

# Projeto de Informática

## *Relatório Técnico*

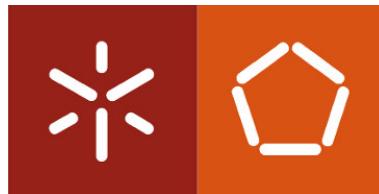
Aplicação de auto-promoção para guias turísticos - *GUIDEme*

28 de janeiro de 2022

### **Return to sleep**

---

|        |                  |
|--------|------------------|
| a83899 | André Moraes     |
| a85370 | Hugo Matias      |
| a84802 | João Nuno Abreu  |
| a84577 | José Pedro Silva |
| a85954 | Luís Ribeiro     |
| a85700 | Pedro Costa      |
| a84783 | Pedro Rodrigues  |
| a84485 | Tiago Magalhães  |



Mestrado Integrado em Engenharia Informática  
Universidade do Minho

# Conteúdo

|          |  |           |
|----------|--|-----------|
| <b>1</b> | <b>Introdução</b>                        | <b>2</b>  |
| <b>2</b> | <b>Contextualização</b>                  | <b>3</b>  |
| <b>3</b> | <b>Análise inicial</b>                   | <b>4</b>  |
| 3.1      | Requisitos . . . . .                     | 4         |
| 3.1.1    | Gestão de conta . . . . .                | 5         |
| 3.1.2    | Visitas/Experiências . . . . .           | 6         |
| 3.1.3    | Outros . . . . .                         | 7         |
| 3.2      | Concepção/Desenho da resolução . . . . . | 8         |
| 3.2.1    | Tecnologias a utilizar . . . . .         | 8         |
| 3.3      | Arquitetura . . . . .                    | 9         |
| 3.3.1    | Descrição da arquitetura . . . . .       | 10        |
| 3.4      | Modelo de dados . . . . .                | 10        |
| 3.5      | Mockups . . . . .                        | 12        |
| <b>4</b> | <b>Desenvolvimento</b>                   | <b>15</b> |
| 4.1      | Screenshots da aplicação . . . . .       | 17        |
| <b>5</b> | <b>Considerações futuras</b>             | <b>20</b> |
| <b>6</b> | <b>Conclusão</b>                         | <b>21</b> |

# 1 Introdução

O presente trabalho foi desenvolvido no âmbito da unidade curricular Projeto de Informática do 1º semestre, do 2º ano do Mestrado em Engenharia Informática da Universidade do Minho. Esta tem como objetivo a aprendizagem prática de conceitos relacionados com a complexidade de desenvolver um projeto *start-up*, tanto a nível técnico como a nível económico.

Este projeto consistiu na especificação de conhecimento de forma a conseguir construir uma aplicação informática que tire partido desse conhecimento, tendo sempre em consideração o estudo do mercado e a projeção económica da mesma. Para tal, foi necessário escolher um tema de estudo, para que a nossa aplicação tivesse uma base de informação bem suportada, de forma a promover algo contemporâneo, progressista e auto-sustentável.

Neste relatório irá ser detalhada a contextualização e trabalhos relacionados com o pretendido; qual a metodologia e tecnologias utilizadas; o processo de implementação da aplicação e decisões relativas ao desenvolvimento do mesmo. Para além disto, será feita também uma apreciação do sistema desenvolvido.

## 2 Contextualização

O interesse contínuo em conhecer e explorar novas áreas foi sempre considerado fundamental para a evolução do Homem. A valorização em aprimorar conhecimento sobre diferentes culturas confere ao ser humano esta necessidade de descobrir.

O Turismo enquadra-se neste domínio, no sentido em que, se caracteriza como sendo um fenómeno de aspecto social, cultural e económico diretamente relacionado com o deslocamento de pessoas para lugares fora do seu ambiente pessoal. O turismo força a uma interação vasta entre pessoas e exige uma variedade de serviços, infraestruturas e investimentos que permitam gerar e aproveitar oportunidades.

Dado a variedade de empregabilidade resultante da sua vasta aplicação em diferentes setores, e também ao interesse natural em viajar, o turismo apresenta-se como um setor económico de extrema importância. Compreendemos que o turismo, apesar da sua abrangência mundial como uma forte fonte de rendimento, é naturalmente influenciado pela alteração situacional.

Assim, a nossa equipa, os **Return to\_sleep**, escolheu o mundo do turismo como foco de estudo devido à sua forte influência na economia mundial e ao poder tecnológico que este permite explorar de forma variada e inovadora.

## 3 Análise inicial

### 3.1 Requisitos

Nesta secção será abordado o estudo e análise dos vários requisitos e serão apresentados os requisitos que foram levantados e que deverão ser suportados pelo sistema a desenvolver.

- **Utilizador comum**

- Criar conta e editar perfil
- Recuperar password
- Iniciar e terminar sessão
- Criar perfil de guia
- Listar visitas/experiências
- Listar guias
- Filtrar listagens
- Abrir em detalhe a visita/experiência
- Comprar bilhete para visita/experiência
- Visualizar perfil de guia
- Avaliar e comentar experiência e guia
- Visualizar bilhetes comprados

- **Guia**

- Criar e cancelar visita/experiência
- Subscrever um plano *premium*

- **Administrador (através do *backoffice*)**

- Personalizar os planos de subscrição *premium*
- Gestão de experiências
- Gestão de utilizadores
- Gestão de bilhetes

### 3.1.1 Gestão de conta

No que diz respeito à gestão de conta é esperado que um utilizador consiga criar e editar a mesma. Para o processo de criação de conta é necessário o utilizador indicar os seguintes campos:

- Primeiro e último nome
- País
- Distrito
- Descrição (biografia)
- Número de telemóvel
- E-mail
- Palavra-passe

No final do registo, é esperado que o utilizador receba um e-mail a confirmar o mesmo.

Quanto à edição de conta, espera-se que o utilizador consiga alterar as fotos de capa e de perfil e todos os campos mencionados acima, exceto o e-mail.

Os campos utilizados para o inicio de sessão são o e-mail e a password e esta deve permanecer ativa após o utilizador fechar a aplicação.

Adicionalmente, é esperado que exista um botão de reposição de password, onde o utilizador receberá um e-mail com uma password temporária que lhe permita iniciar sessão e definir a nova password.

Após a criação de conta e inicio de sessão, espera-se que exista um botão onde o utilizador se possa tornar guia, onde o único requisito será concordar com os termos e condições. No perfil de guia, deverá existir um botão que permita ao guia gerir as subscrições. No que toca às subscrições, o guia deverá conseguir observar e comprar os planos existentes na plataforma, no entanto, não pode ter mais do que um plano ativo em simultâneo.

Na página de perfil é expectável que o utilizador consiga observar as fotos de perfil e capa, o nome, classificação (caso seja guia), localização, número de telemóvel, próximas experiências (caso seja guia), histórico de experiências em que participou e comentários (caso seja guia).

### 3.1.2 Visitas/Experiências

Qualquer utilizador pode comprar bilhete para visitas/experiências, no entanto, apenas um guia as pode criar. Uma experiência pode ser considerada periódica ou pontual. As experiências periódicas serão repetidas durante a semana, ou seja, é definido um horário semanal e este horário é respeitado todas as semanas.

No processo de criação de uma experiência é necessário o utilizador indicar os seguintes campos:

- Tipo de experiência (periódica ou pontual)
- Título
- Localização
- Descrição
- Horário semanal e data de início (caso seja do tipo periódica)
- Data e hora (se for do tipo pontual)
- Duração
- Número limite de pessoas
- Preço
- *Keywords*
- Imagens

Após a criação da experiência, esta passará imediatamente a aparecer nas listagens. Um utilizador comum deverá conseguir observar as experiências ativas no sistema. Deverá ser possível fazer uma filtragem por título, *keywords*, localização, data, classificação e preço.

Uma experiência pode ser cancelada apenas e só por parte do guia e deverá ser feito até 3 dias antes da data da mesma. Após o cancelamento da experiência, esta é considerada inativa, deixa de aparecer nas listagens e deixa de ser possível comprar bilhetes para a mesma. Após o cancelamento de uma experiência, os utilizadores que tinham comprado bilhete para a mesma recebem um e-mail de aviso e devem ser reembolsados.

Qualquer utilizador pode comprar bilhete para uma experiência, exceto o guia que a criou. Para comprar um bilhete, o utilizador deve, no caso de se tratar de uma experiência periódica, indicar a data e a hora que pretende reservar. Após a confirmação da data e hora, o utilizador deverá ser redirecionado para uma página onde é indicada a referência, entidade e montante para realizar o pagamento por multibanco. Após o pagamento, o utilizador recebe um e-mail com o bilhete. Um utilizador não pode comprar mais do que um bilhete para cada experiência. Adicionalmente, o utilizador deverá ser capaz de visualizar os bilhetes comprados.

Após uma experiência, todos os utilizadores com bilhetes para a mesma, deverão ser capazes de comentar e avaliar tanto a experiência como o guia responsável pela mesma.

Na página de detalhe de uma experiência, é esperado que o utilizador consiga observar as imagens, o título, a informação do guia (imagem e nome), localização, classificação, limite de participantes, duração, data/horários, *keywords*, descrição, preço e comentário da mesma.

### **3.1.3 Outros**

#### **Requisitos Não Funcionais**

1. Aplicação de uso intuitivo;
2. Aplicação funcional durante todos os dias da semana;
3. Aplicação multi-plataforma (iOS, Android);

#### **Requisitos de Segurança**

1. Validação de *input*;
2. Mitigar vulnerabilidades de *Injection*;
3. Mitigar vulnerabilidades de quebra na autenticação;
4. Mitigar exposição de dados sensíveis;

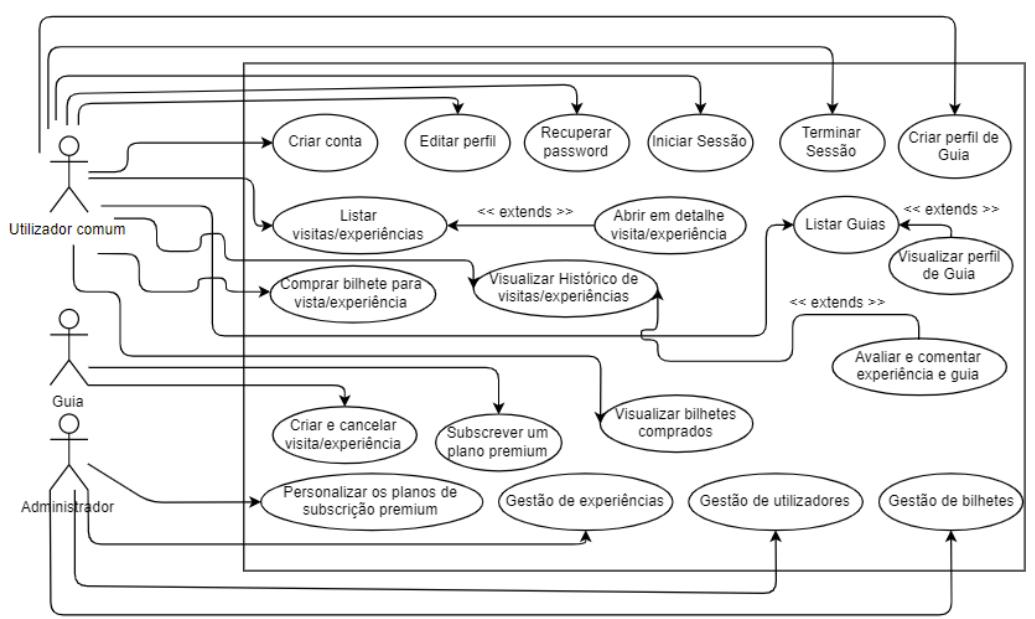


Figura 1: Diagrama de use cases

## 3.2 Concepção/Desenho da resolução

### 3.2.1 Tecnologias a utilizar

A nível do *Frontend*, inicialmente, foi realizada uma recolha das possíveis tecnologias a serem utilizadas, tendo sido *Flutter* e *React Native* as que mais se destacaram por serem *Cross-platform*, com comunidades bem estabelecidas, com boa documentação e bibliotecas relacionadas. Um dos motivos que teve um grande peso na decisão foi o facto de que alguns membros já estavam familiarizados com *frameworks* de *frontend* do ecossistema *JavaScript*. Desse modo, o uso de *Flutter* implicaria que houvesse perda de tempo para aprender uma nova linguagem, o que nos levou a escolher o *React Native*.

Por outro lado, ao nível do *Backend*, a solução mais evidente era o desenvolvimento de uma *REST API*, no entanto, era necessário decidir que tecnologias utilizar para o desenvolvimento da mesma, bem como o tipo de base de dados a utilizar.

Em relação à base de dados, a decisão estaria entre uma base de dados relacional ou não relacional. Considerando os requisitos da plataforma, a

equipa achou importante e necessário a exploração das relações presentes no sistemas (Guia - Experiência - Turista) e decidiu que a melhor solução seria optar por uma base de dados relacional. Posto isto, a equipa decidiu integrar *PostgreSQL*, uma vez que é uma das base de dados relacionais mais utilizadas atualmente.

No que diz respeito à tecnologia para o desenvolvimento da *API*, a equipa propôs duas soluções: *Node.js* e *Django*. Após várias discussões, a equipa decidiu optar por *Django*, uma vez que contém uma *framework* especializada em construção de *APIs* de forma simples (*Django REST framework*) e contém uma fácil configuração de um painel de administrador para gestão da plataforma.

Foi também necessário definir uma tecnologia para armazenamento das imagens da plataforma. Após uma análise preliminar da equipa, obtemos duas possíveis soluções gratuitas, *Firebase* e *Dropbox*. Visto que a integração com o *Firebase* era compatível com a biblioteca utilizada para manipulação de conteúdos multimédia utilizada no *frontend* e esta se encontrava bem documentada e de uso intuitivo, a equipa optou por utilizar esta tecnologia.

Para o envio de *emails*, a equipa conclui que era possível utilizar o *Gmail* como motor de envio de *emails*, visto que oferece um serviço gratuito, ao contrário de outras soluções encontradas.

### 3.3 Arquitetura

De modo a que a aplicação pudesse ser utilizada por qualquer utilizador, seria necessário que a *API* fosse alcançável através da Internet por qualquer dispositivo. Para isto, a equipa decidiu alojar a *API* e os restantes componentes do *backend* num serviço externo. Para a decisão deste serviço externo foram consideradas várias opções, sendo uma delas e a mais óbvia alojar a *API* e a base de dados num serviço da *AWS*. No entanto, estes serviços não são gratuitos, e por isso o grupo decidiu optar por uma solução gratuita, *Heroku*, uma plataforma que garante que a *API* e a base de dados estão disponíveis 24 horas por dia. Adicionalmente, um dos motivos para a escolha do *Heroku* foi que este contém um serviço *PostgreSQL* já configurado.

O desenho da arquitetura completa pode ser observado na Figura 2.

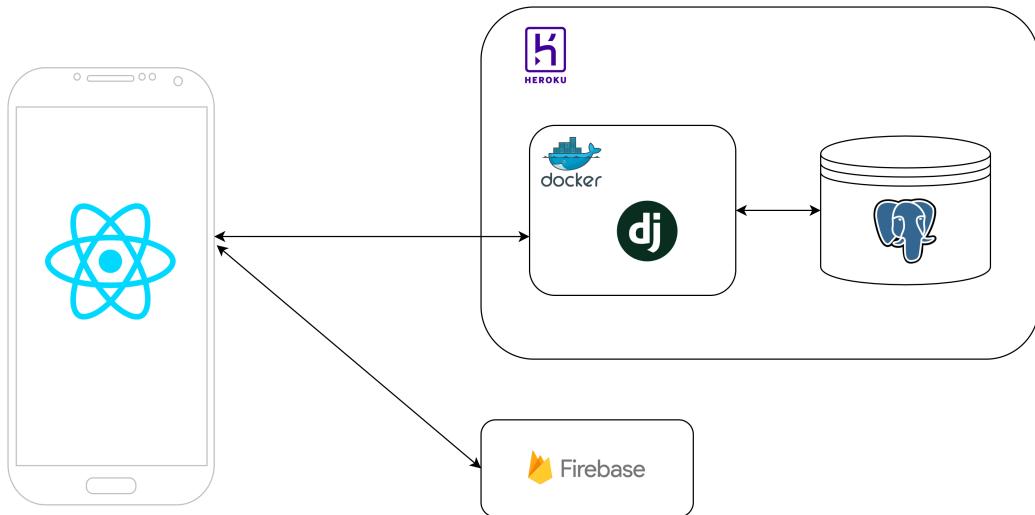


Figura 2: Arquitetura de alto nível do sistema

### 3.3.1 Descrição da arquitetura

Através do modelo do sistema podemos observar que existem 3 componentes principais sendo estas:

- **Frontend:** React Native, fornece interface *mobile* ao cliente.
- **Backend:** Usando Django - Django REST framework - API para tratar da lógica de negócio.
- **Firebase:** Utilizado para guardar as imagens relativas aos utilizadores e experiências.
- **Postgresql:** Base dados que guarda informações relativas aos utilizadores e experiências.

## 3.4 Modelo de dados

Com base na análise dos requisitos que foram apresentados anteriormente, construímos o seguinte modelo de dados:

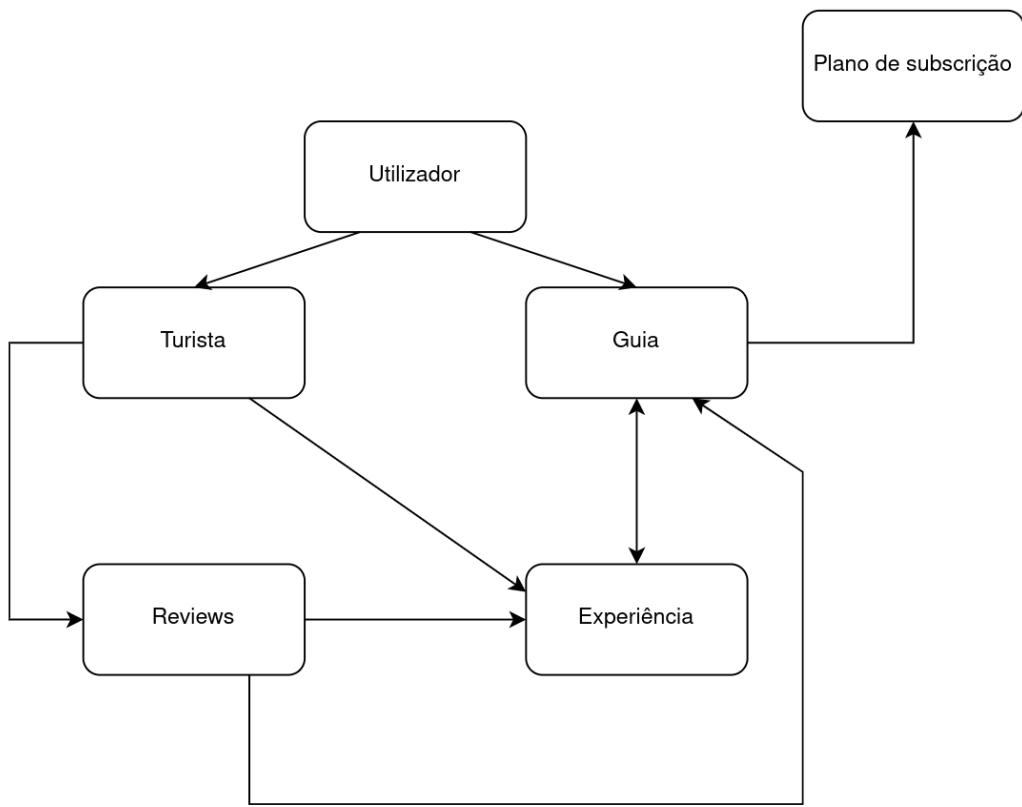


Figura 3: Modelo de dados

Tabela 1: Caracterização de entidades.

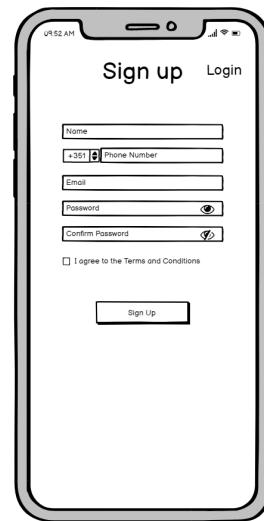
| Entidade         | Representa:   |
|------------------|---|
| Utilizador       | Todos os utilizadores registados na aplicação e são caracterizados por serem turistas ou guias  |
| Reviews          | Representado pelo um <i>rating</i> de 0 a 5 e um comentário associado, as <i>reviews</i> estão relacionadas com os turista e experiências |
| Experiência      | É representado pelo horário, imagens do percurso, <i>keywords</i> , reserva, bem como guias, <i>reviews</i> e turistas a este associados. |
| Plano subscrição | Os guias associados a este planos têm prioridades a nível de apresentação.  |

### 3.5 Mockups

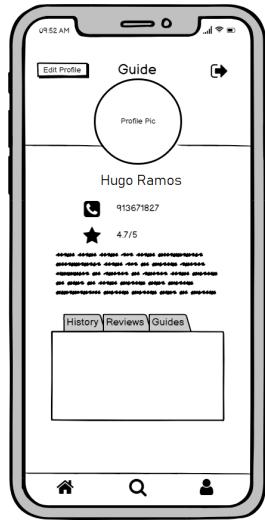
Para estruturar o protótipo da interface utilizamos a ferramenta *Balsamiq*. Estes protótipos apresentam uma baixa fidelidade, isto é -dão apenas uma ideia/análise aproximada, o que abre espaço para que na implementação existam alterações.



(a) Página de Log In



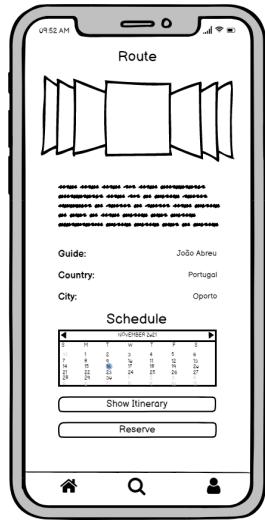
(b) Página de Criar Conta



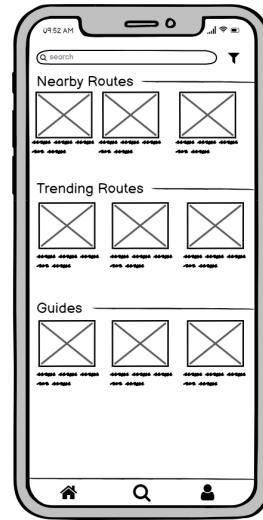
(a) Página de Perfil de um Guia



(b) Página de Perfil de um Turista



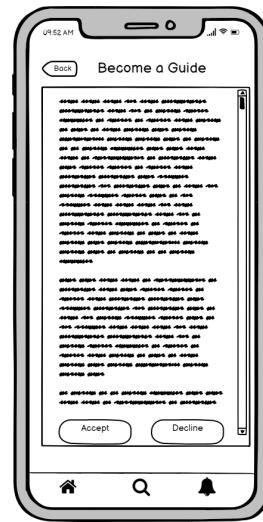
(a) Página de uma Experiência



(b) Página Home



(a) Página de Editar



(b) Página de Termos

## 4 Desenvolvimento

Inicialmente houve uma preocupação da equipa em organizar o projeto e otimizar ao máximo o processo de desenvolvimento. Posto isto, a equipa decidiu alojar o projeto no *Github* em dois repositórios distintos, *frontend* e *backend*. Desde início foram definidas regras específicas para cada repositório, impedindo entradas diretas de código na *branch master*, sendo, por isso, necessário abrir um *pull request* a partir de uma *branch* diferente para nova entrada de código, o que otimizou bastante a manutenção e resolução de conflitos, uma vez que estavam vários elementos a trabalhar ao mesmo tempo no projeto.

Adicionalmente foram adicionadas *pipelines* de testes com recurso ao *Github Actions* onde, para cada *pull request*, é corrida uma pipeline de testes unitários e de *linting* por forma a medir a qualidade do código.

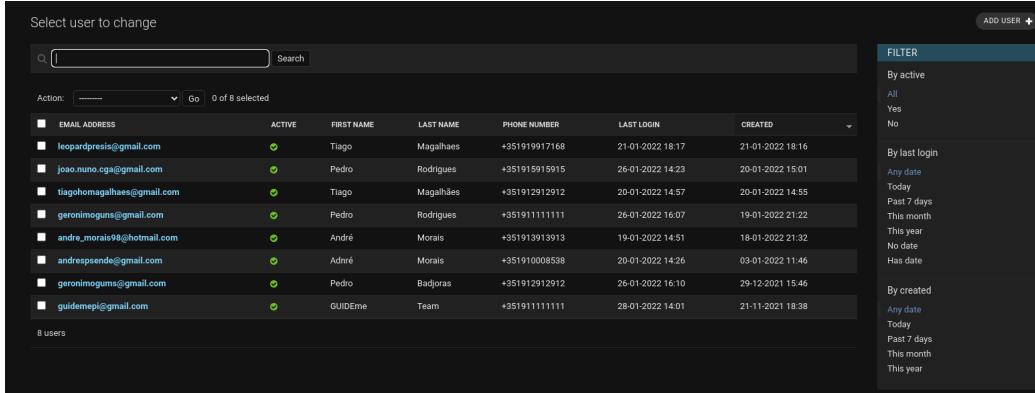
A infraestrutura do *backend* foi montada desde o início para que fosse possível incorporar imediatamente com o *frontend*. Por forma a manter a API em conformidade com os novos desenvolvimentos inseridos no repositório, foi criado um método de *deploy* automático, onde, para cada entrada de código em *master* é feito *deploy* do projeto automaticamente para o *Heroku*, caso o *pull request* passe com sucesso nos testes.

Adicionalmente, para facilitar o processo de desenvolvimento, foi também criado um ambiente local, isto é, foi criado um *docker-compose* que lança 3 *containers* essenciais para a API, nomeadamente, a API em si, a base de dados em *PostgreSQL* e o serviço de e-mails *MailHog*).

Posteriormente, o grupo iniciou o processo de desenvolvimento da aplicação definindo os modelos em *Django* assim como o tema e componentes gerais em *React*.

A API foi desenvolvida com recurso ao *Django REST framework*. Esta *framework* é bastante completa e facilita o desenvolvimento de pontos importantes de uma REST API como a autenticação. O método de autenticação escolhido foi via *token*. Quando um utilizador faz o *login* com as credenciais, é gerado um *token* para aquela conta que só é eliminado com este faz *logout*. Esta estratégia permite que, no *frontend*, o *token* seja guardado e possa ser usado para manter a sessão aberta mesmo quando o utilizador fecha a aplicação. Os *endpoints* da API estão documentados através da ferramenta *swagger* que disponibiliza uma página com os mesmos ([GUIDE](#)*me API schema*).

O painel de admin foi desenvolvido utilizando o módulo de *admin* do Django, que permite criar uma plataforma de gestão de dados, onde é possível criar, ver, editar ou eliminar conteúdo do sistema.



| EMAIL ADDRESS              | ACTIVE | FIRST NAME | LAST NAME | PHONE NUMBER   | LAST LOGIN       | CREATED          |
|----------------------------|--------|------------|-----------|----------------|------------------|------------------|
| leopardpresis@gmail.com    | ✓      | Tiago      | Magalhães | +351919917168  | 21-01-2022 18:17 | 21-01-2022 18:16 |
| joo.nuno.cga@gmail.com     | ✓      | Pedro      | Rodrigues | +351915915915  | 26-01-2022 14:23 | 20-01-2022 15:01 |
| tiagomagalhaes@gmail.com   | ✓      | Tiago      | Magalhães | +351912912912  | 29-01-2022 14:57 | 20-01-2022 14:55 |
| geronimoguns@gmail.com     | ✓      | Pedro      | Rodrigues | +351911111111  | 26-01-2022 16:07 | 19-01-2022 21:22 |
| andre_morais98@hotmail.com | ✓      | André      | Morais    | +351913913913  | 19-01-2022 14:51 | 18-01-2022 21:32 |
| andrespende@gmail.com      | ✓      | Adhré      | Morais    | +3519100008538 | 20-01-2022 14:26 | 03-01-2022 11:46 |
| geronimoguns@gmail.com     | ✓      | Pedro      | Badjoras  | +351912912912  | 26-01-2022 16:10 | 29-12-2021 15:46 |
| guidemepi@gmail.com        | ✓      | GUIDEme    | Team      | +351911111111  | 28-01-2022 14:01 | 21-11-2021 18:38 |

Figura 8: Gestão de Utilizadores - Painel de Admin

De modo a obter um desenvolvimento mais rápido e uma padronização foi utilizado no *frontend* a biblioteca NativeBase que é construída em cima do React Native, que fornece já componentes criados e padronizados tornando o *design* do sistema universal e consistente. Em complemento foi utilizado a *framework* Expo que oferece um conjunto de ferramentas que facilitam o desenvolvimento e o teste de aplicações, fornecendo serviços que normalmente estão disponíveis como *third-party*, dentro destes serviços destacamos o Expo-images que permitiu a manipulação multimédia em junção com o SDK do Firebase, bem como o uso de Expo-secure-store que possibilita guardar os *tokens* localmente cifrados e isolados de outras aplicações. A nível de implementação foram utilizadas ferramentas de análise estática de código como ESLint.

## 4.1 Screenshots da aplicação

17:53

Create Experience

Name

Experience Name

Location

Search

Description

Experience Description

Schedule

Periodic  One day

Select Start Date

Mon Tue Wed Thu Fri Sat Sun

Number of people

0 ↑ ↓

Price (€)

0 ↑ ↓

Duration (hours)

0 ↑ ↓

Upload Images

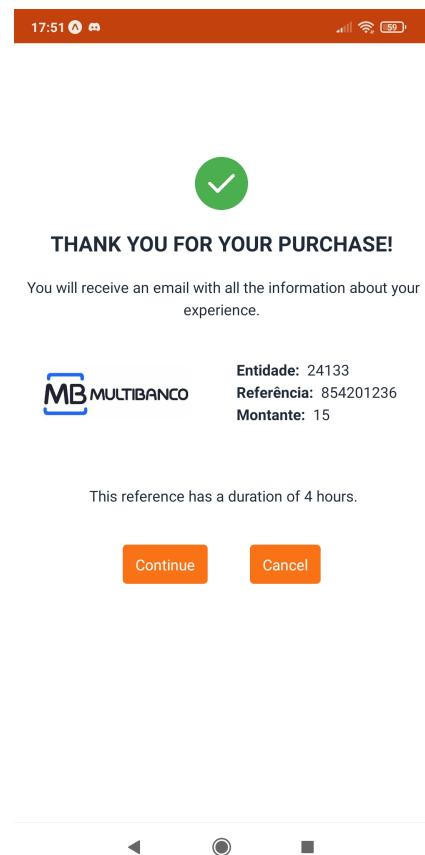
Keywords:

Add some tags...

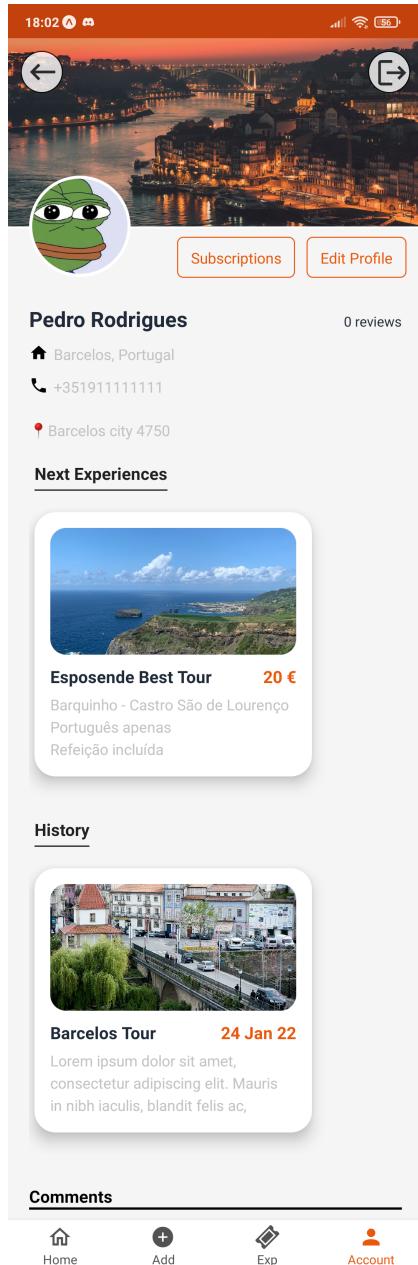
Save

Home Add Exp Account

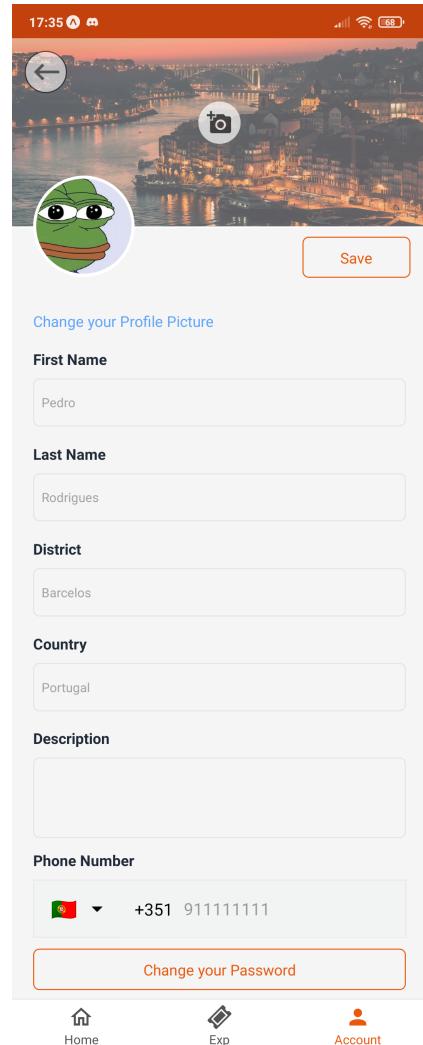
(a) Página de criar experiência



(b) Página da referência gerada



(a) Página de Perfil



(b) Página de Editar

**(a) Página Inicial**

**Nearby Routes**

- Barcelos Tour** **10 €**  
Lorem ipsum dolor sit amet, consectetur adipiscing elit. Mauris in nibh iaculis, blandit felis ac,
- Taipas Tou**  
asdsad asds  
asdsad asds  
asdsad asds

**Premium Routes**

- Barcelos Tour** **10 €**  
Lorem ipsum dolor sit amet, consectetur adipiscing elit. Mauris in nibh iaculis, blandit felis ac,
- Taipas Tou**  
asdsad asds  
asdsad asds  
asdsad asds

**Guides**

- Pedro Rodrigues** **? (0)**  
Braga, Portugal  
Um gajo fixe :) niggers
- Pedro Badjoras**  
Braga, Portugal  
Integer sit amet sagi sapien. Ut ac dapibus

**17:48** **17:50**

**19 Feb 22 12:00**

**Taipas Tou**

**Pedro Badjoras**

**Taipas, Guimarães, Portugal**

**4** **5 hours** **14 Feb 22 12:00**

**Keywords**

- Taipas Tapado Guimarães Praia seca
- Casa do Luís Casa da Linda Outlaw castle

**15 €** **Total Price** **Book Now**

**Comments**

**Home** **Exp** **Account**

**(b) Página de uma experiência**

## 5 Considerações futuras

Esta secção visa em apresentar algumas considerações discutidas em equipa, no desenvolver da aplicação, onde apresentemos possíveis melhorias/sugestões para o futuro da mesma. Achamos relevante dividir estas considerações em melhorias técnicas e funcionalidades adicionais.

A nível de melhoria técnica, a nossa abordagem passa por aprimorar alguns dos serviços implementados de forma a apresentar uma aplicação com menos falhas possíveis.

Assim, considera-se que seria interessante implementar diferentes técnicas que promovam o aumento da segurança e credibilidade da nossa aplicação, nomeadamente na gestão dos bilhetes e das informações pessoais. Aqui, melhorar o sistema de atribuição de bilhetes seria algo a considerar num futuro próximo, juntamente com a implementação da API do multibanco para a venda dos mesmos.

A mudança do serviço externo para alojar a API deve ser adotada, porque apesar do *Heroku* fornecer um plataforma *cloud* grátis, não é tão extensa e credível como os serviços apresentados pela AWS. Seria também interessante mudar a abordagem no tratamento de *emails*, onde a aplicação de um serviço mais robusto seria o ideal.

Adicionalmente, a presente equipa considera que a aplicação poderia ser estendida a nível de funcionalidades. Estas considerações foram discutidas no desenvolver da aplicação, à medida que a equipa foi encontrando limitações e que deviam ser tratadas apenas como futuras extensões ao produto atual.

Por exemplo, poderíamos considerar a implementação de um *chat* integrado, de forma a facilitar a comunicação entre guias e turistas. Inicialmente, achamos que esta implementação não seria necessário de imediato, e as opções encontradas com maior facilidade de implementação não eram de acesso grátis. Poderíamos implementar diferentes temas estéticos na aplicação, como por exemplo, o *dark theme*, muitas vezes pretendido por vários utilizadores. A implementação de vários idiomas, através de implementação de APIs externas seria um aspeto a considerar, dado a projeção aplicacional que queremos atingir com o nosso produto.

Um aspeto que também foi considerado no *Business Case*, como uma estratégia de negócio, seria apostar em criar cursos intensivos de formação, fornecidos por especialistas na área, estendendo os planos de subscrição a nível funcional e económico.

## 6 Conclusão

O presente relatório descreveu, de forma sucinta, o trabalho de especificação de conhecimento de forma a conseguir construir uma aplicação informática que tire partido desse conhecimento.

Durante a elaboração deste trabalho surgiram algumas dificuldades que, com o esforço e entusiasmo da equipa pelo produto final, acabaram por ser ultrapassadas. Entre as quais destaca-se o desafio de lidar com a gestão das experiências, desde a gestão da sua periodicidade até à atribuição de bilhetes, sujeitos a limites de data e ocupação. Outra dificuldade apresentada surge muito da nossa opção a nível aplicacional. O facto de termos optado por desenvolver uma aplicação *mobile* trouxe complexidade adicional em comparação ao desenvolvimento de uma aplicação *web*, onde, por exemplo, dar *debug* aplicacional é extremamente mais difícil.

Após a realização deste trabalho, a equipa **Return to\_sleep** ficou consciente da dificuldade em apresentar um produto aplicacional forte tanto a nível tecnológico com a nível económico. Consideramos que, o facto de termos desenvolvido um *Business Case*, fez com que o desenvolvimento tecnológico fosse um processo muito mais organizado, suportados por uma análise construtiva do mercado.

Consideramos que os objetivos propostos com a realização deste trabalho foram cumpridos, bem como, a consolidação dos conhecimentos em processamento e representação de conhecimento. Por fim, a equipa espera que os conhecimentos obtidos e consolidados sejam de enorme utilidade tendo em conta uma perspetiva futura.