
vcdfvd	gdfgvdfgfd	913287722	256810121	12341256	13-08-2022		Ativo  
ssdvd	ferfre	913287722	--	--	18-09-2022		Ativo  
weef	df	913287722	--	--	13-08-2022		Ativo  

Figura 51 - Página de Colaboradores

5.3.6.2 Gestão de Planos de Cuidados

A funcionalidade dos planos de cuidados, visível na Figura 52 está associada ao plano que o colaborador irá efetuar a um paciente. Este plano inclui assim os diversos serviços que a empresa dispõe para os seus pacientes.

O colaborador irá verificar o plano de cuidados para o paciente, no horário que é atribuído pelo administrador, e posteriormente registar na página de registos dos serviços, Pode registar alguma atividade pendente na página de registo de atividades, bem como observações.

Para poder criar um plano de cuidados, tem que ser obrigatoriamente administrador do sistema.

Paciente ↑	Serviço ↑	Periodicidade	Editado	Estado
Joana	Arrumar a cozinha	Sábado, Domingos, (2 vezes por dia)	13-09-2022	Ativo
Jose Ferreira	Higiene	Dias da semana, (3 vezes por dia)		Ativo

Figura 52 - Página de Planos de Cuidados

Após autenticado, clica no menu do plano de cuidados e no canto superior direito, clica no “+” para inserir um novo plano como se verifica na Figura 53.

Paciente *
Por favor, seleccione uma opção

Serviço *
Por favor, seleccione uma opção

Número de vezes *
0

Periodicidade *
Semana Sábados Domingos Feriados

Gravar Cancelar

Figura 53 - Criar Plano de Cuidados

Para criar um plano, o administrador tem que ter previamente já registado o paciente e o/os serviço/s que irá criar no plano para este paciente, bem como a periodicidade que é necessária para esse serviço.

Quando é necessário atualizar um plano de cuidados, o administrador, no menu, seleciona o ícone de edição, e abrirá uma página onde poderá, editar esse plano, neste caso, mudando o serviço, o número de vezes e a periodicidade. No final poderá gravar ou cancelar as alterações como se observa na Figura 54.

Editar plano de cuidados

Paciente *

Serviço *

Número de vezes *

Periodicidade *
 Semana Sábados Domingos Feriados

Figura 54 - Editar Plano Cuidados

Como é demonstrado na Figura 55 , no menu, ao selecionar o ícone de detalhes, aparece a página de detalhes, onde é possível visualizar mais pormenores sobre este registo, incluindo os dados do paciente, última vez que foi atualizado e a data de início do plano.

Detalhes do plano de cuidados

Paciente ([ver detalhes](#)) : Joana
 Morada: ,
 NIF: 0
 Telemóvel/Telefone: /

Serviço ([ver detalhes](#)) : Arrumar a cozinha

Frequência Sábado, Domingos, (2 vezes por dia)

Adicionado: 30/08/2022 14:37:50

Última atualização: 13/09/2022 17:00:10

Registo ativo: Sim

[Voltar](#)

Figura 55 - Detalhes Plano Cuidados

Em relação aos dados pessoais dos pacientes, por opção da empresa, os colaboradores não têm acesso a nenhuma informação que está registada na aplicação GesSAD como anteriormente referido. A informação é transmitida, pelos administradores a cada colaborador individualmente, nomeadamente o nome, a morada, etc.

5.4 Horário

O horário é uma das principais funcionalidades da aplicação. Permite ao administrador fazer as escalas de serviços do colaborador para aquele paciente. Por outro lado, permite também a cada colaborador, ter acesso para consulta dos registos do serviço, a todos os pacientes da sua escala. Como é possível observar na Figura 56 , a página é visível tanto para Administrador,

como por Técnicos de Saúde, ficando o horário acessível para todos. É possível pesquisar o horário do colaborador e/ou paciente, através da seleção dos dados relevantes nas *combobox*. Além disso, é também possível exportar o horário naquele momento para Excel e/ou PDF.

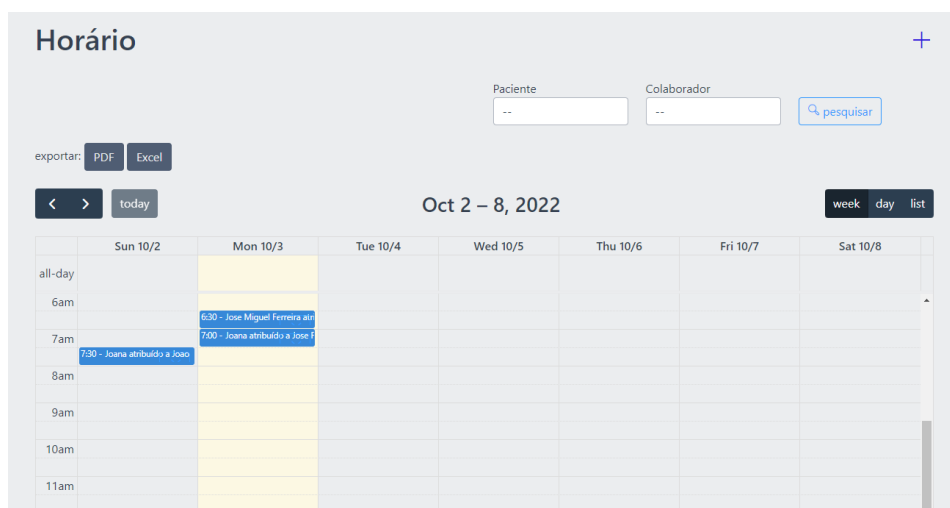


Figura 56 - Horário

A visualização no calendário, pode ser feita por semana ou dia. Estas foram as vistas definidas para serem implementadas. Isto porque os horários são planeados semanalmente e sofrem muitas alterações. É importante referir que esta página é apenas para visualização. A adição/remoção de escalas é feita numa página à parte com um aspeto semelhante, mas acessível apenas os Administradores. Em termos de implementação, foi usada a biblioteca Javascript FullCalendar para a visualização e manipulação das escalas visíveis na Figura 56 .

O excerto de código da Figura 57 ilustra o modo como as escalas estão a ser associadas ao calendário.

```

        foreach (var item in _list.Data)
        {
            var title = $"{item.Patient.Name} atribuído a
{item.Employee.Name}";
            var calendarEvent = new CalendarEvent
            {
                Identifier = item.Id.ToString(),
                Title = title,
                Start = new DateTime(item.ServiceDay.Year,
item.ServiceDay.Month, item.ServiceDay.Day, item.StartHour.Hours,
item.StartHour.Minutes, item.StartHour.Seconds),
                End = new DateTime(item.ServiceDay.Year,
item.ServiceDay.Month, item.ServiceDay.Day, item.EndHour.Hours,
item.EndHour.Minutes, item.EndHour.Seconds),
                SolidColor = CalendarEventSolidColor.Primary,
                DotColor = CalendarEventDotColor.Light,
                Status = CalendarEventStatus.Active
            };
            Events.Add(calendarEvent);
        }

        var fullCalendarInterop = await
JS.InvokeAsync<IJSObjectReference>("import",
"./Pages/Scheduler/Index.razor.js");

        var options = new CalendarOptions
        {
            Id = _uid,
            DefaultView = View,
            CalendarEvents = Events.Where(r => r.Status ==
CalendarEventStatus.Active).ToList()
        };

        var dotNetObjectReference = DotNetObjectReference.Create(this);
        await
fullCalendarInterop.InvokeVoidAsync("FullCalendarInterop.init", options,
dotNetObjectReference);

```

Figura 57 - Exemplo Horário

Neste caso, é necessário criar um objeto `CalendarEvent` por cada escala visível no calendário. Esta classe faz parte do componente desenvolvido internamente que manipula via Javascript o calendário. De seguida é necessário efetuar a comunicação entre o código C# (neste caso, uma razor page) e a classe de Javascript que detém a gestão do calendário. Para tal, é importado o ficheiro javascript em questão:

```

(await
JS.InvokeAsync<IJSObjectReference>("import", "./Pages/Scheduler/Index.razor.js")

```

para que o consigamos usar. De seguida, para executar funções que nele estejam contidas e, neste caso, enviar todos os dados necessários à sua configuração, é executado o excerto de código:

```
fullCalendarInterop.InvokeVoidAsync("FullCalendarInterop.init", options,  
dotNetObjectReference)
```

Ou seja, o "FullCalendarInterop.init" representa o objeto javascript que contém a função init, função essa que detém toda a inicialização do calendário.

Toda a manipulação do calendário é feita de forma semelhante, desde a seleção de escalas no próprio calendário para gravar no sistema, ou mesmo o selecionar de uma escala específica para ser apagada.

5.5 Sumário

Neste capítulo é possível observar pormenores da implementação para a solução proposta. Foram apresentados detalhes dos desenvolvimentos, bem como sugeridas algumas melhorias futuras que serão especificadas em pormenor no capítulo 7.

Capítulo 6 – Teste e Validação da Solução

Neste capítulo, é descrito o processo de avaliação da solução. Esta é uma das etapas cruciais no ciclo de um produto ou serviço, verificando-se se o propósito foi ou não cumprido.

São definidos os indicadores de avaliação, metodologias de avaliação e por último avaliam-se os resultados obtidos.

6.1 Indicadores de avaliação

Os indicadores de avaliação têm como objetivo auxiliar a avaliação do projeto. Descrevem-se os que irão ser alvo de avaliação para a aplicação *web* progressiva a desenvolver.

- **Usabilidade:** pretende-se avaliar a interação do utilizador com a aplicação *web* progressiva, nomeadamente perceber se a aplicação é de fácil utilização e intuitiva. Este tipo de testes permite verificar se o projeto corresponde às necessidades levantadas e identificar possíveis erros e melhorias.
- **Funcionalidade:** pretende-se avaliar se as funcionalidades cumprem os requisitos da organização.
- **Adaptabilidade:** pretende-se compreender a facilidade de adaptação dos utilizadores às funcionalidades do sistema.

Para avaliar os indicadores mencionados, será elaborado um questionário para saber a opinião dos utilizadores sobre a sua experiência.

6.2 Metodologia de Avaliação

Nesta secção apresentam-se os testes de *software* e o inquérito de usabilidade.

6.2.1 Testes de software

Os testes de *software* são importantes na validação de qualquer projeto. Permitem identificar possíveis erros, falhas na especificação dos requisitos e até falhas funcionais. Neste trabalho serão realizados:

- **Testes funcionais** - os testes funcionais têm como objetivo entender o comportamento da aplicação. À medida que o utilizador percorre a aplicação pode identificar possíveis problemas. Para validar o funcionamento da aplicação, serão inseridos dados válidos e inválidos;
- **Testes de usabilidade** - com os testes de usabilidade é possível avaliar a interação do utilizador com a aplicação *web* progressiva.

Estes testes permitem verificar se o projeto corresponde às necessidades levantadas e identificar possíveis erros e melhorias. Foi elaborado um questionário para saber a opinião dos utilizadores sobre a sua experiência com a aplicação *web* progressiva.

6.2.2 Inquéritos de satisfação

De maneira a analisar a satisfação dos utilizadores, desenvolver-se-á um questionário que terá como objetivo verificar se estes se encontram satisfeitos com o sistema desenvolvido. As perguntas serão concretas, evitando a indecisão. Relativamente às respostas, recorrer-se-á à utilização da escala de *likert* (A. Joshi, 2015), sendo esta normalmente constituída por cinco opções:

- Discordo totalmente
- Discordo parcialmente
- Indiferente
- Concordo parcialmente
- Concordo totalmente

A adoção deste tipo de escala permite obter conclusões objetivas de cariz numérico e ou percentual, que possibilitam apresentar conclusões estatisticamente analisadas.

6.3 Quantitative Evaluation Framework

Para a avaliação qualitativa do projeto é utilizado o modelo *Quantitative Evaluation Framework* (QEF). Este modelo no final diz-nos quão próximo dos objetivos definidos se encontra a solução desenvolvida.

Este método apresenta uma visão holística, tendo em consideração as especificidades do produto. Possibilita também uma avaliação do desenvolvimento, onde cada uma das dimensões é composta por um conjunto de fatores. Cada fator tem um certo peso, sendo que, no final, a soma destes em cada dimensão terá que ser igual a um. Cada fator é dividido em requisitos, estes apresentam um peso dentro do fator em que estão inseridos. A atribuição do peso aos requisitos deverá ser de acordo com a sua importância, podendo ser este de 2,4,6,8, ou 10.

A aplicação deste método é importante no sentido em que pode ser aplicado em qualquer fase do produto para monitorizar o seu estado de desenvolvimento. No Anexo K encontra-se o diagrama de QEF.

6.4 Testes Funcionais

Durante o desenvolvimento da aplicação foram efetuados e documentados (apenas alguns) testes para permitir verificar a saída dos dados, usando dados diferentes. Alguns testes permitiram identificar erros, que foram posteriormente corrigidos.

Da Tabela 7 à Tabela 11 é exibida a documentação dos testes efetuados, com descrição dos dados usados, resultados esperados e resultados obtidos.

A Tabela 7 descreve os testes efetuados que permitem verificar se um utilizador se autentica corretamente, se são apresentados os menus corretos, ou se ao inserir credenciais erradas aparecem as mensagens correspondentes.

Tabela 6 - Verificar se a autenticação funciona corretamente

Número do Teste	1		
Descrição	Verificar quando um utilizador se autentica, se funciona corretamente		
Dados do Teste	Resultado Esperado	Resultado Obtido	Observações
- Indicar credenciais válidas de acesso	Espera-se o acesso á aplicação na página definida como "principal"	Autenticação efetuada com sucesso.	
- Indicar credenciais erradas de acesso	Espera-se uma mensagem indicativa de password e/ou nome errados.	Foi apresentada mensagem indicativa de nome e/ou password erradas.	Retificada mensagem de erro

A Tabela 8 pretende verificar se os registos selecionados são corretamente gravados na base de dados, mesmo quando não são selecionados todos os dados.

Tabela 7 - Verificar se os registos são corretamente gravados na base de dados

Número do Teste	2		
Descrição	Verificar se os registos selecionados são corretamente gravados na base de dados		
Dados do Teste	Resultado Esperado	Resultado Obtido	Observações
Registos da tabela auxiliar: Paciente	Clicar em "Gravar" e aparecer mensagem indicativa do sucesso da operação. Ir a "Listagem" e verificar que os registos foram inseridos	"Paciente adicionado com sucesso!". Confirmação que o registo foi inserido na base de dados.	
Registos da tabela auxiliar: Paciente	Clicar em "Gravar" e aparecer mensagem indicativa do insucesso da operação.	"Nome é de preenchimento obrigatório"	Os registos contêm campos obrigatórios, logo o paciente não tinha sido preenchido.

A Tabela 9 tem por objetivo verificar se é possível editar um registro de um paciente, já adicionado na base de dados.

Tabela 8 - Verificar se é possível editar um registro

Número do Teste	3		
Descrição	Verificar se é possível editar um registro		
Dados do Teste	Resultado Esperado	Resultado Obtido	Observações
- Selecionar um registro da tabela "Pacientes" e clicar no botão "Editar". - Alterar o nome para um valor diferente e clicar em "Guardar"	Ao visualizar a tabela "Pacientes" espera-se que o valor tenha sido editado corretamente.	O Paciente foi atualizado com sucesso.	
- Selecionar um registro da tabela "Pacientes" e clicar no botão "Editar". - Alterar o nome para um valor diferente e clicar em "Voltar à lista"	Espera-se que ao clicar em "Voltar à lista" o registro não tenha sido alterado.	O registro não foi alterado.	Atualizado ao registrar a página retornar a lista

A Tabela 10 apresenta os testes efetuados para verificar se os dados estão corretos quando é escolhido um valor numa determinada *combobox*.

Tabela 9 - Verificar a seleção de um elemento de uma *combobox* no Registro de Serviços

Número do Teste	4		
Descrição	Verificar a seleção de um elemento de uma <i>combobox</i> : Registro de Serviços		
Dados do Teste	Resultado Esperado	Resultado Obtido	Observações
- Combobox Registro de Serviços - Selecionar "Paciente"	Visualizar na <i>combobox</i> de Registro de serviços, os Pacientes	Visualização dos Pacientes adicionados	Corrigido para aparecer somente o nome e não o id
- Combobox Registro de Serviços - Selecionar "Serviços"	Visualizar na <i>combobox</i> de Registro de serviços, os serviços	Visualização das Serviços adicionadas	Corrigido para aparecer somente o nome e não o id
- Combobox Registro de Serviços - Selecionar Colaborador"	Visualizar na <i>combobox</i> de Registro de serviços, os colaboradores	Visualização dos Colaboradores adicionados	Corrigido para aparecer somente o nome e não o id

A Tabela 11 apresenta os testes efetuados para verificar os menus de acesso na autenticação

Tabela 101 - Verificar os menus de acesso

Número do Teste	6		
Descrição	Verificar os menus de acesso		
Dados do Teste	Resultado Esperado	Resultado Obtido	Observações
- Login efetuado com role administrador	Ao fazer login, ter acesso a todos os menus	Conforme esperado, acesso a todos os menus	
- Login efetuado com role de Health Technician	Ao fazer login, ter acesso apenas aos menus esperados	Verificado acesso a menus da parte administrativa	Problema foi corrigido

Durante o desenvolvimento deste processo, surgiram alguns erros, que foram validados pela aplicação dos testes, tendo sido posteriormente corrigidos.

6.5 Análise da Usabilidade

O questionário presente na Figura 83, tem como objetivo analisar a usabilidade do utilizador com a aplicação web progressiva. Uma versão de testes da GesSAD App foi disponibilizada para pessoal administrativo e colaboradores. Fez-se um questionário a 5 pessoas intervenientes na empresa como pré-teste, com o objetivo de aferir a informação para posteriormente melhorar.

Como se verifica na Figura 58, os participantes no questionário consideram que a aplicação tem uma boa usabilidade relativamente à forma da organização, à interação e navegação na aplicação e ao atendimento das necessidades atuais.

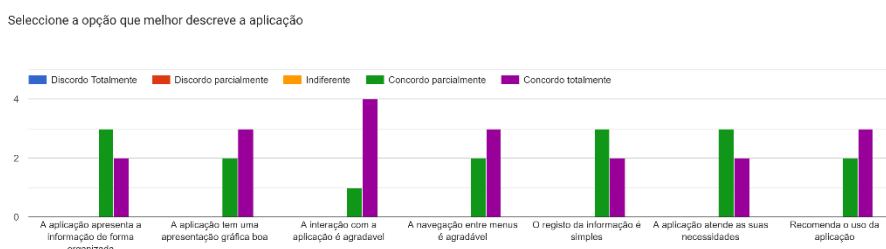


Figura 58 – Usabilidade da Aplicação

Em relação à funcionalidade da aplicação, como podemos observar pela Figura 59 os participantes no questionário consideram no geral que a aplicação facilita o registo das atividades/ocorrências, bem como a informação disponibilizada ser suficiente para as tarefas a desempenharem habitualmente.

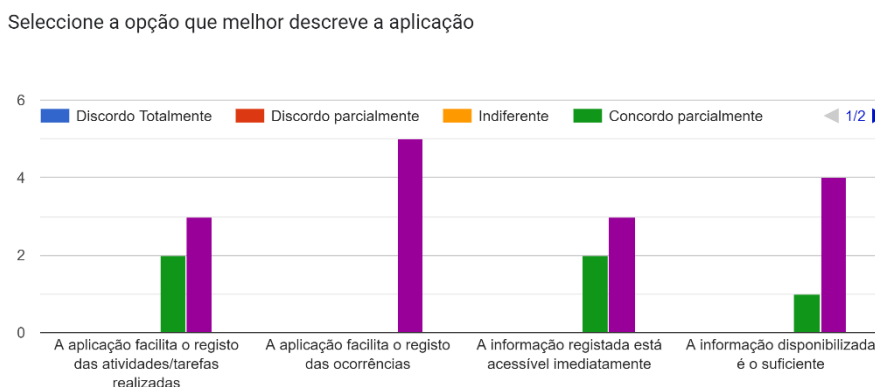


Figura 59 - Funcionalidade da Aplicação

Relativamente à adaptabilidade da aplicação, segundo a Figura 60 podemos retirar a informação que se adaptam bem a aplicação.

Para cada frase indique a dificuldade que teve

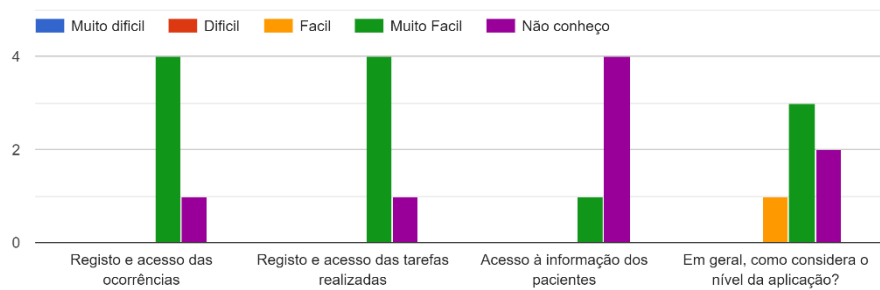


Figura 60 - Adaptabilidade da Aplicação

Com estes resultados, é possível concluir que a aplicação satisfaz as necessidades, mas existe espaço para melhoria e algum trabalho futuro que se indicará no capítulo seguinte.

6.6 Sumário

Neste capítulo foi definida a abordagem metodológica seguida para a avaliação trabalho desenvolvido. Foram também apresentados os resultados obtidos e as conclusões alcançadas.

Capítulo 7 - Conclusão e Trabalho Futuro

Neste capítulo, é descrita a conclusão e trabalho futuro.

7.1 Conclusão e Perspetivas de Trabalho Futuro

Esta dissertação teve como principal objetivo o desenvolvimento de uma aplicação *web* progressiva para uma empresa de apoio domiciliário. Esta aplicação vai facilitar a monitorização da informação dos diversos serviços prestados aos seus clientes, melhorar a comunicação entre os vários setores da empresa, aumentar a eficiência da gestão e reduzir custos.

A elaboração da análise de valor permitiu colocar o trabalho num contexto real de negócio, tendo sido identificado o seu potencial para os clientes alvo e para os utilizadores desses clientes. Por outro lado, o levantamento do Estado da Arte sobre o tema do Apoio Domiciliário, foi importante para perceber o que se encontra atualmente feito no que diz respeito a este tipo de aplicações. Realizou-se um levantamento das soluções existentes no mercado e concretizou-se a sistematização das respetivas características.

O principal objetivo foi assim cumprido. O *backoffice* suporta a informatização dos principais processos da empresa, que permite a administração libertar a maior parte dos papéis utilizados para começar a utilizar a aplicação, os colaboradores podem fazer os seus registos e anotações no fim ou durante os seus serviços. A administração consegue visualizar em tempo real estes registos e com isso ganhar tempo para outras funções.

Mediante a avaliação da solução feita, dos questionários feitos aos colaboradores da empresa, comprovam que a aplicação foi bem recebida, e em conjunto foram apontados detalhes e desenvolvimentos futuros, já referidas nas secções anteriores.

No que diz respeito ao trabalho futuro, há muitos aspetos que podem ajudar a melhorar a aplicação, alguns deles já levantados pelos próprios colaboradores da empresa, nomeadamente a administração, a implementação dos questionários, melhorar o calendário implementado para permitir uma melhor visualização nos telemóveis dos colaboradores, melhorar a segurança da API, neste momento os identificadores das entidades estão a ser criados como um número, de futuro mudar para Guid, ou Nulid (Nulid, 2022) por exemplo, para uma melhor segurança no acesso a base de dados. Relativamente ao funcionamento offline, a gravação dos dados pessoais dos clientes, colaboradores e mesmo funcionários terá de ser assegurada, será repensada e melhorada juntamente com os restantes pontos referidos acima.

A aplicação não foi testada em IOS até ao momento, será algo a ter em conta pois poderá haver colaboradores que necessitem com este sistema operativo mais tarde.

Em termos pessoais, gostei de desenvolver esta dissertação. Todo o processo de análise, desde o tipo de tecnologias que poderiam ou não ser usadas, até à própria idealização em conjunto com a empresa da aplicação em si. Há muitas melhorias a fazer, mas penso que o principal objetivo foi atingido, a informatização dos processos, registos por parte dos funcionários está funcional, mas existe sempre espaço para melhorar.

Referências

- A. Joshi, S. K. (2015). *Likert Scale: Explored and Explained*.
- Agile, S. (Fevereiro de 2021). Domain Modeling in Agile at Large Scale. Obtido de <https://www.scaledagileframework.com/domain-modeling/>
- Alexander Osterwalder, Y. P. (2014). *Value Proposition Design: How to Create Products and Services Customers Want*.
- Aliexpress. (s.d.). Obtido de <https://pt.aliexpress.com/>
- AltexSoft. (Maio de 2018). Progressive Web Apps: Core Features, Architecture, Pros and Cons. Obtido de <https://www.altexsoft.com/blog/engineering/progressive-web-apps/>
- Android. (2022). *Android*. Obtido de Android: https://www.android.com/intl/pt_pt/
- AppDevice. (2019). Help2care. Obtido de <https://apps.apple.com/us/app/id1439443105>
- Apple. (2022). *App Store - Apple (PT)*. Obtido de App Store: <https://www.apple.com/pt/app-store/>
- Apple. (s.d.). *MacOs*. Obtido de <https://www.apple.com/pt/macOS/monterey/>
- Asisges. (2022). *Software de gestão de serviços de apoio domiciliário*. Obtido de Asisges: <https://www.asisges.com/pt-pt/>
- ATES, & CNIS. (Março de 2021). Impacto da pandemia de Covid19 nas IPSS e seus utentes em Portugal. Obtido de <http://cnis.pt/wp-content/uploads/2021/03/Relat%C3%B3rio-COVID-19-nas-IPSS.pdf>
- Bitbucket. (2022). *Bitbucket*. Obtido de <https://bitbucket.org/>
- Blazor. (2022). *DnetIndexedDb*. Obtido de <https://github.com/amuste/DnetIndexedDb>
- Bonfim, C., & Veiga, S. (1996). *Serviços de Apoio Domiciliário*. Lisboa.
- Brothers, A. (2021). *PWA vs Native App – Differences and Similarities for Business, Performance and User Experience*. Obtido de <https://asperbrothers.com/blog/pwa-vs-native-app/>
- Chris. (Janeiro de 2021). Apple Safari Ships Service Worker and Progressive Web App (PWA) Support on iOS 11.3. Obtido de <https://love2dev.com/blog/apple-ships-service-workers/>
- Clockwise. (2021). *How Much Does It Cost To Build a Web Application From Scratch*. Obtido de <https://clockwise.software/blog/how-much-does-it-cost-to-build-a-web-application/>
- Conforto, V. e. (Janeiro de 2009). *VidaConforto - Apoio Domiciliário*. Obtido de VidaConforto - Apoio Domiciliário: <https://vidaconforto.pt/>
- Corella, F. (2022). *Storing Cryptographic Keys in Persistent Browser Storage*. Obtido de <https://pomcor.com/2017/06/02/keys-in-browser/>

- CssChopper. (2021). *Web App Vs Progressive Web App: Why Are They Different From Each Other?* Obtido de <https://www.csschopper.com/blog/web-app-vs-progressive-web-app/>
- Cunha, L. M. (2007). *Modelos Rasch e Escalas de Likert e Thurstone na*. Obtido de https://repositorio.ul.pt/bitstream/10451/1229/1/18914_ULFC072532_TM.pdf
- Divante. (2019). *The PWA Book*. Obtido de <https://www.divante.com/pwabook/chapter/02-the-history-of-pwas>
- Dziuba, A. (2022). *PWA vs. Native Apps: Which One Should You Choose?* Obtido de https://relevant.software/blog/pwa-vs-native-apps-which-one-should-you-choose/#PWA_vs_Native_Apps_Comparison
- Edward, F. R. (Junho de 2010). *Stakeholder Theory: The State of the Art. Stakeholder Theory: The State of the Art*.
- F3M. (2020). *F3M - Apoio Domiciliário. Apoio Domiciliário*. Obtido em 6 de Janeiro de 2022, de <https://www.f3m.pt/pt/software/ipss/apoio-domiciliario#>
- FileDoc. (2019). Obtido em 19 de Abril de 2020, de https://www.filedoc.eu/Cache/binImagens/Filedoc_brochura_2019-2082.pdf
- Flipkart. (s.d.). Obtido de <https://www.flipkart.com/>
- Forcepoint. (s.d.). *What is Bring Your Own Device (BYOD)?* Obtido de <https://www.forcepoint.com/cyber-edu/bring-your-own-device-byod>
- Freeman, R. E. (1984). *Strategic management : a stakeholder approach*. Boston: Pitman, 1984.
- Gibb, R. (2016). *What is a Web Application?* Obtido de <https://blog.stackpath.com/web-application/>
- Git. (2022). *Git*. Obtido de <https://git-scm.com/>
- GlobalStats. (Janeiro de 2022). *Desktop Browser Market Share Worldwide*. Obtido de <https://gs.statcounter.com/browser-market-share/desktop/worldwide/#monthly-202101-202201-bar>
- Google. (2022). *Formulários do Google*. Obtido de <https://www.google.com/forms/about/>
- Google. (2022). *Google Play*. Obtido de Google Play: <https://play.google.com/store>
- Google. (s.d.). *Google Chrome*. Obtido de <https://www.google.com/intl/pt-PT/chrome/>
- Google. (s.d.). *Google Developers*. Obtido de <https://developer.android.com/distribute/console?hl=pt-br>
- Grevi, G., & Gnesotto, N. (2008). *O Mundo em 2025*.
- Hauser, J., & Gerry Katz, A. G. (Dezembro de 2010). *Quality Function Deployment. Quality Function Deployment*.
- Help2care. (2019). *Help2care*. Obtido de <http://www.help2care.pt/help2care/project>

- IEEE Xplore. (s.d.). Obtido de IEEE Xplore: <https://ieeexplore.ieee.org/Xplore/home.jsp>
- INE. (2015). *Envelhecimento da População Residente em Portugal e na União Europeia*. Obtido de https://www.ine.pt/ngt_server/attachfileu.jsp?look_parentBoui=232024193&att_display=n&att_download=y
- INE. (Novembro de 2016). A proporção de utilizadores de internet em mobilidade duplicou em 4 anos - 2016. Obtido de https://www.ine.pt/xportal/xmain?xpid=INE&xpgid=ine_destaquas&DESTAQUESdest_boui=250254698&DESTAQUESmodo=2
- INE. (Novembro de 2017). Práticas de gestão contam significativamente para o desempenho económico das empresas - 2016. Obtido de https://www.ine.pt/xportal/xmain?xpid=INE&xpgid=ine_destaquas&DESTAQUESdest_boui=302954141&DESTAQUESmodo=2&xlang=pt
- Ine. (21 de Novembro de 2019). *80% dos utilizadores de internet participam em redes sociais - 2019*. Obtido de https://www.ine.pt/xportal/xmain?xpid=INE&xpgid=ine_destaquas&DESTAQUESdest_boui=354447153&DESTAQUESmodo=2
- Ine. (Junho de 2020). Ine. *Estimativas de população residente em portugal 2020*.
- InoveSaude. (3 de Janeiro de 2022). *Gestão de serviços de apoio domiciliario*. Obtido de <https://inovesaude.com/pt/gestao-de-servicos-de-apoio-domiciliario>
- Itabashi et al. (2005). A Support System for Home Care Service Based on Multi-agent System. *A Support System for Home Care Service Based on Multi-agent System*. Obtido de <https://ieeexplore.ieee.org/document/1689213>
- Keyoti. (2022). *Conveyor allows you to easily test web applications from external tablets and phones on your network*. Obtido de [Conveyor allows you to easily test web applications from external tablets and phones on your network: https://keyoti.com/products/conveyor/index.html](https://keyoti.com/products/conveyor/index.html)
- Koen et al. (2002). *Fuzzy Front End : Effective Methods , Tools , and Techniques*.
- Lage. (2005). *Envelhecer em Portugal. Psicologia, saúde e prestação de cuidados*. Climepsi Editores.
- Larman, C. (2001). *Applying UML and Patterns*. Obtido de https://books.google.pt/books?id=r8i-4En_aa4C&printsec=frontcover&hl=pt-PT&source=gbs_ge_summary_r&cad=0#v=onepage&q&f=false
- LePage, P. (Abril de 2020). *The Cache API: A quick guide*. Obtido de <https://web.dev/cache-api-quick-guide/>
- Lewis, J., & Fowler, M. (2014). *Microservices a definition of this new architectural term*. Obtido de <https://martinfowler.com/articles/microservices.html>

Linux. (s.d.). *Linux*. Obtido de Linux: <https://www.linux.org/>

Madeinweb. (s.d.). OS DESAFIOS DO DESENVOLVIMENTO DE APLICAÇÕES. Obtido em Fevereiro de 2022, de <https://www.madeinweb.com.br/os-desafios-do-desenvolvimento-de-aplicativos/>

MaisEficaz. (Maio de 2018). *MySenior - Programa para gestão de Lares de Idosos*. Obtido de MySenior: <https://mysenior.com/>

Marktest. (Julho de 2020). 7,2 milhões de portugueses usam smartphone. Obtido de <https://www.marktest.com/wap/a/n/id~2700.aspx>

Metatheke. (2022). *Ankira*. Obtido em 2 de Janeiro de 2022, de Ankira - Plataforma para a gestão de lares de idosos: <https://ankira.pt/>

Microsoft. (2022). *ASP.NET Core Blazor authentication and authorization*. Obtido de ASP.NET Core Blazor authentication and authorization: <https://learn.microsoft.com/en-us/aspnet/core/blazor/security/?view=aspnetcore-6.0>

Microsoft. (2022). *Blazor, Build client web apps with C#*. Obtido de Build client web apps with C#: <https://dotnet.microsoft.com/en-us/apps/aspnet/web-apps/blazor>

Microsoft. (2022). *Edge*. Obtido de <https://www.microsoft.com/pt-pt/edge>

Microsoft. (2022). *Entity Framework Core*. Obtido em Agosto de 2022, de <https://docs.microsoft.com/en-us/ef/core/>

Microsoft. (2022). *Introduction to ASP.NET Core Blazor*. Obtido de <https://docs.microsoft.com/en-us/aspnet/core/blazor/?view=aspnetcore-6.0>

Microsoft. (2022). *System.Linq.Dynamic.Core*. Obtido de System.Linq.Dynamic.Core: <https://www.nuget.org/packages/System.Linq.Dynamic.Core>

Microsoft. (2022). *Tutorial: Create a web API with ASP.NET Core*. Obtido de Tutorial: Create a web API with ASP.NET Core: <https://learn.microsoft.com/en-us/aspnet/core/tutorials/first-web-api?view=aspnetcore-6.0&tabs=visual-studio>

Microsoft. (2022). *Visual Studio Community*. Obtido de Visual Studio 2022 Community: <https://visualstudio.microsoft.com/vs/community/>

Microsoft, D. (Janeiro de 2022). Build Progressive Web Applications with ASP.NET Core Blazor WebAssembly. Obtido de <https://docs.microsoft.com/en-us/aspnet/core/blazor/progressive-web-app?view=aspnetcore-6.0&tabs=visual-studio>

Mozilla. (2022). *IndexedDB API*. Obtido de https://developer.mozilla.org/en-US/docs/Web/API/IndexedDB_API

Mozilla, D. (Novembro de 2021). Making PWAs work offline with Service workers. Obtido de https://developer.mozilla.org/en-US/docs/Web/Progressive_web_apps/Offline_Service_workers

Mozilla. (2022). *Mozilla Firefox*. Obtido de <https://www.mozilla.org/pt-PT/firefox/new/>

MySQL. (2022). *MySQL*. Obtido de <https://www.mysql.com/>

- Ngrok. (2022). *Ngrok - Online in One line*. Obtido de Ngrok - Online in One line: <https://ngrok.com/>
- Nova, K. (2018). What is a monolithic application? Obtido de <https://blog.heptio.com/what-is-a-monolithic-application-e375f5ad5ecb>
- Nulid. (2022). Obtido de <https://github.com/RobThree/Nulid>
- Oliveira, J. (2010). *Psicologia do Envelhecimento e do Idoso (4ª ed.)*. Porto: LivPsic.
- Opera. (2022). *Opera*. Obtido de <https://www.opera.com/pt>
- Osterwalder, A. (Janeiro de 2004). The Business Model Ontology – A Proposition in a Design Science Approach.
- Paquin, R. L., & Joyce, A. (Junho de 2016). The triple layered business model canvas: A tool to design more sustainable business models. *The triple layered business model canvas: A tool to design more sustainable business models*.
- Pinterest. (s.d.). *Pinterest*. Obtido de <https://www.pinterest.pt/>
- Portugal2020. (s.d.). *Portugal2020 - Aposte nas suas ideias*. Obtido de <https://portugal2020.pt/>
- PowerDMS. (23 de Dezembro de 2020). Why you should adopt a paperless policy management system. *Why you should adopt a paperless policy management system*. Obtido em 4 de Janeiro de 2022, de <https://www.powerdms.com/policy-learning-center/why-you-should-adopt-a-paperless-policy-management-system>
- Público, J. (29 de Março de 2018). Portugal é um dos países mais envelhecidos da UE à frente de Espanha. Obtido em Dezembro de 2021, de <https://www.publico.pt/2018/03/29/sociedade/noticia/portugal-e-um-dos-paises-mais-envelhecidos-da-ue-a-frente-de-espanha-1808478>
- QuestionPro. (2022). *QuestionPro: Free Online Survey Software and Tools*. Obtido de <https://www.questionpro.com/>
- Republica, D. d. (30 de Janeiro de 2013). Portaria n.º 38/2013. *Portaria n.º 38/2013*. Obtido de <https://files.dre.pt/1s/2013/01/02100/0060500608.pdf>
- Ruparel, C., & Vidhya, J. (2018). Cross-platform development: Challenges and opportunities. Obtido de https://www.researchgate.net/publication/326108814_Cross-platform_development_Challenges_and_opportunities
- Rykov, S. (Novembro de 2021). Progressive Web App Development: How to Cook PWA in 2022. Obtido de <https://dev.to/sergeyrykov/progressive-web-app-development-how-to-cook-pwa-in-2022-15g2>
- Safari. (s.d.). *Safari - Apple*. Obtido de <https://www.apple.com/safari/>
- Sam Richard, P. L. (2020). *O que são Progressive Web Apps?* Obtido de <https://web.dev/what-are-pwas/>


- Santoni, M. (Agosto de 2021). Progressive Web Apps browser support & compatibility. Obtido de <https://www.goodbarber.com/blog/progressive-web-apps-browser-support-compatibility-a883/>
- ScandiPWA. (19 de Julho de 2019). History of Progressive Web Apps. Obtido de <https://medium.com/progressivewebapps/history-of-progressive-web-apps-4c912533a531>
- Serilog. (2022). *Flexible, structured events — log file convenience*. Obtido de <https://serilog.net/>
- Silva, B. W. (2011). História do QFD. *História do QFD*. Obtido de <https://www.bwsconsultoria.com/2011/05/historia-do-qfd.html>
- Solanas et al. (November de 2017). Recent Advances in Healthcare Software. Obtido de <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8106887>
- Sommerville, I. (2011). *Engenharia de Software*.
- Statcounter. (2022). *Mobile Operating System Market Share Portugal*. Obtido de <https://gs.statcounter.com/os-market-share/mobile/portugal>
- Statista. (Fevereiro de 2022). Market share held by leading mobile internet browsers worldwide from January 2012 to December 2021. Obtido de <https://www.statista.com/statistics/263517/market-share-held-by-mobile-internet-browsers-worldwide/>
- Store, A. (s.d.). *Apple Store Developer*. Obtido de App Store: <https://developer.apple.com/support/compare-memberships/>
- Strategyzer. (4 de Janeiro de 2022). *Value Proposition Canvas – Download the Official Template*. Obtido de <https://www.strategyzer.com/canvas/value-proposition-canvas>
- SurveyLegend. (2022). *SurveyLegend*. Obtido de <https://www.surveylegend.com/>
- SurveyMonkey. (2022). *SurveyMonkey*. Obtido de <https://pt.surveymonkey.com/?>
- Swagger. (2022). *API Development*. Obtido de <https://swagger.io/>
- Totvs. (Março de 2020). *Como a gestão de documentos impulsiona as empresas*. Obtido de <https://www.totvs.com/blog/negocios/gestao-de-documentos/>
- Trends, D. (2021). Utilização da internet. Obtido de <https://www.hootsuite.com/pages/digital-trends-2021>
- TutorialsTeacher. (2022). *Dependency Injection*. Obtido de TutorialsTeacher: <https://www.tutorialsteacher.com/ioc/dependency-injection>
- Twitter. (s.d.). *Twitter*. Obtido de <https://twitter.com>
- Ward, A. (Fevereiro de 2021). How to Create a Progressive Web App. Obtido de <https://www.scorchsoft.com/blog/how-to-create-progressive-web-app/>
- WebAssembly. (2022). *WebAssembly*. Obtido de WebAssembly: <https://webassembly.org/>

Windows. (s.d.). *Windows*. Obtido de Windows: <https://www.microsoft.com/pt-pt/windows/?r=1>

YML. (Fevereiro de 2021). Why Progressive Web Apps Are The Future of Mobile Web. Obtido de <https://medium.com/ymedialabs-innovation/why-progressive-web-apps-are-the-future-of-mobile-web-5195f381d856>

Yoopies. (2021). A procura por apoio domiciliário a idosos cresce neste fim de ano. Obtido de <https://yooopies.pt/c/apoio-domiciliario-a-idosos-cresce-no-fim-do-ano>

Anexo A - Ficha de Contactos



VIDA CONFORTO
APOIO DOMICILIÁRIO


CONTATOS DOS FAMILIARES

CONTATOS DA VIDA CONFORTO

(Diretora Técnica)
António Ferreira (Diretor)
VidaConforto

Figura 61 - Ficha de Contactos

Anexo C - Ficha de Registo de Horas dos colaboradores




FOLHA DE TRABALHO

Nome: _____ Mês: _____

Utente: _____

Dia	Hora de Entrada	Hora de Saída	Nº de Horas
1			
2			
3			
4			
5			
6			
7			
8			
9			
10			
11			
12			
13			
14			
15			
16			
17			
18			
19			
20			
21			
22			
23			
24			
25			
26			
27			
28			
29			
30			
31			
Total Horas:			

_____ (Cliente) _____ (Funcionário)



FOLHA DE TRABALHO

Nome: _____ Mês: _____


Utente: _____

Dia	Hora de Entrada	Hora de Saída	Nº de Horas
1			
2			
3			
4			
5			
6			
7			
8			
9			
10			
11			
12			
13			
14			
15			
16			
17			
18			
19			
20			
21			
22			
23			
24			
25			
26			
27			
28			
29			
30			
31			
Total Horas:			

_____ (Cliente) _____ (Funcionário)

Figura 63 - Ficha de registo de horas

Anexo E - Ficha de Registo de Serviços (A)



VIDA CONFORTO
APOIO DOMICILIÁRIO

SERVIÇO DE APOIO DOMICILIÁRIO
Processo de Desenvolvimento Individual - Ficha de Registo de Serviços (A)


Utente: _____
Ano: _____ Mês: _____

Dia	Hora		Alimentação					AVD's					Outras Actividades										Hora Saída	Rubrica colaborador(o)							
			Preparação refeição	Req. Almoço	Almoço	Lanche	Jantar	Banho		Higiene Pessoal	Higiene Oral	Cuidado de Próteses	(B) Banho / (D) Depilação	(U) Unhas / (P) Pé / (C) Escar cabelo	Higiene Habitacional	Tratamento de Respas	Aquisição bens e géneros alimentícios	Acompanhamento ao Exterior	Acompanhamento a Consultas	Administração de medicação / prescrição médica	Proteções Ósseas	Creme Hidratante			Insulina	Glicemia	Tensão Arterial (1)				
								WC	Cama																		WC	Cama	S	D	
1	H	M																										H	M		
2	H	M																											H	M	
3	H	M																											H	M	
4	H	M																											H	M	
5	H	M																											H	M	
6	H	M																											H	M	
7	H	M																											H	M	
8	H	M																											H	M	
9	H	M																											H	M	
10	H	M																											H	M	
11	H	M																											H	M	
12	H	M																											H	M	
13	H	M																											H	M	
14	H	M																											H	M	
15	H	M																											H	M	
16	H	M																											H	M	
17	H	M																											H	M	
18	H	M																											H	M	
19	H	M																											H	M	
20	H	M																											H	M	
21	H	M																											H	M	
22	H	M																											H	M	
23	H	M																											H	M	
24	H	M																											H	M	
25	H	M																											H	M	
26	H	M																											H	M	
27	H	M																											H	M	
28	H	M																											H	M	
29	H	M																											H	M	
30	H	M																											H	M	
31	H	M																											H	M	

(1) - Valores normais: S - Sistólica (< 139); D - Diastólica (< 89)

Figura 65 - Ficha de registo de serviços (A)

Anexo F - Ficha de Registo de Serviços (B)



VIDA CONFORTO
APOIO DOMICILIÁRIO

SERVIÇO DE APOIO DOMICILIÁRIO
Processo de Desenvolvimento Individual - Ficha de Registo de Serviços (B)


Utente: _____
Ano: _____ Mês: _____

Dia	Hora		Eliminações / Conteúdo							Mobilizações			Posicionamentos e Transferências (1)										Hora Saída		Rubrica colaborador(a)									
			Apelo WC	Colocação de Fralda	Casca Fralda	Colocação de Pano	Fralda Seca	Urina	Feces	Urina e Feces	Número de mudas	Deitado	Levante	Dambulatório	Hora	Posição	Hora	Posição	Hora	Posição	Hora	Posição				Hora	Posição	Hora	Posição	Hora	Posição	Hora	Posição	
1	H	M																														H	M	
2	H	M																														H	M	
3	H	M																														H	M	
4	H	M																														H	M	
5	H	M																														H	M	
6	H	M																														H	M	
7	H	M																														H	M	
8	H	M																														H	M	
9	H	M																														H	M	
10	H	M																														H	M	
11	H	M																														H	M	
12	H	M																														H	M	
13	H	M																														H	M	
14	H	M																														H	M	
15	H	M																														H	M	
16	H	M																														H	M	
17	H	M																														H	M	
18	H	M																														H	M	
19	H	M																														H	M	
20	H	M																														H	M	
21	H	M																														H	M	
22	H	M																														H	M	
23	H	M																														H	M	
24	H	M																														H	M	
25	H	M																														H	M	
26	H	M																														H	M	
27	H	M																														H	M	
28	H	M																														H	M	
29	H	M																														H	M	
30	H	M																														H	M	
31	H	M																														H	M	

[1] DLD - Decúbito Lateral Direito; DLE - Decúbito Lateral Esquerdo; DD - Decúbito Dorsal; T - Transferências

Figura 66 - Ficha de registo de serviços (B)

Anexo G - Ficha de Registo de Serviços (C)



SERVIÇO DE APOIO DOMICILIÁRIO
 Processo de Desenvolvimento Individual - Ficha de Registo de Serviços (C)

Utente: _____
 Ano: _____ Mês: _____

Dia	Hora	Medicação						Data		Observações	Hora Saída	Rubrica colaborador (s)
		Aljum	Pequeno Almoço	Almoço	Lanche	Jantar	Deitar	Data	Hora			
1	H M										H M	
2	H M										H M	
3	H M										H M	
4	H M										H M	
5	H M										H M	
6	H M										H M	
7	H M										H M	
8	H M										H M	
9	H M										H M	
10	H M										H M	
11	H M										H M	
12	H M										H M	
13	H M										H M	
14	H M										H M	
15	H M										H M	
16	H M										H M	
17	H M										H M	
18	H M										H M	
19	H M										H M	
20	H M										H M	
21	H M										H M	
22	H M										H M	
23	H M										H M	
24	H M										H M	
25	H M										H M	
26	H M										H M	
27	H M										H M	
28	H M										H M	
29	H M										H M	
30	H M										H M	
31	H M										H M	

Figura 67 - Ficha de registo de serviços (C)

Anexo H - Ficha de Inscrição do Utente

FICHA DE INSCRIÇÃO		
DATA	NÚMERO	
CARACTERIZAÇÃO PESSOAL	NOME	
	MORADA	
	CÓDIGO POSTAL	-
	TELEFONE	TELEMÓVEL
	ESTADO CIVIL	NASCIMENTO
	IDADE	NATURALIDADE
	PROFISSÃO	
	BI / CC	NIF
	NISS	SNS
	PESSOA (S) PRÓXIMA (S)	NOME (filha)
MORADA		
CÓDIGO POSTAL		-
TELEFONE		TELEMÓVEL
PARENTESCO		
PROFISSÃO		
E-MAIL		
PESSOA (S) PRÓXIMA (S)	NOME (outro familiar)	
	MORADA	
	CÓDIGO POSTAL	-
	TELEFONE	TELEMÓVEL
	PARENTESCO	NIF
	PROFISSÃO	
	E-MAIL	

Figura 68 - Ficha de inscrição do utente

	2ª a 6ª	Sábado	Domingo	Feriados	Nº vezes/dia			
					1 X	2 X	3 X	Outra
SERVIÇOS SOLICITADOS	Cuidados de Higiene e Conforto Pessoal	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	Higiene Habitacional	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	Preparação de Refeições no Domicílio e Acompanhamento de Refeições	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	Compras e Entregas ao domicílio/Aquisição de bens e serviços					<input type="checkbox"/>		
	Acompanhamento a consultas/tratamento					<input type="checkbox"/>		
	Serviço Helpphone					<input type="checkbox"/>		

DADOS RECOLHIDOS	N DE IDENTIFICAÇÃO FISCAL	<input type="checkbox"/>
		<input type="checkbox"/>
		<input type="checkbox"/>
		<input type="checkbox"/>
		<input type="checkbox"/>
		<input type="checkbox"/>
		<input type="checkbox"/>
		<input type="checkbox"/>

AGREGADO FAMILIAR	
UTENTE	<input type="checkbox"/>
CÔNJUGE	<input type="checkbox"/>

MÉDICO ASSISTENTE

Nota:

1. A VidaConforto garante a total confidencialidade dos dados recolhidos junto dos clientes
2. Autorizo a utilização dos meus dados pessoais que constam desta ficha, na situação a cima descrita ou em situações devidamente fundamentadas.
3. Teve conhecimento do Regulamento Interno

Candidato:

Diretor/a:

Figura 69 - Ficha de inscrição do utente

Anexo I - Questionário de qualidade



SERVIÇO DE APOIO DOMICILIÁRIO

Por favor, dedique alguns minutos a responder a este pequeno questionário. As suas respostas serão trabalhadas de forma confidencial.

Utente:	
Colaborador(a):	DATA: / /
Direção Técnica:	

Por cada item identificado abaixo, faça p.f. um círculo no número no lado direito que melhor se adequar à sua opinião sobre a qualidade do serviço prestado. Utilize a escala para selecionar o nível da sua satisfação qualitativa.

DESCRIÇÃO DA COMPETÊNCIA DOS PROFISSIONAIS NO DESEMPENHO DA PRESTAÇÃO DOS SERVIÇOS	Escala				
	Pobre	Boa			Excelente
1. Como avalia a apresentação e postura do(a) Colaborador(a)	1	2	3	4	5
2. Como avalia a técnica e a humanização utilizada na higiene pessoal do beneficiário	1	2	3	4	5
3. Como avalia a colaboração do(a) Colaborador(a) com as ideias do beneficiário	1	2	3	4	5
4. Como avalia a atitude do(a) Colaborador(a) perante um imprevisto	1	2	3	4	5
5. Como avalia o perfil do(a) Colaborador(a) tendo em conta o desempenho das suas funções	1	2	3	4	5
6. Como avalia a pontualidade e assiduidade do(a) Colaborador(a)	1	2	3	4	5
7. Os colaboradores vestem-se de forma apropriada ao desempenho das suas funções	1	2	3	4	5
8. Como avalia o relacionamento do(a) Colaborador(a) com o beneficiário e os seus familiares	1	2	3	4	5
9. De modo geral como avalia a competência do(a) Colaborador(a)	1	2	3	4	5
10. Do relacionamento que tem/teve com a Diretora Técnica , como avalia a sua competência	1	2	3	4	5
11. Como avalia a frequência das visitas realizadas pela Diretora Técnica	1	2	3	4	5
12. Como avalia o desempenho e a qualidade do Back-office (administrativo e comercial) da VidaConforto	1	2	3	4	5
13. Como avalia a relação qualidade/preço dos serviços prestados da VidaConforto	1	2	3	4	5
14. De modo geral como avalia a VidaConforto	1	2	3	4	5
15. Recomendaria a VidaConforto ?	Sim			Não	

Dada que a sua opinião é importante, pretende-se com o presente questionário somente corrigir eventuais procedimentos menos corretos, bem como a atribuição de maior responsabilização aos profissionais da **VidaConforto**. Toda a informação prestada tem carácter sigiloso.

Figura 70 - Questionário de Qualidade

Anexo J - Casos de Uso

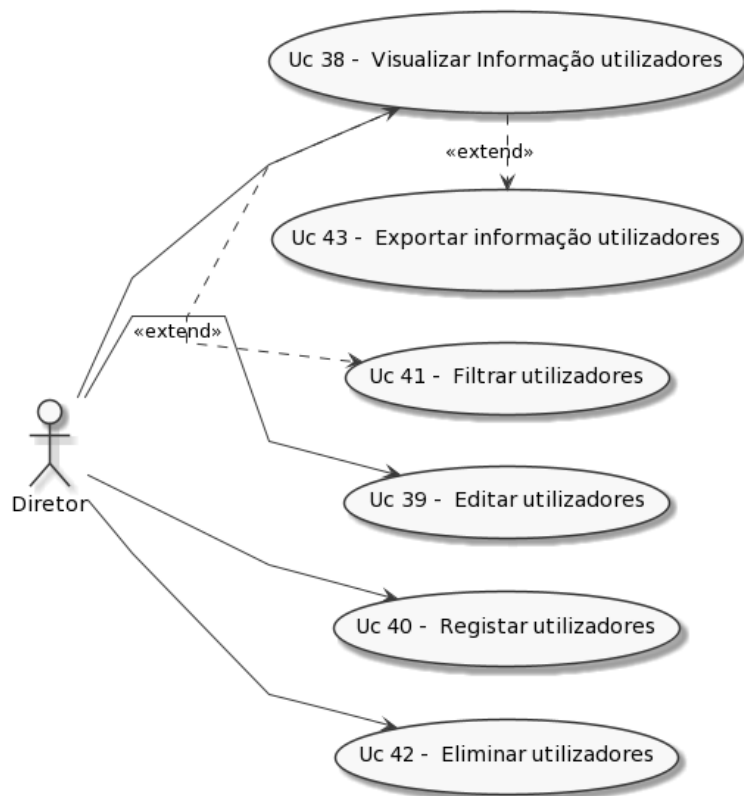


Figura 71 - Gerir Utilizadores

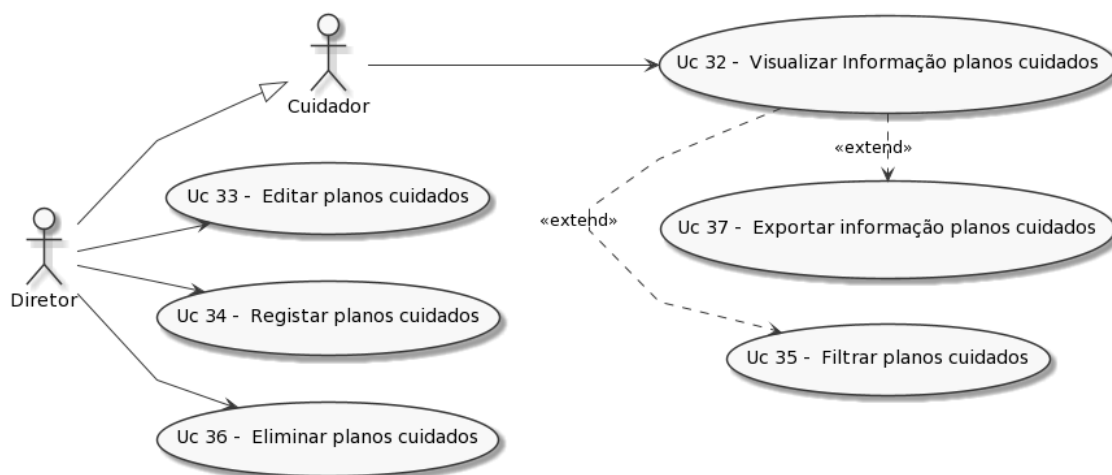


Figura 72 - Gestão de plano de cuidados

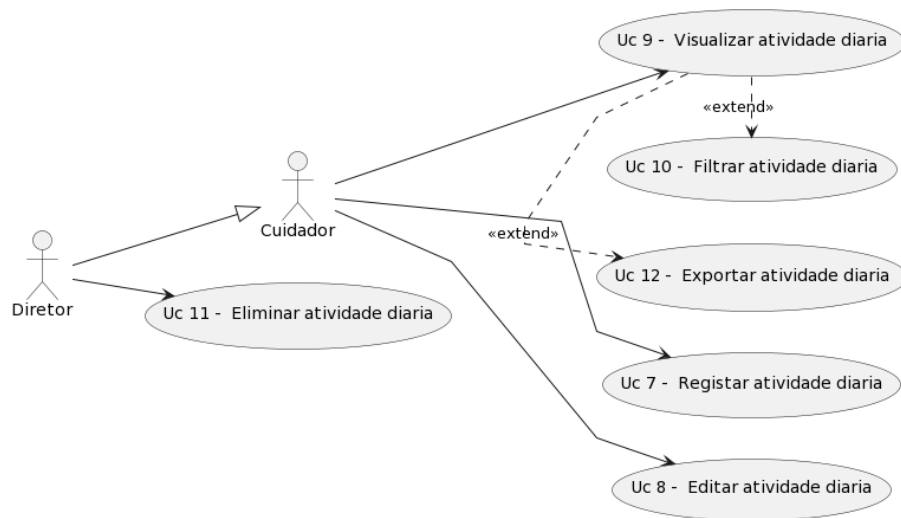


Figura 73 - Gestão Atividade Diária

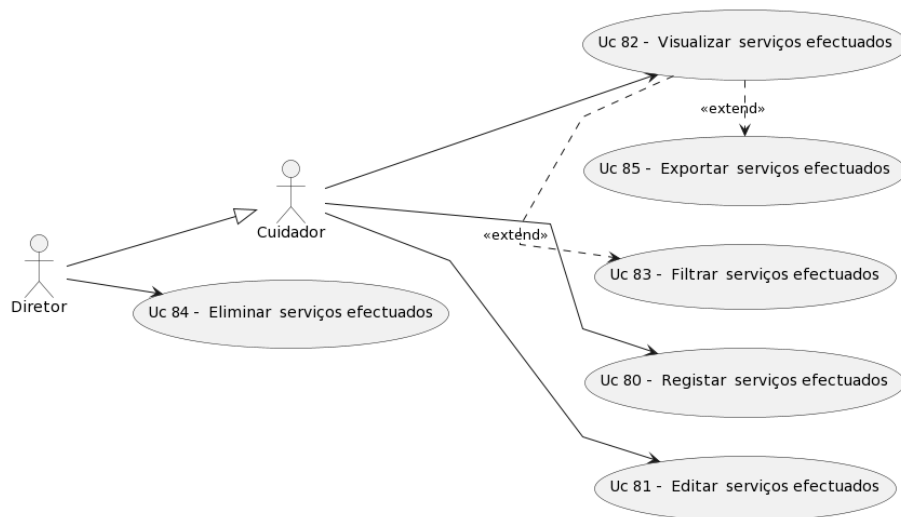


Figura 74 - Gestão de Serviços

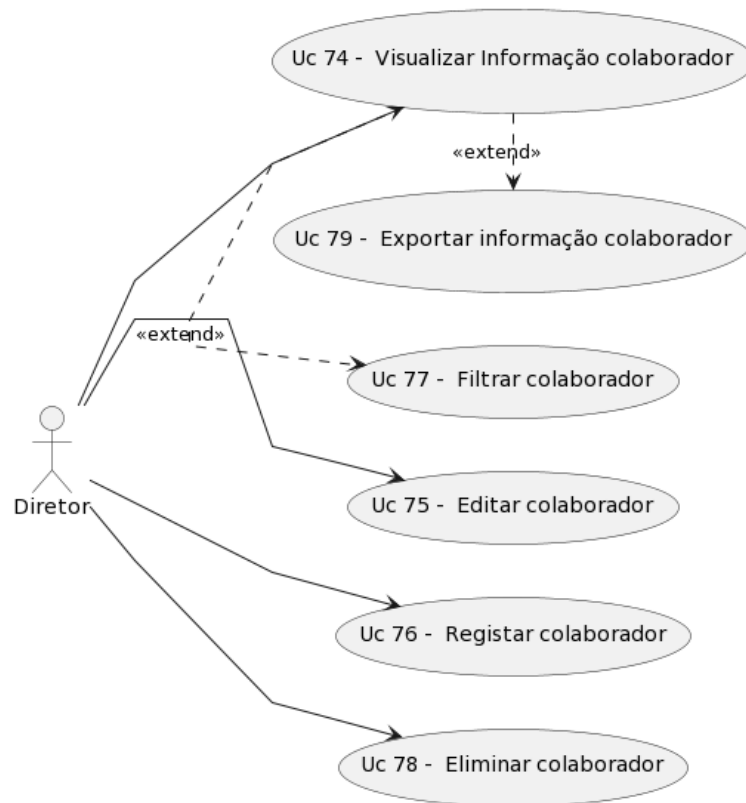


Figura 75 - Gestão de Colaboradores

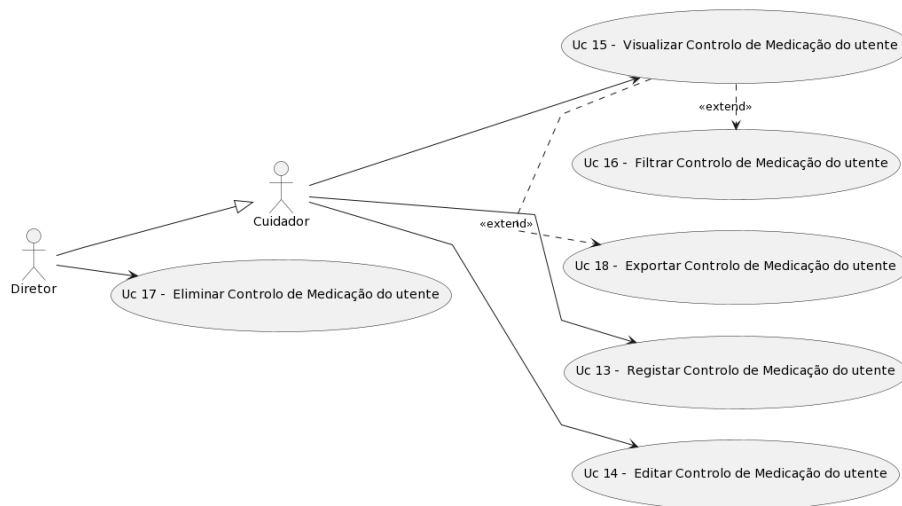


Figura 76 - Gestão de Controlo de Medicação

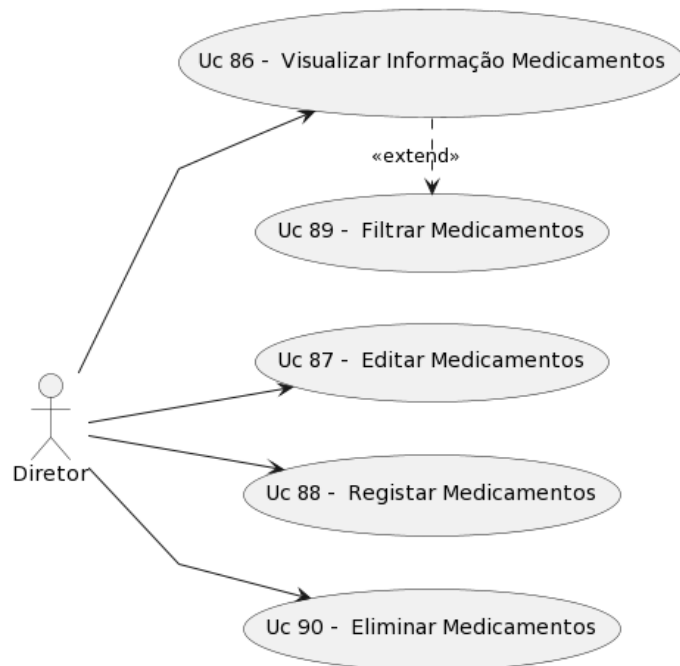


Figura 77 - Gestão de Medicamentos

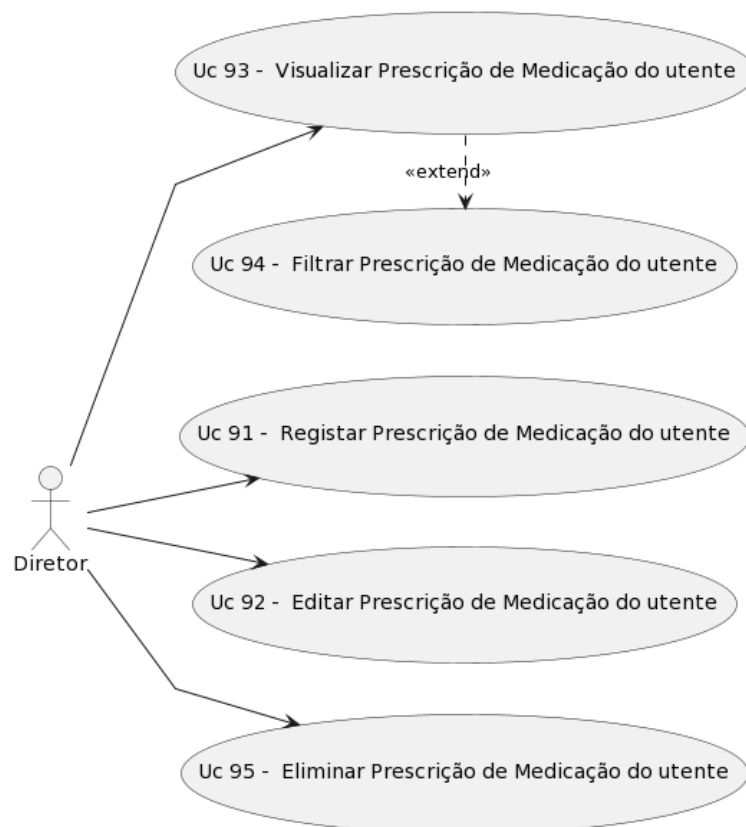


Figura 78 - Gestão de Prescrição de Medicamentos

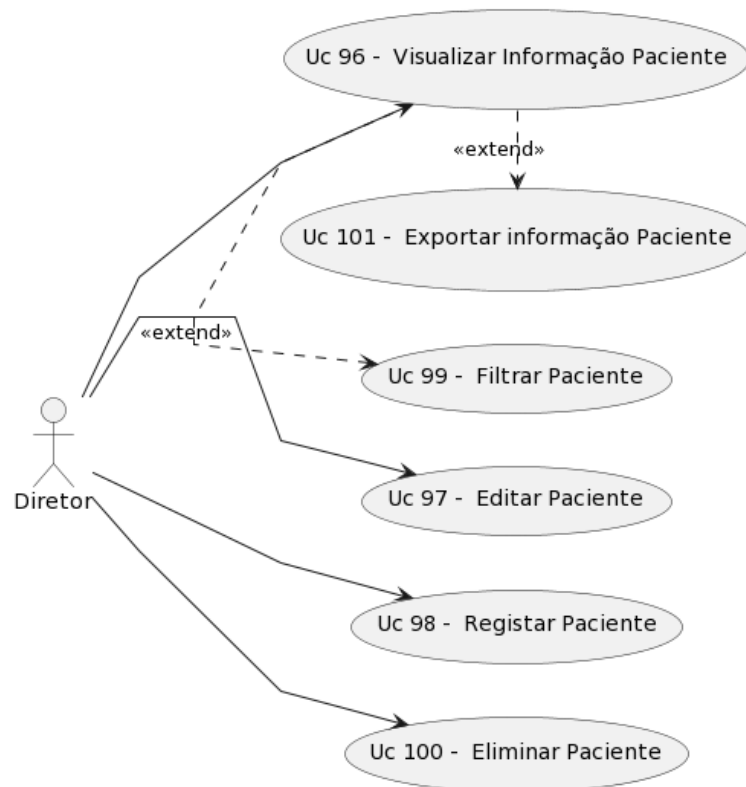


Figura 79 - Gestão de Paciente

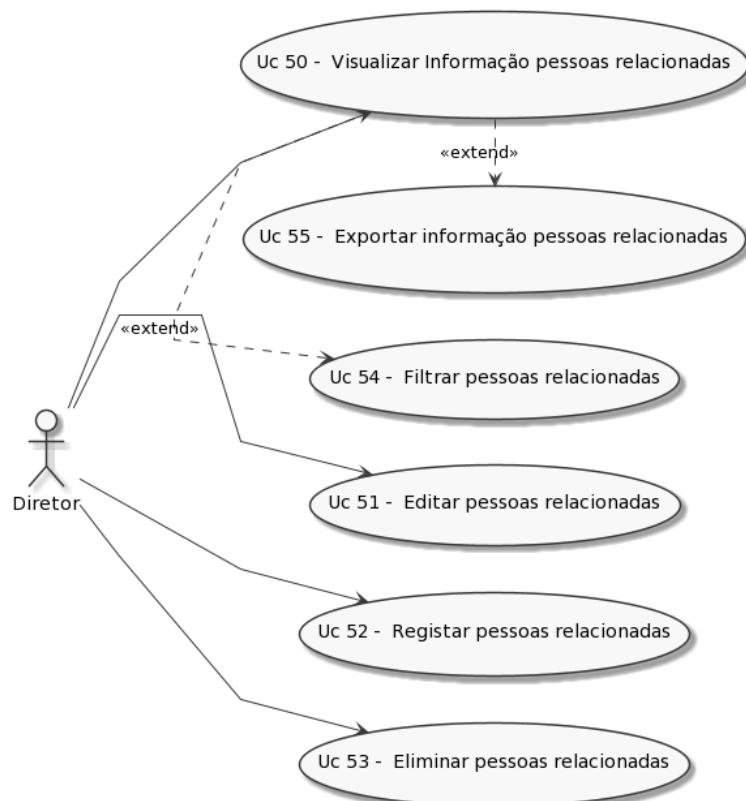


Figura 80 - Gestão de Pessoas Relacionadas

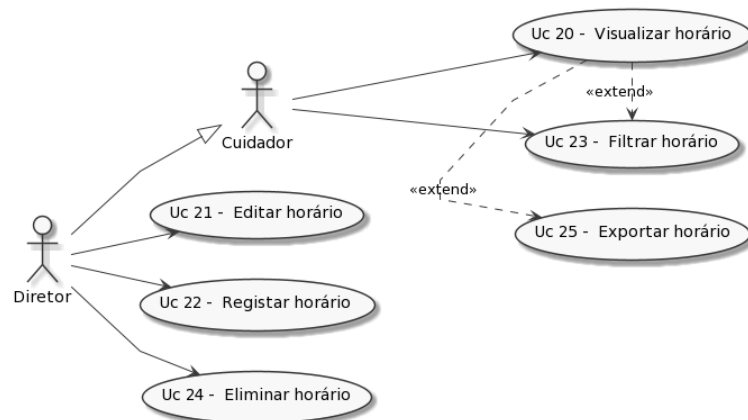


Figura 81 - Gestão de Horário

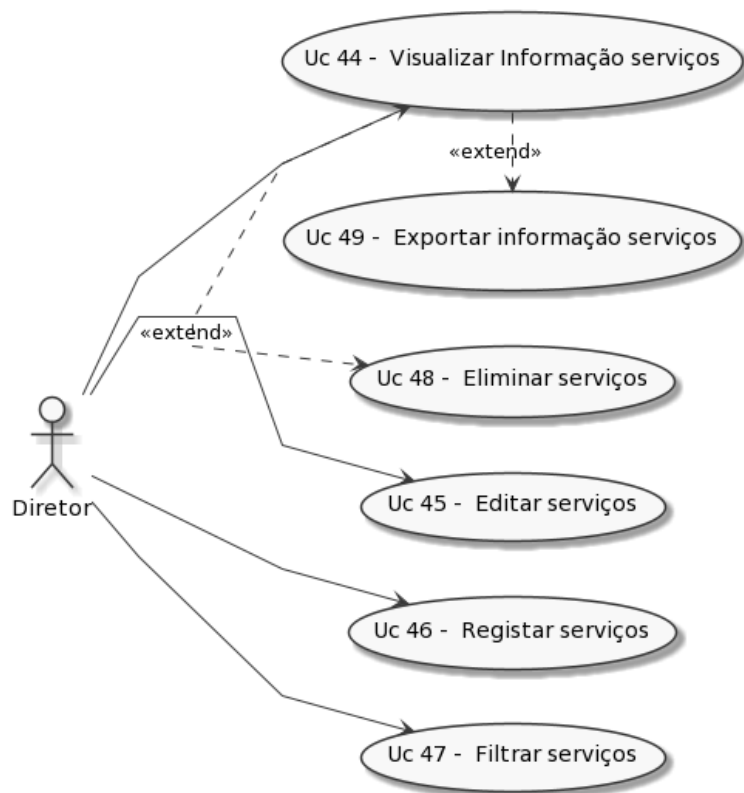


Figura 82 - Gestão de Serviços

Anexo K - Quantitative Evaluation Framework (QEF)

Tabela 112 - Quantitative Evaluation Framework

q	D	ci	Dimensões	Qi	Wj (Peso do factor j in Dim ij) [0,1]	Factores	rwk (Peso do requisito k in Factor j) {2, 4, 6, 8, 10}	Requisitos	wf _k % Cumprimento do requisito k) [0,100]		
5%	1,33	1,775	Funcionalidade	2,88462	0,62	Funcional (Casos de Uso)	6	FF01 - Utilizador faz login	50		
							4	FF02 - Utilizador gere a atividade diária	0		
							4	FF03 - Utilizador gere o controlo de medicação	0		
							8	FF04 - Utilizador visualiza e exporta estatísticas	0		
							8	FF05 - Utilizador gere utilizadores	0		
							8	FF06 - Utilizador gere informação utentes	0		
							8	FF07 - Utilizador gere horário	0		
							8	FF08 - Utilizador regista medicação	0		
							6	FF09 - Utilizador gere ocorrências	0		
							8	FF10 - Utilizador preenche questionários	0		
							4	FF11 - Utilizador gere pessoas relacionadas	0		
							4	FF12 - Utilizador gere plano medicação utente	0		
							2	FF13 - Utilizador gere planos de cuidados	0		
							8	FF14 - Utilizador gere questionários	0		
							8	FF15 - Utilizador exporta dados	0		
							10	FF16 - Utilizador gere serviços	0		
						10	FUI01 - Aplicação é intuitiva	0			
						10	FUI02 - Aplicação tem uma boa experiência de design	0			
						8	FUI03 - Organização de conteúdos	0			
						6	FUI04 - Aplicação tem um bom controlo e navegação	0			
						8	FUI05 - Permissões de acessos	0			
						4	FUI06 - Modo offline	0			
						10	FUI07 - Aplicação tem pesquisa	0			
						10	FCD01 - Organização de informação	0			
						6	FCD02 - Textos bem redigidos e concisos	0			
						8	FCD04 - Conteúdos relacionados com o tema	0			
						0	0,20	Tempo de carregamento	8	EAC01 - As páginas carregam rápido	0
						10	Eficiência	12,5	0,80	Navegação	8
8	EN02 - Interface rápida e responsiva, com informações de progresso sempre que relevantes	0									
8	EN03 - Aplicação sem erros, e erros inesperados devem ser bem tratados	0									
8	EN04 - Autenticação	50									

Anexo L – Inquérito de Satisfação do Utilizador

Inquérito de Satisfação do Utilizador

Com este questionário pretende-se recolher a opinião sobre a experiência de utilização do Utilizador com a aplicação web progressiva GesSad

Cargo/Função

Paciente
 Administrador
 Técnico de Saúde

Idade

A sua resposta _____

Usabilidade da Aplicação

Selecione a opção que melhor descreve a aplicação

	Discordo Totalmente	Discordo parcialmente	Indiferente	Concordo parcialmente	Concordo totalmente
A aplicação apresenta a informação de forma organizada	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
A aplicação tem uma apresentação gráfica boa	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
A interação com a aplicação é agradável	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
A navegação entre menus é agradável	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
O registo da informação é simples	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
A aplicação atende as suas necessidades	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Recomenda o uso da aplicação	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Funcionalidade

Selecione a opção que melhor descreve a aplicação

	Discordo Totalmente	Discordo parcialmente	Indiferente	Concordo parcialmente	Concordo totalmente
A aplicação facilita o registo das atividades/tarefas realizadas	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
A aplicação facilita o registo das ocorrências	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
A informação registada está acessível imediatamente	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
A informação disponibilizada é o suficiente	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Adaptabilidade

Para cada frase indique a dificuldade que teve

	Muito difícil	Difícil	Fácil	Muito Fácil	Não conheço
Registo e acesso das ocorrências	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Registo e acesso das tarefas realizadas	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Acesso à informação dos pacientes	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Em geral, como considera o nível da aplicação?	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

[Obter link](#)

Nunca entre palavras-passe através dos Google Forms.
 Este conteúdo não foi criado nem aprovado pela Google. Denunciar abuso - Termos de Utilização - Política de privacidade

Preencha previamente as respostas e, em seguida, clique em "Obter link".

Google Formulários

Figura 83 - Inquérito de Satisfação