



UNIVERSIDADE
LUSÓFONA

Sistema para Monitorização de Plantas - Frontend e Backend

Trabalho Final de Curso

Relatório Final

Nome do Aluno: Pedro Miguel Martins Rocha - 22003766

Nome do Orientador: Professor João Pedro Leal Abalada de Matos Carvalho

Trabalho Final de Curso | LEI | 28/06/2023

www.lusofona.pt

Direitos de cópia

(Sistema para Monitorização de Plantas - Frontend e Backend), Copyright de (Pedro Miguel Martins Rocha), ULHT.

A Escola de Comunicação, Arquitetura, Artes e Tecnologias da Informação (ECATI) e a Universidade Lusófona de Humanidades e Tecnologias (ULHT) têm o direito, perpétuo e sem limites geográficos, de arquivar e publicar esta dissertação através de exemplares impressos reproduzidos em papel ou de forma digital, ou por qualquer outro meio conhecido ou que venha a ser inventado, e de a divulgar através de repositórios científicos e de admitir a sua cópia e distribuição com objetivos educacionais ou de investigação, não comerciais, desde que seja dado crédito ao autor e editor.

Resumo

O presente documento resulta de um projeto a ser desenvolvido no contexto de Trabalho Final de Curso (TFC) do autor, que tem por título ‘Sistema para Monitorização de Plantas - Frontend e Backend’. Este projeto incide no desenvolvimento de uma aplicação web que permite suportar e facilitar a monitorização de diversas plantas, permitir a existência de apoio e controlo detalhado na gestão dessas mesmas plantas, seja numa pequena horta ou em campos com maior dimensão de forma sustentável.

Diversas aplicações têm surgido no âmbito da internet das coisas e no que toca à componente de monitorização do mundo que nos rodeia tal não é exceção. Diversos estudos abordam, por exemplo, o tema da importância do aumento da eficiência de rega perante um cenário de progressiva escassez de água entre o setor da agricultura e o consumo urbano. Neste projeto pretende-se implementar um website de monitorização de plantas a partir de tecnologias frontend (HTML5, CSS3 e JavaScript) com tecnologias backend (Python e *framework* Django).

Este sistema tem como objetivo ajudar pessoas que queiram ter uma pequena plantação em casa poderem ser acompanhadas a nível de quantidades de água, humidade, fertilizantes e herbicidas, deteção de pragas e formas sustentáveis que cada planta possa ser plantada pela pessoa que cultiva.

Partindo deste conhecimento são detetadas algumas falhas e problemas nestes sistemas e é procurado responder com soluções apropriadas e adequadas. O produto final deste projeto visa uma aplicação web, que permite aos utilizadores estarem constantemente informados sobre os procedimentos que cada planta deve ter permitindo que haja organização e estruturação, com o objetivo de haver um melhor cuidado a ter com as plantas, evitando, assim, a degradação das mesmas.

O sistema desenvolvido vai ter como base a facilitação da partilha de informações para com os utilizadores, ajudando na poupança de recursos como, por exemplo, a água, uma vez que é um bem essencial fazendo uma correta gestão de modo que não haja desperdícios.

Este TFC é um complemento do TFC DEISI270 do colega Tiago Rodrigues (Sistema de Monitorização de Plantas), onde o TFC do Tiago corresponde à recolha de dados relativamente às plantas como, por exemplo, a quantidade de humidade e a temperatura.

E, portanto, este TFC corresponde a uma aplicação web que através da recolha dos dados, estes serão apresentados ao utilizador para poder fazer a monitorização correta das plantas que decide cuidar.

Palavras-chave: Monitorização, Escassez de Água, Sustentabilidade, Plantas, Facilitar Partilha de Informação.

Abstract

This document is the result of a project to be developed in the context of the author's Final Course Work (FCW), which is entitled 'Plant Monitoring System - Frontend and Backend'. This project focuses on the development of a web application that allows supporting and facilitating the monitoring of different plants, allowing the existence of support and detailed control in the management of these same plants, whether in a small garden or in larger fields in a sustainable way.

Several applications have emerged within the scope of the internet of things and when it comes to monitoring the world around us, this is no exception. Several studies address, for example, the issue of the importance of increasing irrigation efficiency in a scenario of progressive water scarcity between the agriculture sector and urban consumption. In this project we intend to implement a plant monitoring website from frontend technologies (HTML5, CSS3 and JavaScript) with backend technologies (Python and Django *framework*).

This system aims to help people who want to have a small plantation at home to be able to monitor the amount of water, humidity, fertilizers and herbicides, pest detection and sustainable ways that each plant can be planted by the person who cultivates.

Based on this knowledge, some flaws and problems in these systems are detected and an attempt is made to respond with appropriate and adequate solutions. The final product of this project aims at a web application, which allows users to be constantly informed about the procedures that each plant must have, allowing for organization and structuring, with the aim of taking better care of the plants, thus avoiding their degradation.

The developed system will be based on facilitating the sharing of information with users, helping to save resources such as, for example, water, since it is an essential good, managing it correctly so that there is no waste.

This TFC is a complement to TFC DEISI270 by colleague Tiago Rodrigues (Plant Monitoring System), where Tiago's TFC corresponds to the collection of data on plants, such as, for example, the number of fertilizers and sun exposure.

And, therefore, this TFC corresponds to a web application that, through the collection of data, these will be presented to the user to be able to correctly monitor the plants that he decides to care for.

Keywords: Monitoring, Water Scarcity, Sustainability, Plants, Facilitating Information Sharing.

Índice

Resumo	iii
Abstract	v
Índice	vii
Lista de Figuras	ix
Lista de Tabelas	xi
1 Identificação do Problema	1
2 Viabilidade e Pertinência.....	2
3 Benchmarking.....	3
4 Engenharia.....	6
4.1 Levantamento e Análise de Requisitos.....	6
4.1.1 Requisitos técnicos	6
4.1.2 Requisitos funcionais.....	7
4.1.3 Requisitos não funcionais.....	8
4.2 Diagrama de Casos de Uso.....	9
4.3 Diagrama de Atividades	10
4.4 Modelos Relevantes.....	11
4.4.1 Diagrama Entidade-Relação	11
4.4.2 Diagrama de Classes	12
4.4.3 Diagrama de Sequência	13
4.5 Estrutura em árvore.....	14
4.6 Mockups.....	14
5 Solução Proposta.....	19
5.1 Introdução.....	19
5.2 Arquitetura.....	19
5.3 Tecnologias e Ferramentas Utilizadas	21
5.3.1 <i>Frontend</i>	21
5.3.2 <i>Backend</i>	21
5.4 Implementação	22
5.5 Abrangência	22

6	Método e Planeamento	23
6.1	Calendário <i>Frontend</i> e <i>Backend</i>	24
6.1.1	Calendário <i>Frontend</i>	24
6.1.2	Calendário <i>backend</i>	25
6.2	Desenvolvimento do Relatório	26
7	Resultados	28
8	Conclusões e Trabalhos Futuros	30
8.1	Conclusões	30
8.2	Trabalhos Futuros	31
	Bibliografia.....	32
	Anexo 1 – Estudo de viabilidade, pertinência e relevância.....	33
	Anexo 2 – Resultados do questionário	38
	Anexo 3 – Mockups, story board e mapa aplicacional.....	42
	Anexo 4 – Progresso de trabalho	45
	Anexo 5 – Protótipo Interativo	48
	Anexo 6 – Plano de Testes e Validação	52
	Glossário	55

Lista de Figuras

Figura 1 – Plantix	3
Figura 2 – Picture This	4
Figura 3 – WaterMe - Gardening Reminders	4
Figura 4 – Diagrama de Casos de Uso.....	9
Figura 5 – Diagrama de Atividades	10
Figura 6 – Diagrama Entidade-Relação.....	11
Figura 7 – Diagrama de Classes	12
Figura 8 – Diagrama de Sequência (<i>login</i>).....	13
Figura 9 – Estrutura em Árvore	14
Figura 10 – Página Dashboard1	15
Figura 11 – Página Dashboard2	15
Figura 12 – Tela Início1.....	16
Figura 13 – Tela Início2.....	16
Figura 14 – Página Plantas.....	17
Figura 15 – Tela O Meu Jardim.....	17
Figura 16 – Tela Notificações	18
Figura 17 – Tela Gráfica Água.....	18
Figura 18 – Model View Controller (MVC)	19
Figura 19 – Model View Template (MVT).....	20
Figura 20 – Calendário proposto em formato Gantt.....	23
Figura 21 – Calendário Gantt (frontend)	24
Figura 22 – Calendário Gantt (backend).....	25
Figura 23 – Questionário (Introdução)	33
Figura 24 – Pergunta 1 (Dados Demográficos).....	33
Figura 25 – Pergunta 2 (Dados Demográficos).....	34
Figura 26 – Pergunta 3 (Dados Demográficos).....	34
Figura 27 – Pergunta 1 (Viabilidade e Pertinência do Trabalho).....	35
Figura 28 – Pergunta 2 (Viabilidade e Pertinência do Trabalho).....	35
Figura 29 – Pergunta 3 (Viabilidade e Pertinência do Trabalho).....	36
Figura 30 – Pergunta 4 (Viabilidade e Pertinência do Trabalho).....	36
Figura 31 – Pergunta 5 (Viabilidade e Pertinência do Trabalho).....	36
Figura 32 – Pergunta 6 (Viabilidade e Pertinência do Trabalho).....	37
Figura 33 – Pergunta 7 (Viabilidade e Pertinência do Trabalho).....	37
Figura 34 – Pergunta 8 (Viabilidade e Pertinência do Trabalho).....	37
Figura 35 – Resposta Pergunta 1 (Dados Demográficos)	38
Figura 36 – Resposta Pergunta 2 (Dados Demográficos)	38
Figura 37 – Resposta Pergunta 3 (Dados Demográficos)	39
Figura 38 – Resposta pergunta 1 (Viabilidade e Pertinência do Trabalho).....	39
Figura 39 – Resposta pergunta 2 (Viabilidade e Pertinência do Trabalho).....	39

Figura 40 – Resposta pergunta 3 (Viabilidade e Pertinência do Trabalho).....	40
Figura 41 – Resposta pergunta 4 (Viabilidade e Pertinência do Trabalho).....	40
Figura 42 – Resposta pergunta 5 (Viabilidade e Pertinência do Trabalho).....	40
Figura 43 – Resposta pergunta 6 (Viabilidade e Pertinência do Trabalho).....	41
Figura 44 – Resposta pergunta 7 (Viabilidade e Pertinência do Trabalho).....	41
Figura 45 – Resposta pergunta 8 (Viabilidade e Pertinência do Trabalho).....	41
Figura 46 – Página de Login.....	42
Figura 47 – Splashscreen.....	42
Figura 48 – Página Previsão do Tempo.....	43
Figura 49 – Detalhe da Planta.....	43
Figura 50 – Storyboard.....	44
Figura 51 – Mapa Aplicacional.....	44
Figura 52 – Calendário Gantt Relatório Intercalar 1º Semestre.....	45
Figura 53 – Calendário Gantt Relatório Intermédio.....	46
Figura 54 – Calendário Gantt Relatório Intercalar 2º Semestre.....	47
Figura 55 – Página de Login (Protótipo).....	48
Figura 56 – Página Início (Protótipo).....	49
Figura 57 – Página Plantas (Protótipo).....	49
Figura 58 – Página O Meu Jardim (Protótipo).....	50
Figura 59 – Ecrã Perfil (Protótipo).....	50
Figura 60 – Meteorologia onde se encontra o Utilizador (Protótipo).....	51
Figura 61 – Página Ver Mais Plantas (Protótipo).....	51
Figura 62 – Detalhe da Planta (Protótipo).....	51

Lista de Tabelas

Tabela 1 – Análise benchmarking.....	5
Tabela 2 – Requisitos Técnicos	6
Tabela 3 – Requisitos Funcionais.....	7
Tabela 4 – Requisitos Não Funcionais	8
Tabela 5 – Resultados.....	29
Tabela 6 – Testes para Solução Desenvolvida.....	53

1 Identificação do Problema

A transformação digital surgiu com o intuito de facilitar e automatizar processos tecnológicos ao nível de todas as vertentes que todos conhecemos, desde o mundo empresarial, ensino, entre outros. O processo de monitorização de plantas é algo que está ainda em fase de crescimento relativamente ao mundo digital ao qual, enquanto cidadãos comuns, que utilizam estas aplicações como forma de agilizar a resolução para alguns problemas diários, sentimos uma incerteza relativamente à fiabilidade dos dados apresentados.

Deste modo, perante esta incerteza, é necessário fazer um levantamento de problemas e possíveis oportunidades e, de certo modo, poder inverter a situação atual. Assim, identificou-se uma oportunidade de melhoria no que diz respeito à monitorização de todo o tipo de plantas. Por exemplo, temos ou tivemos algum tipo de planta nas nossas casas e, por algum motivo, de um dia para o outro esta murchou e acabou mesmo por morrer. Ou porque colocámos água a mais e era uma planta que precisava de pouca água ou vice-versa, ou precisava de humidade e no sítio onde a coloquei não havia humidade suficiente. Não sabemos qual a resposta a este problema. Isto poderá ser recorrente e leva a uma certa desmotivação por parte do cuidador das plantas.

Posto isto, e por forma a resolver todos os potenciais problemas decorrentes do que foi acima referido, é importante o desenvolvimento de uma aplicação web com o intuito de ajudar todas as pessoas interessadas que queiram ter uma plantação de pequena, média ou de grande dimensão na sua residência de forma a cuidar das suas plantas de forma sustentável, uma vez que a incorreta gestão de água nas plantas e/ou utilização de fertilizantes e herbicidas de forma errada poderá levar à degradação das plantas e, conseqüentemente, à sua “morte”.

A implementação desta nova funcionalidade contribuirá para a resolução de problemas que as pessoas que cuidam de plantas se veem confrontadas e poderem, de certo modo, terem um nível de confiança e fiabilidade maior sobre os dados que são apresentados aos utilizadores e, assim, fazer uma “agricultura” de forma sustentável e também se poderá escalar depois na agricultura de precisão.

2 Viabilidade e Pertinência

O projeto em questão tem capacidades para garantir viabilidade de desenvolvimento para além do âmbito de trabalho académico, visto que existe a possibilidade de implementação e criação de novas plantas que possam ser descobertas no futuro.

Nesta fase inicial, o projeto tem como objetivo dar ao utilizador a conhecer os diversos tipos de plantas que existem, bem como dicas para ajudar na culturação, pragas e doenças associadas a cada tipo, saber de quanto em quanto tempo as plantas devem ser regadas, de modo a fazer uma gestão correta da utilização da água bem como um desenvolvimento e/ou crescimento da planta de forma mais sustentável.

Como referido anteriormente, nesta fase a aplicação web tem como principal objetivo responder a questões e necessidades de pessoas que tenham plantas em casa para cuidar de forma mais sustentável, utilizando os recursos de forma correta, nas quantidades certas de modo a não existir falta/excesso dos mesmos. Com isto, o crescimento e a manutenção de cada planta serão de forma constante, uma vez que é fornecida toda a informação para o utilizador poder seguir, evitando oscilações no desenvolvimento.

A metodologia aplicada no desenvolvimento deste projeto permite que o mesmo seja altamente adaptável a qualquer tipo de planta, permitindo assim um uso mais universal desta aplicação web.

Realizou-se um questionário anónimo (com auxílio da plataforma Google Forms) a várias pessoas que pudessem dar feedback face aos conhecimentos que possuem em termos de facilidade no acesso a informação em relação à monitorização de plantas [Anexo 1 – Estudo de viabilidade, pertinência e relevância].

No total participaram 68 pessoas, de onde se pode concluir que um website relacionado com monitorização de plantas que respondam aos problemas acima apresentados seriam uma mais valia para quem cuida/trata de todo o tipo de plantas ou planeia fazê-lo no futuro [Anexo 2 – Resultados do questionário].

3 Benchmarking

Através de uma análise de mercado é possível averiguar que atualmente existe pouca oferta no que diz respeito a aplicações relativamente a monitorização de plantas, no entanto, através de influencers digitais, são cada vez mais pessoas a querer ter um espaço dedicado para as suas plantas. Por outro lado, a nível de Web, após uma pesquisa intensa e exaustiva, a conclusão é de que não existe nenhum website relacionado com este tema de TFC.

Posto isto, após procurar e analisar os aspetos mais diferenciadores de algumas aplicações e refletir melhorias para a minha solução na realização do website/aplicação web.

Vamos tomar como exemplo soluções como as aplicações “Plantix”, “Picture This” e “WaterMe - Gardening Reminders”.

Plantix [Figura 1] [Plan22] é uma aplicação móvel de consultoria agrícola para agricultores e jardineiros. O Plantix foi desenvolvido pela PEAT GmbH (*Gesellschaft mit beschränkter Haftung*), uma start-up de IA (Inteligência Artificial) com sede em Berlim. A aplicação afirma diagnosticar danos de pragas, doenças de plantas e deficiências de nutrientes que afetam as culturas e oferece medidas de tratamento correspondentes.

Os utilizadores podem participar da comunidade online onde encontram cientistas, agricultores e especialistas em plantas para discutir questões relativamente à saúde vegetal. Os agricultores podem visualizar o clima local, obter bons conselhos agrícolas durante todo o ano e receber alertas de doenças quando uma doença se espalha na sua zona e nos seus arredores.

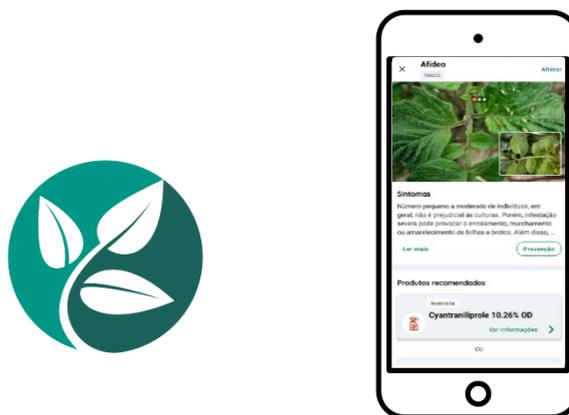


Figura 1 – Plantix

Picture This [Figura 2] [PiTh22] é uma aplicação móvel para identificar plantas disponível para download em telemóveis Android e iOS. A app utiliza inteligência artificial para reconhecer mais de 10 mil espécies de plantas, entre flores, árvores, suculentas e catos, a partir da câmara do telemóvel.

Após fazer a captura, o *software* traz o nome da espécie, imagens e informações extras sobre ela, como dicas de cuidados. A aplicação ainda é capaz de analisar plantas e diagnosticar possíveis doenças ou problemas.

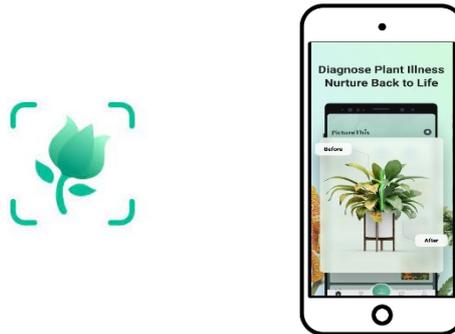


Figura 2 – Picture This

WaterMe - Gardening Reminders [Figura 3] [WaGa22] é uma aplicação móvel para o sistema operativo iOS que permite adicionar as suas plantas na aplicação, lembretes para regar e fertilizar. O utilizador recebe uma notificação todos os dias a relembrar quais são as plantas que precisam de cuidados.

A aplicação suporta vários tipos de lembretes por planta, lembra para regar, fertilizar, aparar e muito mais, é fácil de ver quais são e quando as plantas precisam de ser cuidadas. A app WaterMe sincroniza os dados em todos os seus dispositivos iOS do utilizador via iCloud suportando trabalhar rapidamente com interface de arrastar e soltar.

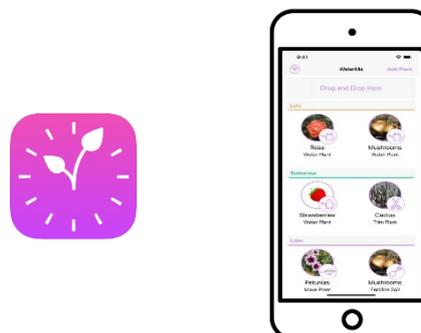


Figura 3 – WaterMe - Gardening Reminders

Na seguinte tabela [Tabela 1] é possível analisar de uma forma mais sistemática algumas funcionalidades das aplicações existentes no mercado, acima detalhados, versus o que é proposto que a aplicação web a ser desenvolvida venha a ter.

	Deteção de pragas/doenças nas plantas	Live Chat	Notificação para rega e fertilização das plantas	Sincronização com serviços 'Cloud'	Quantidade de água e fertilizantes a ser utilizado nas plantas	Utilização de inteligência artificial	Página "O meu jardim"	Temperatura do local onde se encontra o utilizador
Plantix	✓	✓	✗	✗	✗	✓	✓	✓
Picture This	✓	✗	✗	✗	✗	✓	✓	✗
WaterMe - Gardening Reminders	✗	✗	✓	✓	✗	✗	✓	✗
Sistema para Monitorização de Plantas - Frontend e Backend	✓	✗	✓	✗	✓	✗	✓	✓

Tabela 1 – Análise benchmarking

4 Engenharia

Atualmente, todos os sistemas informáticos têm como objetivo responder a objetivos e/ou necessidades dos utilizadores, sendo estes utilizadores, singulares ou organizações.

Deste modo, existe uma enorme importância para a correta e precisa definição de requisitos de sistema, uma vez que são estes que funcionam como guia para gestores de projeto, programadores, técnicos de sistemas e elementos de qualidade, durante o desenvolvimento do projeto.

4.1 Levantamento e Análise de Requisitos

Numa fase inicial do projeto é importante definir que requisitos devem ser implementados para que respondam às necessidades dos utilizadores e que cumpram na íntegra a visão de projeção relativamente ao projeto sobre o seu estado final.

Foram então estipulados requisitos em três formas: requisitos técnicos [Tabela 2], requisitos funcionais [Tabela 3] e requisitos não funcionais [Tabela 4].

Enumera-se de seguida o resultado da recolha de requisitos:

4.1.1 Requisitos técnicos

Requisitos	Escala 'Have'
R.T.1 - A aplicação necessita obrigatoriamente de uma conexão à internet, válida.	must have
R.T.2 - Nesta versão da aplicação, apenas serão suportados os browsers "Microsoft Edge" e "Google Chrome".	must have
R.T.3 - Nesta versão da aplicação, apenas serão suportados <i>desktops</i> .	must have
R.T.4 - A aplicação deverá estar sempre atualizada.	must have

Tabela 2 – Requisitos Técnicos

4.1.2 Requisitos funcionais

Requisitos	Escala de tamanhos	Escala 'Have'	Story points
R.F.1 - O utilizador deverá conseguir efetuar <i>login</i> de modo a entrar na sua conta pessoal.	XS	must have	2
R.F.2 - O utilizador deverá conseguir aceder a todas as notificações para saber quais notificações é que viu e quais as que não viu.	S	could have	3
R.F.3 - O utilizador deverá conseguir efetuar <i>logout</i> de modo a sair da sua conta.	XS	must have	2
R.F.4 - Na página destinada às plantas deverá ser possível adicionar uma planta ao jardim do utilizador.	M	should have	4
R.F.5 - Na página destinada às plantas, ao selecionar uma planta, deverá ser possível ver os seus detalhes.	XL	must have	9
R.F.6 - Na página destinada às plantas deverá ser possível remover uma planta ao jardim do utilizador.	M	should have	4
R.F.7 - Na página inicial, o utilizador deverá conseguir visualizar a meteorologia de modo a ter uma perceção da previsão do tempo para os dias seguintes.	S	should have	5
R.F.8 - A aplicação deverá registar o dia em que o utilizador começou a plantar uma determinada planta de modo a poder enviar notificações.	M	nice to have	6
R.F.9 - Na página destinada ao jardim, o utilizador deverá conseguir selecionar e adicionar as plantas que pretender com o estatuto "Favorito".	L	nice to have	6
R.F.10 - O utilizador tem acesso a um perfil na aplicação de modo a poder guardar a sua informação em serviços 'cloud'.	M	nice to have	5
R.F.11 - O utilizador na página inicial deverá conseguir saber algumas curiosidades sobre plantas.	S	should have	3

Tabela 3 – Requisitos Funcionais

4.1.3 Requisitos não funcionais

Requisitos	Escala 'Have'
R.N.F.1 - O utilizador deverá ter credenciais válidas, para conseguir entrar na sua conta pessoal.	must have
R.N.F.2 - A plataforma deve respeitar a legislação em vigor, assim como as normativas da Comissão Nacional de Proteção de Dados (RGPD).	must have
R.N.F.3 - A aplicação deverá ter embutido uma cifra de encriptação quando o utilizador efetuar o seu <i>login</i> .	nice to have
R.N.F.4 - A aplicação deve ser de fácil utilização com uma interface intuitiva e fácil de navegar.	should have
R.N.F.5 - Todas as notificações deverão ter um nível de prioridade associado, numa escala de 1 a 5, sendo 1 'Muito Baixo' e 5 'Muito Crítico'.	should have
R.N.F.6 - Todas as notificações devem ter um estado associado a estas, podendo alterar entre "Lida" e "Não Lida", caso o utilizador tenha aberto ou não a notificação.	could have
R.N.F.7 - A aplicação deverá ter um impacto positivo para com todos os tipos de cuidadores de plantas, facilitando o acesso a informação e dados.	must have
R.N.F.8 - O utilizador deverá conseguir consultar a meteorologia de modo a saber o que pode fazer com as plantas que possui no jardim (colocar fertilizante, colocar pouca água, por exemplo).	nice to have
R.N.F.9 - A aplicação deve ser capaz de lidar com um número significativo de utilizadores (10 mil) sem sofrer quaisquer problemas de ligação ou a aplicação ficar lenta.	should have
R.N.F.10 - O processo de adicionar e remover as plantas deve ser fácil e intuitivo para os utilizadores.	could have

Tabela 4 – Requisitos Não Funcionais

4.2 Diagrama de Casos de Uso

Através da implementação de um diagrama de casos de uso [Figura 4] consegue-se mostrar as interações esquemáticas entre o sistema e os ambientes, respetivos. Este diagrama resume todos os detalhes dos utilizadores perante o sistema (são também conhecidos como atores) e as interações destes com o sistema.

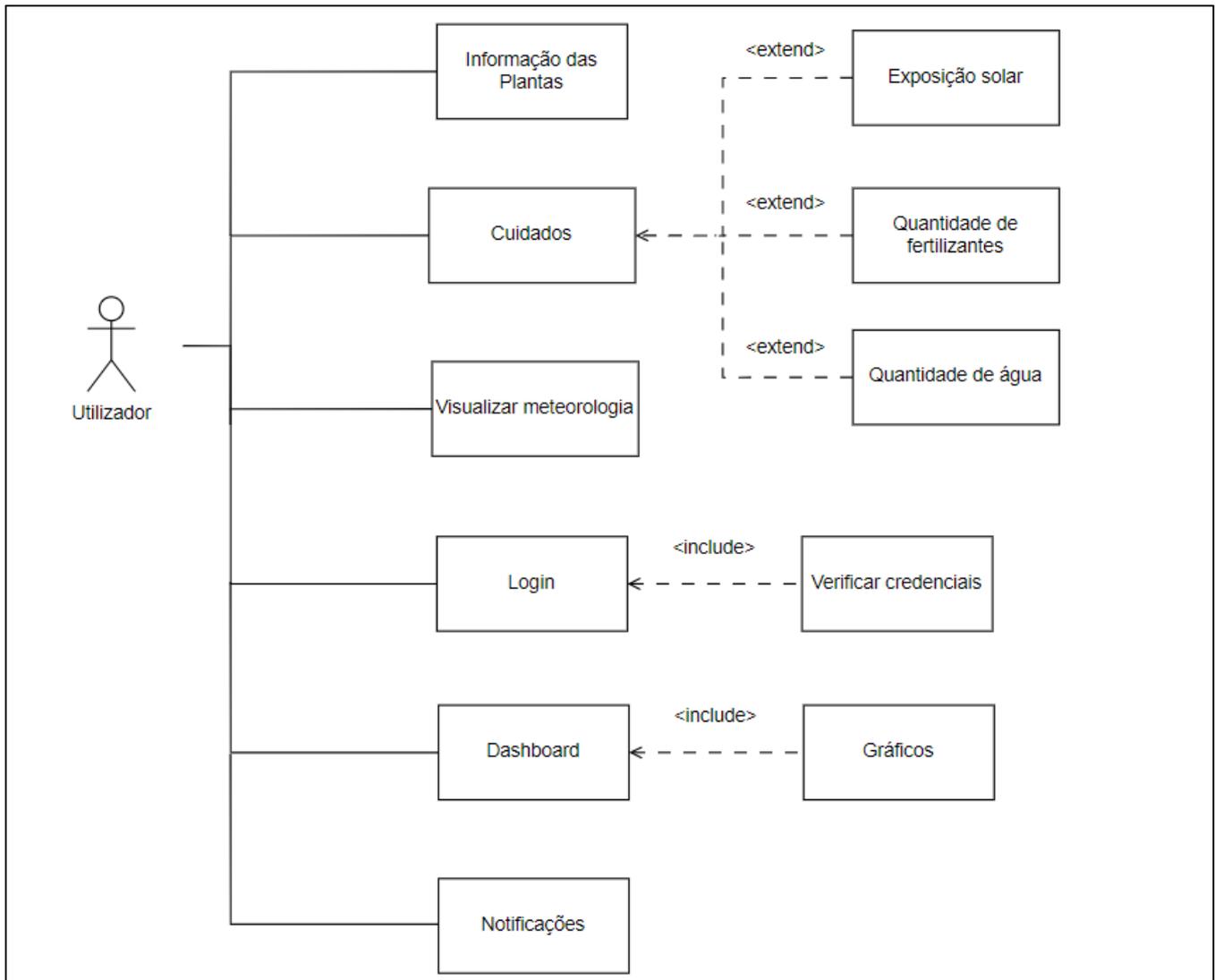


Figura 4 – Diagrama de Casos de Uso

4.3 Diagrama de Atividades

Os diagramas de atividades [Figura 5] representam os fluxos conduzidos por processamentos que corresponde essencialmente a um gráfico de fluxo, na qual é mostrado o fluxo de controlo de uma atividade para outra.

Figura 5 – Diagrama de Atividades

4.4 Modelos Relevantes

4.4.1 Diagrama Entidade-Relação

Os diagramas E.R. (Entidade-Relação) são uma forma de representar entidades, relações e atributos entre as diferentes variáveis. Estes diagramas são usados para modelação e criação de bases de dados relacionais, em termos de regras lógicas e de negócio, como num modelo lógico de dados, e em termos da tecnologia específica a ser implementada, como num modelo físico de dados.

Deste modo, elaborou-se um diagrama E.R. [Figura 6], onde os retângulos representam as entidades e as linhas representam a conexão que existe entre essas respetivas entidades.

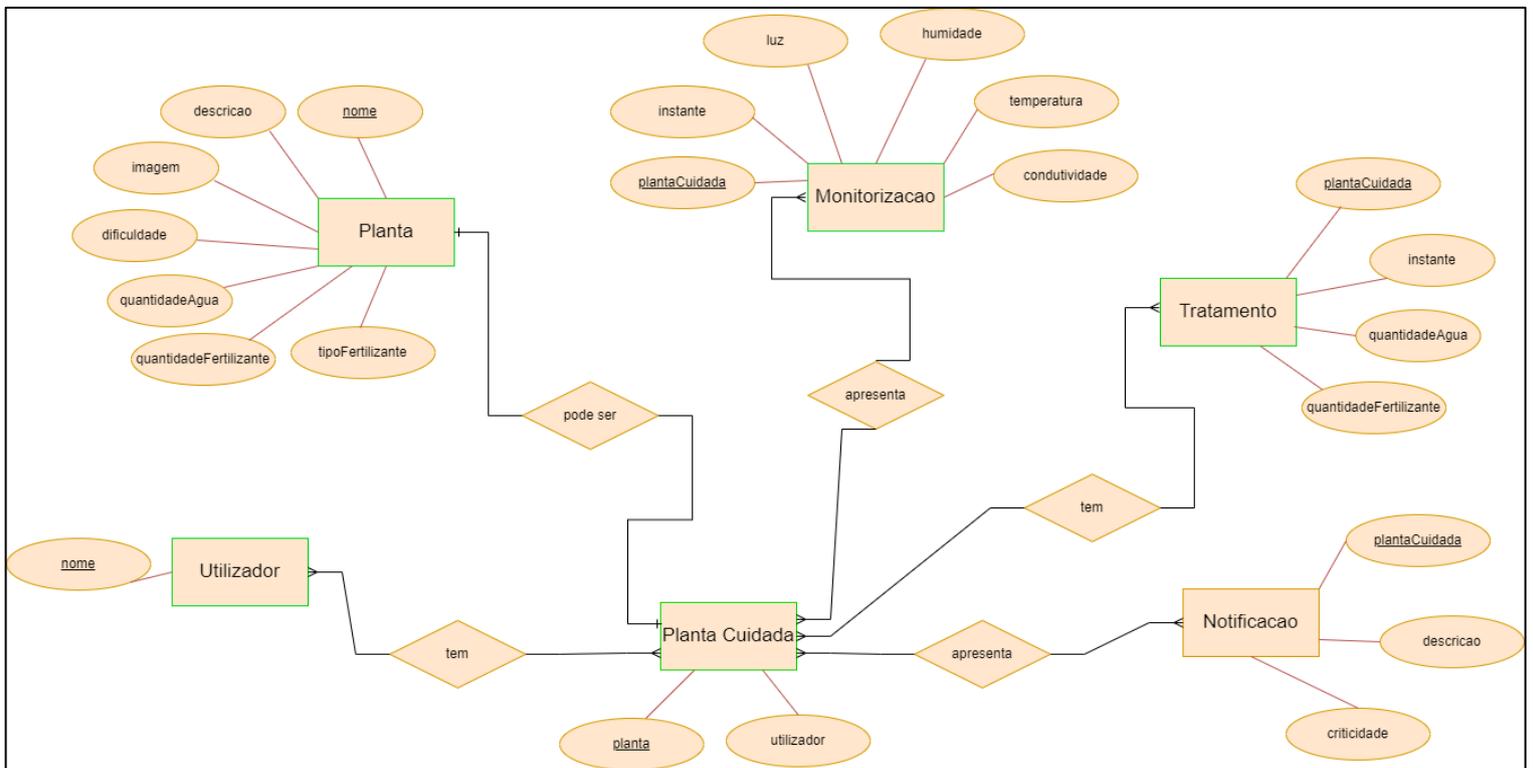


Figura 6 – Diagrama Entidade-Relação

4.4.2 Diagrama de Classes

O diagrama de classes corresponde a uma representação gráfica da estrutura de uma determinada aplicação, representando classes, atributos, métodos e relações entre eles. Este diagrama é amplamente utilizado na modelagem de programação orientada a objetos e ajuda a visualizar e comunicar melhor com a arquitetura do sistema.

Esta aplicação web não contém uma estrutura de programação orientada a objetos porém, para a melhor compreensão da programação desta aplicação, foi elaborado um diagrama de classes como ilustrado na figura abaixo [Figura 7].

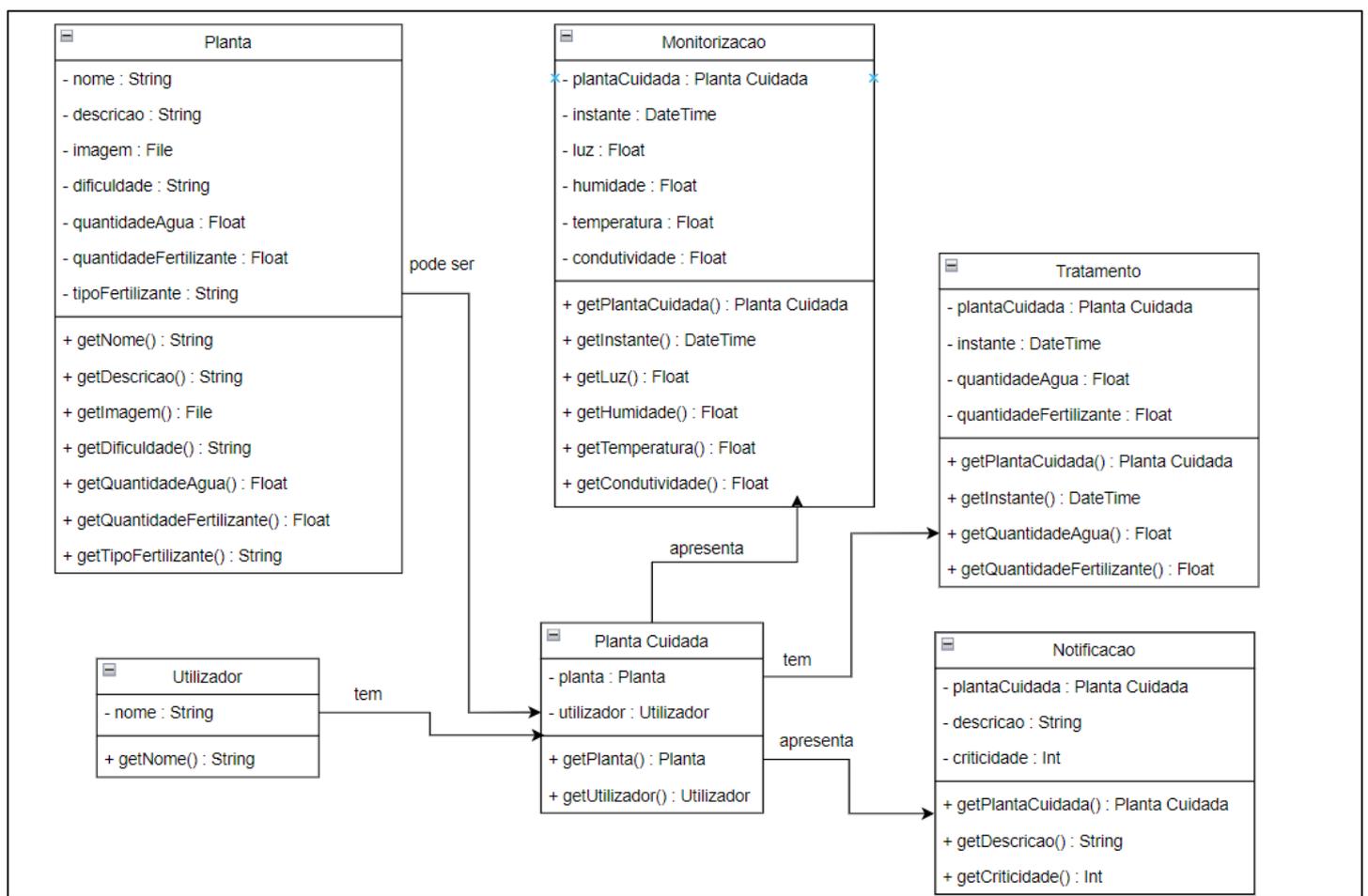


Figura 7 – Diagrama de Classes

4.4.3 Diagrama de Sequência

Os diagramas de sequência [Figura 8], comumente usados pelos desenvolvedores, modelam as interações entre objetos em um único caso de uso. Eles ilustram como as diferentes partes de um sistema interagem entre si para realizar uma função e a ordem em que as interações ocorrem quando um determinado caso de uso é executado.

Em palavras mais simples, um diagrama de sequência mostra diferentes partes de um sistema que trabalha em uma 'sequência' para se fazer algo.

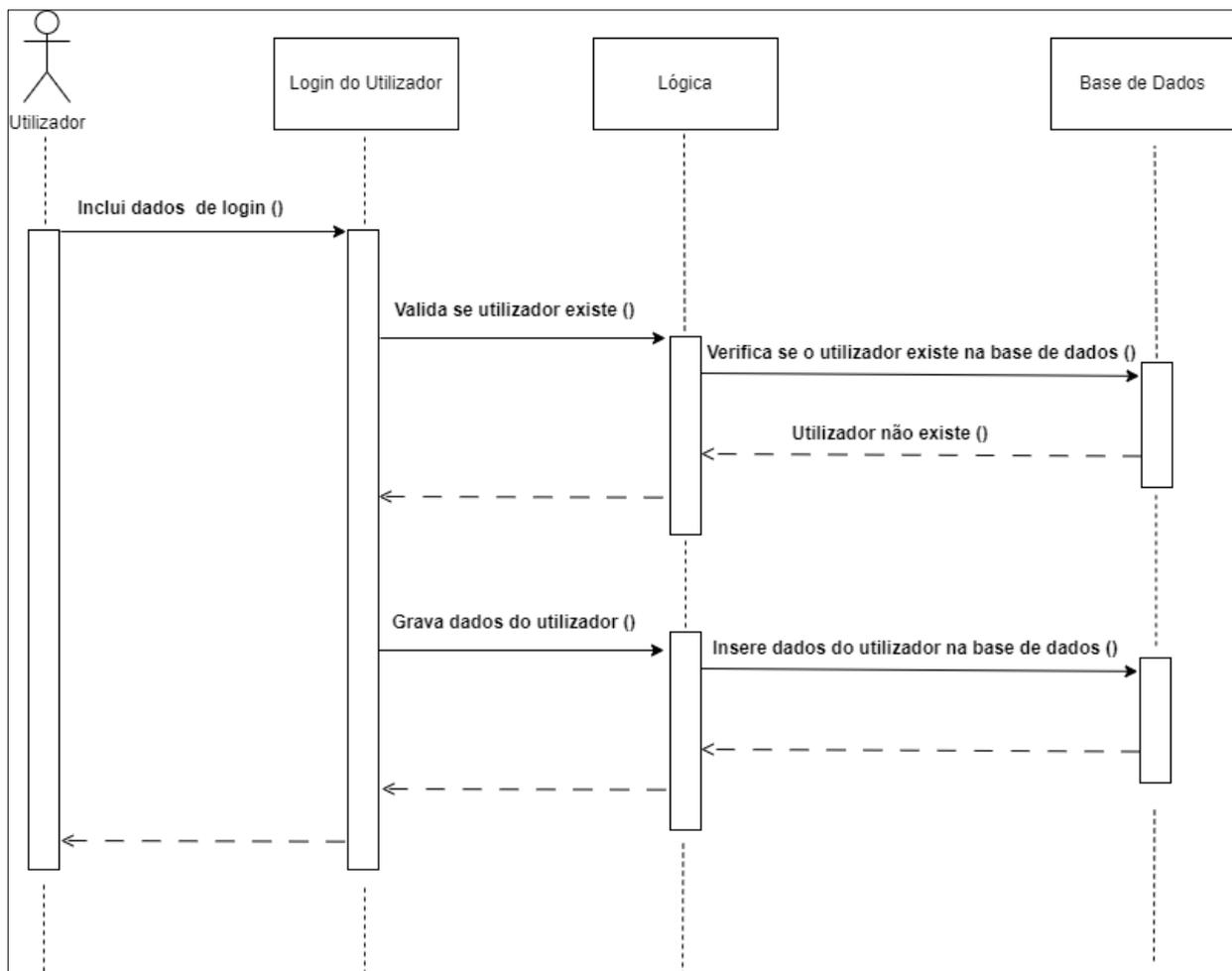


Figura 8 – Diagrama de Sequência (*login*)

4.5 Estrutura em árvore

As estruturas em árvore [Figura 9] são úteis em projetos web porque ajudam a representar a estrutura de uma aplicação.

Estes diagramas apresentam inúmeras vantagens, nomeadamente:

1. Permite o desenvolvimento sistemático e lógico de um sistema de estratégias, assim como de análise matemática de combinações. Assim, procura-se resolver um problema ou desenvolver meios para se alcançar um objetivo, tornando menos provável que os pontos essenciais sejam omitidos ou esquecidos;
2. Facilita o diálogo e a concordância entre os membros do grupo;
3. Ser convincente em apresentações porque estes identificam e mostram claramente os detalhes de questões complexas.

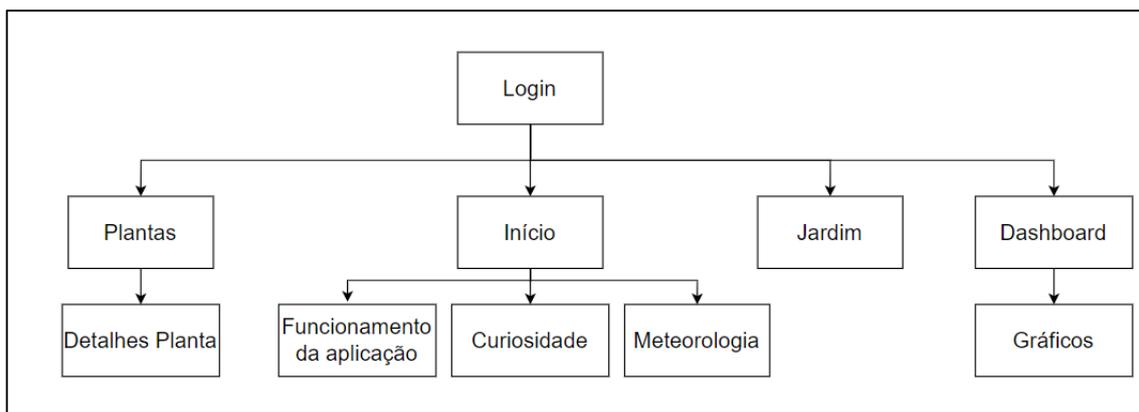


Figura 9 – Estrutura em Árvore

4.6 Mockups

Os Mockups serão usados como guia no desenvolvimento da aplicação. Nesta secção estarão representados os ecrãs de maior relevância e os restantes ecrãs serão colocados em anexo [Anexo 3 – Mockups, story board e mapa aplicacional]

Nas figuras abaixo [Figura 10] e [Figura 11] é mostrado o ecrã de dashboard, onde ficará o utilizador, depois de fazer login. Neste ecrã o utilizador consegue ver para que páginas pode ser redirecionado e apresenta gráficos de quantidade de água e de fertilizantes relativamente às plantas que possui no seu jardim.

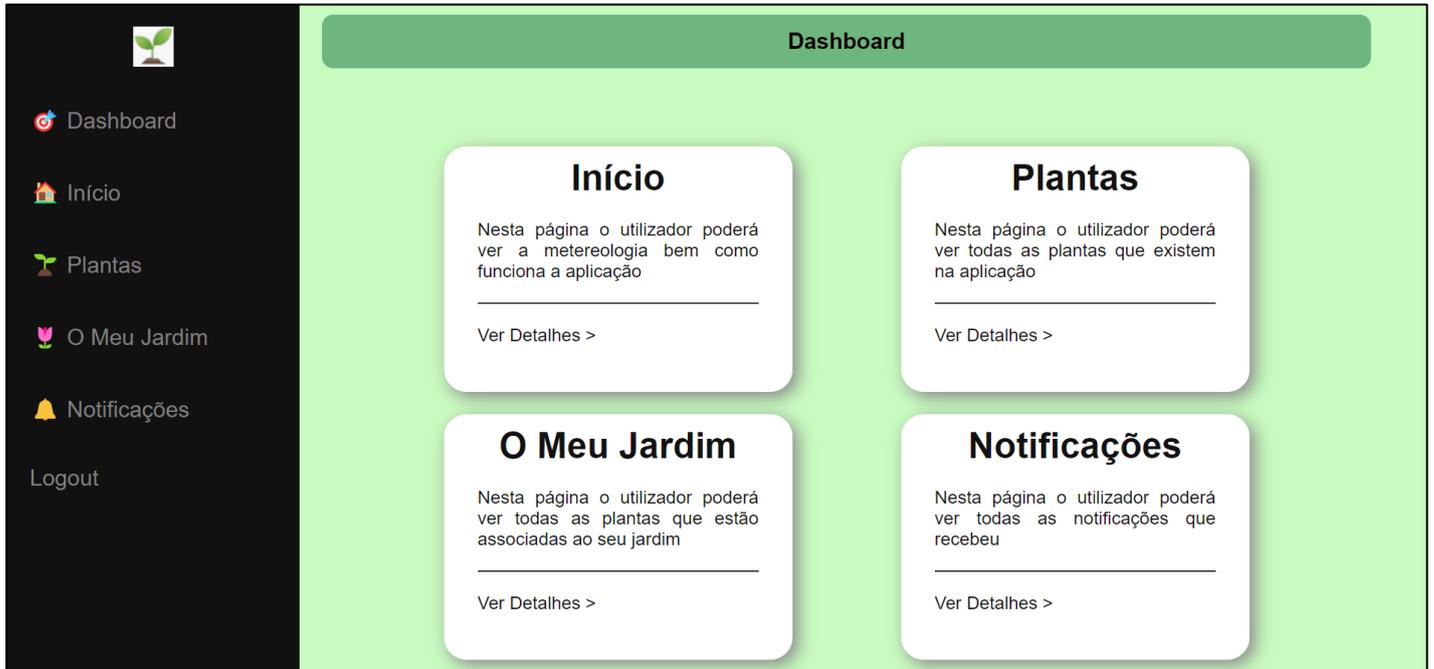


Figura 10 – Página Dashboard1



Figura 11 – Página Dashboard2

Nas seguintes figuras [Figura 12] e [Figura 13] é mostrada a página de início, onde o utilizador ao selecionar esta página através do dashboard ou do menu de navegação se depara com as curiosidades sobre as plantas, bem como a meteorologia, onde o utilizador consegue visualizar a meteorologia atual e a previsão para os 5 dias seguintes.

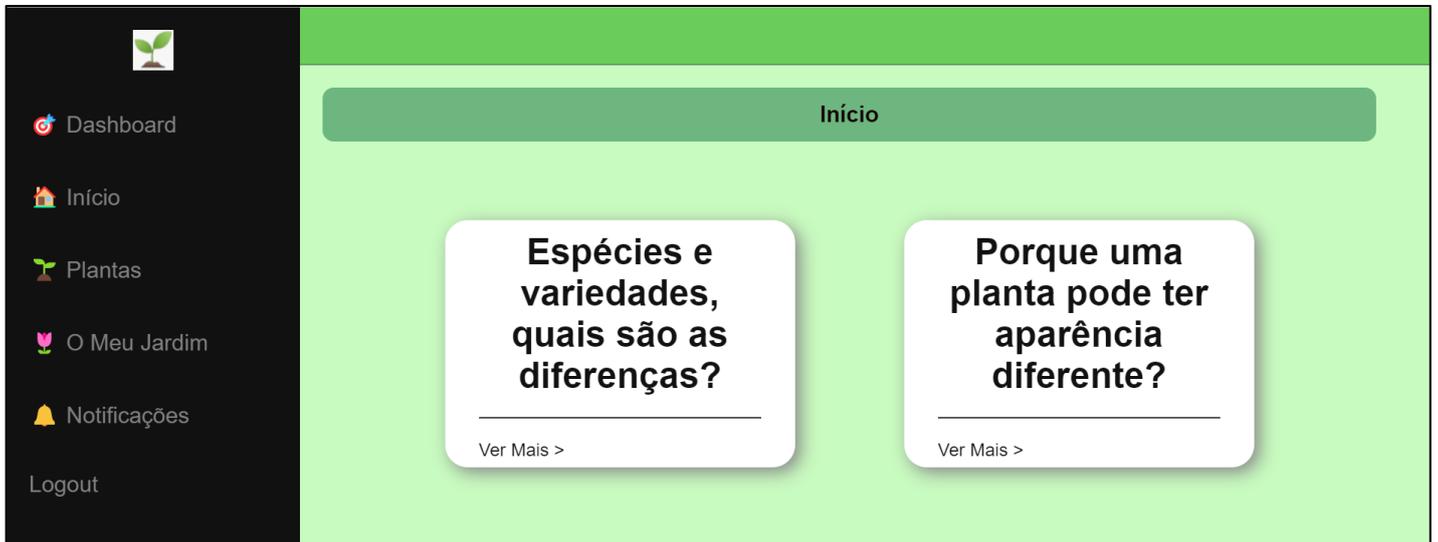


Figura 12 – Ecrã Início1

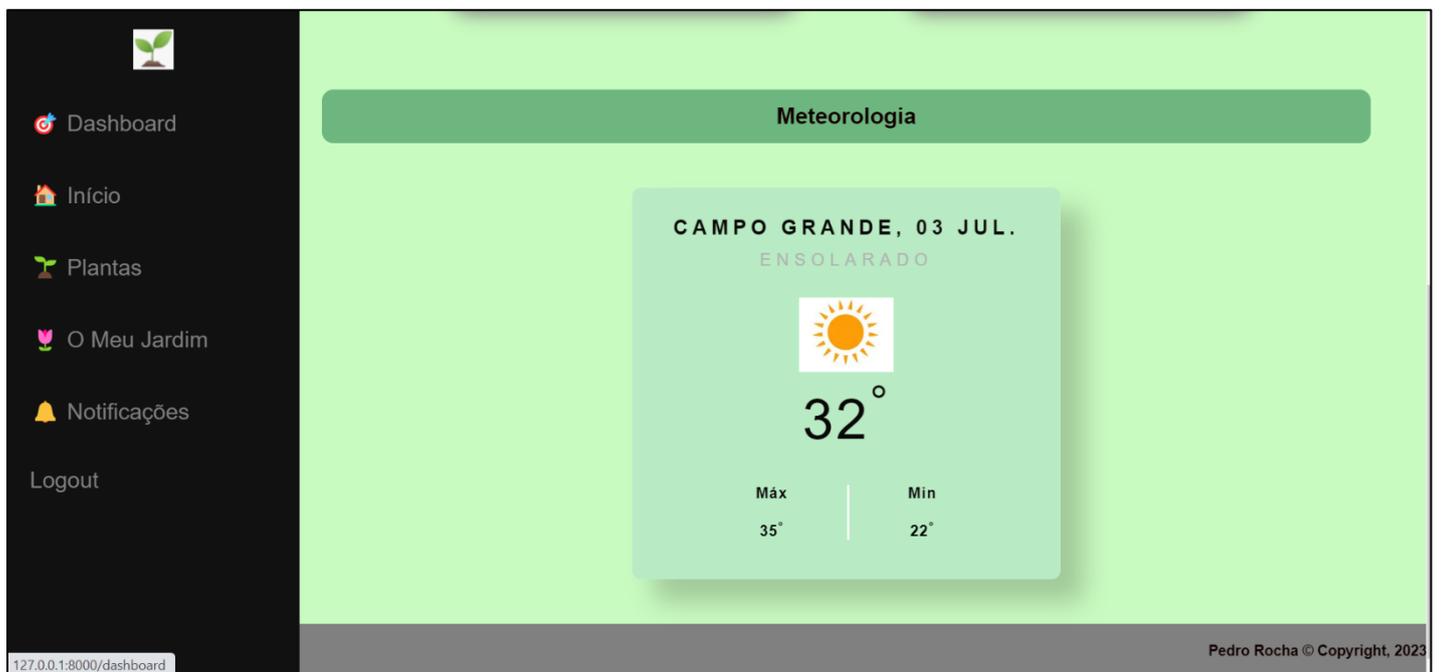


Figura 13 – Ecrã Início2

Na figura seguinte [Figura 14] é mostrada uma lista de todas as plantas registadas na base de dados. O utilizador ao carregar no botão de adicionar a planta seleccionada é adicionada ao seu jardim. Carregando em qualquer uma destas plantas levará o utilizador para os detalhes em específico da planta seleccionada [Figura 49].

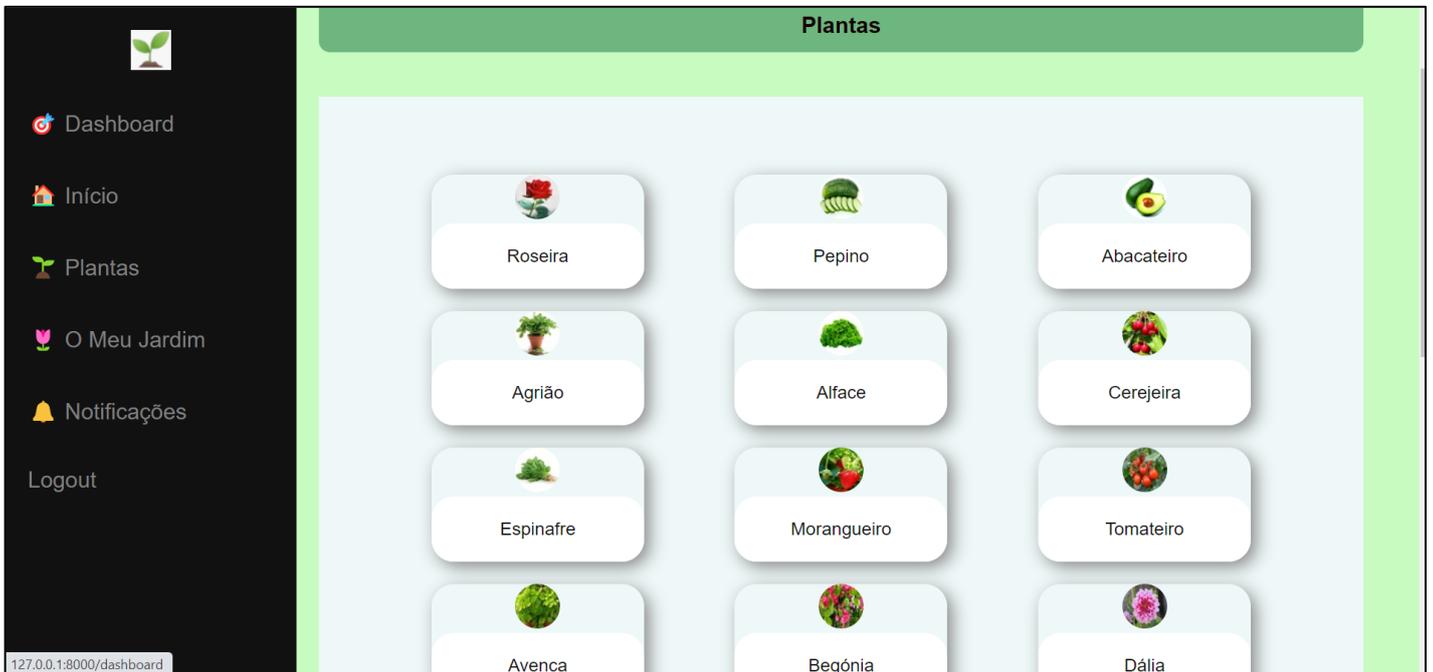


Figura 14 – Página Plantas

Na seguinte figura [Figura 15] é apresentado o jardim do utilizador, isto é, as plantas que o utilizador adicionou ao seu jardim para cuidar das mesmas.

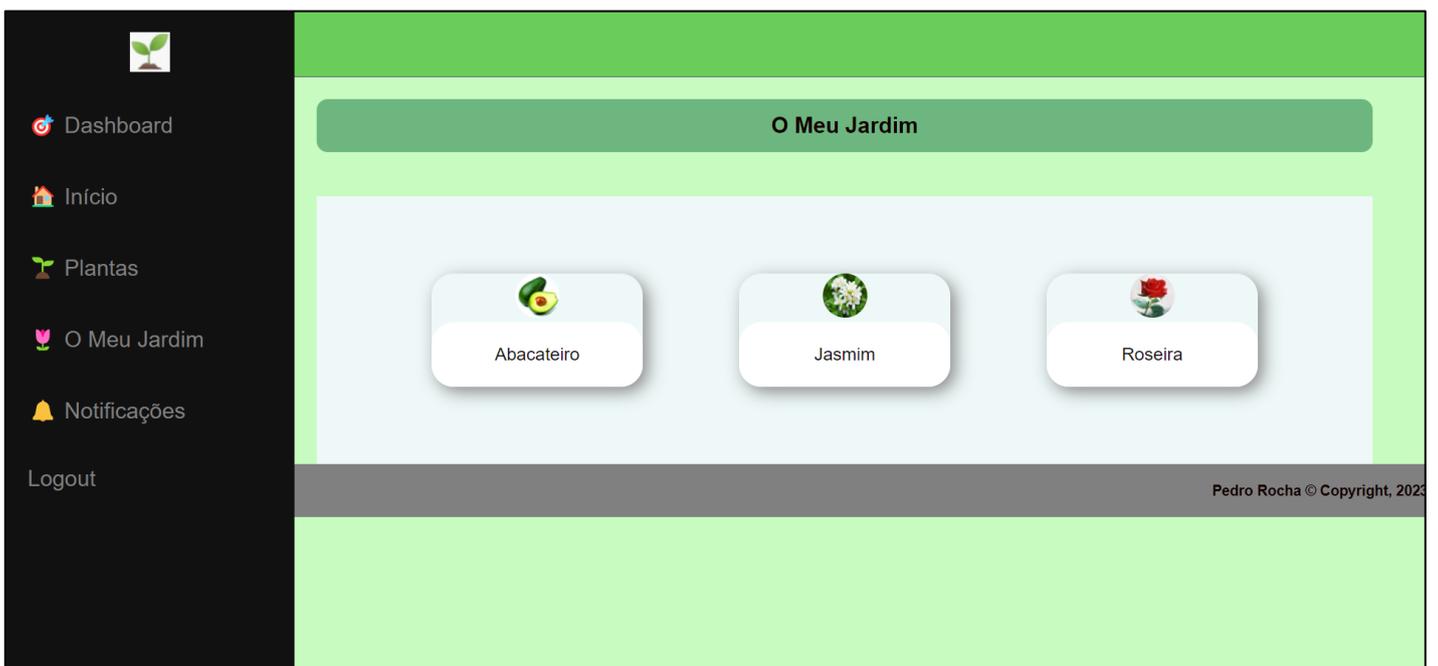


Figura 15 – Ecrã O Meu Jardim

Na figura abaixo [Figura 16] é apresentado as notificações do utilizador, onde o cuidador pode ver a data que foi enviada a notificação, bem como a descrição e o nível de criticidade. As notificações podem ser filtradas ainda por cor consoante o nível da notificação (1 – Nada Crítico e 5 – Muito Crítico, 1 – Azul e 5 – Vermelho) e notificações já lidas (cor cinzenta).



Figura 16 – Ecrã Notificações

Na seguinte figura [Figura 17], o cuidador consegue observar um gráfico temporal face às necessidades da planta a nível de água.

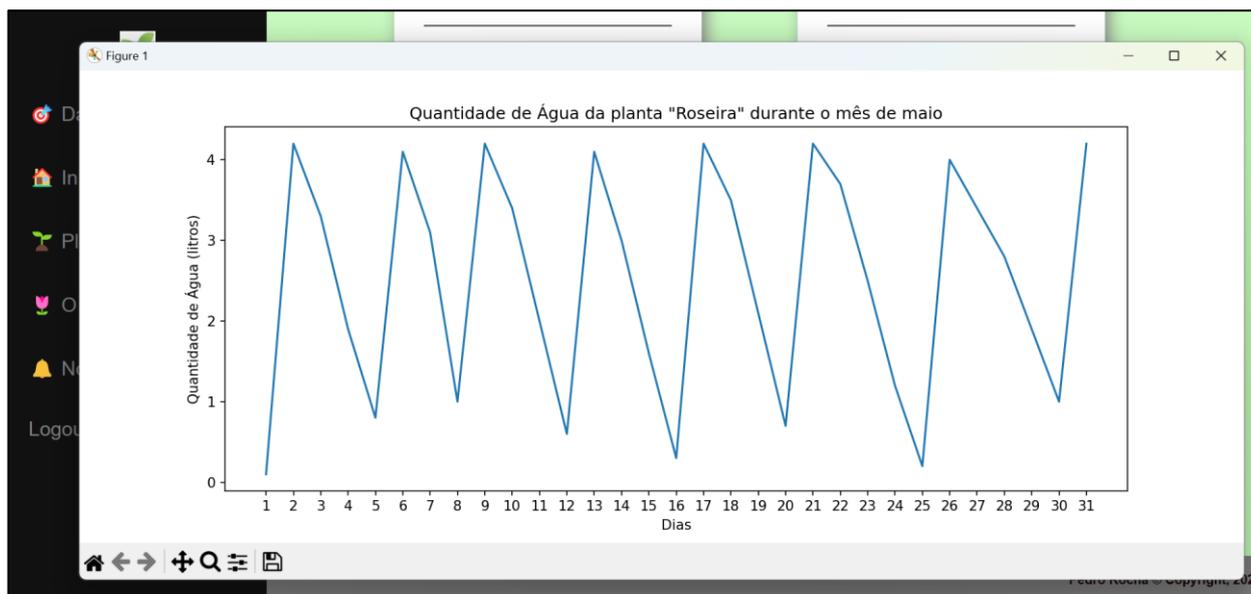


Figura 17 – Ecrã Gráfica Água

5 Solução Proposta

5.1 Introdução

A utilização e escolha das tecnologias para realizar em contexto de TFC deve-se, ao facto de, já ter utilizado na vertente académica, na unidade curricular de Programação Web. Este trabalho está a ser desenvolvido juntamente com outro TFC do colega Tiago Rodrigues (DEISI270 – Sistema de monitorização de plantas), em que o TFC do Tiago corresponde aos dados (backend) e este TFC está encarregue de mostrar esses dados aos utilizadores/cuidadores de plantas (frontend).

A aplicação encontra-se, por agora, disponível para teste ou consulta:

- O código encontra-se disponível para consulta no repositório de GitHub [GiHu23];
- A correr na plataforma PythonAnywhere[PytA23];
- O seu funcionamento pode ser visto neste vídeo explicativo [YoVi23].

5.2 Arquitetura

O MVC (Model View Controller) [Figura 18] [MVC22] é um padrão de desenho software, que traz várias vantagens a nível da manutenção, facilitando a compreensão do código, visto as componentes estarem devidamente separadas. Esta arquitetura torna o código mais suscetível a testes, uma vez que é possível isolar o problema, caso exista.

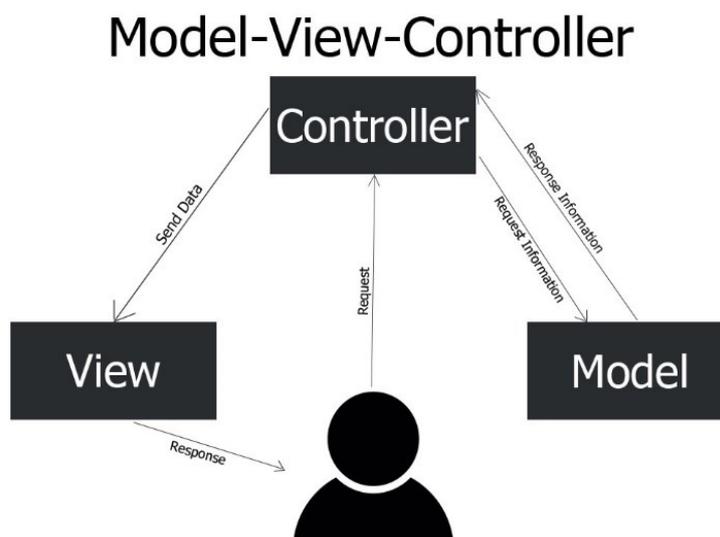


Figura 18 – Model View Controller (MVC)

Para a arquitetura da solução desenvolvida no presente projeto é utilizado o padrão de desenho de software MVT (Model View Template) [MVT22], utilizado pela *framework* Django. Este é semelhante ao MVC porém, o Django utiliza os seus próprios acordos, tal como demonstrado na figura abaixo [Figura 19].

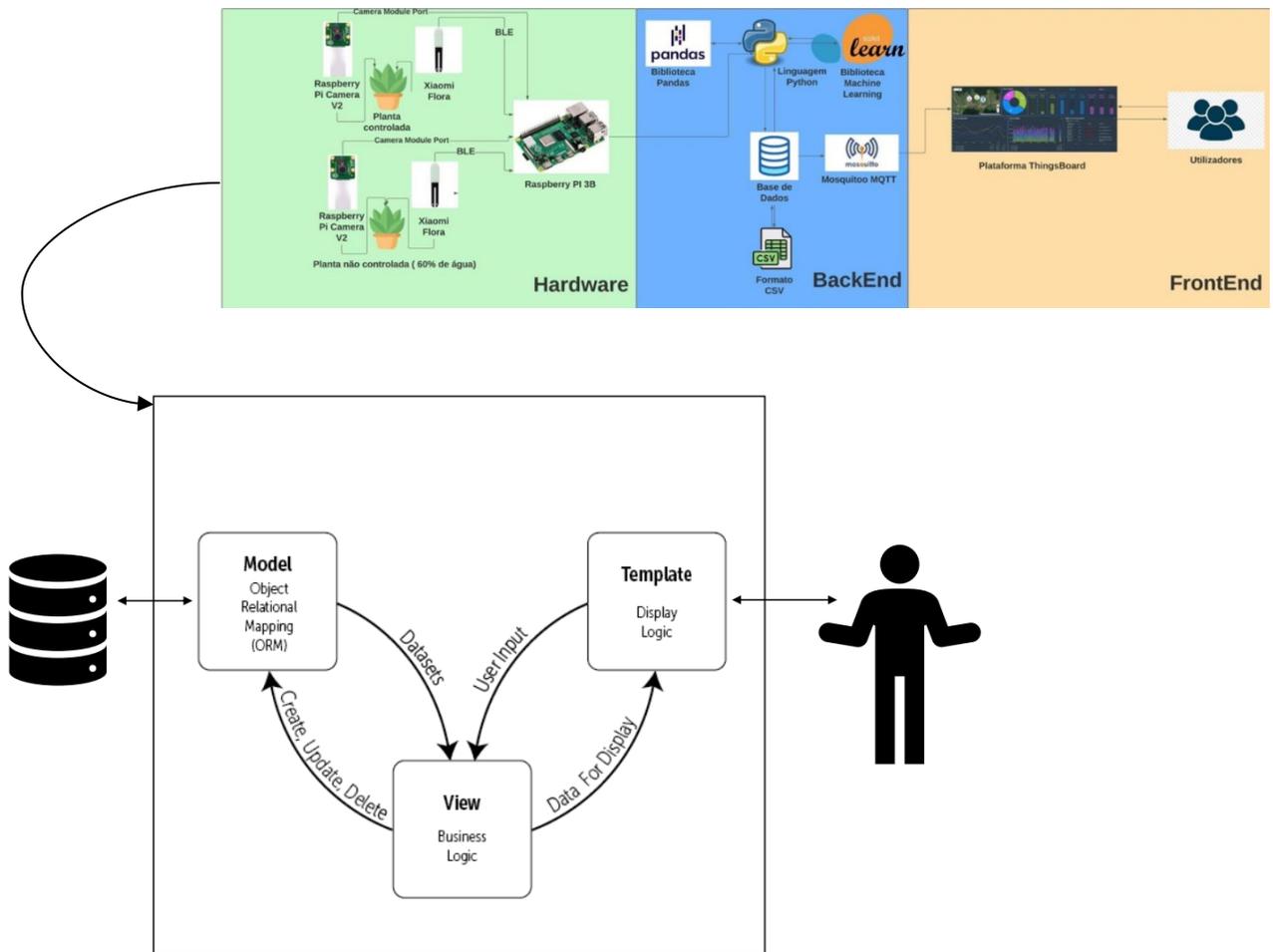


Figura 19 – Model View Template (MVT)

No Models é onde são criadas as classes e funções necessárias para a manipulação de dados, tanto os que estão guardados na base de dados, como variáveis locais para utilização no Template ou Views. O Models faz ligação com a base de dados e é através desta que podemos fazer as respetivas manipulações.

Nas Views, são preparados os dados que serão utilizados no Template, acedendo a funções alocadas no Models e, diretamente, as classes lá criadas, agregando a página html com as variáveis criadas, para depois poderem ser acedidas dentro do Template HTML.

Em resumo, o utilizador faz um pedido via Url, tendo associado uma View, que irá preparar toda a informação necessária para juntar e apresentar o Template. Todos os dados serão guardados numa base de dados SQLite3.

5.3 Tecnologias e Ferramentas Utilizadas

5.3.1 Frontend

Para o *frontend*, as tecnologias utilizadas são as seguintes:

- HTML5 [HTML23HTML23HTML23]: HyperText Markup Language (HTML) é o bloco de construção mais básico da web. Define o significado e a estrutura do conteúdo da web.
- CSS3 [CSS23]: É chamado de linguagem Cascading Style Sheet (CSS), e é usado para estilizar elementos escritos numa linguagem de marcação como HTML. O CSS separa o conteúdo da representação visual do website.
- JavaScript [JaSc23]: Linguagem de alto nível, cuja interação com HTML e CSS constitui uma grande parte do *frontend*. Sendo uma das suas principais características, o event listening que permite a existência de uma interação muito grande com o utilizador.
- Bootstrap [Boot23]: É uma *framework* HTML5, CSS3 e JavaScript mais popular para o desenvolvimento responsive.

5.3.2 Backend

Para o *backend*, as tecnologias utilizadas são as seguintes:

- Python [Pytn23]: Linguagem de alto nível, utilizado em diversos campos da informática, como data analytics, programação web ou automação de tarefas.
- Django [Djan23]: A Plataforma Web será desenvolvida em Django, estrutura Web de alto nível que incentiva o desenvolvimento rápido e um *design* limpo e pragmático, na linguagem de programação Python, que permite desenvolver uma aplicação web modular, robusta, segura e escalável.
- SQLite3 [SQLi23]: É uma biblioteca de linguagem C que implementa um mecanismo de base de dados SQL pequena, rápida, autónoma e de alta confiabilidade cheio de recursos disponíveis.
- PythonAnywhere [PyAn23]: É um ambiente de desenvolvimento integrado (IDE) online e serviço de hospedagem na web (Plataforma como serviço) baseado na linguagem de programação Python.

5.4 Implementação

A solução vai ser desenvolvida recorrendo à tecnologia Python. Esta é uma linguagem de alto nível lançada no ano de 1991, muito popular atualmente, e que suporta tanto programação orientada a objetos como programação imperativa.

A escolha desta tecnologia, para além da sua versatilidade vai também permitir o uso de *frameworks* de desenvolvimento web como o caso do Django.

Esta *framework* é conhecida por ter um tempo de execução extremamente rápido, pela sua ótima versatilidade e pela sua capacidade de escalabilidade.

Uma vez que o conceito desta solução está estruturado em cima de uma base de dados, e como a linguagem é Python, vai ser utilizada a biblioteca SQLite3, já que é a opção defeito do Django para a construção da base de dados.

O website em si vai ser desenvolvido com a maior personalização possível utilizando tecnologias HTML5, CSS3 e JavaScript, sendo que não é descartada a possibilidade de utilização de alguns módulos de bootstrap para agilizar processos.

5.5 Abrangência

Na vertente académica, esta solução vai envolver metodologias já trabalhadas em algumas áreas curriculares já lecionadas até à data, sendo estas:

- Algoritmia e Estruturas de Dados – Conhecimentos relacionados com manipulação e análise de dados.
- Bases de Dados – Conhecimentos relacionados com bases de dados, desde a familiarização com a linguagem SQL à construção de bases de dados e criação de queries.
- Engenharia de Requisitos e Testes – Conhecimentos relacionados com o levantamento de requisitos, planeamento e gestão do projeto.
- Programação Web – Conhecimentos relacionados com desenvolvimento *frontend* e *backend*.
- Interação Humano-Máquina – Conhecimentos relacionados com *design* de interfaces do utilizador e questões relacionadas com acessibilidade.

6 Método e Planeamento

De forma que seja possível parametrizar e controlar que as sucessivas implementações do TFC se mantenham dentro do prazo previsto, na figura abaixo [Figura 20] segue a proposta cronológica de ordem de trabalho.

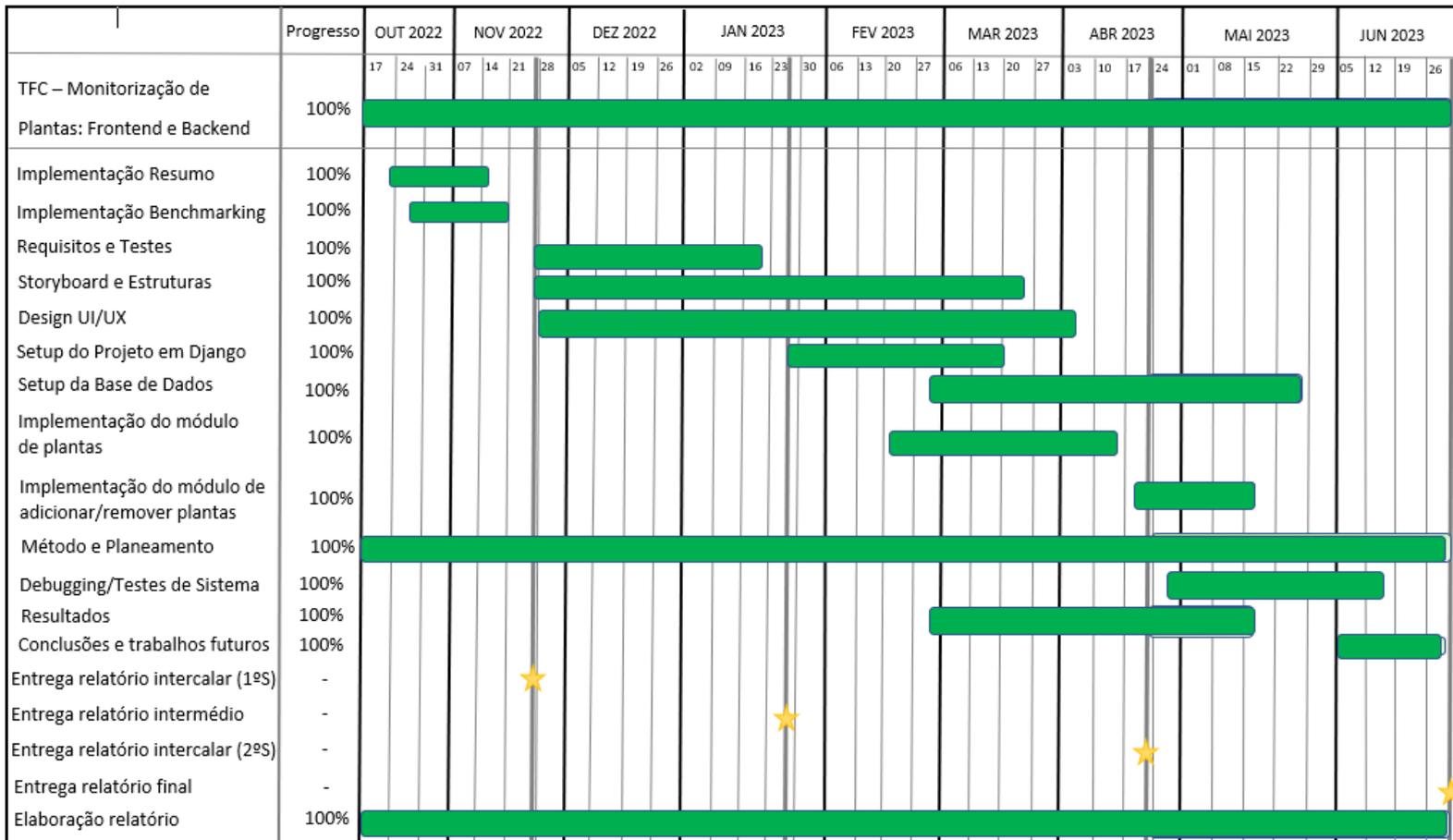


Figura 20 – Calendário proposto em formato Gantt

6.1 Calendário Frontend e Backend

6.1.1 Calendário Frontend

Na figura abaixo [Figura 21] segue a proposta cronológica de ordem de trabalho, face à parte *frontend* do mesmo.

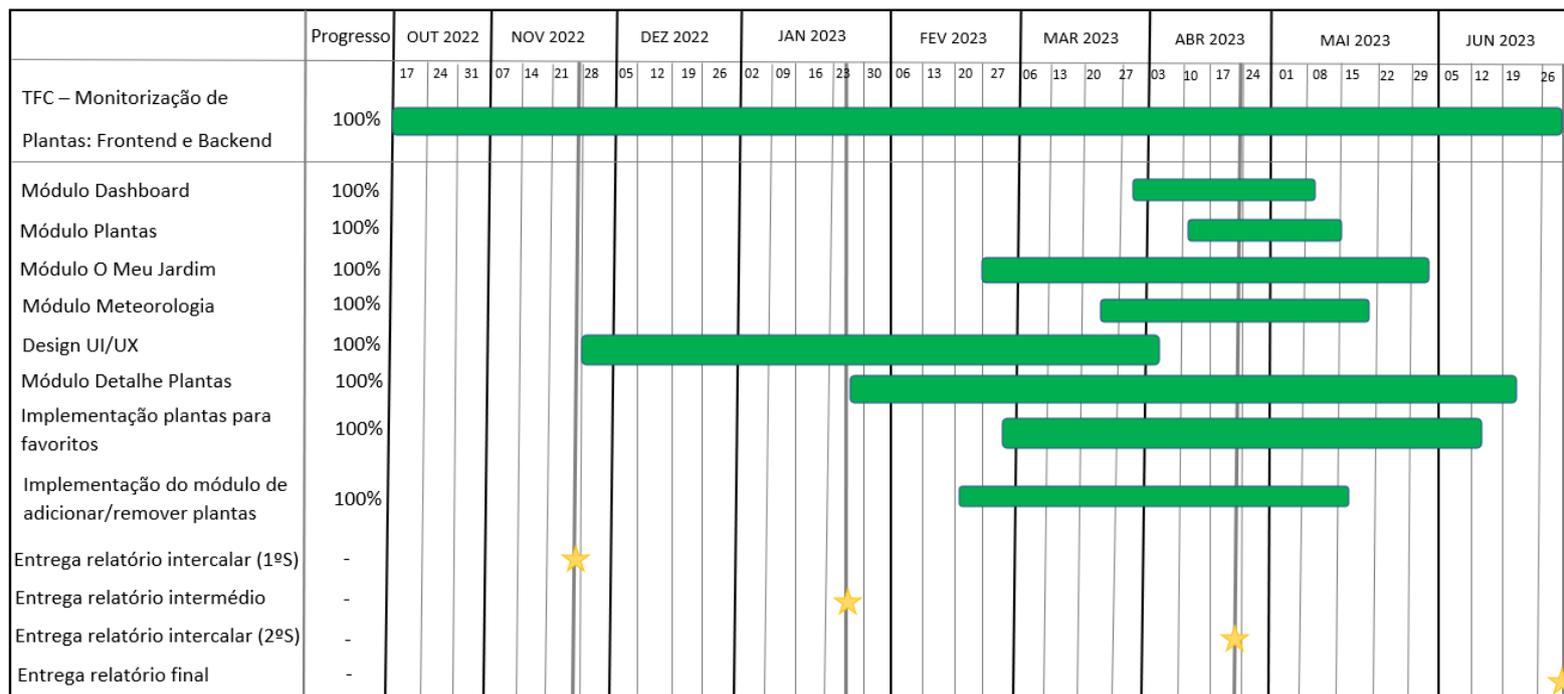


Figura 21 – Calendário Gantt (frontend)

6.1.2 Calendário *backend*

Na seguinte figura [Figura 22] segue a proposta cronológica de ordem de trabalho, face à parte *backend* do mesmo.

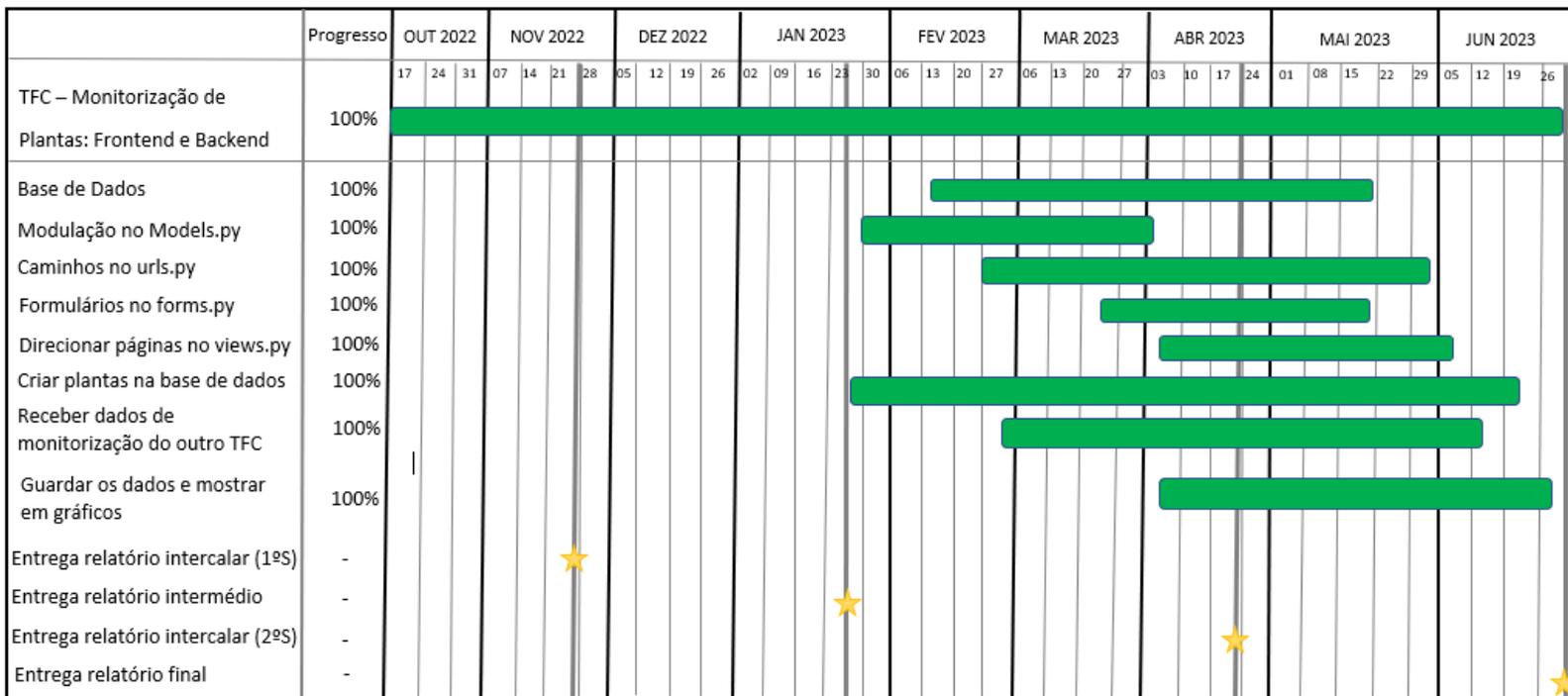


Figura 22 – Calendário Gantt (backend)

6.2 Desenvolvimento do Relatório

Neste projeto existiu um trabalho de forma incremental, todo o processo de análise e desenvolvimento da aplicação foi feito de maneira a receber feedback para melhoria da mesma. Para isto, foram feitas reuniões uma vez por semana em que, nestas reuniões estavam presentes o professor João Carvalho e o professor João Pavia, juntamente com os colegas referentes a um dos orientadores ou a ambos.

No início deste trabalho, o principal foco foi perceber os requisitos essenciais para que a aplicação fosse simples e funcional. Para perceber os requisitos essenciais, foi através das reuniões semanais com os professores João Carvalho e João Pavia que podemos receber feedback e orientações para a melhoria quer do relatório quer da aplicação web.

Depois começamos a desenvolver o Diagrama de Entidade-Relação, de modo a perceber como iríamos guardar os diferentes tipos de dados, e como as diferentes partes da aplicação se interligavam, este foi, inevitavelmente, sofrendo algumas alterações depois, durante o desenvolvimento da aplicação.

Para a validação dos requisitos bem como do design geral da aplicação foi feito um protótipo interativo que foi mostrado a ambos os professores, que pode ser visto no seguinte anexo [Anexo 5 – Protótipo Interativo].

1. Introdução e Resumo

Nesta fase é feito a ideologia do tema sistema monitorização de plantas como a justificação da escolha do tema incluído.

2. Identificação do Problema

Neste tópico do relatório é elaborado a identificação de problemas atuais e futuros perante a construção deste sistema como também a identificação global dos problemas da agricultura e perda de água na rega de plantas.

3. Estudo da Viabilidade e Pertinência

Neste tópico do relatório é pretende-se que procure saber mais sobre um sistema de monitorização de plantas e como poderá conter um futuro após a conclusão deste trabalho final ou até mesmo a possibilidade de conjugar com outro sistema existente.

4. Benchmarking

Após a procura de websites de sistemas de monitorização de plantas existentes deve-se mencionar nesta secção relatório essas mesmas soluções encontradas e comparar com a nossa proposta.

5. Levantamento de Requisitos

Neste tópico deve-se refletir sobre possíveis requisitos do TFC incluído requisitos funcionais , requisitos não funcionais e requisitos técnicos. Após a reflexão é necessário enumerar os requisitos de forma estruturada e atribuir possíveis custos de tempo bem como a importância neste website de monitorização de plantas.

6. Descrição da Implementação e Tecnologias Utilizadas

Neste tópico do relatório deve-se fazer uma breve descrição da lista de ferramentas e tecnologias a utilizar na construção do website.

7. Método e Planeamento

Neste tópico é apresentado o plano de trabalho proposto no formato Gantt de modo a orientar o progresso e tempo de trabalho.

8. Resultados

Neste tópico é apresentado os resultados obtidos através de testes à aplicação conforme os requisitos que foram enumerados no ponto 4.1 do relatório.

9. Conclusões e Trabalhos Futuros

Neste tópico é apresentado as conclusões sobre o trabalho realizado ao longo do ano bem como perspectiva para trabalhos futuros que poderão aperfeiçoar e melhorar a aplicação em si.

7 Resultados

Para validar as funcionalidades da aplicação foram criados testes de usabilidade, visíveis no [Anexo 6 – Plano de Testes e Validação] com base nas funcionalidades idealizadas de implementar, e demonstradas na secção [Levantamento e Análise de Requisitos]. A seguinte tabela [Tabela 5] é semelhante à tabela presente no anexo de Plano de Testes com a diferença de ter uma coluna extra, isto é, a coluna com o resultado obtido.

Estes testes, para além de contribuírem para alinharmos a aplicação com as necessidades da organização, e, posteriormente, ser alvo de melhoria, servem para a identificação de bugs.

Título	Descrição	Requisito(s)	Resultado Esperado	Resultado Obtido
Autenticação	O utilizador entra na aplicação através da sua conta	R.F.1, R.N.F.1	Login bem sucedido, utilizador é levado ao seu dashboard.	Login bem sucedido e o utilizador é redirecionado para o seu dashboard.
Aceder aos detalhes das plantas	O utilizador utiliza o menu de navegação para aceder à página das plantas e visualizar os seus detalhes	R.F.5	O utilizador é levado à lista das plantas	É levado à lista de plantas com sucesso.
Adicionar e remover plantas do jardim do utilizador	O utilizador consegue adicionar e remover as plantas do seu jardim através do botões associados aos mesmos	R.F.4, R.F.6, R.N.F.10	O utilizador adicione e remove as plantas	O utilizador adicione e remove as plantas com facilidade e simplicidade

Visualizar notificações	O utilizador consegue visualizar as notificações que recebeu	R.F.2, R.N.F.5, R.N.F.6	O utilizador visualiza e observa as notificações que tem	Consegue ver o nível de criticidade da notificação bem como se já viu ou não
Visualizar a meteorologia	O utilizador utiliza o menu de navegação para aceder à página início onde visualiza a meteorologia	R.F.7, R.N.F.8	O utilizador consegue ver o tempo para o dia e para os 5 dias seguintes	O utilizador consegue visualizar a meteorologia com sucesso
Visualizar curiosidades	Na página início o utilizador consegue aceder a curiosidades	R.F.11	O utilizador vê as curiosidades “Espécies e variedades” e “Aparência diferentes nas plantas”	O utilizador consegue visualizar as curiosidades com sucesso
Registar dia de plantação	A aplicação regista o dia em que o utilizador começou o tratamento de uma planta	R.F.8	A planta ao ser adicionada ao jardim é registado o dia de modo a ser possível enviar notificações	Não implementado
Cifra de encriptação	A aplicação deverá ter embutido uma cifra de encriptação quando efetuar login	R.N.F.3	A aplicação deverá ter embutido uma cifra de encriptação	Não implementado
Perfil	O utilizador apresenta um perfil com os seus dados pessoais	R.F.10	O utilizador tem a sua informação guardada na secção perfil	Não implementado

Tabela 5 – Resultados

8 Conclusões e Trabalhos Futuros

8.1 Conclusões

Ao analisar todos os temas disponíveis de Trabalho Final de Curso, o projeto Sistema de Monitorização de Plantas – Frontend e Backend foi o trabalho que mais se destacou desde o início.

Para a implementação deste projeto é de destacar tecnologias como a framework Django, onde foram utilizadas linguagens de programação como Python, e linguagens de marcação como o HTML e o CSS.

Para colocar a aplicação acessível a qualquer pessoa foi utilizado o PythonAnywhere.

Na consecução deste trabalho de projeto utilizámos conhecimentos, conteúdos, conceitos e ferramentas de Unidades Curriculares de toda a licenciatura, nomeadamente:

- Engenharia de Requisitos, onde foram utilizados os conceitos de planeamento e definição de requisitos;
- Os conteúdos da disciplina de Base de Dados também foram bastante utilizados, uma vez que praticamente toda a informação está alocada numa base de dados e são utilizados mecanismos de manipulação de dados entre a base de dados e a aplicação;
- Programação Web, retirámos os conceitos base para a realização do trabalho, como Django e toda a linguagem de programação utilizada.
- Foram também de enorme utilidade as ferramentas e os conhecimentos adquiridos na disciplina de Interação Humano-Máquina, onde foi utilizado o programa Axure para a realização dos Mockups, os conhecimentos sobre testes de usabilidade.

8.2 Trabalhos Futuros

Num futuro próximo seria interessante desenvolver esta aplicação para outro tipo de dispositivos eletrónicos, mais concretamente para telemóveis e tablets, por forma a abranger o máximo de utilizadores possíveis.

Seria igualmente importante existir uma conta admin em que os utilizadores pudessem enviar sugestões de melhoria da aplicação bem como o registo de mais plantas na base de dados do servidor.

Na secção da meteorologia, os dados serem dinâmicos que serão alimentados através de uma API de um site de meteorologia como, por exemplo, o ipma.

Seria igualmente interessante, de modo a garantir uma continuidade deste TFC, existir sensores que pudesse medir a quantidade de água e de fertilizante em tempo real alertando para excesso ou falta dos mesmos.

Bibliografia

- [Plan22] Plantix, <https://plantix.net/en/> , acedido em Nov. 2022.
- [PiTh22] Picture This, <https://www.picturethisai.com/> , acedido em Nov. 2022.
- [WaGa22] WaterMe - Gardening Reminders, <https://appadvice.com/app/waterme-gardening-reminders/1089742494> , acedido em Nov. 2022.
- [Axur23] Link Axure, <https://ms1jnu.axshare.com> , acedido em Jan. 2023.
Email: plantas@gmail.pt , Password: plantas
- [GiHu23] Link GitHub, <https://github.com/PedroRocha-a22003766/TFC> , acedido em Jun. 2023.
- [YoVi23] Link Video YouTube, https://youtu.be/QhBdPtZWQ_0 , acedido em Jun. 2023.
- [MVC22] Model View Controller, <https://folk.universitetetioslo.no/trygver/themes/mvc/mvc-index.html> , acedido em Dez. 2022.
- [MVT22] Model View Template, <https://www.javatpoint.com/django-mvt> , acedido em Dez. 2022.
- [HTML23] HTML5, <https://developer.mozilla.org/en-US/docs/Web/HTML> , acedido em Abr. 2023.
- [CSS23] CSS3, <https://developer.mozilla.org/en-US/docs/Web/CSS> , acedido em Abr. 2023.
- [JaSc23] JavaScript, <https://developer.mozilla.org/pt-BR/docs/Web/JavaScript> , acedido em Abr. 2023.
- [Boot23] Bootstrap, https://www.w3schools.com/whatis/whatis_bootstrap.asp , acedido em Abr. 2023.
- [Pytn23] Python, https://www.w3schools.com/python/python_intro.asp , acedido em Abr. 2023.
- [Djan23] Django, <https://www.djangoproject.com/> , acedido em Abr. 2023.
- [SQLi23] SQLite, <https://www.sqlite.org/index.html> , acedido em Abr. 2023.
- [PyAn23] PythonAnywhere, <https://www.pythonanywhere.com/> , acedido em Mai. 2023.

Anexo 1 – Estudo de viabilidade, pertinência e relevância

Questionário de Viabilidade, Pertinência e Relevância: Aplicação Web de Monitorização de Plantas

Este inquérito pretende conhecer a opinião das partes interessadas (pessoas com pouca ou muita experiência como cuidadores de plantas de todos os tipos) e, com base nos resultados, promover a partilha de informação relevante e fiável ao tratamento de diversas plantas de modo a que possam cuidar das mesmas de forma sustentável e rigorosa.

Agradecemos a sua disponibilidade para participar, respondendo a este questionário. (tempo estimado de resposta: 2-3 minutos).

Para qualquer informação adicional, contacte pedro.mgmrocha@gmail.com

 pedro.mgmrocha@gmail.com (não partilhado) [Mudar de conta](#) 

Figura 23 – Questionário (Introdução)

Dados Demográficos

Nesta secção serão recolhidos dados pessoais para fins estatísticos.

Idade *

0 - 15 anos

16 - 30 anos

31 - 45 anos

46 - 60 anos

61 - 75 anos

+ 75 anos

Figura 24 – Pergunta 1 (Dados Demográficos)

Género *

Feminino

Masculino

Outro

Figura 25 – Pergunta 2 (Dados Demográficos)

Escolaridade *

Nenhuma

4º ano

9º ano

12º ano

Licenciatura

Mestrado

Doutoramento

Figura 26 – Pergunta 3 (Dados Demográficos)

Viabilidade, Pertinência e Relevância do Trabalho

Nesta secção serão feitas perguntas de modo a compreender a relevância do trabalho.

Cuida/cuidou de plantas? *

Sim

Não

Figura 27 – Pergunta 1 (Viabilidade e Pertinência do Trabalho)

Se respondeu sim à pergunta acima, com que frequência as plantas que cuidava * acabavam por "morrer" sem saber o porquê de tal ter acontecido?

Nunca

Raramente

Algumas vezes

Muitas vezes

Respondi não na pergunta acima.

Figura 28 – Pergunta 2 (Viabilidade e Pertinência do Trabalho)

Como classifica, na atualidade, o acesso a aplicações web de monitorização de plantas? *

- Nada acessíveis
- Pouco acessíveis
- Algo acessíveis
- Muito acessíveis

Figura 29 – Pergunta 3 (Viabilidade e Pertinência do Trabalho)

Como classifica a necessidade de criação de uma aplicação web dirigida a todas as pessoas que cuidam de plantas ou que o pensem fazer no futuro? *

- Muito Desnecessário
- Desnecessário
- Necessário
- Muito Necessário

Figura 30 – Pergunta 4 (Viabilidade e Pertinência do Trabalho)

Como classifica a necessidade de existir uma funcionalidade de notificar o utilizador para saber se é preciso regar ou colocar fertilizantes? *

- Muito Desnecessário
- Desnecessário
- Necessário
- Muito Necessário

Figura 31 – Pergunta 5 (Viabilidade e Pertinência do Trabalho)

Como classifica a necessidade de existir gráficos com a monitorização feita quer *
de água, quer de fertilizantes às diferentes plantas do jardim associado a cada
utilizador?

- Muito Desnecessário
- Desnecessário
- Necessário
- Muito Necessário

Figura 32 – Pergunta 6 (Viabilidade e Pertinência do Trabalho)

Como avalia a necessidade de existir um sistema de filtrar as plantas por *
categoria a que está associada (ex.: pessegueiro -> frutas e cebola -> legumes) ?

- Muito Desnecessário
- Desnecessário
- Necessário
- Muito Necessário

Figura 33 – Pergunta 7 (Viabilidade e Pertinência do Trabalho)

Como avalia a necessidade de existir uma página da aplicação somente *
destinada para as plantas que o utilizador/cuidar decide cuidar?

- Muito Desnecessário
- Desnecessário
- Necessário
- Muito Necessário

Figura 34 – Pergunta 8 (Viabilidade e Pertinência do Trabalho)

Anexo 2 – Resultados do questionário

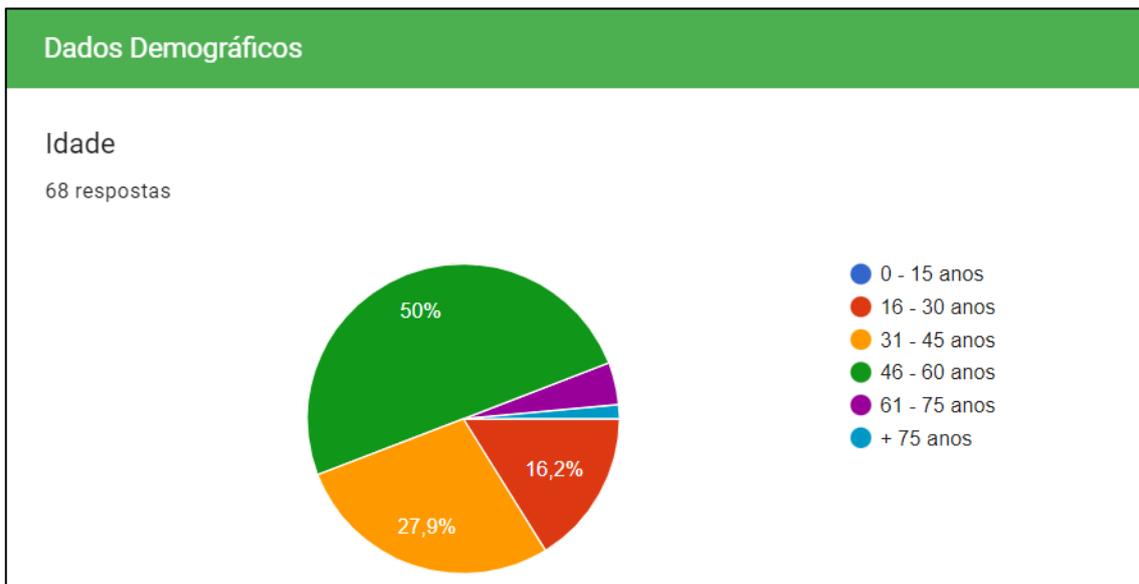


Figura 35 – Resposta Pergunta 1 (Dados Demográficos)

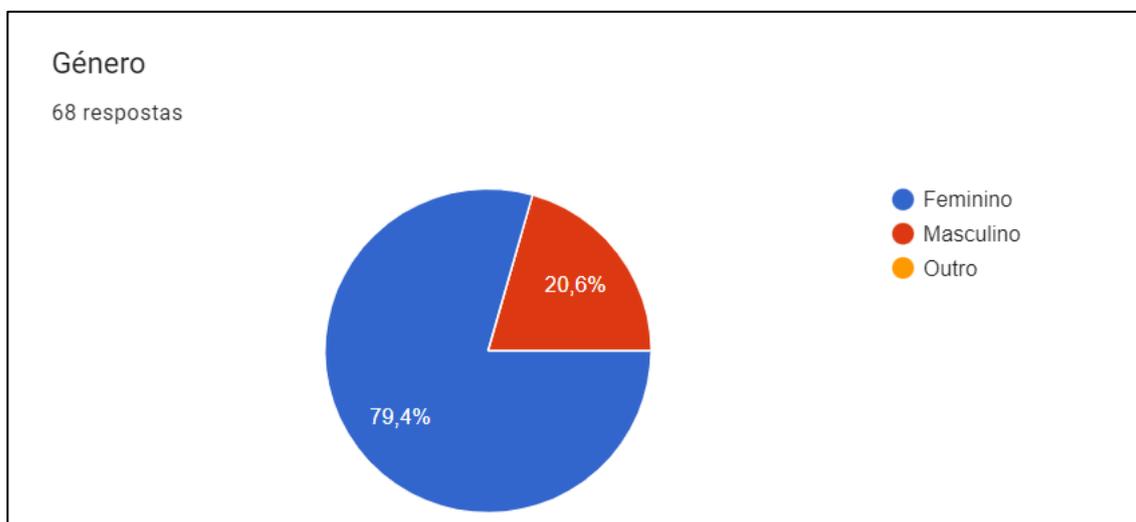


Figura 36 – Resposta Pergunta 2 (Dados Demográficos)

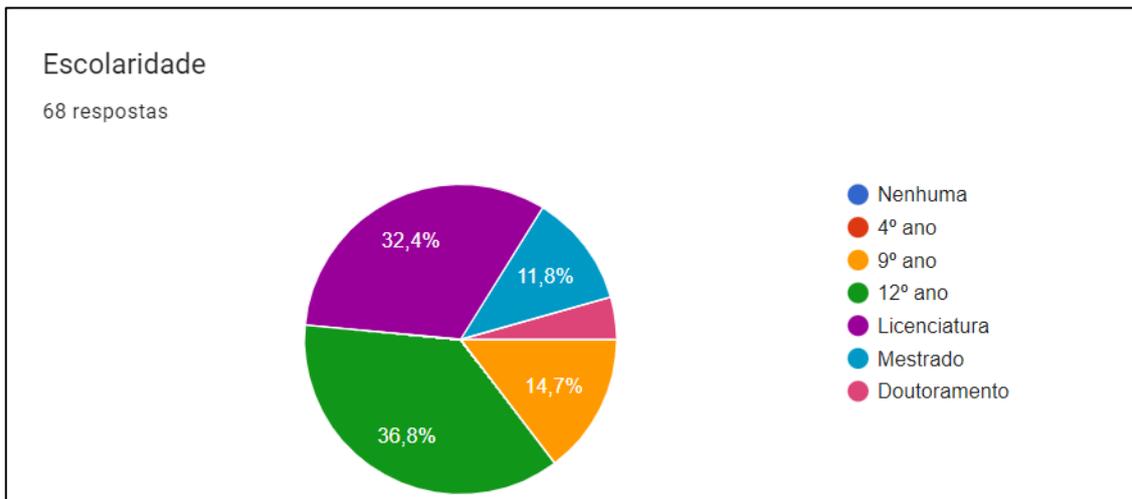


Figura 37 – Resposta Pergunta 3 (Dados Demográficos)

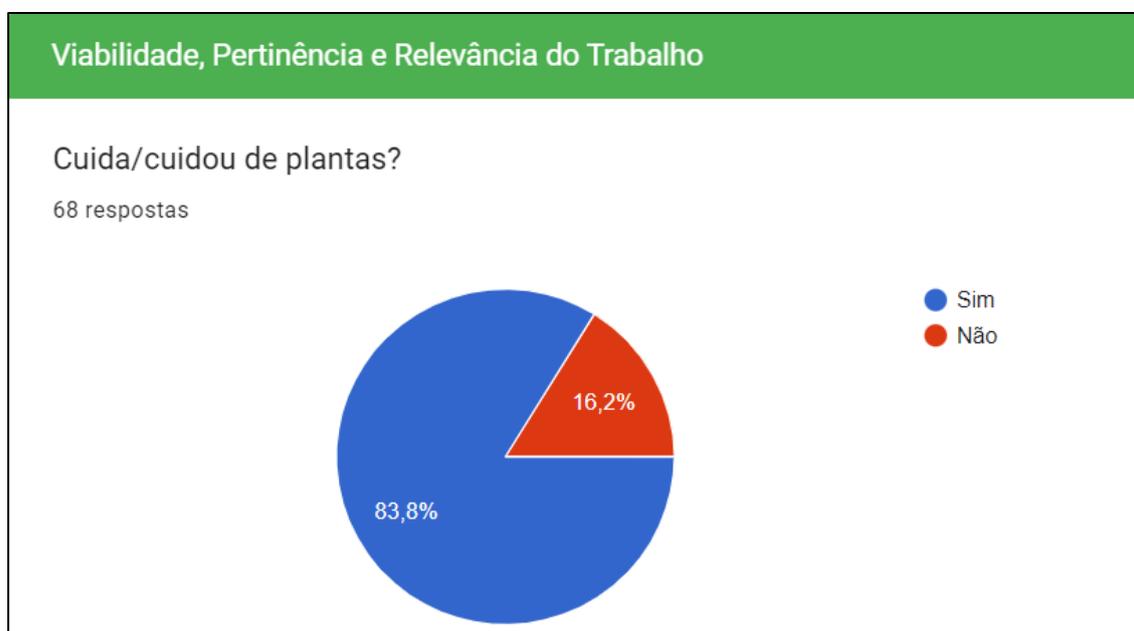


Figura 38 – Resposta pergunta 1 (Viabilidade e Pertinência do Trabalho)

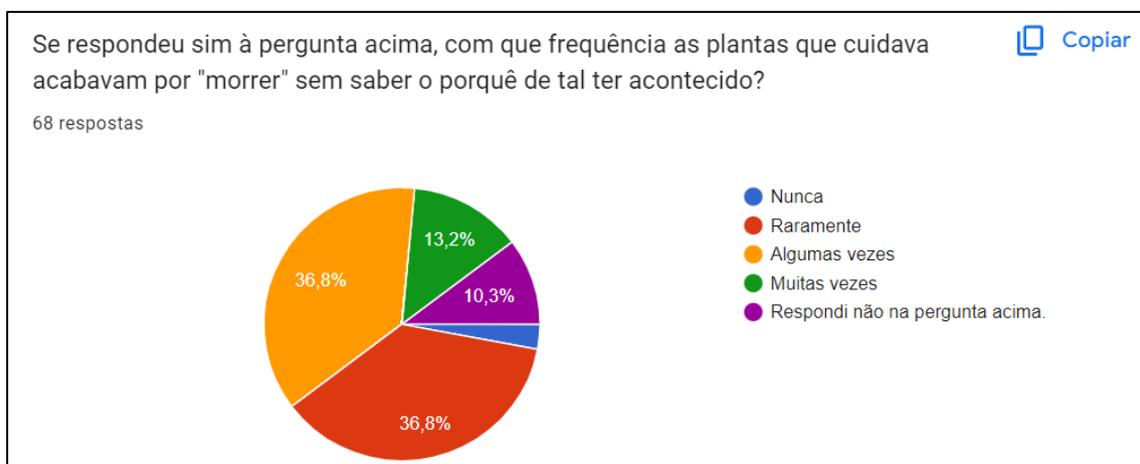


Figura 39 – Resposta pergunta 2 (Viabilidade e Pertinência do Trabalho)

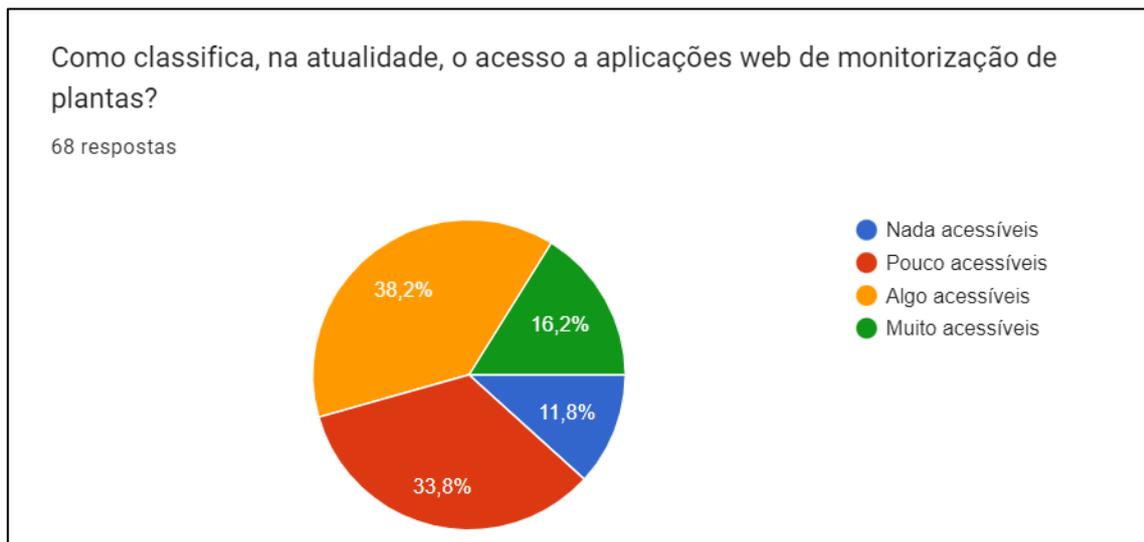


Figura 40 – Resposta pergunta 3 (Viabilidade e Pertinência do Trabalho)

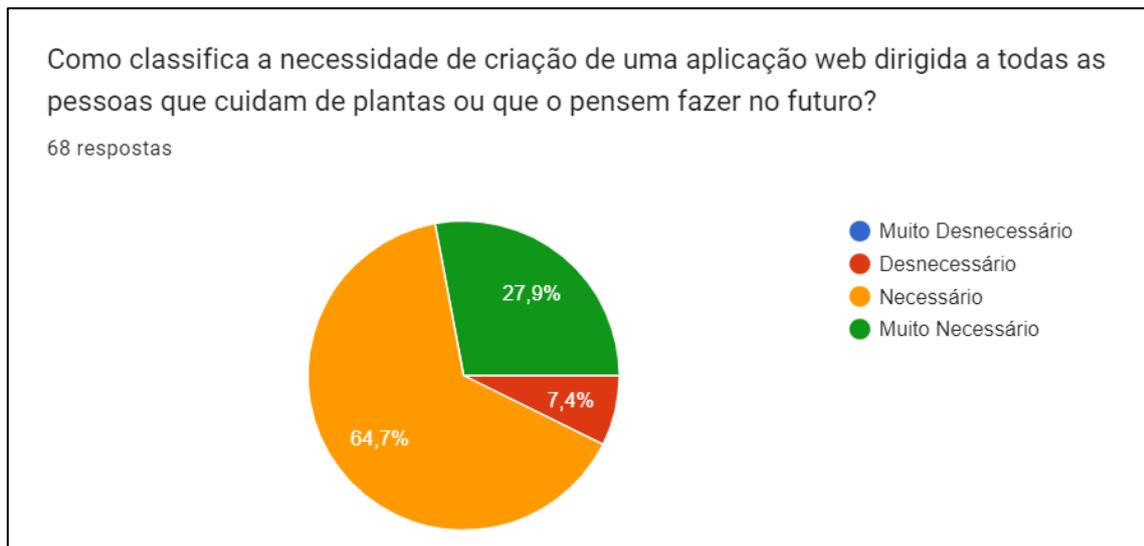


Figura 41 – Resposta pergunta 4 (Viabilidade e Pertinência do Trabalho)



Figura 42 – Resposta pergunta 5 (Viabilidade e Pertinência do Trabalho)

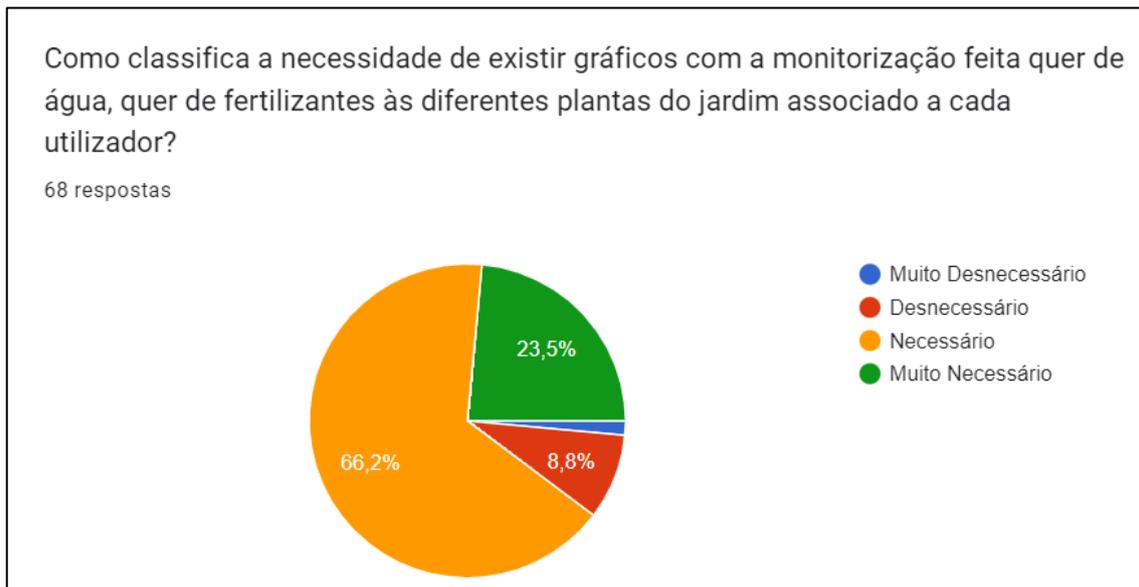


Figura 43 – Resposta pergunta 6 (Viabilidade e Pertinência do Trabalho)

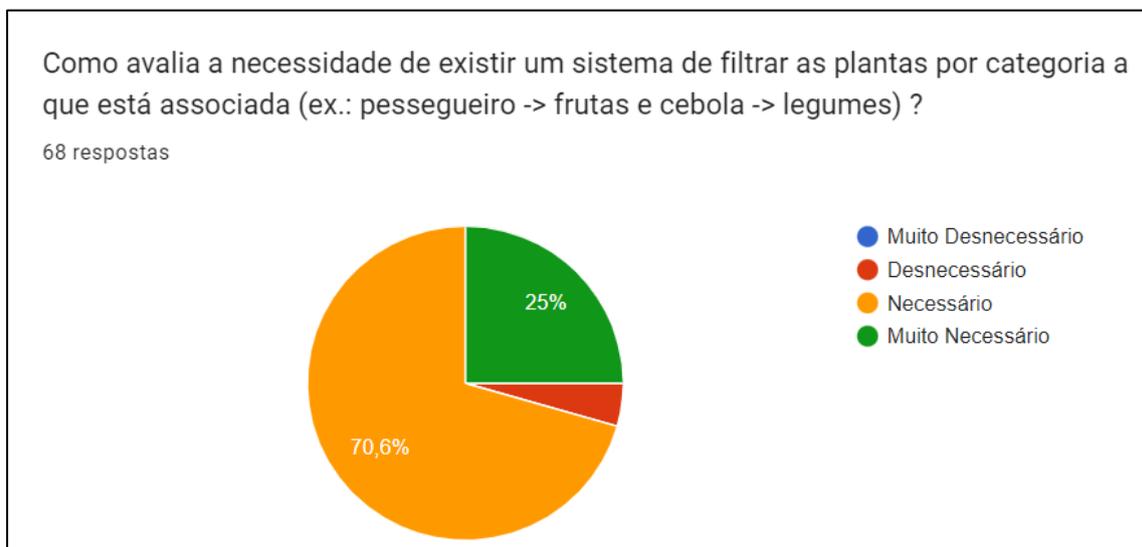


Figura 44 – Resposta pergunta 7 (Viabilidade e Pertinência do Trabalho)

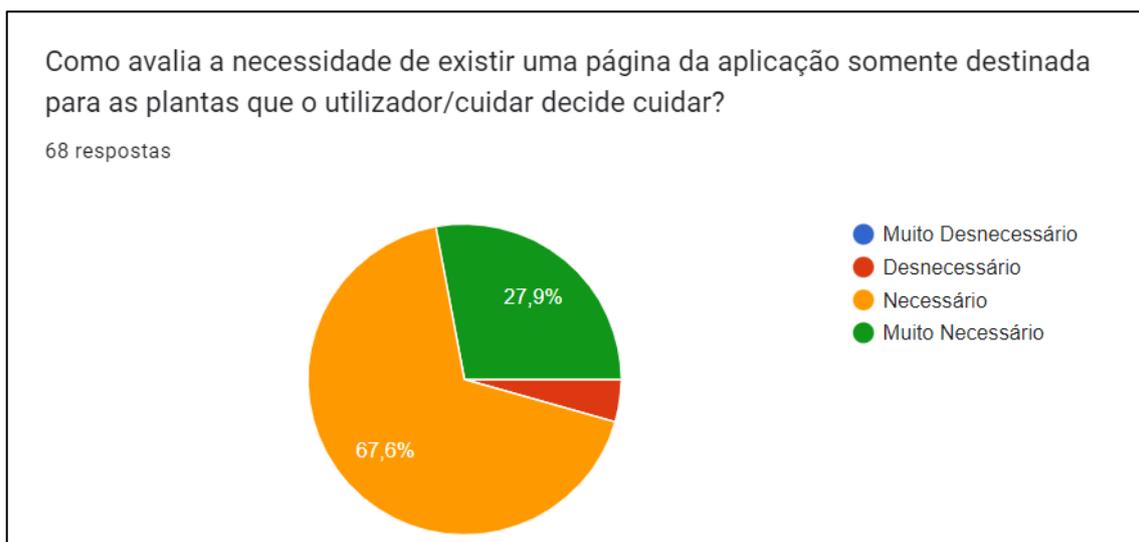


Figura 45 – Resposta pergunta 8 (Viabilidade e Pertinência do Trabalho)

Anexo 3 – Mockups, story board e mapa aplicacional

Nesta secção serão apresentados os restantes Mockups relativamente ao tópico [Mockups], situado na secção 4.6 do relatório.

Na figura abaixo [Figura 46] é apresentada a página de Login onde o utilizador introduz as credenciais Email e Password de modo a poder entrar no dashboard pessoal.

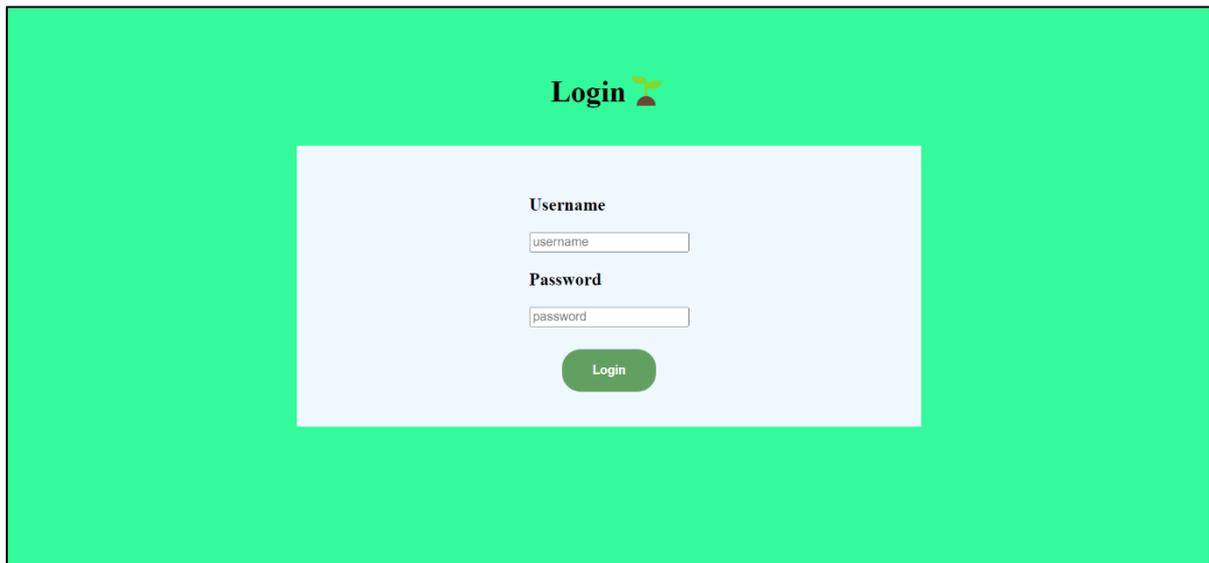


Figura 46 – Página de Login

Na figura seguinte [Figura 47] é apresentado o ecrã de splashscreen da aplicação, onde o utilizador se encontra mal a aplicação começa a correr.



Figura 47 – Splashscreen

Após o utilizador clicar na secção de meteorologia [Figura 13Figura 11], o utilizador é direcionado para esta página [Figura 48] onde pode visualizar o tempo para os 5 dias seguintes.



Figura 48 – Página Previsão do Tempo

Por fim, na figura seguinte [Figura 49], encontramos os detalhes de cada planta que contém: Nome, descrição, fotografia bem como as condições exigidas juntamente com um guia personalizado dos cuidados.

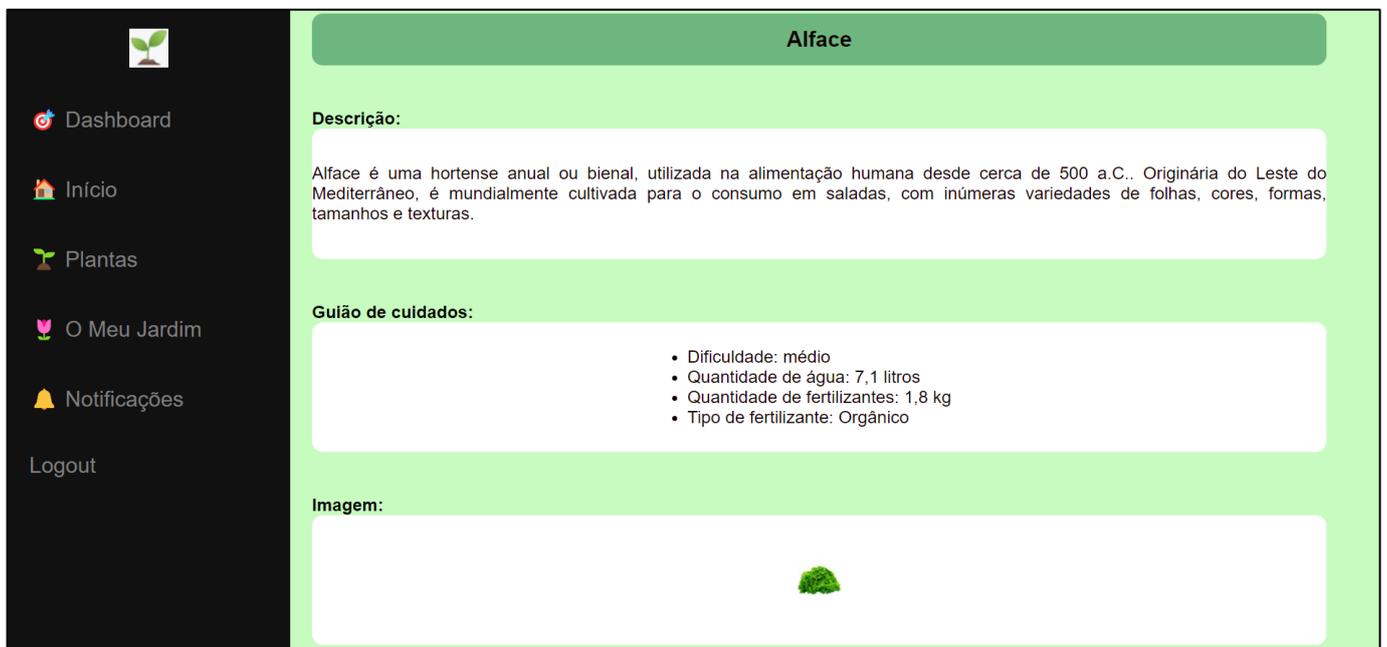


Figura 49 – Detalhe da Planta

Na figura seguinte [Figura 50] podemos ver o *Storyboard* da aplicação web que demonstra que ao clicarmos num determinado botão ou imagem que nos irá guiar por os conteúdos presentes em diferentes páginas da aplicação.

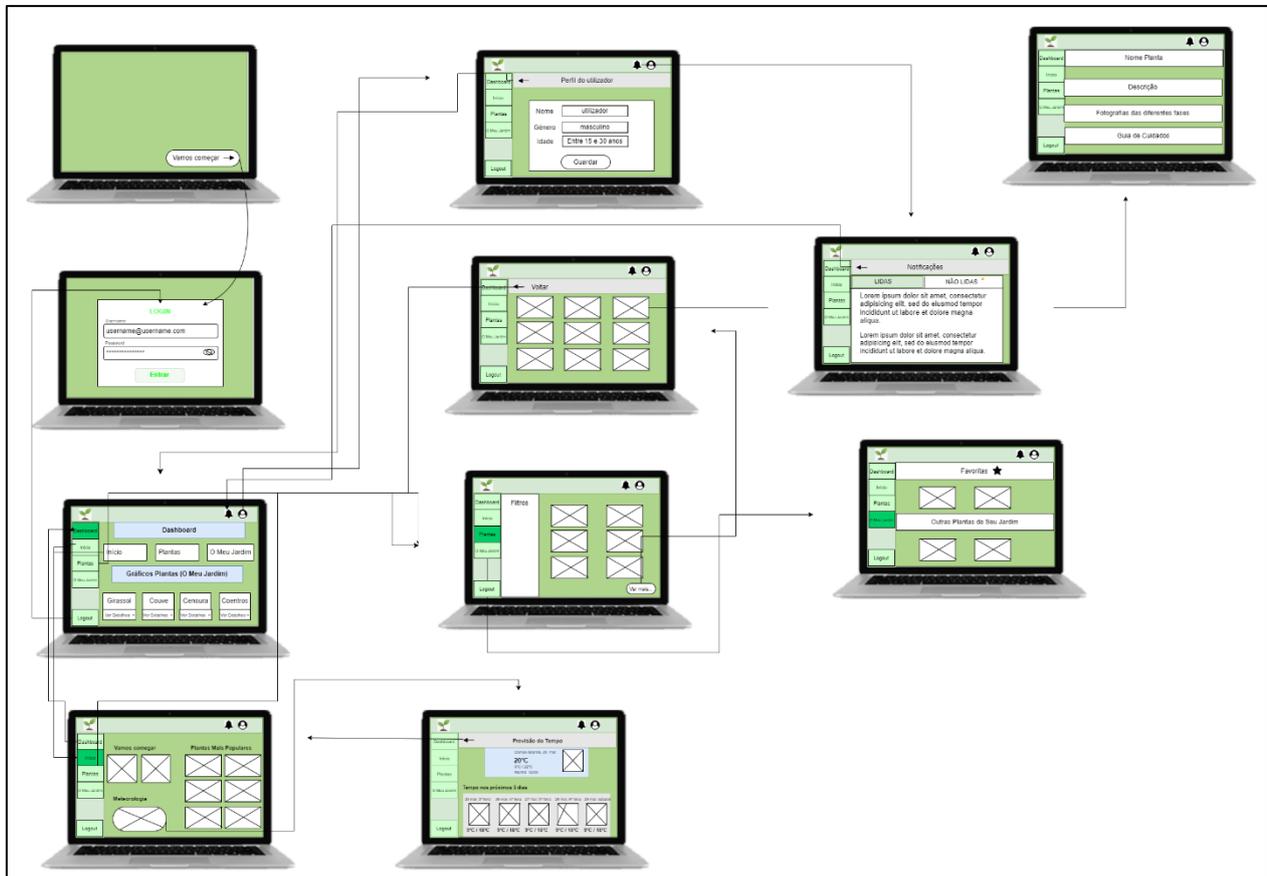


Figura 50 – Storyboard

Na seguinte figura [Figura 51] podemos ver o Mapa Aplicacional da aplicação web que demonstra que ao clicarmos num determinado botão ou imagem que nos irá guiar por outras páginas da aplicação.

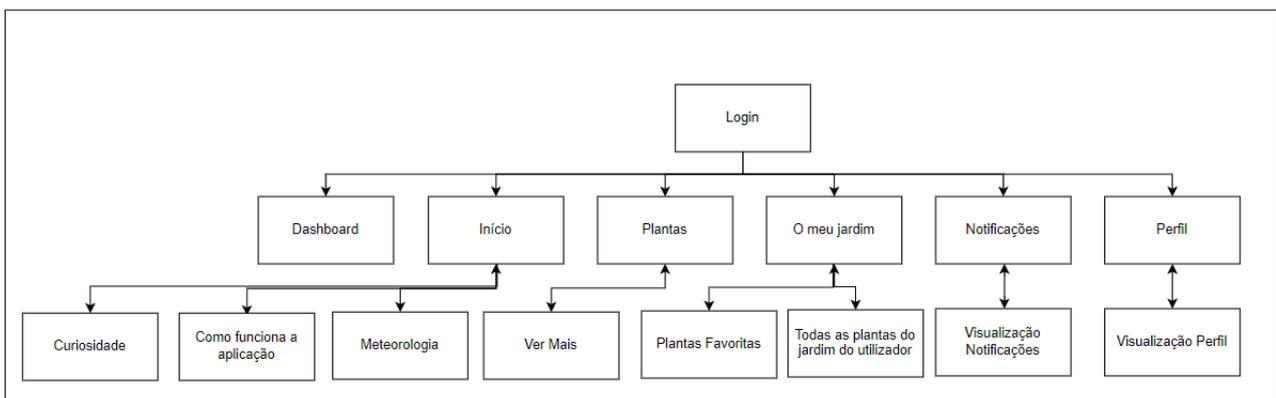


Figura 51 – Mapa Aplicacional

Anexo 4 – Progresso de trabalho

Nesta secção será representada o progresso de trabalho relativamente ao TFC desenvolvido.

Na primeira figura [Figura 52] representa a calendarização do relatório intercalar do 1º Semestre e o planeamento esperado.

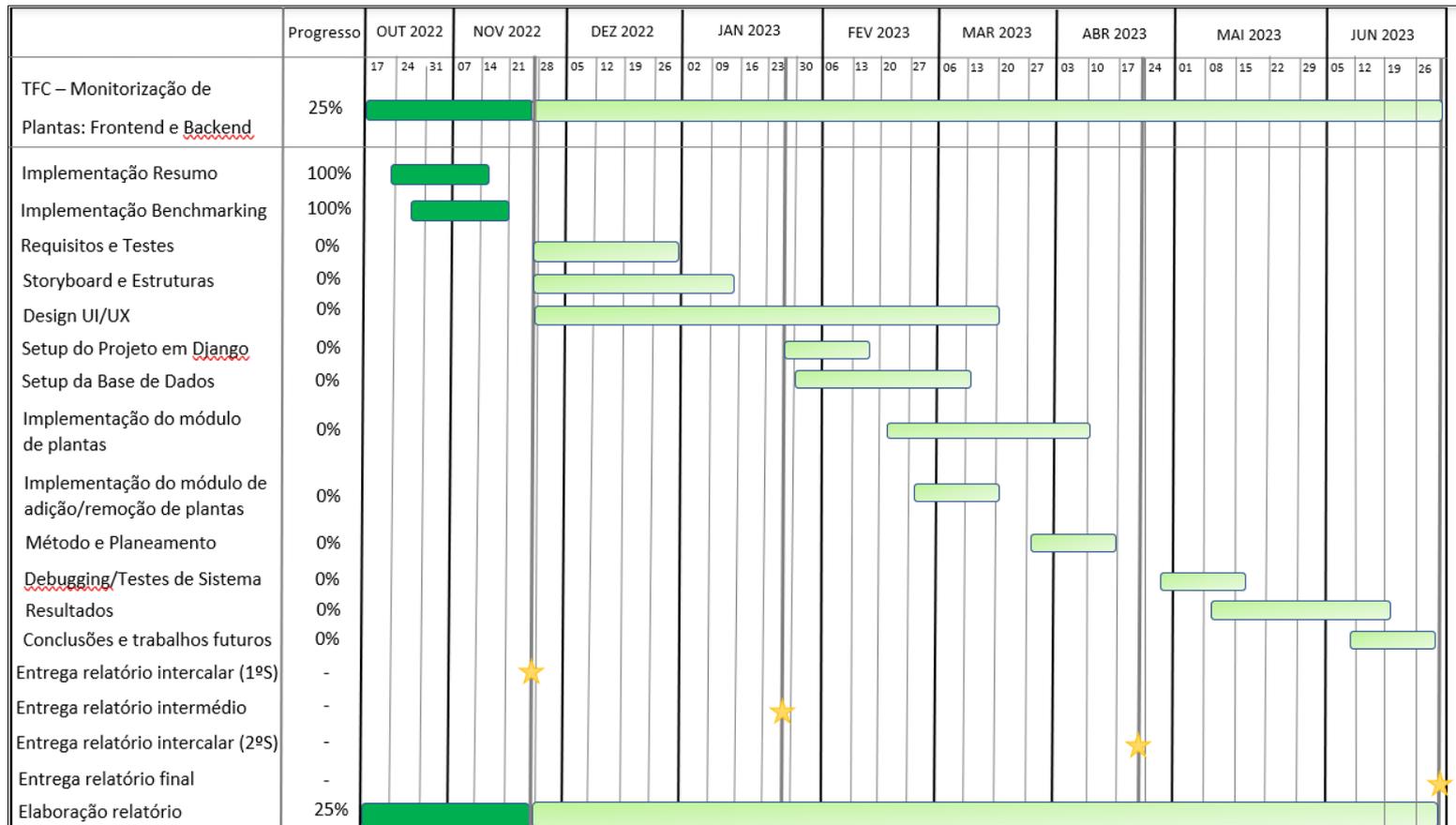


Figura 52 – Calendário Gantt Relatório Intercalar 1º Semestre

Na segunda figura [Figura 53] representa o calendário relativo à entrega do relatório intercalar e pode-se constatar que o progresso está a ser cumprido e respeitado tal como no primeiro calendário proposto.

Comparando com o calendário anterior [Figura 52], verifica-se que o planeamento está a correr de acordo com o planeamento inicialmente estipulado. Em alguns pontos nomeadamente a recolha de requisitos só foi finalizada durante o período entre a entrega intermédia e a entrega intercalar do 2º semestre.

O módulo em que o utilizador tem a possibilidade de adicionar ou remover as plantas decidi deixar para a entrega final, uma vez que a parte relativa aos dados e à construção da base de dados são processos mais demorados e mais complexos e, por isso, o adiamento deste módulo na aplicação web.

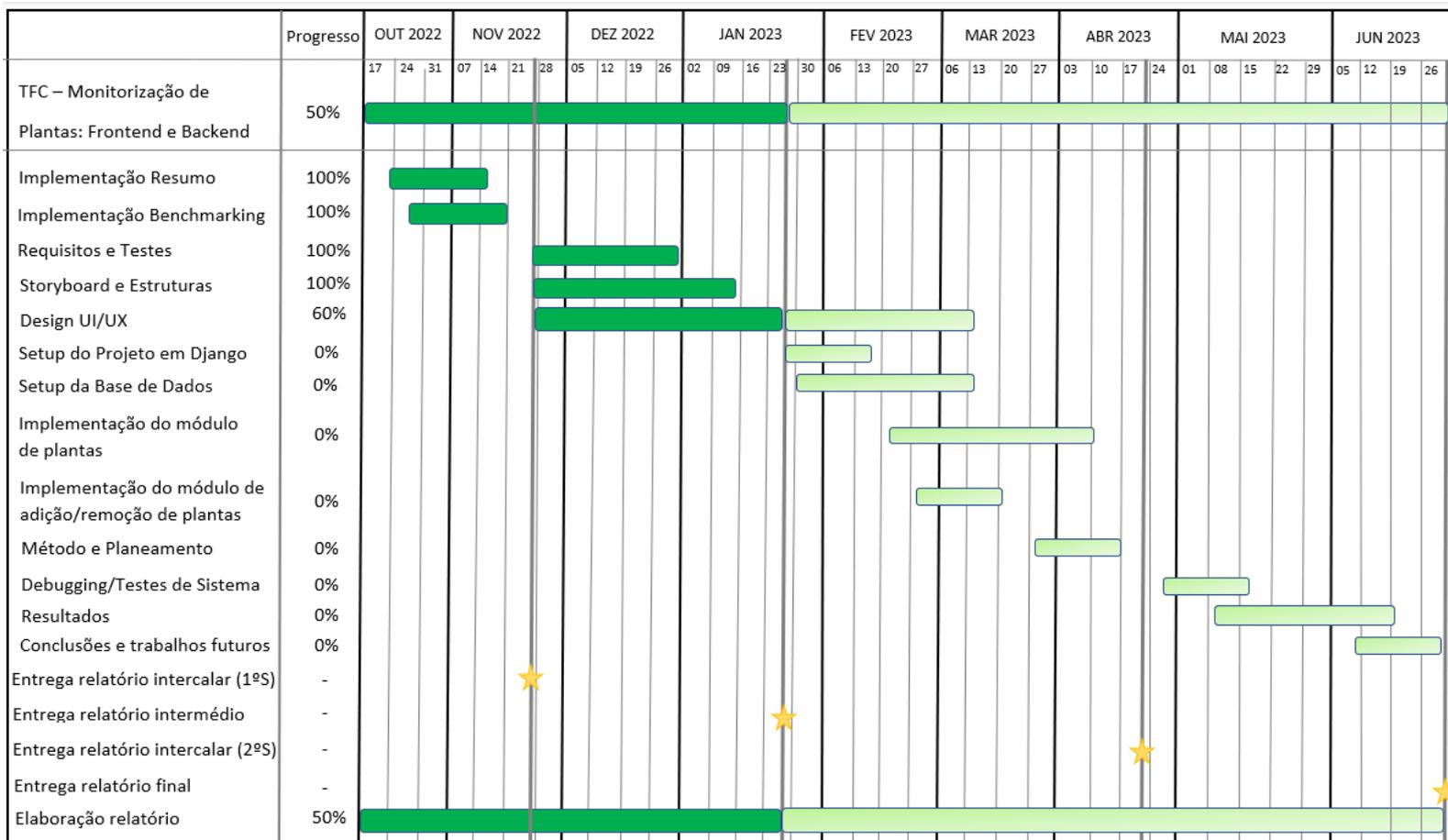


Figura 53 – Calendário Gantt Relatório Intermédio

Na terceira figura [Figura 54] representa o calendário relativo à entrega do relatório intercalar e pode-se constatar que o progresso está a ser cumprido e respeitado tal como no primeiro calendário proposto, com exceção de determinados ajustes que serão inumerados de seguida.

O setup da base de dados em SQLite3, opção defeito do Django, precisou de um ajuste de tempo de planeamento, já que são dados de 200 plantas de diferentes categorias e daí ser necessário estender o tempo de execução.

Em relação ao método e planeamento do calendário da entrega do relatório intermédio [Figura 53], foi iniciado a apresentação de resultados, porém não foi totalmente concluída neste relatório intercalar de 2º semestre sendo que será finalizado na entrega final do relatório.

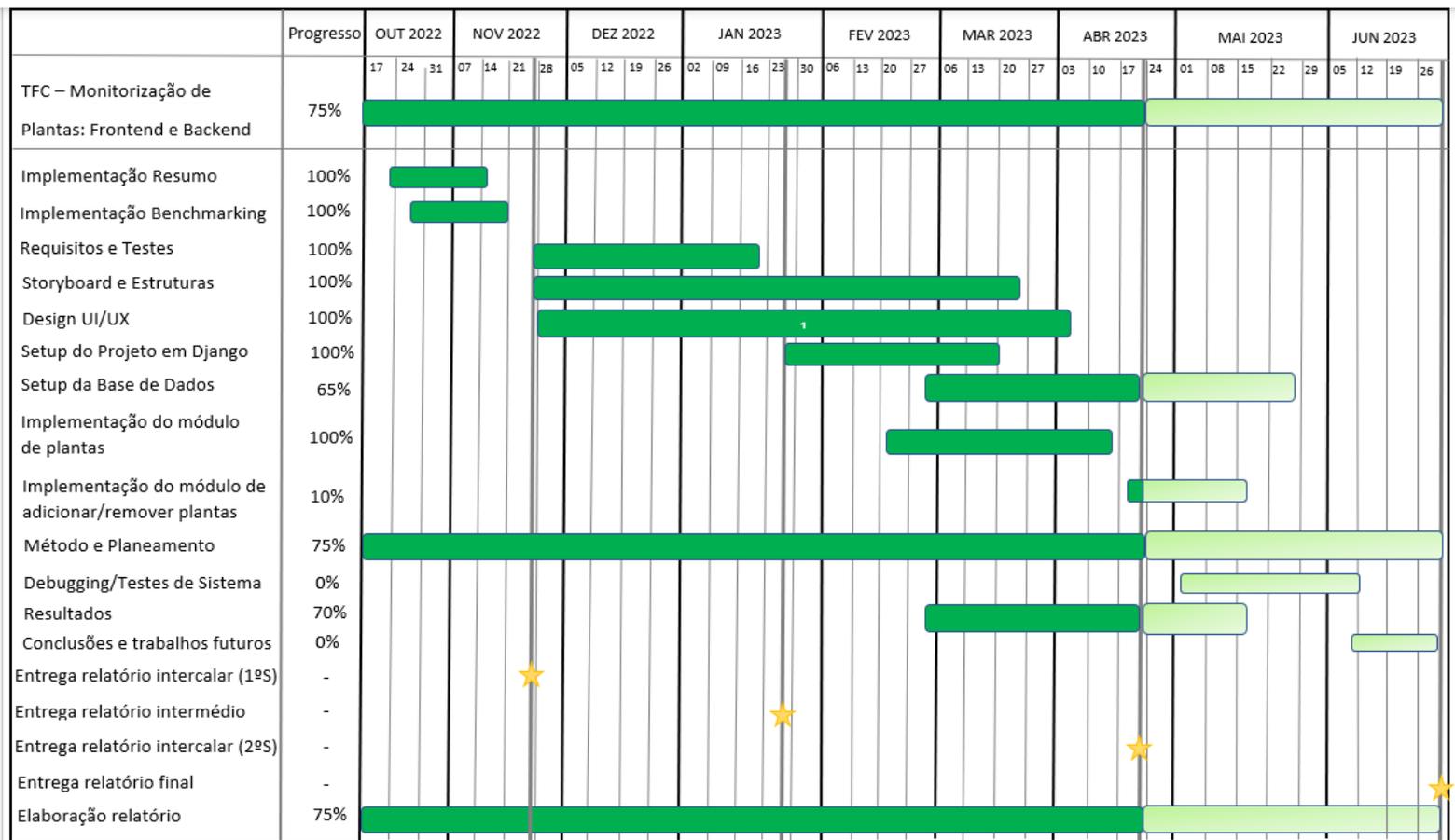


Figura 54 – Calendário Gantt Relatório Intercalar 2º Semestre

Anexo 5 – Protótipo Interativo

Este protótipo foi desenhado para a aplicação web de monitorização de plantas foi desenvolvida na plataforma Axure RP10 e tem como objetivo representar por meio de prototipagem o que a aplicação irá fazer e a forma como os utilizadores poderão mexer e navegar nas diferentes páginas web bem como no click de imagens, links e botões.

O protótipo interativo pode ser acedido aqui [\[Axur23\]](#).

Nas figuras seguintes [Figura 55], [Figura 56], [Figura 57], [Figura 58], [Figura 59], [Figura 60], [Figura 61] e [Figura 62], serão apresentadas mockups relativos ao protótipo desenvolvido.

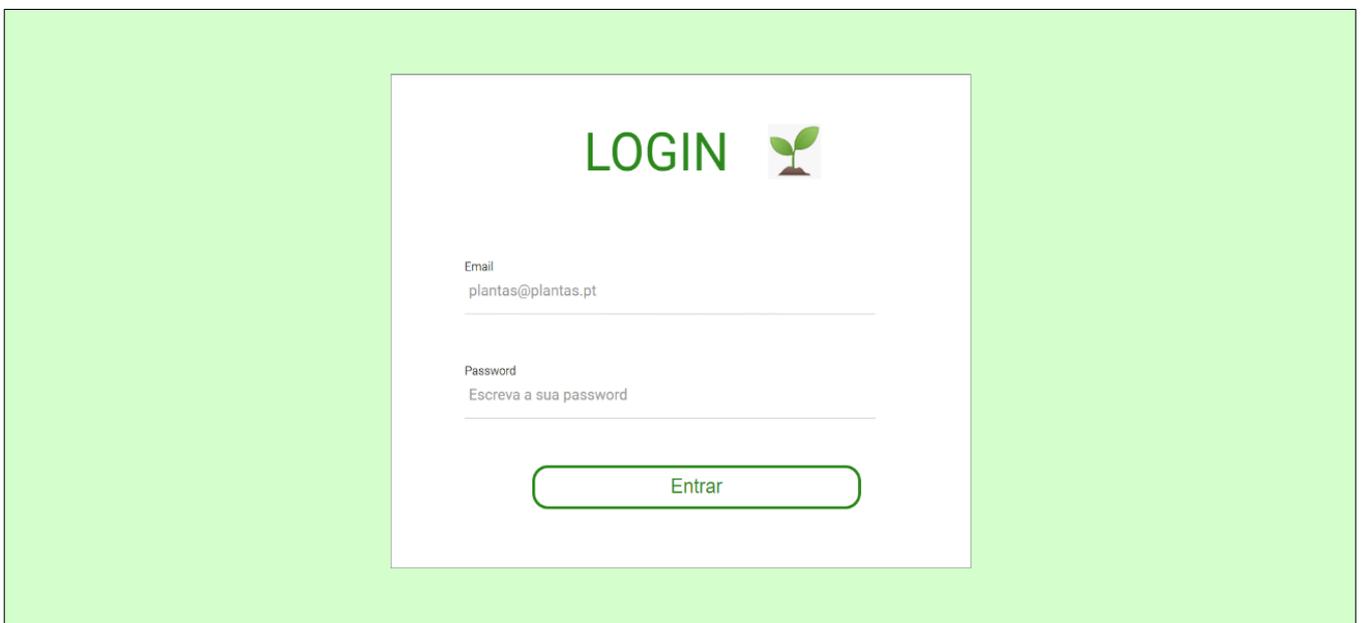


Figura 55 – Página de Login (Protótipo)

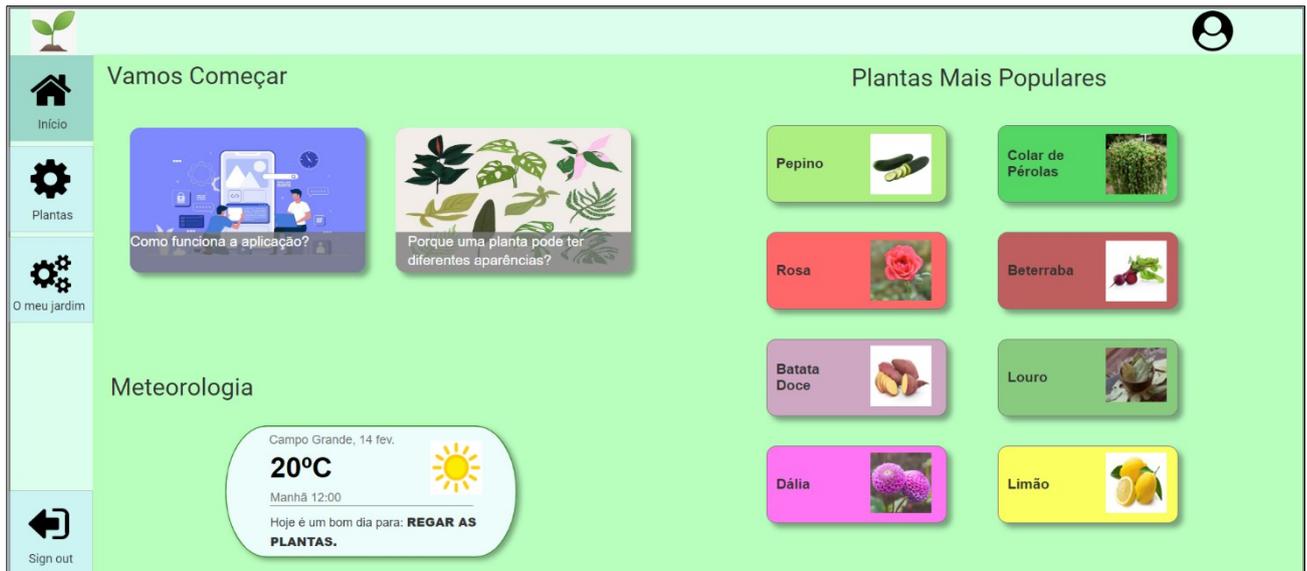


Figura 56 – Página Início (Protótipo)

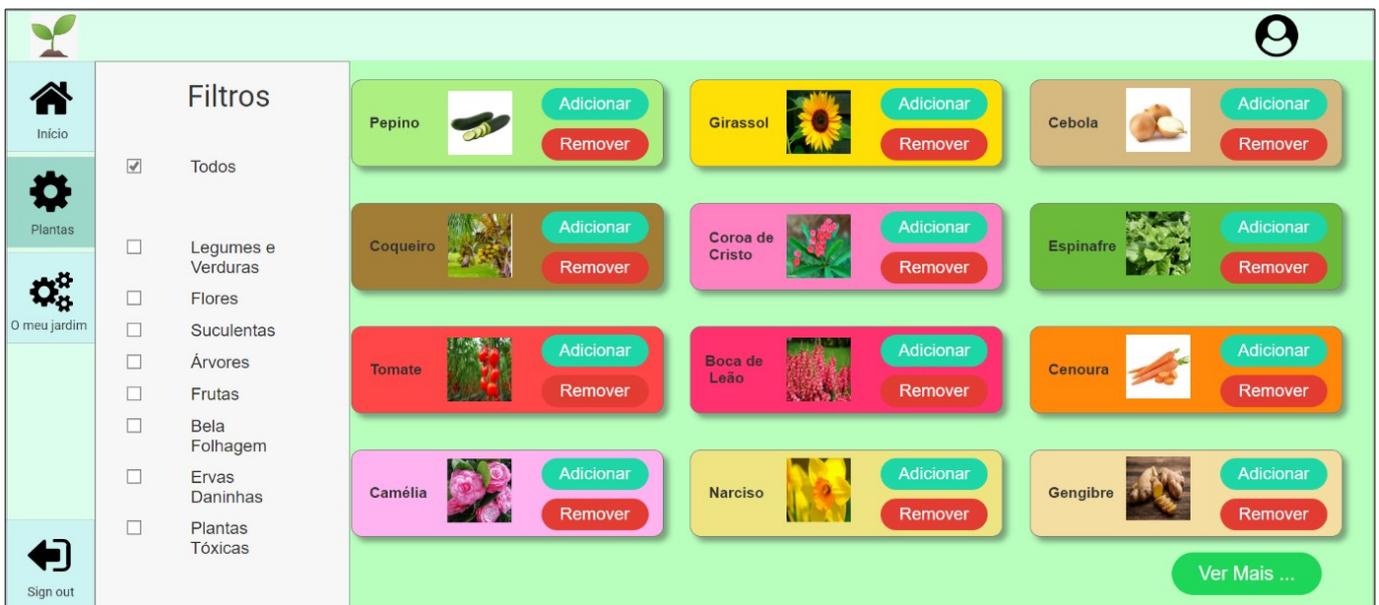


Figura 57 – Página Plantas (Protótipo)



Figura 58 – Página O Meu Jardim (Protótipo)



Figura 59 – Ecrã Perfil (Protótipo)



Figura 60 – Meteorologia onde se encontra o Utilizador (Protótipo)

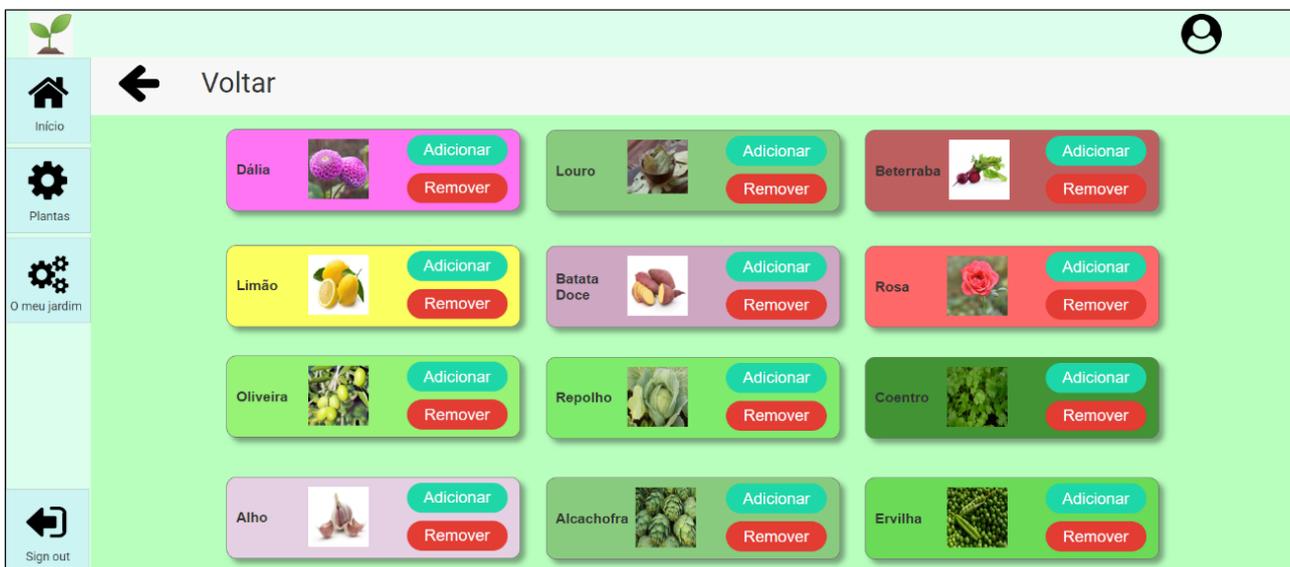


Figura 61 – Página Ver Mais Plantas (Protótipo)



Figura 62 – Detalhe da Planta (Protótipo)

Anexo 6 – Plano de Testes e Validação

De modo a testar a solução desenvolvida, quando for o seu tempo, seguiremos o plano de testes descrito na secção “Plano de Testes”. Estes testes assegurarão o correto funcionamento da plataforma antes de passar ao plano de validação, que é apresentado na secção “Plano de Validação”.

Plano de Testes

Para assegurar o bom funcionamento da plataforma antes de ser validada pelos utilizadores, esta será sujeita a testes por parte da equipa de desenvolvimento. Estes testes visam garantir que o maior número possível de requisitos [Levantamento e Análise de Requisitos] foi concretizado, e da maneira correta.

Uma vez que o desenvolvimento da solução ainda está apto para ser testado, foi concebida uma tabela de testes, onde os resultados obtidos serão mostrados. A seguinte tabela [Tabela 6] foi concebida com base na validação anterior, do protótipo interativo.

Título	Descrição	Requisito(s)	Resultado Esperado
Autenticação	O utilizador entra na aplicação através da sua conta	R.F.1, R.N.F.1	Login bem sucedido, utilizador é levado ao seu dashboard.
Aceder aos detalhes das plantas	O utilizador utiliza o menu de navegação para aceder à página das plantas e visualizar os seus detalhes	R.F.5	O utilizador é levado à lista das plantas
Adicionar e remover plantas do jardim do utilizador	O utilizador consegue adicionar e remover as plantas do seu jardim através do botões associados aos mesmos	R.F.4, R.F.6, R.N.F.10	O utilizador adicione e remova as plantas

Visualizar notificações	O utilizador consegue visualizar as notificações que recebeu	R.F.2, R.N.F.5, R.N.F.6	O utilizador visualiza e observa as notificações que tem
Visualizar a meteorologia	O utilizador utiliza o menu de navegação para aceder à página início onde visualiza a meteorologia	R.F.7, R.N.F.8	O utilizador consegue ver o tempo para o dia e para os 5 dias seguintes
Visualizar curiosidades	Na página início o utilizador consegue aceder a curiosidades	R.F.11	O utilizador vê as curiosidades “Espécies e variedades” e “Aparência diferentes nas plantas”
Registar dia de plantação	A aplicação regista o dia em que o utilizador começou o tratamento de uma planta	R.F.8	A planta ao ser adicionada ao jardim é registado o dia de modo a ser possível enviar notificações
Cifra de encriptação	A aplicação deverá ter embutido uma cifra de encriptação quando efetuar login	R.N.F.3	A aplicação deverá ter embutido uma cifra de encriptação
Perfil	O utilizador apresenta um perfil com os seus dados pessoais	R.F.10	O utilizador tem a sua informação guardada na secção perfil
Guardar nos favoritos	O utilizador ao adicionar uma planta ao seu jardim pode adicioná-la como sendo “favorita”	R.F.9	Ao adicionar uma planta ao jardim esta pode ser considerada “favorita”

Tabela 6 – Testes para Solução Desenvolvida

Plano de Validação

Como descrito na secção [Solução Proposta], foi desenvolvido um protótipo interativo para validar o fluxo de trabalho da plataforma disponível no [Anexo 5 – Protótipo Interativo] realizado com o intuito de ajudar o utilizador a familiarizar-se e explorar todas as funcionalidades da aplicação de uma maneira eficaz.

Estes testes têm a finalidade de alinhar a aplicação com as necessidades da organização, e, posteriormente, ser alvo de melhoria. No caso da validação da solução desenvolvida o procedimento será bastante semelhante.

Glossário

LEI	Licenciatura em Engenharia Informática
TFC	Trabalho Final de Curso
IA	Inteligência Artificial
GmbH	Gesellschaft mit beschränkter Haftung
R.T.	Requisito Técnico
R.F.	Requisito Funcional
R.N.F.	Requisito Não Funcional
E.R.	Entidade-Relação
HTML	Hyper Text Markup Language
CSS	Cascading Style Sheet
MVC	Model View Controller
MVT	Model View Template
IDE	Integrated Development Environment
PaaS	Plataform as a Service