



UNIVERSIDADE  
LUSÓFONA

# Despensa Virtual

## Trabalho Final de curso

Relatório Final

Nome do Aluno: FREDERICO NINA MIGUEL AZEVEDO

Nome do Aluno: TIAGO TORRES MESQUITA DA SILVA

Nome do Orientador: PEDRO PERDIGÃO

Trabalho Final de Curso | LIG | 02 de Julho 2022

[www.lusofona.pt](http://www.lusofona.pt)

## **Direitos de cópia**

*Relatório Intercalar trabalho Final de Curso – Despensa Virtual, Copyright de Frederico Nina Miguel Azevedo e Tiago Torres Mesquita da Silva, 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**

Este trabalho tem como objetivo o desenvolvimento de uma nova aplicação móvel cuja principal finalidade é combater o desperdício alimentar que possa eventualmente existir em cada uma das nossas habitações.

Para tal, a aplicação terá de identificar cada produto - a que acresce a introdução da sua respetiva data de validade - que o utilizador tenha ou venha a adquirir.

Os produtos serão, preferencialmente, introduzidos e catalogados na aplicação através do seu respetivo código de barras usando, para isso, a câmara do *smartphone*. Facilita-se assim o processo de identificação e rápida inserção de qualquer artigo escolhido.

A aplicação, ao identificar um produto, irá permitir ao utilizador - caso este assim o deseje -, definir a data de validade do mesmo.

Assim, o utilizador, além de poder ter todos os produtos adquiridos introduzidos na aplicação para sua consulta e melhor gestão e organização destes, pode também controlar o prazo de validade de cada um e ter todas as informações sobre os artigos que tem na sua despensa no mesmo lugar.

Desta forma, o desperdício alimentar poderá ser minimizado ou, idealmente, eliminado.

## **Abstract**

This project aims to develop a new mobile application whose main objective is to combat food waste that may eventually occur in each of our homes.

For this purpose, the application will have to identify each product – may also introduce its respective expiration data - that the user has or will purchase.

The products will, preferably, be introduced and cataloged in the application through their respective barcode, using for this purpose a smartphone camera. This facilitates the process of identifying and quickly inserting any chosen item.

The application, when identifying a product, will allow the user - if desired - to define its expiration date.

Thus, the user, in addition to having all purchased products inserted into the application for consultation and better management and organization, can also control the validity period of each one have all the information about the items in their pantry in the same place.

In this way, food waste can be minimized or, ideally, eliminated.

# Índice

Resumo .....	iii
Abstract.....	iv
Índice.....	1
Lista de Figuras.....	2
Lista de Tabelas .....	3
1 Identificação do Problema .....	4
2 Viabilidade e Pertinência.....	5
3 Levantamento e Análise dos Requisitos .....	10
4 Solução Proposta.....	12
4.1 Código de Barras .....	13
4.2 Tecnologias Utilizadas .....	13
4.2.1 Ionic.....	14
4.2.2 NodeJS.....	14
4.2.3 ExpressJS .....	14
4.2.4 Cors.....	14
4.2.5 JWT .....	15
4.2.6 MongoDB.....	16
4.3 Arquitetura da Aplicação .....	17
4.4 Modelo de Entidade Relação.....	18
5 Benchmarking .....	19
6 Calendário .....	22
7 Trabalhos Futuros.....	23
8 Conclusão.....	24
Bibliografia .....	25
Anexo 1 – Questionário.....	26
Glossário .....	27

## Lista de Figuras

Figura 1 - Respostas à Questão 2	5
Figura 2 - Respostas à Questão 3	5
Figura 3 - Respostas à Questão 4	6
Figura 4 - Respostas à Questão 7	6
Figura 5 - Respostas à Questão 8	6
Figura 6 - Respostas à Questão 9	7
Figura 7 - Respostas à Questão 10	7
Figura 8 - Exemplo de código de barras	13
Figura 9 - JWT Token	15
Figura 10 - Arquitetura Aplicacional	17
Figura 11 - Modelo de Entidade-Relação	18
Figura 12 - App 'Datas de Validade'	19
Figura 13 - App 'Food Check List'	19
Figura 14 - App 'My Pantry'	20
Figura 15 - App 'Fridge Buddy'	21
Figura 16 - Cronograma de TFC	22

## **Lista de Tabelas**

Tabela 1 - Requisitos Não Funcionais .....	10
Tabela 2 - Requisitos Funcionais .....	11
Tabela 3 - Tecnologias utilizadas .....	13

# 1 Identificação do Problema

É frequente adquirirmos produtos (alimentares e não só), que iremos guardar em diversos lugares das nossas casas: seja numa despensa, num armário, no frigorífico, entre outros. Essa dispersão, aliada ao facto de cada produto ter a sua respetiva data de validade, diferentes entre si, leva a que, por vezes, deixemos passar o prazo de muitos artigos que adquirimos. Inevitavelmente, enviamos para o lixo um produto que nunca foi aberto e que acabou por ficar esquecido e fora da validade.

Feitas pesquisas, sondagens e realizados inquéritos públicos - que iremos explicar detalhadamente na parte de Viabilidade e Pertinência deste projeto -, chegámos à conclusão de que uma grande parte das pessoas inquiridas (44,1%) não sabe com que frequência verifica a validade dos produtos que tem em casa. Associado a este facto, estariam recetivas a utilizar uma aplicação que lhes desse essa possibilidade (82,4%).

A maior vantagem que a maioria das pessoas identificou numa aplicação como esta - que irá ser desenvolvida - é a possibilidade de conseguir minimizar e até eliminar por completo o desperdício alimentar que, inevitavelmente, acaba por acontecer em cada casa.

Após uma pesquisa no mercado, tanto de aplicações para Android como para IOS, verificámos que as aplicações existentes, que possam ser concorrentes à nossa, têm diversas lacunas e problemas. Como exemplo, algumas delas têm um UI demasiado complexos e confusos, que podem levar, logo à partida, a um afastamento dos seus potenciais utilizadores. Noutras, a publicidade intrusiva e constante é bastante limitadora do propósito da própria aplicação, mas iremos abordar com mais profundidade esta temática no capítulo de Benchmarking deste relatório.

Detetado o problema existente - e com o propósito de dar resposta a este -, iremos desenvolver uma aplicação móvel que seja de fácil compreensão e utilização, com funcionalidades que vão de encontro ao que os utilizadores pretendem. O objetivo é que funcione de uma forma natural e intuitiva, ajudando facilmente a combater o desperdício alimentar.

## 2 Viabilidade e Pertinência

De forma a reforçar tanto a viabilidade como a pertinência deste projeto elaborámos um Questionário público e anónimo para analisar a recetividade do mercado a esta aplicação móvel que será criada.

A seguir podemos ver o resultado e as estatísticas das respostas mais relevantes – num universo de 34 pessoas inquiridas:

2. Com que frequência verifica a validade dos produtos que tem em sua casa?

34 respostas

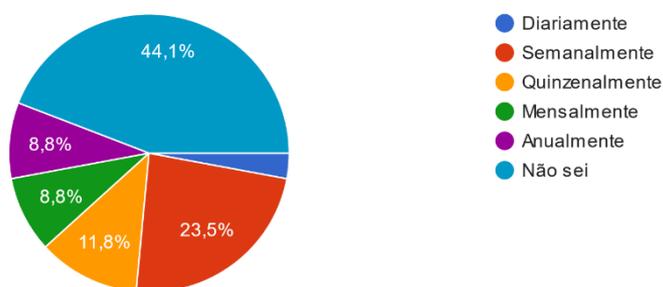


Figura 1 - Respostas à Questão 2

3. Tem por hábito usar alguma aplicação para gerir as suas compras?

34 respostas

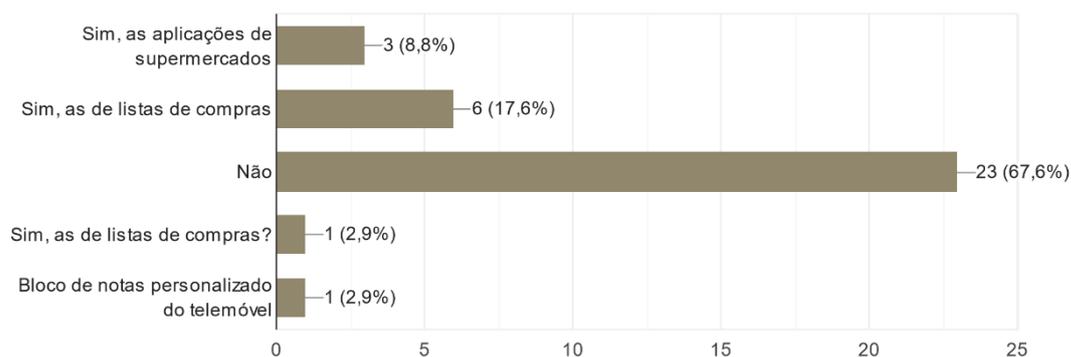
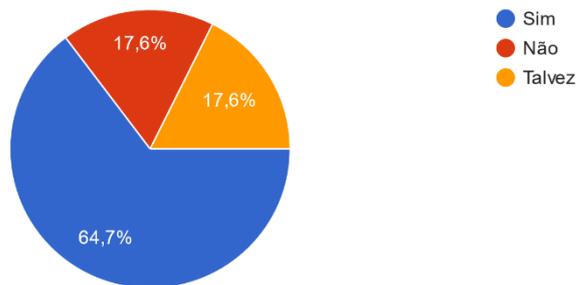


Figura 2 - Respostas à Questão 3

4. Utilizaria uma aplicação para saber o que tem na despensa?

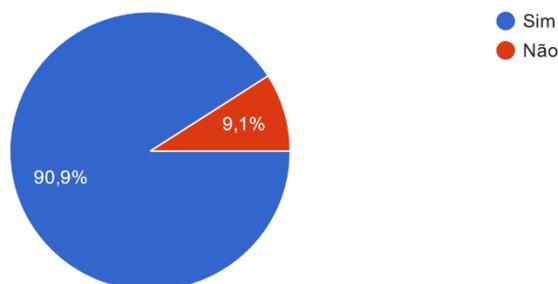
34 respostas



**Figura 3 - Respostas à Questão 4**

7. Daria permissão para que a app usasse a câmara do seu smartphone de modo a identificar um produto através do código de barras?

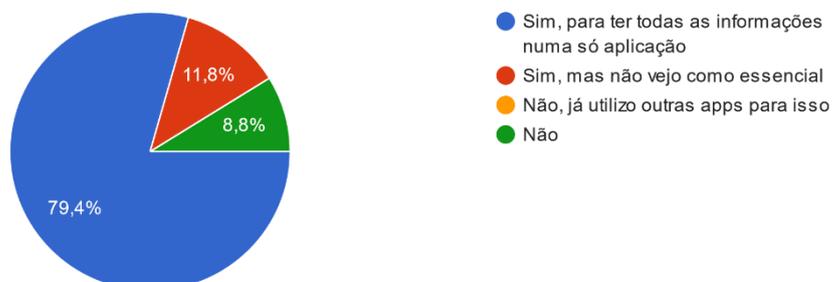
33 respostas



**Figura 4 - Respostas à Questão 7**

8. Pensa que uma aplicação deste género deveria também disponibilizar uma funcionalidade para fazer a sua lista de compras?

34 respostas



**Figura 5 - Respostas à Questão 8**

9. Para que género de produtos utilizaria esta aplicação?

34 respostas

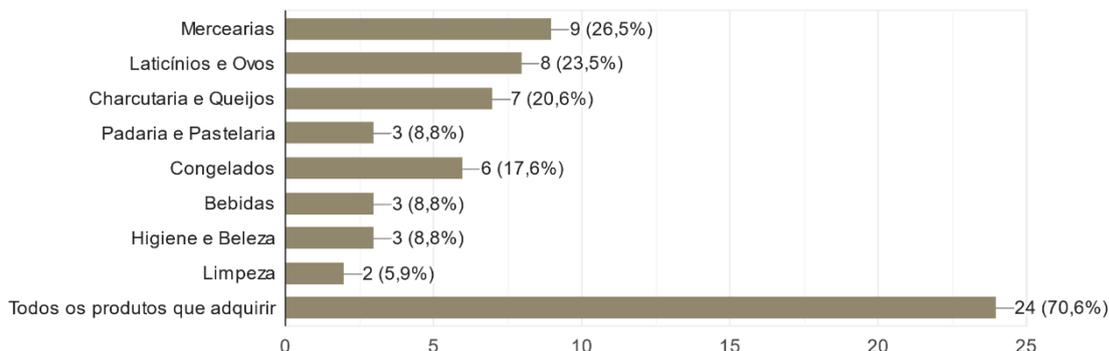


Figura 6 - Respostas à Questão 9

10. Que vantagens vê num projeto como este?

34 respostas

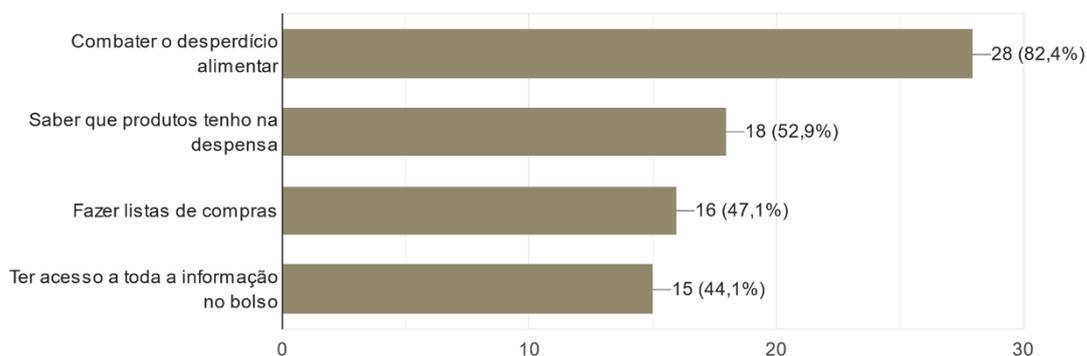


Figura 7 - Respostas à Questão 10

Fazendo uma análise aos resultados do questionário podemos tirar as seguintes conclusões:

- A faixa etária dominante que deu resposta a este inquérito foram pessoas entre os 18 e os 40 anos - representando 67,7% do total da nossa amostra.
- O maior grupo - representando 44,1% das pessoas inquiridas - não sabem com que frequência verificam a validade dos produtos que têm em casa, ver [Figura 1](#). O facto de o segundo maior grupo (com 23,5%) ter respondido que verifica a validade dos seus produtos semanalmente não vai, necessariamente, invalidar que tenha interesse em usar uma aplicação como a nossa. O tempo despendido nessas tarefas

pode ser facilmente reduzido ao consultar a aplicação e ver a sua lista de produtos e respetivas validades.

- A grande maioria dos inquiridos (67,6%) não utiliza aplicações para gerir as suas compras. Apenas 17,6% tem alguma, como as de listas de compras, ver [Figura 2](#).
- Quando perguntados sobre se utilizariam uma aplicação para saber o que têm na despensa, a maioria (64,7%) respondeu afirmativamente, como se pode ver na [Figura 3](#). Esta larga percentagem dá-nos um bom indicador da recetividade que a nossa aplicação poderá ter no mercado.
- Já por questões relacionadas com privacidade e segurança, perguntámos se os inquiridos dariam permissão para que a aplicação usasse a câmara dos seus smartphones, de modo a identificar os produtos através dos seus códigos de barras. Apenas 9,1% respondeu que não daria permissão. Os restantes 90,9% não iriam ter qualquer problema com esta questão - como se pode observar na [Figura 4](#).
- Para tentar perceber se os potenciais utilizadores da nossa aplicação estariam ainda interessados numa funcionalidade que lhes permitisse fazer a sua lista de compras obtivemos as seguintes respostas - ver [Figura 5](#):
  - 79,4% disseram 'sim, para ter todas as informações numa só aplicação'.
  - 11,8% responderam 'sim, embora não seja uma funcionalidade essencial'.
  - 8,8% 'não' estão interessados em ter essa funcionalidade.
- Das várias categorias existentes para os produtos, quisemos saber se haveria alguma de maior interesse para os eventuais utilizadores da nossa aplicação. Com o resultado de 70,6%, podemos constatar que a maioria iria utilizar a aplicação para listar todos os produtos que adquirir – ver [Figura 6](#). Desta forma, iriam ter um maior controlo dos artigos, evitando assim que alguns possam ficar esquecidos com o tempo.
- Por fim, quisemos saber quais as vantagens que as pessoas poderiam ver num projeto como este. Tendo a pergunta a possibilidade de respostas várias – cada inquirido pôde selecionar mais do que uma opção - obtivemos os seguintes resultados ([Figura 7](#)):
  - 'Combater o desperdício alimentar' - 82,4% das respostas dadas (correspondente a 28 escolhas).
  - 'Saber que produtos tenho na despensa' - 52,9% das respostas dadas (correspondente a 18 escolhas).
  - 'Fazer listas de compras' - 47,1 % das respostas dadas (correspondente a 16 escolhas).
  - 'Ter acesso a toda a informação no bolso' - 44,1% das respostas dadas (correspondente a 15 escolhas).

Pelas respostas dadas, podemos concluir que, num universo de 34 inquiridos, a maioria - 28 destes - vê como a maior vantagem num projeto como este o combate ao desperdício alimentar.

Analisando os resultados de todas as respostas ao nosso inquérito podemos reforçar a ideia de que existe uma lacuna no mercado acerca de aplicações que permitam aos utilizadores ter controlo dos produtos que adquirem.

A solução que apresentamos tem o potencial de ser continuada e não se esgotar na sua base – um trabalho académico. Será uma ferramenta que os utilizadores poderão usar no seu dia a dia, enquanto ‘ajuda’ nas compras e na listagem dos artigos que têm em casa.

Em última análise, a aplicação poderá igualmente ajudar as pessoas que a usam a poupar não só os recursos como também dinheiro – ao não adquirirem produtos que estavam ‘perdidos’ nos armários ou não deixando terminar o prazo de algum deles.

### 3 Levantamento e Análise dos Requisitos

Requisitos não funcionais são aqueles que não estão conectados diretamente com as funções específicas do sistema, pois influenciam diretamente na execução do sistema, especificam ou restringem as propriedades emergentes do sistema, como usabilidade, confiabilidade, disponibilidade, desempenho, portabilidade, segurança e proteção. (SOMMERVILLE, 2007, p. 82).

Código	Requisitos Não Funcionais
RNF 01	Plataformas <u>Android</u> e <u>IOS</u>
RNF 02	Desenvolvimento da aplicação em <u>ionic</u>
RNF 03	Base de dados local e também na <i>Cloud</i>
RNF 04	Aplicação gratuita e sem publicidade

**Tabela 1 - Requisitos Não Funcionais**

Requisitos funcionais vão estabelecer funções ou serviços do sistema, como o sistema deverá agir em determinadas situações. A classificação de requisitos funcionais é exigida que se faça obrigatoriamente completa (SOMMERVILLE, 2007, p. 81).

Código	Requisitos Funcionais
Utilizadores / Registo	
RF 01	O sistema permitirá que os utilizadores sejam registados
RF 02	O registo do utilizador deverá conter email e password
Utilizadores / Login	
RF 03	O sistema permitirá o acesso dos utilizadores previamente registados através de email e password
RF 04	O sistema permitirá a possibilidade de redefinição de password através de email

Utilizadores / Funcionalidades	
RF 05	O sistema pedirá ao utilizador permissão para utilizar a camara do smartphone
RF 06	O sistema irá permitir a leitura de produtos através dos seus códigos de barras
RF 07	O sistema irá permitir após a leitura dos códigos de barras adicionar os produtos à lista
RF 08	O sistema irá permitir adicionar, editar e remover os produtos adicionados 'manualmente'
RF 09	O sistema irá permitir ordenar os produtos adicionados por data, por 'adicionados recentemente', por nome, por data de validade e por categoria
RF 10	O sistema irá permitir atribuir um produto a uma categoria
RF 11	O sistema irá permitir adicionar produtos a uma lista de compras (Figura 5)
RF 12	O sistema irá permitir consultar os detalhes de cada produto
RF 13	O sistema irá permitir filtrar os produtos inseridos na lista
RF 14	O sistema irá permitir adicionar produtos sem definir qualquer data de validade
RF 15	O sistema irá permitir adicionar produtos, definindo a data de validade, apenas colocando o número de dias desejado

**Tabela 2 - Requisitos Funcionais**

## 4 Solução Proposta

Para o desenvolvimento da aplicação que nos propomos a desenvolver, acreditamos que a melhor plataforma de desenvolvimento para este projeto é Ionic, pois esta oferece um SDK de código aberto, com muitas funcionalidades e a possibilidade de desenvolver o *frontend* de aplicações móveis em Angular.

A nossa solução proposta vai consistir no seguinte:

- Compatível com o sistema Android e IOS.
- Identificar o nome dos produtos através da câmara do smartphone, através dos seus respetivos códigos de barras.
- Possibilidade de adicionar produtos através do código de barras.
- Adicionar e registar os produtos identificados na base de dados da aplicação.
- Registar a data de expiração dos produtos.
- Permitir a catalogação dos vários produtos em categorias a definir.
- Permitir a remoção de produtos da lista se o utilizador assim o pretender.
- Permitir que o utilizador crie uma lista de compras.
- UI de fácil navegação e intuitiva para os utilizadores.
- Aplicação gratuita e sem publicidade, oferecendo todas as funcionalidades disponíveis ao utilizador sem necessidade de subscrições.

## 4.1 Código de Barras

O código de barras de um produto é uma sequência numérica composta por 13 algarismos. Este tipo de codificação universal pode identificar, de forma rápida e clara, um conjunto de dados:



Figura 8 - Exemplo de código de barras

- País de origem do Produto
- Identificação a Empresa
- Identificação do Produto
- Dígito Verificador

O código de barras está preparado para ser lido através de um leitor de código de barras, permitindo assim aceder à informação do produto. Os 3 primeiros números existentes no código de barras identificam o país em que o produto foi concebido, em Portugal é o número 560.

## 4.2 Tecnologias Utilizadas

Frontend:	Backend:
<u>Ionic</u>	NodeJS
Capacitor	ExpressJS
Angular	Passport
	Cors
	JWT
	MongoDB

Tabela 3 - Tecnologias utilizadas

#### 4.2.1 Ionic

Ionic é um suporte que trabalha por cima do Capacitor para criar aplicações híbridas. Permite desenvolver uma interface de aplicação comparável com páginas web, ou seja, usando HTML, CSS e JavaScript. Essas aplicações são executadas dentro do WebView.

A virtude da estrutura ionic é ser de código aberto em que este só precisa de ser escrito uma vez e pode ser executado em vários dispositivos móveis. Existem vários plugins que podem ser utilizados no desenvolvimento da aplicação.

Esta é uma estrutura favorável ao programador, devido à sua simplicidade.

#### 4.2.2 NodeJS

O Node.js pode ser definido como um ambiente de execução Javascript server-side. Isso significa que com o Node.js é possível criar aplicações Javascript sem depender de um browser para a sua execução, como estamos acostumados.

O principal motivo de escolha do NodeJS é a sua alta capacidade de escala. Além disso, a sua arquitetura, flexibilidade e baixo custo, resultam numa boa escolha para implementação de Microsserviços e componentes da arquitetura Serverless. Inclusive, os principais fornecedores de produtos e serviços Cloud já têm suporte para desenvolvimento de soluções escaláveis utilizando o Node.js.

#### 4.2.3 ExpressJS

O Express.js é uma Framework rápida e uma das mais utilizadas em conjunto com o Node.js, facilitando o desenvolvimento de aplicações back-end.

O Express é uma framework incrível, pelo facto de ter diversas características que facilitam o desenvolvimento de aplicações.

#### 4.2.4 Cors

Esta tecnologia permite auxiliar na segurança aplicada a uma API. Isso é feito ignorando os cabeçalhos Access-Control-Allow-Origin, que especificam quais origens que podem aceder à API.

Em outras palavras, CORS é um recurso de segurança do navegador que restringe solicitações HTTP de origem cruzada com outros servidores e especifica quais domínios que podem aceder aos seus recursos.

#### 4.2.5 JWT

O JWT (JSON Web Token) é um método RCT 7519 padrão da indústria para realizar autenticação entre duas partes por meio de um token assinado que autentica uma requisição web. Esse token é um código em Base64 que armazena objetos JSON com os dados que permitem a autenticação da requisição.

Uma vez que os dados enviados pelo cliente tenham sido autenticados no servidor, este criará um token JWT assinado com um segredo interno da API e enviará este token de volta ao cliente.

Com o token autenticado, o cliente possui acesso aos endpoints da aplicação que antes lhes eram restritos. Para realizar esse acesso é preciso informar esse token no header Authorization da requisição e, por convenção, após a palavra Bearer.

A assinatura de um JSON Web Token é o seu componente mais sensível por tratar justamente da segurança deste token. Por conta disto existe uma fórmula padrão para que o token seja adequadamente assinado, exigindo que o token seja uma hash em Base64 gerada de um algoritmo de criptografia, por exemplo SHA256 ou SHA512, e essa hash precisa ser feita a partir do header e do payload do token.

O JSON Web Token é utilizado em dois principais cenários, quando queremos realizar um processo de autorização em nossa aplicação ou quando queremos realizar troca de informações.

Autorização: Este é o cenário mais comum para o uso do JWT. Depois que o usuário estiver autenticado, cada requisição subsequente incluirá o JWT, permitindo que o utilizador tenha acesso a rotas, serviços e recursos permitidos com esse token.

Troca de Informações: JSON Web Tokens são uma boa maneira de transmitir informações com segurança entre as partes. Como os JWTs podem ser assinados, por exemplo, usando pares de chaves pública e privada podemos garantir que os remetentes são quem dizem ser.

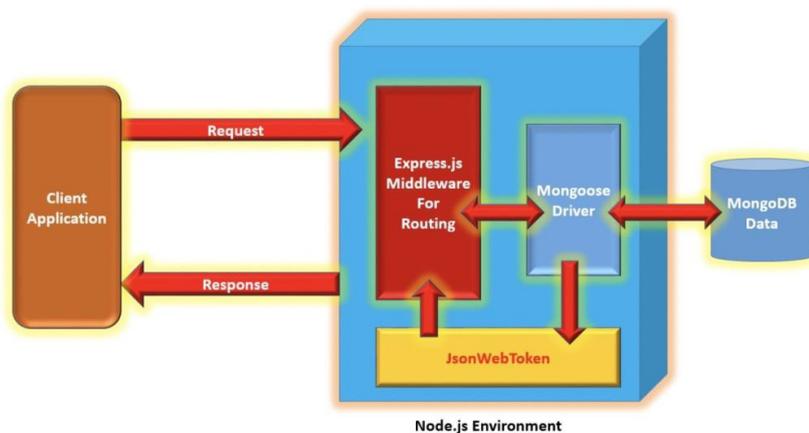


Figura 9 - JWT Token

#### 4.2.6 MongoDB

MongoDB é um programa de gestão de base de dados de código aberto. MongoDB é uma ferramenta que pode gerir informações orientadas a documentos, armazenar ou recuperar informações.

A MongoDB suporta várias formas de dados, sendo uma das muitas tecnologias de base de dados não relacionais. Em vez de usar tabelas e linhas como nas bases de dados relacionais, a arquitetura da MongoDB é composta por coleções e documentos, com cada um desses documentos a serem constituídos em atributos de chave/valor. Um único documento pode ser pensado ao equivalente a uma linha em uma tabela, com cada chave sendo semelhante a um nome de coluna e o valor de cada chave sendo semelhante ao valor da respetiva linha. Um documento não está restrito a um determinado esquema ou colunas em uma tabela pois dois documentos podem partilhar elementos semelhantes, como um campo de ID, além de ter elementos completamente diferentes.

A MongoDB também permite alterações dinâmicas, devido a ser fácil fazer alterações ágeis sem precisar de remodelar a base de dados. Além disso, a hierarquia de documentos é mapeada facilmente para hierarquias de objetos no código da aplicação, simplificando as operações CRUD.

As organizações podem usar a Mongo DB para suas consultas ad-hoc, indexação, balanceamento de carga, agregação, execução de JavaScript no lado do servidor e outros recursos.

### 4.3 Arquitetura da Aplicação

Tudo o que esteja relacionado com serviços de backend (por exemplo, operações CRUD na base de dados) vai estar interligado a uma API. Essa API vai estar interligada com a MongoDB, uma base de dados não relacional. A API vai funcionar como SaaS (Software as a Service) na cloud da Mongo. Por sua vez, existe uma conexão à API da Open Food Facts que é responsável por identificar o produto através do respetivo código de barras do mesmo. O frontend foi realizado em Ionic, na qual cada dispositivo tem associado um Local Storage. Por fim, a conexão entre a aplicação e a API NodeJs, é efetuada com o auxílio do JWT TOKEN, que é responsável por autenticar cada um dos utilizadores. Com esta arquitetura é ainda possível reutilizar a API para outros tipos de plataformas, como por exemplo uma WEB App.

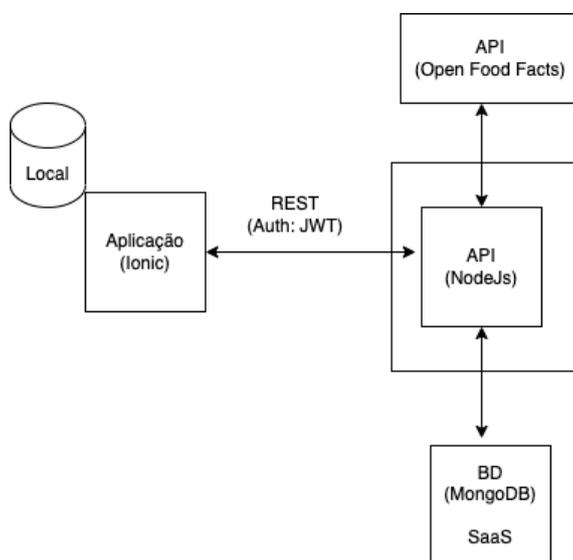


Figura 10 - Arquitetura Aplicacional

## 4.4 Modelo de Entidade Relação

No modelo de entidade relação podemos realçar que existem 4 entidade diferentes, uma delas é o utilizador que vai ter um email, uma password e uma ou várias listas. Essas listas podem ser lista de compras ou lista dos produtos que se encontram na despensa do utilizador.

Cada lista vai ter associada a si uma entidade chamada produto. Cada produto tem 8 atributos, que podem ser o seu nome, quantidade, imagem, um booleano que irá identificar se o produto já se encontra fora de validade, quantidades de dias que falta para o produto expirar, o dia em que o produto foi adicionado, o dia em que acaba a validade do produto, o respetivo código de barras e por fim uma enumeração de 0-10, que irá identificar qual a categoria do produto podendo ser bebida, mercearia, laticínios, talho, peixaria, fruta, legume, charcutaria, padaria, congelado ou outros.

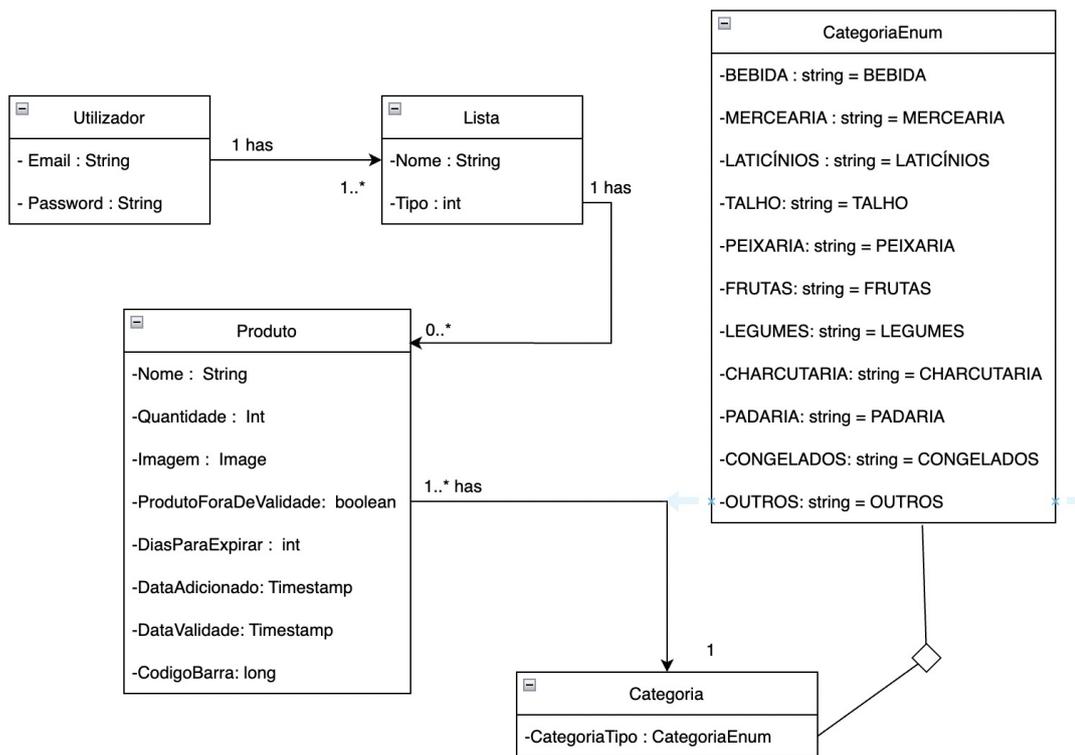


Figura 11 - Modelo de Entidade-Relação

## 5 Benchmarking

Este capítulo destina-se à apresentação da análise comparativa da solução proposta face a alternativas e potenciais concorrentes existentes no mercado.

Algumas das aplicações encontradas e analisadas foram as seguintes:

### 1) Datas de Validade

- Apenas disponível para Android
- Possível adicionar produtos utilizando a câmara, identificando os códigos de barras dos mesmos
- Não regista o nome do produto através do código de barras, terá de ser introduzido manualmente
- Só após o nome do produto ser inserido manualmente é que a app irá reconhecer futuramente o mesmo, pelo nome previamente inserido
- Após o artigo ser identificado, é possível atribuir uma categoria, uma data de compra, uma data de validade, uma data de abertura, um preço, um tipo de recipiente, unidades de quantidade e notas
- Oferece apenas possibilidade de exportar a lista de produtos para Dropbox
- Funciona sem ser preciso ligação à internet
- Notificações sobre produtos a expirar nem sempre funciona
- UI confusa e de difícil navegação entre os écrans disponibilizados
- Não é necessário login para utilizar todas a funcionalidades da aplicação
- Aplicação gratuita



Figura 12 - App 'Datas de Validade'

### 2) Food Check List

- Apenas disponível para Android
- Possível adicionar produtos utilizando a câmara, identificando os códigos de barras dos mesmos
- Não regista o nome do produto através do código de barras, terá de ser introduzido manualmente
- Só após o nome do produto ser inserido manualmente é que a app irá reconhecer futuramente o mesmo, pelo nome previamente inserido
- Após o artigo ser identificado é possível atribuir uma categoria, uma data de fabrico, uma data de validade, um armazém, um preço, unidades de quantidade



Figura 13 - App 'Food Check List'

e nome da loja onde comprou o produto. A opção de adicionar uma imagem é uma funcionalidade *Premium* apenas disponível por assinatura

- Notificações, sobre produtos individuais em categorias a expirar, é um recurso *Premium* apenas disponível por assinatura
- Oferece apenas possibilidade de exportar a lista de produtos para Excel
- Funciona sem ser preciso ligação à internet
- É possível fazer login com uma conta Google, mas adicionar outros utilizadores é uma funcionalidade *Premium*
- Aplicação gratuita, mas para usufruir de algumas funcionalidades é necessária uma assinatura

### 3) My Pantry

- Apenas disponível para Android
- Não é possível adicionar produtos utilizando a câmara. Assim, a funcionalidade de identificar os códigos de barras dos mesmos não está disponível.
- O nome do produto terá de ser introduzido manualmente ou por comando de voz
- Para aceder às definições da aplicação é preciso fazer login com o Google ou criar uma conta de utilizador
- Receber notificações é uma funcionalidade *Premium* apenas disponível por assinatura
- Não oferece a possibilidade de exportar a lista de produtos
- Publicidade intrusiva, aparecendo *pop-ups* que só desaparecem passados alguns segundos
- Funciona sem ser preciso ligação à internet
- UI confusa e de difícil navegação entre os écrans
- Aplicação gratuita, mas para usufruir de algumas funcionalidades é necessária uma assinatura



Figura 14 - App 'My Pantry'

#### 4) Fridge Buddy



Figura 15 - App 'Fridge Buddy'

- Apenas disponível para IOS
- Possível adicionar produtos utilizando a câmara, identificando os códigos de barras dos mesmos
- Após o artigo ser identificado, é possível atribuir uma categoria, uma data de validade e unidades de quantidade
- Receber notificações com restrições no limite de dias a escolher pelo utilizador, apenas com a funcionalidade *Premium*, disponível por assinatura, é que pode definir 4 ou mais dias para ser previamente notificado.
- Funciona sem ser preciso ligação à internet
- UI minimalista e de fácil utilização
- Para exportar a lista de produtos é preciso ter uma assinatura
- Aplicação gratuita, mas para usufruir de algumas funcionalidades é necessária uma assinatura

Após fazer um levantamento de prós e contras das aplicações já existentes dentro do segmento de mercado no qual nos queremos inserir, consideramos que a nossa aplicação será uma mais-valia para os potenciais utilizadores. Antes de mais, o facto de ser uma aplicação totalmente gratuita – e não só com algumas funcionalidades disponíveis nesta modalidade, como algumas das nossas eventuais concorrentes -, é uma vantagem para angariar mais facilmente e satisfazer com sucesso os utilizadores.

A isto, junta-se a simplicidade e acessibilidade facilitada a todas as opções que iremos disponibilizar. Bastará o código de barras para ter acesso ao 'nome' do produto, sem que este tenha de ser colocado manualmente, por exemplo. Após a passagem deste código, através do *smartphone*, será possível colocar o artigo na respetiva categoria, colocar data de validade, observações e respetiva quantidade.

A interface será, igualmente, atrativa, o que consideramos que não sucede com todas as aplicações do mesmo género atualmente presentes no mercado. Esta será desenvolvida para Android e IOS.

Com estas, a nosso ver, vantagens em relação às aplicações semelhantes – nenhuma se propõe a ter tantas opções e todas gratuitas como a nossa (e sem publicidade) – os utilizadores terão mais facilidade em combater (e, eventualmente, terminar) o desperdício alimentar.

## 6 Calendário

Com o propósito de que o plano de trabalhos proposto seja executado, dentro dos prazos estipulados, realizámos um cronograma para uma melhor orientação e gestão do projeto. Neste, além das datas, podemos também observar, em cada um dos *milestones* definidos, as atividades que nos propomos a desenvolver.

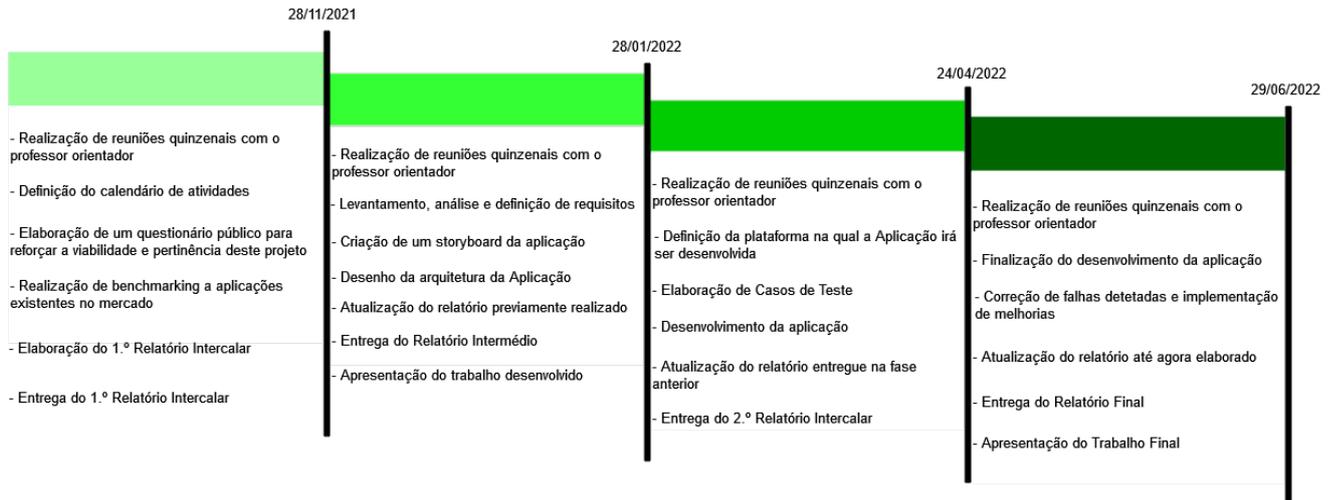


Figura 16 - Cronograma de TFC

## 7 Trabalhos Futuros

Sugerem-se como trabalhos futuros, após a angariação de alguns utilizadores, estudos e análises estatísticas de forma a criar um mecanismo que permita estabelecer padrões associados à validade média dos produtos alimentares após a inserção dos mesmos, bem como a possibilidade de oferecer soluções/conselhos acerca da alimentação do utilizador, com base nos produtos que se encontram na sua despensa.

Para que o uso da aplicação seja mais *user friendly*, sugere-se a possibilidade de adicionar produtos a partir da voz, assim como criar um sistema de notificações que permita informar o utilizador quando um produto se encontra quase fora de validade.

Por fim, a aplicação ficará à disposição de futuros alunos que se encontram a fazer o TFC na Universidade Lusófona de Humanidades e Tecnologias. A nossa tarefa, no futuro, passará por ser um apoio externo para eventuais dúvidas ou sugestões.

## 8 Conclusão

O principal objetivo desta aplicação é proporcionar aos seus utilizadores a disponibilidade de ter uma despensa virtual no seu telemóvel, onde é possível controlar de uma forma organizada os produtos que os mesmos têm em casa e a sua validade, evitando assim o desperdício.

No decorrer do processo de desenvolvimento do projeto existiram algumas limitações que dificultaram a sua realização. Em primeiro lugar, a compreensão da conexão dos utilizadores entre o Backend e Frontend. Para a sua resolução, e depois do auxílio do professor orientador, foi feita uma pesquisa intensa acerca do JWT Token, algo não lecionado durante a licenciatura. Em segundo lugar, foi a conexão de uma API (Open Food Facts) ao Backend, já que a aplicação utiliza a mesma para identificar o produto a partir do código de barras. Por último, outra limitação foi a instalação do Barcode Scanner na qual para a sua resolução foi realizada uma investigação para compreender como a desenvolver.

Com este trabalho foi possível colocar em prática diversas aprendizagens e conhecimentos técnicos adquiridos ao longo da licenciatura, como por exemplo o desenvolvimento de uma aplicação em angular como foi lecionado na unidade curricular de Sistemas Móveis Empresariais.

O desenvolvimento desta aplicação foi, sem dúvida, o projeto mais desafiante do percurso académico, já que para além das capacidades técnicas desenvolvidas, foram adquiridas bastantes *soft skills*, tais como: capacidade de adaptação a mudança, trabalho em equipa e investigação individual.

Por fim, após alguma reflexão relativamente ao processo de desenvolvimento do projeto, existiram algumas fases que se realizaria de uma forma diferente, tal como a forma como o projeto foi iniciado. Foi reconhecido que era de enorme importância ter começado por realizar um planeamento prévio da estrutura dos dados/entidades necessários para o desenvolvimento do projeto, ao invés de se ter começado pela realização do código referente ao frontend.

## Bibliografia

- [DEISI21] DEISI, Regulamento de Trabalho Final de Curso, Set. 2021.
- [ULHT21] Universidade Lusófona de Humanidades e Tecnologia, [www.ulusofona.pt](http://www.ulusofona.pt), acessido em Out. 2021.
- [ADO21] Adobe XD, [www.adobe.com](http://www.adobe.com), acessido em Nov. 2021.
- [AXU21] Axure RP, [www.axure.com](http://www.axure.com), acessido em Nov. 2021.
- [IONIC21] Ionic Framework, [www.ionicframework.com](http://www.ionicframework.com), acessido em Nov. 2021.
- [OUT21] Outsystems, [www.outsystems.com](http://www.outsystems.com), acessido em Nov. 2021.
- [UDD21] Udemy, [www.udemy.com](http://www.udemy.com), acessido em Nov. 2021.
- [Store21] PlayStore, [www.play.google.com/store](http://www.play.google.com/store), acessido em Nov. 2021.
- [IOS21] App Store, [www.apple.com](http://www.apple.com), acessido em Nov. 2021.
- [GOOGLE21] Google Forms, [www.google.com/forms](http://www.google.com/forms), acessido em Nov. 2021.
- [NodeJS22] NodeJS, [www.nodejs.dev](http://www.nodejs.dev), acessido em Fev. 2022.
- [Capacitor22] Capacitor, [www.capacitorjs.com](http://www.capacitorjs.com), acessido em Fev. 2022.
- [ExpressJS22] ExpressJS, [www.expressjs.com](http://www.expressjs.com), acessido em Fev. 2022.
- [MongoDB22] MongoDB, [www.mongodb.com](http://www.mongodb.com), acessido em Fev. 2022.

## **Anexo 1 – Questionário**

Link para o questionário anónimo: <https://forms.gle/XZbtqf76rK7g4ZAq9>

## **Glossário**

LIG	Licenciatura em Informática de Gestão
TFC	Trabalho Final de Curso
Android	Sistema operativo de base Unix/Linux com colaboração da Google
IOS	Sistema operativo da Apple
UI	User Interface
SDK	Software Development Kit
API	Application Programming Interface
CRUD	Create Read Update Delete
IONIC	Framework para desenvolvimento de Aplicações