

## Projeto em Sistema de Informação



### ***AeroControl***

---

**Grupo:** PL-TV-B

**Data de entrega:** 2023-02-13

---

Nº 2211893      Pedro Henriques

Nº 2211904      Pedro Norberto

Nº 2211900      Rafael Bento

---

# ÍNDICE

<b>ÍNDICE DE FIGURAS .....</b>	<b>4</b>
<b>ÍNDICE DE TABELAS .....</b>	<b>5</b>
<b>1 INTRODUÇÃO .....</b>	<b>7</b>
1.1 SUMÁRIO EXECUTIVO .....	8
<b>2 DEFINIÇÃO DA LÓGICA DE NEGÓCIO .....</b>	<b>10</b>
<b>3 ANÁLISE DE IMPACTO .....</b>	<b>11</b>
<b>4 ANÁLISE CONCORRENCIAL .....</b>	<b>12</b>
4.1.1 FLYTAP .....	12
4.1.2 LISBOA AEROPORTO .....	14
4.1.3 ANA AEROPORTOS DE PORTUGAL .....	16
4.1.4 COMPARAÇÃO DOS SISTEMAS .....	18
4.1.5 ENQUADRAMENTO DA ANÁLISE CONCORRENCIAL NO SI .....	18
<b>5 TECNOLOGIAS UTILIZADAS .....</b>	<b>19</b>
5.1 WEBSITE/API .....	19
5.2 APLICAÇÃO MÓVEL .....	19
<b>6 ARQUITETURA DO SISTEMA .....</b>	<b>20</b>
<b>7 METODOLOGIAS E DIVISÃO DE TAREFAS .....</b>	<b>21</b>
<b>8 ROLES E REQUISITOS DO SI .....</b>	<b>22</b>
8.1 REQUISITOS FUNCIONAIS .....	23
8.2 REQUISITOS NÃO FUNCIONAIS DE USABILIDADE (USABILITY) .....	32
8.3 REQUISITOS NÃO FUNCIONAIS DE FIABILIDADE (RELIABILITY) .....	34
8.4 REQUISITOS NÃO FUNCIONAIS DE SEGURANÇA (SECURITY) .....	35
8.5 REQUISITOS NÃO FUNCIONAIS DE EFICIÊNCIA (PERFORMANCE) .....	37
8.6 REQUISITOS NÃO FUNCIONAIS DE DISPONIBILIDADE (AVAILABILITY) .....	38
8.7 REQUISITOS NÃO FUNCIONAIS DE AMBIENTE (PORTABILITY) .....	39
8.8 REQUISITOS NÃO FUNCIONAIS DE DESENVOLVIMENTO (IMPLEMENTATION) .....	40
<b>9 USER STORIES .....</b>	<b>43</b>

Cofinanciado por:



<b><u>10</u></b>	<b><u>DIAGRAMA DE ENTIDADE-RELACIONAMENTO.....</u></b>	<b><u>53</u></b>
<b><u>11</u></b>	<b><u>DIAGRAMA DE CLASSES .....</u></b>	<b><u>55</u></b>
<b><u>12</u></b>	<b><u>WIREFRAMES/MOCKUPS .....</u></b>	<b><u>56</u></b>
<b><u>13</u></b>	<b><u>IMPLEMENTAÇÃO.....</u></b>	<b><u>84</u></b>
<b><u>14</u></b>	<b><u>FASE DE TESTES .....</u></b>	<b><u>85</u></b>
<b>14.1</b>	<b>TESTES UNITÁRIOS .....</b>	<b>86</b>
<b>14.2</b>	<b>TESTES FUNCIONAIS.....</b>	<b>89</b>
<b>14.3</b>	<b>TESTES DE ACEITAÇÃO.....</b>	<b>92</b>
<b><u>15</u></b>	<b><u>PROBLEMAS/DIFICULDADES E SOLUÇÕES.....</u></b>	<b><u>93</u></b>
<b>15.1</b>	<b>APLICAÇÃO MÓVEL .....</b>	<b>94</b>
<b>15.2</b>	<b>API .....</b>	<b>95</b>
<b>15.3</b>	<b>WEBSITE.....</b>	<b>96</b>
<b><u>16</u></b>	<b><u>CONCLUSÃO .....</u></b>	<b><u>98</u></b>

## ÍNDICE DE FIGURAS

Figura 1 – Diagrama arquitetural do sistema .....	20
Figura 2 - Diagrama de Entidade-Relacionamento (Relações) .....	53
Figura 3 – Diagrama de Entidade-Relacionamento (Atributos) .....	54
Figura 4 - Diagrama de Classes .....	55
Figura 5 - Framework SDLC (Software Development Life Cycle) .....	84
Figura 6 - Teste unitário - Criar companhia .....	86
Figura 7 - Teste unitário - Ler companhia da base de dados.....	86
Figura 8 - Teste unitário - atualização de uma companhia .....	86
Figura 9 - Teste unitário - Apagar registo do item dos perdidos e achados associado ao ticket de suporte..	87
Figura 10 - Teste unitário - Sem campos preenchidos .....	87
Figura 11 - Teste unitário - Autenticação com a password errada .....	88
Figura 12 - Teste unitário - Autenticação com dados corretos .....	88
Figura 13 - Teste funcional - função “before” .....	89
Figura 14 - Teste funcional - Submissão do formulário sem dados preenchidos.....	89
Figura 15 - Teste funcional - Submissão do formulário com formato de email incorreto .....	90
Figura 16 – Submissão do formulário com todos os dados devidamente preenchidos e corretos .....	90
Figura 17 - Teste funcional - Autenticação de um administrador .....	90
Figura 18 - Teste funcional - Autenticação de um funcionário .....	91

## ÍNDICE DE TABELAS

Tabela 1 – Descrição da FlyTap.....	12
Tabela 2 – Descrição do Lisboa Aeroporto.....	14
Tabela 3 – Descrição do ANA Aeroportos de Portugal.....	16
Tabela 4 – Resumo das características/funcionalidades dos Sistemas concorrenciais.....	18
Tabela 5 – Roles dos diferentes end-users do SI.....	22
Tabela 6 – Requisitos funcionais do front-office.....	23
Tabela 7 – Requisitos funcionais do back-office.....	24
Tabela 8 – Requisitos funcionais da aplicação android.....	25
Tabela 9 – Cruzamento dos requisitos funcionais do front-office e respetivos roles.....	26
Tabela 10 - Cruzamento dos requisitos funcionais do back-office e respetivos roles.....	28
Tabela 11 - Cruzamento dos requisitos funcionais da aplicação android e respetivos roles.....	30
Tabela 12 – Requisitos Não funcionais de Usabilidade do website.....	32
Tabela 13 - Requisitos Não Funcionais de Usabilidade da aplicação android.....	33
Tabela 14 - Requisitos Não Funcionais de Fiabilidade do website.....	34
Tabela 15 - Requisitos Não Funcionais de Fiabilidade da aplicação android.....	34
Tabela 16 - Requisitos Não Funcionais de Segurança do website.....	35
Tabela 17 - Requisitos Não Funcionais de Segurança da aplicação android.....	35
Tabela 18 - Requisitos Não Funcionais de Segurança da API.....	36
Tabela 19 - Requisitos Não Funcionais de Eficiência do website.....	37
Tabela 20 - Requisitos Não Funcionais de Eficiência da aplicação android.....	37
Tabela 21 - Requisitos Não Funcionais de Eficiência da API.....	37
Tabela 22 - Requisitos Não Funcionais de Disponibilidade do website.....	38
Tabela 23 - Requisitos Não Funcionais de Disponibilidade da aplicação android.....	38
Tabela 24 - Requisitos Não Funcionais de Ambiente website.....	39
Tabela 25 - Requisitos Não Funcionais de Ambiente da aplicação android.....	39
Tabela 26 - Requisitos Não Funcionais de Desenvolvimento do website.....	40
Tabela 27 - Requisitos Não Funcionais de Desenvolvimento da aplicação android.....	41
Tabela 28 - Requisitos Não Funcionais de Desenvolvimento da API.....	42
Tabela 29 - User Stories - Front-office.....	43
Tabela 30 - User Stories - Back-office.....	46
Tabela 31 - User Stories - Aplicação android.....	50
Tabela 32 - Wireframe/Mockup do ecrã principal - Website.....	57
Tabela 33 - Wireframes/Mockups do ecrã de abertura (Splash Screen) – Aplicação Móvel.....	58
Tabela 34 - Wireframe/Mockup do ecrã de registo - Website.....	59
Tabela 35 - Wireframe/Mockup do ecrã de registo - Aplicação Móvel.....	60
Tabela 36 - Wireframe/Mockup do ecrã da página de erro - Website.....	61
Tabela 37 – Wireframe/Mockup do ecrã de Login - Website.....	62
Tabela 38 - Wireframe/Mockup do ecrã de Login - Aplicação Móvel.....	63
Tabela 39 - Mockup do ecrã da conta autenticado/não autenticado - Aplicação Móvel.....	63
Tabela 40 - Wireframe/Mockup do ecrã de repor a palavra-passe - Website.....	64
Tabela 41 - Wireframe/Mockup do ecrã de repor a palavra-passe - Aplicação Móvel.....	65
Tabela 42 - Wireframe/Mockup do ecrã da pesquisa de voos - Website.....	66
Tabela 43 - Wireframe/Mockups dos ecrãs da pesquisa de voos - Aplicação Móvel.....	67
Tabela 44 - Wireframe/mockup do ecrã da compra de um bilhete de voo - Website.....	68
Tabela 45 - Wireframe/Mockups dos ecrãs da compra de um bilhete de voo - Aplicação Móvel.....	69

Cofinanciado por:



Tabela 46 - Wireframe/Mockup do ecrã de editar os dados da conta - Website .....	70
Tabela 47 - Wireframe/Mockups dos ecrãs de atualizar os dados da conta - Aplicação Móvel.....	71
Tabela 48 - Wireframe/Mockup do ecrã da lista de bilhetes - Website .....	72
Tabela 49 - Wireframes/Mockup do ecrã da lista de bilhetes - Aplicação Móvel.....	73
Tabela 50 - Wireframe/Mockup da informação do bilhete comprado - Aplicação Móvel .....	73
Tabela 51 - Wireframe/Mockup do ecrã da lista de tickets de suporte - Website .....	74
Tabela 52 - Wireframe/Mockup do ecrã da lista de tickets de suporte - Aplicação Móvel .....	75
Tabela 53 - Wireframe/Mockup do ecrã do ticket de suporte - Website .....	76
Tabela 54 - Wireframe/Mockup do ecrã do ticket de suporte - Aplicação Móvel.....	77
Tabela 55 - Wireframe/Mockup do ecrã de suporte através de email - Website.....	78
Tabela 56 - Wireframe/Mockup do ecrã de suporte através de email - Aplicação Móvel .....	79
Tabela 57 - Wireframe/Mockup do ecrã da lista e informação dos restaurantes - Website.....	80
Tabela 58 - Wireframes/Mockups do ecrã da lista e da Informação dos restaurantes - Aplicação Móvel.....	81
Tabela 59 - Wireframe/Mockup do ecrã da lista e informação das lojas - Website .....	82
Tabela 60 - Wireframes/Mockups dos ecrãs da lista e das informações das lojas – Aplicação Móvel.....	83

# 1 INTRODUÇÃO

Este projeto consiste em desenvolver:

- Uma **aplicação web** front-office, back-office, RBAC e testes de software para a UC de *PSI* (Plataformas de Sistemas de Informação);
- Uma **aplicação móvel Android** com acesso a *webservice* REST/RESTful para a UC de *AMSI* (Acesso Móvel a Sistemas de Informação);
- Uma **API de serviços** para a UC de *SIS* (Serviços e Interoperabilidade de Sistemas);
- Uma **BD Relacional ou NoSql** de suporte aos sistemas anteriores.

Estes sistemas irão se interligar da seguinte maneira:

- A **aplicação Android** comunica com a **API** para o cumprimento dos respetivos requisitos funcionais e não funcionais;
- A **aplicação Web** comunica com a **API** de forma interna ou via HTTP, para o cumprimento dos respetivos requisitos funcionais e não funcionais;
- A **BD** de suporte ao sistema resultante é acedida unicamente através da componente **API**.

Cofinanciado por:



UNIÃO EUROPEIA  
Fundo Social Europeu

## 1.1 Sumário executivo

### Secção 1

- Introdução ao projeto, o que é, para que serve e qual é o objetivo?

### Secção 2

- Descrição do sistema.

### Secção 3

- Impactos positivos e negativos do sistema desenvolvido para o utilizador.

### Secção 4

- Comparação e análise das funcionalidades de softwares semelhantes.

### Secção 5

- Descrição das tecnologias utilizadas no projeto.

### Secção 6

- Descrição das componentes desenvolvidas e da forma que comunicam entre si.

### Secção 7

- Descrição das metodologias utilizadas no projeto.

### Secção 8

- Descrição das roles existentes no sistema;
- Requisitos funcionais e não funcionais do sistema desenvolvido.

### Secção 9

- Descrição detalhada das user storys.

### Secção 10

- Diagrama de Entidade-Relacionamento (DER) do sistema desenvolvido.

### Secção 11

- Diagrama de classes.

### Secção 12

Cofinanciado por:



UNIÃO EUROPEIA  
Fundo Social Europeu



- Diversos Wireframes e Mockups dos ecrãs do website e da aplicação móvel.

### **Secção 13**

- Implementação, descrição do processo de desenvolvimento do projeto.

### **Secção 14**

- Descrição da fase de testes do projeto e descrição dos testes realizados:
  - Unitários;
  - Funcionais;
  - Aceitação;

### **Secção 15**

- Descrição dos problemas obtidos em cada componente ao longo do desenvolvimento do projeto.

### **Secção 16**

- Conclusão do projeto e do relatório.

## 2 DEFINIÇÃO DA LÓGICA DE NEGÓCIO

Este projeto tem vários objetivos, mas o principal é fornecer ao cliente um software para que o mesmo possa gerir um aeroporto.

O projeto irá conter um website com uma componente back-office na qual irá ser feita uma melhor gestão por parte do administrador e trabalhador. Na back-office será possível gerir todos os dados relativos à base de dados, como por exemplo, gerir os trabalhadores do aeroporto para que os mesmo possam ajudar a gerir o aeroporto, gerir os gerentes dos restaurantes para que possam gerir os menus dos seus restaurantes, gerir os objetos dos perdidos e achados, gerir o sistema de pontos/promoções do seu sistema, etc. A back-office irá ter também uma log para que o administrador possa ver todas as atividades efetuadas pelos trabalhadores.

O website terá também uma parte front-office que será utilizada pelos clientes do aeroporto. Nesta componente do website será possível visualizar uma lista com os voos do aeroporto, comprar bilhetes para o voo pretendido e alterar os dados do bilhete se necessário, contactar o suporte do aeroporto, consultar os restaurantes do aeroporto bem como os seus menus.

Este projeto terá ainda uma app mobile na qual terá todas as funcionalidades da front-office do website e irá ainda incluir algumas funcionalidades diferentes, sendo estas, a possibilidade de conversão do bilhete para QR Code e permitir ao cliente contactar diretamente os funcionários através de um chat (não em tempo real) para que o mesmo possa informar sobre os itens dos perdidos e achados.

Como já foi mencionado o sistema terá quatro roles. Sendo estas administrador, trabalhador, cliente, gerente.

- O administrador que será o cargo mais alto e será responsável pela gestão de toda a informação da back-office;
- O trabalhador do aeroporto que terá acesso a uma gestão parcial na back-office de forma a não sobrecarregar o administrador com toda a gestão;
- O cliente, ou seja, o utilizador habitual da front-office e da app mobile, o utilizador que vai comprar os bilhetes;
- O gerente, o cargo fornecido a cada gerente(s) de cada restaurante(s) para que o mesmo possa gerir o menu do seu próprio restaurante.

### 3 ANÁLISE DE IMPACTO

Com a criação deste sistema de gestão de um aeroporto, podemos contar com impactos positivos e negativos.

Começando pelos impactos positivos, a nossa aplicação visa ser direta e segura permitindo ao utilizador poupar tempo e a ser mais autónomo, oferece uma grande quantidade de informação sobre o aeroporto permitindo consultar tudo que seja necessário e tem a possibilidade de reservar voos sem a necessidade de se deslocar ao local.

Com a criação de mais um software de gestão do aeroporto, iremos estar a criar mais concorrência dando aos utilizadores um maior leque de escolha.

Nos aspetos negativos, podemos contar com a alta concorrência e poderá não ter todas as funcionalidades que os outros concorrentes têm.

#### **Impactos positivos:**

- Fácil e simples de usar;
- Oferece uma grande quantidade de informação sobre o aeroporto;
- Possibilidade de reservar voos online sem a necessidade de deslocamento.

#### **Impactos negativos:**

- Pode não ter alguma funcionalidade que os concorrentes têm, como por exemplo, a possibilidade de mudar de idioma.

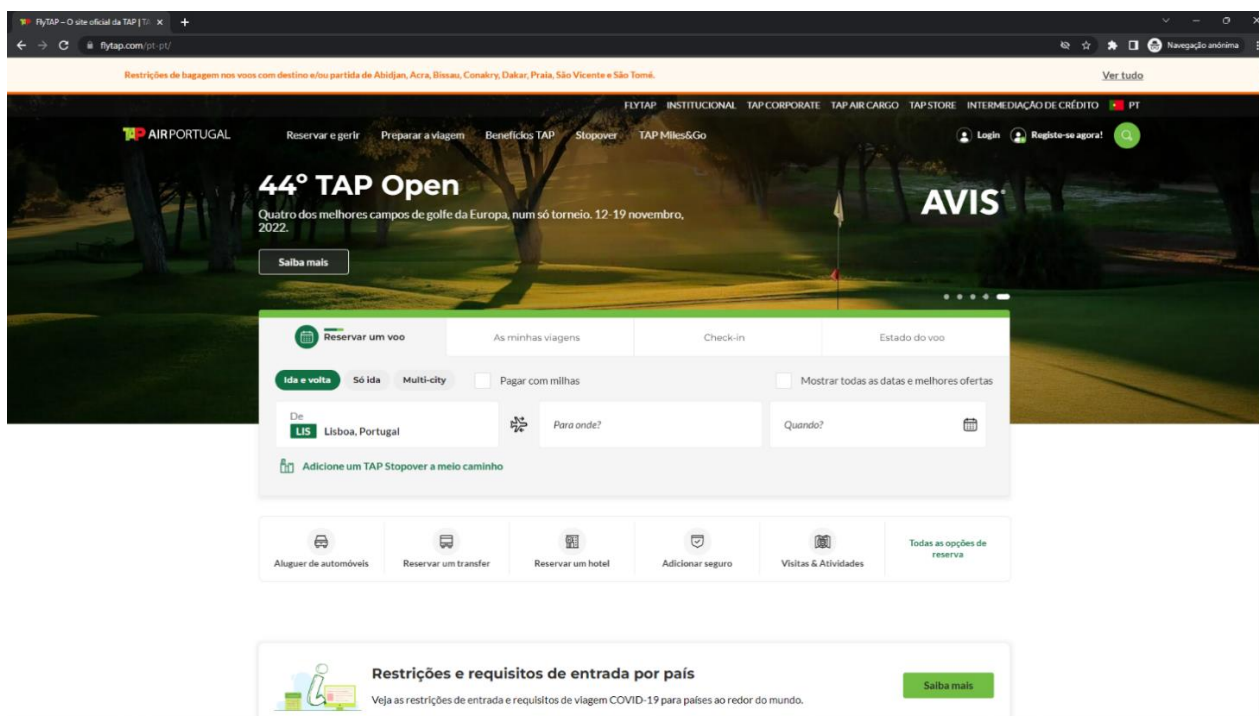
## 4 ANÁLISE CONCORRENCIAL

Análise de 3 sistemas que disponibilizam funcionalidades idênticas ou relacionadas com o AeroControl.

### 4.1.1 FlyTap

A próxima tabela resume as características do website *FlyTap*.

Tabela 1 – Descrição da FlyTap



Nome:	FlyTap
Site:	<a href="https://www.flytap.com/pt-pt/">https://www.flytap.com/pt-pt/</a>
Descrição:	A FlyTap é um sistema de reservas de voos.
Vantagens:	<ul style="list-style-type: none"><li>• Criação/Gestão de conta pessoal;</li><li>• Reservar/Gerir as reservas de voos;</li><li>• Descobrir destinos;</li><li>• Promoções;</li><li>• Várias informações sobre, por exemplo, o que existe a bordo (refeições, entretenimento, etc);</li><li>• Check-in automático/online;</li></ul>

Cofinanciado por:

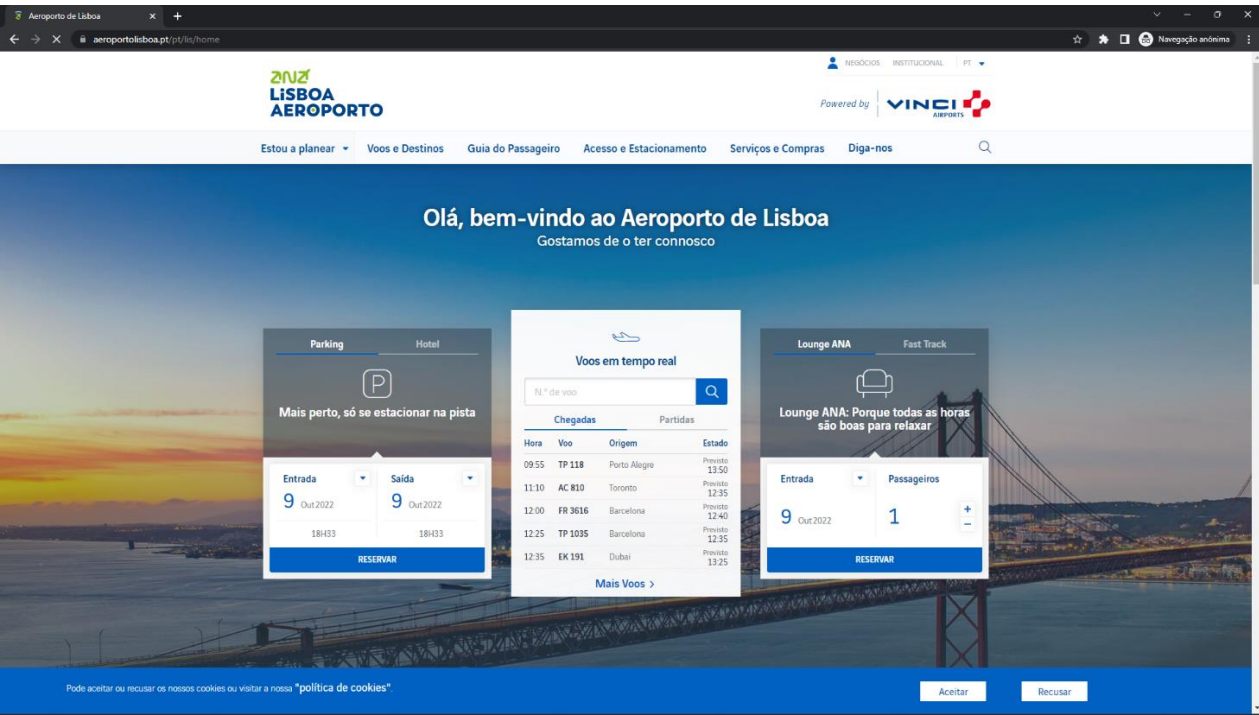


	<ul style="list-style-type: none"> <li>• Painel de suporte;</li> <li>• Sugestões e Reclamações;</li> <li>• Newsletter;</li> <li>• Sistema de tradução para várias línguas;</li> <li>• Informações dos voos em tempo real.</li> </ul>
<b>Desvantagens:</b>	Como existe muita informação/funcionalidades um utilizador pode sentir-se sobrecarregado quando visita o website.
<b>O que falta:</b>	Um suporte automatizado.

4.1.2 Lisboa Aeroporto

A próxima tabela resume as características do website Lisboa Aeroporto.

Tabela 2 – Descrição do Lisboa Aeroporto



Nome:	Lisboa Aeroporto
Site:	<a href="https://www.aeroporto lisboa.pt/pt/lis/home">https://www.aeroporto lisboa.pt/pt/lis/home</a>
Descrição:	Site oficial do Aeroporto de Lisboa – Partidas e chegadas em tempo real, companhias aéreas e destinos, acessos, reserva de estacionamento e muito mais.
Vantagens:	<ul style="list-style-type: none"><li>• Criação/Gestão de conta pessoal;</li><li>• Reservar/Gerir as reservas de voos;</li><li>• Várias informações sobre companhias aéreas, destinos, guia do passageiro, acesso e estacionamento, direções, serviços, compras e restaurantes;</li><li>• Sugestões e Reclamações;</li><li>• Informações de voos em tempo real;</li><li>• Reservas de parking, lounge ANA, Fast Track;</li><li>• Suporte automatizado;</li></ul>

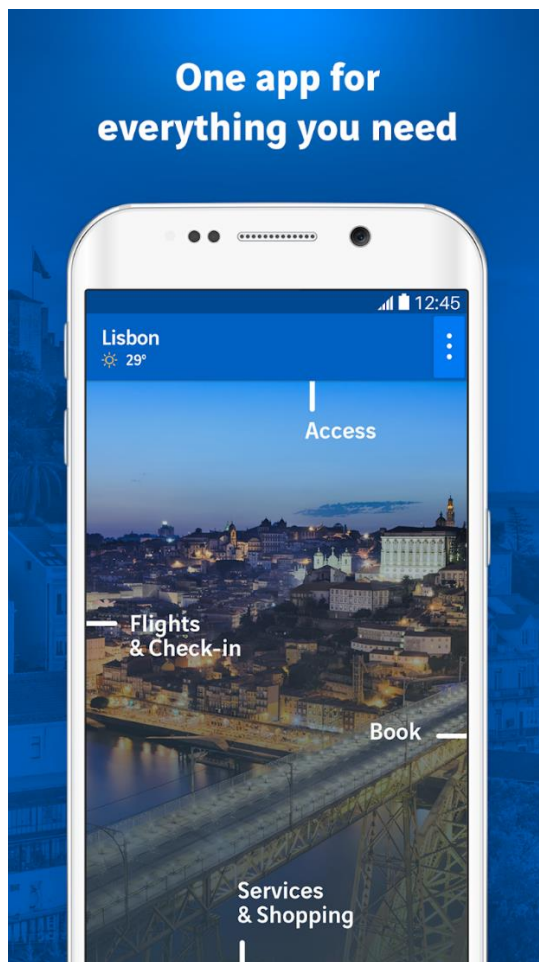
Cofinanciado por:

	<ul style="list-style-type: none"> <li>• Sistemas de tradução para inglês e espanhol.</li> </ul>
<b>Desvantagens:</b>	<ul style="list-style-type: none"> <li>• Notou-se que o website é lento a carregar, tornando a navegação difícil;</li> <li>• Preços dos transportes públicos desatualizados;</li> <li>• A parte de reservas não está no website principal o que pode originar confusão;</li> <li>• Algumas secções do website não são user friendly.</li> </ul>
<b>O que falta:</b>	<ul style="list-style-type: none"> <li>• Suporte de traduções para mais línguas;</li> <li>• Newsletter.</li> </ul>

#### 4.1.3 ANA Aeroportos de Portugal

A próxima tabela resume as características da aplicação ANA Aeroportos de Portugal.

Tabela 3 – Descrição do ANA Aeroportos de Portugal



<b>Nome:</b>	ANA Aeroportos de Portugal
<b>Site:</b>	<a href="https://play.google.com/store/apps/details?id=com.innovagency.ana&amp;hl=pt_PT&amp;gl=US">https://play.google.com/store/apps/details?id=com.innovagency.ana&amp;hl=pt_PT&amp;gl=US</a>
<b>Descrição:</b>	A aplicação ANA disponibiliza informações sobre os aeroportos de Lisboa, Porto, Faro, Madeira, Ponta Delgada, Porto Santo, Horta, Santa Maria e Flores.
<b>Vantagens:</b>	<ul style="list-style-type: none"><li>• Área de Oportunidades de Negócio;</li><li>• Horários de chegadas e partidas em tempo real;</li><li>• Sistemas de subscrição de voos para receber todas as informações de alteração;</li><li>• Check-in dentro da aplicação;</li></ul>

Cofinanciado por:



	<ul style="list-style-type: none"> <li>• Reservar o estacionamento ANA, Lounge e Fast Track;</li> <li>• Informações sobre todas as lojas; restaurantes e serviços que completam a viagem;</li> <li>• Sistema de promoções e descontos;</li> <li>• Informações sobre as direções e transportes para chegar ou sair do aeroporto.</li> </ul>
<b>Desvantagens:</b>	<ul style="list-style-type: none"> <li>• A UI/UX não é muito user friendly;</li> <li>• Algumas das secções estão sem imagens;</li> <li>• Preços dos transportes públicos desatualizados;</li> <li>• Não permite seguir dois voos (em caso de haver um voo com escala).</li> </ul>
<b>O que falta:</b>	Mostrar os principais itens dos restaurantes e lojas.

#### 4.1.4 Comparação dos Sistemas

De seguida está feita uma comparação entre os 3 sistemas analisados.

*Tabela 4 – Resumo das características/funcionalidades dos Sistemas concorrenciais*

<b>Funcionalidades</b>	<b>Sistema 1</b>	<b>Sistema 2</b>	<b>Sistema 3</b>
<i>Criação/Gestão de conta pessoal</i>	X	X	-
<i>Reservar/Gerir as reservas de voos</i>	X	X	-
<i>Descobrir destinos</i>	X	X	-
<i>Sistema de promoções/descontos</i>	X	-	-
<i>Extensa informação sobre o aeroporto</i>	X	X	X
<i>Check-in automático/online</i>	X	-	X
<i>Painel de suporte</i>	X	-	-
<i>Newsletter</i>	X	-	-
<i>Sugestões e Reclamações</i>	X	X	-
<i>Informação em tempo real de voos</i>	X	X	X
<i>Reservas de parking, lounge, Fast Track</i>	X	X	X
<i>Suporte automatizado</i>	-	X	-
<i>Tradução para várias/poucas línguas</i>	X	X	-
<i>Subscrição de um voo para receber informações sobre alguma alteração</i>	-	-	X

#### 4.1.5 Enquadramento da análise concorrencial no SI

A análise destes três sistemas é importante no desenvolvimento do SI, porque dá-nos conhecimento sobre algumas funcionalidades, que não foram pensadas antes, que podem ser implementadas no AeroControl, nomeadamente, reservas de parking, lounge e Fast-Track, tradução para várias línguas, etc.

## 5 TECNOLOGIAS UTILIZADAS

Sendo o projeto um projeto com as três componentes tal como mencionado anteriormente, Website, API e aplicação móvel, foram utilizadas diversas tecnologias, algumas lecionadas e outras exploradas através de pesquisa.

### 5.1 Website/API

- Linguagens de programação – PHP, JavaScript;
- Yii2 Framework – Framework utilizada para o desenvolvimento do projeto;
- MySQL – Sistema de base de dados utilizado.
- Codeception – Framework que permite fazer testes funcionais e não funcionais ao website;
- Extensões adicionadas:
  - Kartik
    - DatePicker, DateTimePicker, TimePicker – Extensões que permitem adicionar inputs de data e/ou hora a um formulário;
    - Active-Form - Extensão que permite utilizar um formulário melhorado da Yii2;
    - File-Input – Extensão que permite adicionar ficheiros a um formulário;
  - Log-Reader – Ferramenta da Yii2 no back-office que permite ao administrador visualizar no website o que foi feito por outros funcionários ou administradores;
  - Admin-LTE – Template já feita que foi utilizada na back-office do projeto.

### 5.2 Aplicação Móvel

- Linguagens de programação – Java, SQL;
- Glide – Biblioteca que permite carregar imagens através da Internet;
- Android-GIF-Drawable – Biblioteca que permite utilizar GIFs na aplicação;
- Material Design – Guia de design da Google que fornece diretrizes para a criação de interfaces de utilizador;
- Volley – Biblioteca utilizada para realizar os pedidos à API, esta que já é totalmente assíncrona;
- SQLite – Sistema de base de dados que armazena tudo o que é necessário para o bom funcionamento da aplicação no dispositivo, para que o utilizador possa aceder quando não possui internet;
- Data-Binding – Biblioteca que permite ao desenvolvedor fazer uma ligação automática entre a atividade e o ficheiro da interface do utilizador.

## 6 ARQUITETURA DO SISTEMA

Na seguinte imagem é possível observar o diagrama arquitetural do sistema.

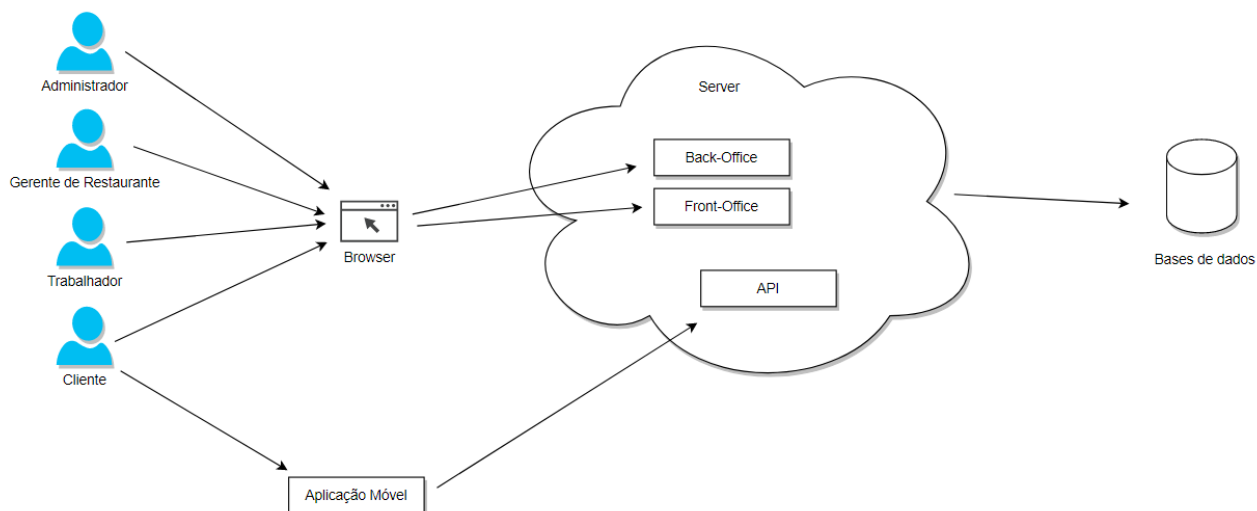


Figura 1 – Diagrama arquitetural do sistema

Tal como é possível observar, todos os utilizadores comunicam com o navegador (Back-office e/ou Front-office), que posteriormente estará conectado ao servidor e este quando necessário efetuará a comunicação com a base de dados. Por outro lado, na aplicação móvel apenas existe contacto com o cliente. A aplicação posteriormente comunicará com a API, localizada no servidor, que fará o contacto com a base de dados.

## 7 METODOLOGIAS E DIVISÃO DE TAREFAS

Para o desenvolvimento do projeto foram utilizadas algumas ferramentas:

- GitHub, para que o grupo pudesse trabalhar em simultâneo e para que tivéssemos um melhor controlo de versões, caso houvesse algum problema em uma versão por exemplo e utilizámos o git flow para diferenciar as branches que estão a ser desenvolvidas e que já estão finalizadas;
- Jira, com o intuito de ajudar a gerir o projeto através de sprints, e a gerir a divisão de tarefas, utilizámos o SCRUM de forma a otimizar o desenvolvimento ágil do grupo para melhor controlo.

Através do Jira foram feitas 2 sprints, sendo que a segunda é onde consta uma maior carga de trabalho, não só, porque, durou mais tempo, como também, foi a última componente do projeto final a entregar, ou seja, aquela que não demos máxima prioridade inicialmente.

Dentro destas sprints foram divididas tarefas pelos três elementos do grupo, de forma que todos os elementos trabalhassem um pouco em todas as funcionalidades.

## 8 ROLES E REQUISITOS DO SI

Tabela 5 – Roles dos diferentes end-users do SI

Role	Descrição detalhada	Back-office	Front-office	App mobile
<b>Administrador</b>	Gestão de toda a informação que está na base de dados, ou seja, gestão de toda a informação que consta na tabela 7 de requisitos funcionais do back-office.  Acesso a uma log, ou seja, um registo de todas as atividades realizadas na back-office.	X		
<b>Trabalhador</b>	Gestão parcial do back-office: <ul style="list-style-type: none"> <li>• Gerir perdidos e achados;</li> <li>• Gerir os voos;</li> <li>• Gerir as informações dos aviões;</li> <li>• Gerir as informações dos clientes.</li> </ul>	X		
<b>Gerentes</b>	Gestão dos menus do seu respetivo restaurante.	X		
<b>Cliente</b>	Consultar lista de voos;		X	X
	Consultar lista de restaurantes;		X	X
	Consultar lista da ementa do restaurante;		X	X
	Consultar as lojas que estão no aeroporto;		X	X
	Comprar bilhetes de voos;		X	X
	Gerir os seus bilhetes;		X	X
	Acesso a um sistema de promoções;		X	X
	Contactar o suporte através de email;		X	X
	Converter o bilhete para o QR Code;			X
	Enviar ticket para o suporte sobre os perdidos e achados.		X	X

## 8.1 Requisitos Funcionais

Tabela 6 – Requisitos funcionais do front-office

#	Requisito	Prioridade
RF-FO-01	O utilizador deve poder consultar a lista de voos.	Alta
RF-FO-02	O utilizador deve poder consultar a lista de restaurantes registados no aeroporto.	Alta
RF-FO-03	O utilizador deve poder comprar bilhetes de voos.	Alta
RF-FO-04	O utilizador deve poder consultar as ementas dos restaurantes registados.	Média
RF-FO-05	O utilizador deve poder alterar os dados do bilhete de voo.	Alta
RF-FO-06	O utilizador deve poder ter acesso aos bilhetes comprados.	Alta
RF-FO-07	O utilizador deve poder receber o bilhete do voo por email.	Média
RF-FO-08	O utilizador deve poder contactar o suporte através de email.	Média
RF-FO-09	O utilizador deve poder consultar a lista de lojas registadas no aeroporto.	Baixa
RF-FO-10	O utilizador deve poder comprar o bilhete para mais do que um passageiro.	Média
RF-FO-11	O utilizador deve poder escolher um entre vários métodos de pagamento.	Alta
RF-FO-12	O utilizador deve poder enviar um ticket de forma a contactar o trabalhador sobre os perdidos e achados.	Alta

Tabela 7 – Requisitos funcionais do back-office

#	Requisito	Prioridade
RF-BO-01	O trabalhador deve poder gerir os perdidos e achados.	Alta
RF-BO-02	O administrador deve gerir os trabalhadores do aeroporto.	Alta
RF-BO-03	O trabalhador deve poder gerir as promoções.	Média
RF-BO-04	O trabalhador deve poder gerir os voos.	Alta
RF-BO-05	O trabalhador deve poder gerir os aviões.	Alta
RF-BO-06	O administrador deve poder gerir os restaurantes registadas na base de dados.	Alta
RF-BO-07	O administrador deve poder gerir os gerentes dos restaurantes.	Alta
RF-BO-08	O gerente do restaurante deve poder atualizar a ementa do seu restaurante.	Alta
RF-BO-09	O trabalhador deve poder gerir os clientes.	Alta
RF-BO-10	O administrador deve poder ter acesso ao Log do servidor.	Alta
RF-BO-11	O administrador deve poder gerir as lojas registadas na base de dados.	Baixa
RF-BO-12	O administrador deve poder gerir as companhias que estão no aeroporto.	Média
RF-BO-13	O trabalhador deve poder gerir os métodos de pagamento possíveis a utilizar.	Média
RF-BO-14	O trabalhador deve poder comunicar com o cliente através de um sistema de tickets sobre os perdidos e achados.	Média
RF-BO-14	O trabalhador deve poder gerir os aeroportos de destino dos voos.	Média

Cofinanciado por:





Tabela 8 – Requisitos funcionais da aplicação android

#	Requisito	Prioridade
RF-AM-01	O utilizador deve poder consultar a lista de voos.	Alta
RF-AM-02	O utilizador deve poder consultar a lista de restaurantes registados no aeroporto.	Alta
RF-AM-03	O utilizador deve poder comprar bilhetes de voos.	Alta
RF-AM-04	O sistema deve poder converter o bilhete do utilizador para QR Code.	Média
RF-AM-05	O utilizador deve poder enviar um ticket de forma a contactar o trabalhador sobre os perdidos e achados.	Média
RF-AM-06	O utilizador deve poder consultar as ementas dos restaurantes registados.	Média
RF-AM-07	O utilizador deve poder alterar os dados do bilhete de voo.	Alta
RF-AM-08	O utilizador deve poder ter acesso aos bilhetes comprados.	Alta
RF-AM-09	O utilizador deve poder receber o bilhete do voo por email.	Média
RF-AM-10	O utilizador deve poder consultar a lista de lojas registadas no aeroporto.	Baixa
RF-AM-11	O utilizador deve poder comprar o bilhete para mais do que um passageiro.	Média
RF-AM-12	O utilizador deve poder escolher um entre vários métodos de pagamento.	Alta

Tabela 9 – Cruzamento dos requisitos funcionais do front-office e respetivos roles

#	Requisito	Administrador	Trabalhador	Gerente	Cliente
RF-FO-01	O utilizador deve poder consultar a lista de voos.				X
RF-FO-02	O utilizador deve poder consultar a lista de restaurantes registados no aeroporto.				X
RF-FO-03	O utilizador deve poder comprar bilhetes de voos.				X
RF-FO-04	O utilizador deve poder consultar as ementas dos restaurantes registados.				X
RF-FO-05	O utilizador deve poder alterar os dados do bilhete de voo.				X
RF-FO-06	O utilizador deve poder ter acesso aos bilhetes comprados.				X
RF-FO-07	O utilizador deve poder receber o bilhete do voo por email.				X
RF-FO-08	O utilizador deve poder contactar o suporte através de email.				X
RF-FO-09	O utilizador deve poder consultar a lista de lojas registadas no aeroporto.				X

<b>RF-FO-10</b>	O utilizador deve poder comprar o bilhete para mais do que um passageiro.	X
<b>RF-FO-11</b>	O utilizador deve poder escolher um entre vários métodos de pagamento.	X
<b>RF-FO-12</b>	O utilizador deve poder enviar um ticket de forma a contactar o trabalhador sobre os perdidos e achados.	X

Tabela 10 - Cruzamento dos requisitos funcionais do back-office e respetivos roles

#	Requisito	Administrador	Trabalhador	Gerente	Cliente
RF-BO-01	O trabalhador deve poder gerir os perdidos e achados.	X	X		
RF-BO-02	O administrador deve gerir os trabalhadores do aeroporto.	X			
RF-BO-03	O trabalhador deve poder gerir as promoções.	X			
RF-BO-04	O trabalhador deve poder gerir os voos.	X	X		
RF-BO-05	O trabalhador deve poder gerir os aviões.	X	X		
RF-BO-06	O administrador deve poder gerir os restaurantes registadas na base de dados.	X			
RF-BO-07	O administrador deve poder gerir os gerentes dos restaurantes.	X			
RF-BO-08	O gerente do restaurante deve poder atualizar a ementa do seu restaurante.	X		X	
RF-BO-09	O trabalhador deve poder gerir os clientes.	X	X		
RF-BO-10	O administrador deve poder ter acesso ao Log do servidor.	X			

<b>RF-BO-11</b>	O administrador deve poder gerir as lojas registadas na base de dados.	X	
<b>RF-BO-12</b>	O administrador deve poder gerir as companhias que estão no aeroporto.	X	
<b>RF-BO-13</b>	O trabalhador deve poder gerir os métodos de pagamento possíveis a utilizar.	X	
<b>RF-BO-14</b>	O trabalhador deve poder comunicar com o cliente através de um sistema de tickets sobre os perdidos e achados.	X	X
<b>RF-BO-14</b>	O trabalhador deve poder gerir os aeroportos de destino dos voos.	X	X

Tabela 11 - Cruzamento dos requisitos funcionais da aplicação android e respetivos roles

#	Requisito	Administrador	Trabalhador	Gerente	Cliente
RF-AM-01	O utilizador deve poder consultar a lista de voos.				X
RF-AM-02	O utilizador deve poder consultar a lista de restaurantes registados no aeroporto.				X
RF-AM-03	O utilizador deve poder comprar bilhetes de voos.				X
RF-AM-04	O sistema deve poder converter o bilhete do utilizador para QR Code.				X
RF-AM-05	O utilizador deve poder enviar um ticket de forma a contactar o trabalhador sobre os perdidos e achados.				X
RF-AM-06	O utilizador deve poder consultar as ementas dos restaurantes registados.				X
RF-AM-07	O utilizador deve poder alterar os dados do bilhete de voo.				X
RF-AM-08	O utilizador deve poder ter acesso aos bilhetes comprados.				X
RF-AM-09	O utilizador deve poder receber o bilhete do voo por email.				X

---

<b>RF-AM-10</b>	O utilizador deve poder consultar a lista de lojas registadas no aeroporto.	X
<b>RF-AM-11</b>	O utilizador deve poder comprar o bilhete para mais do que um passageiro.	X
<b>RF-AM-12</b>	O utilizador deve poder escolher um entre vários métodos de pagamento.	X

---

## 8.2 Requisitos Não Funcionais de Usabilidade (Usability)

Tabela 12 – Requisitos Não funcionais de Usabilidade do website

#	Requisito	Prioridade
RNF-USA-01	O website deve ter uma interface gráfica simples de perceber.	Alta
RNF-USA-02	O website deve ter uma opção de ajuda em páginas menos perceptíveis.	Baixa
RNF-USA-03	O website tem de ter um design responsivo.	Alta
RNF-USA-04	O website deve estar disponível para pessoas com necessidades especiais.	Baixa
RNF-USA-05	Os utilizadores devem ser corretamente informados de como efetuar as operações que pretendem para evitar a ocorrência de erros.	Média
RNF-USA-06	O website deve ter um tema escuro.	Baixa
RNF-USA-07	Deve de ser possível ativar qualquer funcionalidade do website com no máximo 3 cliques do rato.	Baixa
RNF-USA-08	Todos os botões do website devem ter as mesmas características a nível de tamanho e cor.	Média
RNF-USA-09	Todas as funcionalidades do sistema devem poder ser acedidas não só com o rato, mas também com o teclado.	Baixa



*Tabela 13 - Requisitos Não Funcionais de Usabilidade da aplicação android*

#	Requisito	Prioridade
<b>RNF-USA-01</b>	A aplicação deve ter uma interface gráfica simples de perceber.	Alta
<b>RNF-USA-02</b>	A aplicação deve ter uma opção de ajuda nos ecrãs menos perceptíveis.	Baixa
<b>RNF-USA-03</b>	Os utilizadores devem ser corretamente informados de como efetuar as operações que pretendem para evitar a ocorrência de erros.	Média
<b>RNF-USA-04</b>	A aplicação deve ter um tema escuro.	Baixa
<b>RNF-USA-05</b>	Deve de ser possível ativar qualquer funcionalidade da aplicação com no máximo 3 cliques.	Baixa
<b>RNF-USA-06</b>	Todos os botões do website devem ter as mesmas características a nível de tamanho e cor.	Média
<b>RNF-USA-07</b>	A interface não deve bloquear quando uma operação é realizada.	Alta

### 8.3 Requisitos Não Funcionais de Fiabilidade (Reliability)

*Tabela 14 - Requisitos Não Funcionais de Fiabilidade do website*

#	Requisito	Prioridade
<b>RNF-FIA-01</b>	Todas as páginas do website devem poder ser acedidas 98% das vezes, sem falhas.	Alta
<b>RNF-FIA-02</b>	O website deve poder ser acedido por mais de uma pessoa em simultâneo.	Média
<b>RNF-FIA-03</b>	O website deverá apresentar um design responsivo 97% das vezes.	Alta

*Tabela 15 - Requisitos Não Funcionais de Fiabilidade da aplicação android*

#	Requisito	Prioridade
<b>RNF-FIA-01</b>	Todos os ecrãs da aplicação devem poder ser acedidos 98% das vezes, sem falhas.	Alta
<b>RNF-FIA-02</b>	A aplicação deve poder ser acedida em dispositivos diferentes em simultâneo.	Média

## 8.4 Requisitos Não Funcionais de Segurança (Security)

Tabela 16 - Requisitos Não Funcionais de Segurança do website

#	Requisito	Prioridade
RNF-SEG-01	O website deve encriptar a password do utilizador no registo.	Alta
RNF-SEG-02	O utilizador deve necessitar de credenciais para a compra de bilhetes.	Alta
RNF-SEG-03	Devem ser efetuadas 2 cópias de segurança dos dados do sistema, sendo que uma delas deve encontrar-se em local externo.	Alta
RNF-SEG-04	O website deve-se encontrar protegido contra XXS (Cross-Site Scripting), SQL Injection, DDoS, Man-in-the-Middle e Brute-Forces ataques.	Média
RNF-SEG-05	A base de dados deve ser protegida para acesso apenas de utilizadores autorizados.	Alta
RNF-SEG-06	Deve ser usado o GitHub para controlo de versões.	Alta
RNF-SEG-07	O website deve ter um controlo de acesso baseado em funções (RBAC).	Alta

Tabela 17 - Requisitos Não Funcionais de Segurança da aplicação android

#	Requisito	Prioridade
RNF-SEG-01	O utilizador deve necessitar de credenciais para a compra de bilhetes.	Alta
RNF-SEG-02	Devem ser efetuadas 2 cópias de segurança dos dados do sistema, sendo que uma delas deve encontrar-se em local externo.	Alta
RNF-SEG-03	Deve ser usado o GitHub para controlo de versões.	Alta

Tabela 18 - Requisitos Não Funcionais de Segurança da API

#	Requisito	Prioridade
<b>RNF-SEG-01</b>	A API deve encriptar a password do utilizador no registo.	Alta
<b>RNF-SEG-02</b>	Devem ser efetuadas 2 cópias de segurança dos dados do sistema, sendo que uma delas deve encontrar-se em local externo.	Alta
<b>RNF-SEG-03</b>	Deve ser usado o GitHub para controlo de versões.	Alta
<b>RNF-SEG-04</b>	A API deve-se encontrar protegida contra DDoS.	Média

Cofinanciado por:



UNIÃO EUROPEIA  
Fundo Social Europeu

## 8.5 Requisitos Não Funcionais de Eficiência (Performance)

*Tabela 19 - Requisitos Não Funcionais de Eficiência do website*

#	Requisito	Prioridade
RNF-EFI-01	O tempo de carregamento das páginas do website não deve superar 2 segundos.	Alta
RNF-EFI-02	O website deve suportar mais que 1000 utilizadores, ao mesmo tempo, sem haver diminuição na performance.	Alta
RNF-EFI-03	Deve ser possível atualizar a aplicação sem perder os dados disponibilizados.	Alta

*Tabela 20 - Requisitos Não Funcionais de Eficiência da aplicação android*

#	Requisito	Prioridade
RNF-EFI-01	O tempo de carregamento da aplicação não deve superar 2 segundos.	Alta
RNF-EFI-02	Deve ser possível atualizar a aplicação sem perder os dados disponibilizados.	Alta

*Tabela 21 - Requisitos Não Funcionais de Eficiência da API*

#	Requisito	Prioridade
RNF-EFI-01	A API deve suportar vários pedidos sem diminuir a performance.	Alta

## 8.6 Requisitos Não Funcionais de Disponibilidade (Availability)

*Tabela 22 - Requisitos Não Funcionais de Disponibilidade do website*

#	Requisito	Prioridade
RNF-DIS-01	O cliente deve poder aceder diariamente ao website.	Alta
RNF-DIS-02	O gerente de um restaurante pode alterar o menu a qualquer hora.	Alta
RNF-DIS-03	O administrador deve poder adicionar um novo trabalhador sempre que pretender.	Alta

*Tabela 23 - Requisitos Não Funcionais de Disponibilidade da aplicação android*

#	Requisito	Prioridade
RNF-DIS-01	O cliente deve poder aceder diariamente à aplicação.	Alta

## 8.7 Requisitos Não Funcionais de Ambiente (Portability)

Tabela 24 - Requisitos Não Funcionais de Ambiente website

#	Requisito	Prioridade
<b>RNF-AMB-01</b>	O website deve ser compatível em todos os browsers mais populares (Chrome, Edge, Safari, Firefox).	Média

Tabela 25 - Requisitos Não Funcionais de Ambiente da aplicação android

#	Requisito	Prioridade
<b>RNF-AMB-01</b>	A aplicação deve ser capaz de correr desde a versão 5.0 do Android até à atual.	Alta
<b>RNF-AMB-02</b>	A aplicação deve utilizar a webservice REST/RESTful (API).	Alta

## 8.8 Requisitos Não Funcionais de Desenvolvimento (Implementation)

Tabela 26 - Requisitos Não Funcionais de Desenvolvimento do website

#	Requisito	Prioridade
RNF-DES-01	O website deve ser desenvolvido utilizando HTML, CSS, JavaScript, PHP.	Alta
RNF-DES-02	O website deve utilizar a Framework Yii2.	Alta
RNF-DES-03	O website deve utilizar a library JQuery.	Baixa
RNF-DES-04	O website deve utilizar a Framework Bootstrap.	Baixa
RNF-DES-05	O website deve utilizar a Framework Scrum para o gerenciamento do mesmo.	Alta
RNF-DES-06	O website deve utilizar a ferramenta Jira para monitorizar o Scrum.	Alta
RNF-DES-07	O tempo de desenvolvimento não deve ultrapassar os 4 meses.	Alta
RNF-DES-08	Os testes do website devem ser implementados recorrendo à Framework Codeception.	Alta



Tabela 27 - Requisitos Não Funcionais de Desenvolvimento da aplicação android

#	Requisito	Prioridade
<b>RNF-DES-01</b>	A aplicação deve ser desenvolvida em Java.	Alta
<b>RNF-DES-02</b>	A aplicação deve ser desenvolvida com o IDE Android Studio.	Alta
<b>RNF-DES-03</b>	A aplicação deve utilizar a library Volley.	Alta
<b>RNF-DES-04</b>	A aplicação deve utilizar a Framework Scrum para o gerenciamento da mesma.	Alta
<b>RNF-DES-05</b>	A aplicação deve utilizar a ferramenta Jira para monitorizar o Scrum.	Alta
<b>RNF-DES-06</b>	O tempo de desenvolvimento não deve ultrapassar os 4 meses.	Alta
<b>RNF-DES-07</b>	A aplicação deve registar dados em ficheiros locais (shared preferences) e em base de dados SQL local (ex.: cache de informação ...).	Alta
<b>RNF-DES-08</b>	A aplicação deve ser desenvolvida consoante a arquitetura MVC.	Alta

Tabela 28 - Requisitos Não Funcionais de Desenvolvimento da API

#	Requisito	Prioridade
<b>RNF-DES-01</b>	A API deve devolver as respostas em JSON.	Alta
<b>RNF-DES-02</b>	A API deve utilizar a Framework Scrum para o gerenciamento da mesma.	Alta
<b>RNF-DES-03</b>	A API deve utilizar a ferramenta Jira para monitorizar o Scrum.	Alta
<b>RNF-DES-04</b>	O tempo de desenvolvimento não deve ultrapassar os 4 meses.	Alta
<b>RNF-DES-05</b>	As funcionalidades da API devem ser baseadas em <i>messaging</i> ( <i>publish/subscribe</i> ).	Alta

Cofinanciado por:



## 9 USER STORIES

Tabela 29 - User Stories - Front-office

<b>Título:</b> US-FO-1 – Efetuar consulta de voos.	<b>SP:</b> 13
<b>Descrição:</b> Como utilizador, quero poder consultar uma lista de todos os voos para que possa escolher um determinado voo.	
<b>Critérios de Aceitação:</b> <ul style="list-style-type: none"><li>• O utilizador tem de inserir o destino e a origem do voo.</li><li>• O utilizador tem de escolher se são voos diretos ou com escalas;</li></ul>	
<b>Título:</b> US-FO-2 – Efetuar consulta dos restaurantes do Aeroporto	<b>SP:</b> 3
<b>Descrição:</b> Como utilizador, quero poder consultar uma lista com todos os restaurantes do aeroporto, de forma a saber quais existem dentro do aeroporto.	
<b>Título:</b> US-FO-3 – Efetuar compra de bilhetes de voos.	<b>SP:</b> 8
<b>Descrição:</b> Como utilizador, quero poder comprar bilhetes de um determinado voo, de forma a poder posteriormente viajar.	
<b>Critérios de Aceitação:</b> <ul style="list-style-type: none"><li>• O utilizador tem de estar registado no sistema;</li><li>• O utilizador tem de estar autenticado;</li><li>• O utilizador tem de preencher todos os campos relativos ao voo selecionado.</li></ul>	
<b>Título:</b> US-FO-4 – Efetuar consulta das ementas dos restaurantes.	<b>SP:</b> 5
<b>Descrição:</b> Como utilizador, quero poder consultar uma lista com a ementa do restaurante selecionado, de forma a saber as ofertas do restaurante.	
<b>Critérios de Aceitação:</b> <ul style="list-style-type: none"><li>• O utilizador tem de escolher um restaurante primeiro.</li></ul>	

<b>Título:</b> US-FO-5 – Efetuar alterações no bilhete do voo.	<b>SP:</b> 8
--	--------------

**Descrição:** Como utilizador, quero poder fazer alterações no bilhete, para que possa alterar as suas datas ou de forma a cancelar o próprio bilhete.

**Critérios de Aceitação:**

- O utilizador tem de estar autenticado;
- O utilizador tem de ter comprado o bilhete anteriormente;
- Para que haja alteração de data tem de haver um voo na data que o utilizador necessita.

<b>Título:</b> US-FO-6 – Efetuar consulta dos bilhetes comprados pelo utilizador.	<b>SP:</b> 5
---	--------------

**Descrição:** Como utilizador, quero poder visualizar todos os meus bilhetes comprados, de forma a poder, no futuro, consultar o histórico dos meus bilhetes.

**Critérios de Aceitação:**

- O utilizador tem de estar autenticado;
- O utilizador tem de ter comprado bilhetes.

<b>Título:</b> US-FO-7 – Receção do bilhete do voo através de email.	<b>SP:</b> 8
--	--------------

**Descrição:** Como utilizador, quero poder receber o bilhete do voo através do email, para que o possa imprimir.

**Critérios de Aceitação:**

- O utilizador tem de estar autenticado;
- O utilizador tem de comprar o bilhete anteriormente.

<b>Título:</b> US-FO-8 – Contactar o suporte.	<b>SP:</b> 5
---	--------------

**Descrição:** Como utilizador, quero poder contactar o suporte através de um email, para que possa obter ajuda no que necessitar.

**Critérios de Aceitação:**

- O utilizador tem de preencher os dados pessoais necessários e a mensagem que pretende enviar.

<b>Título:</b> US-FO-9 – Efetuar a consulta de lojas registadas no aeroporto.	<b>SP:</b> 3
---	--------------

**Descrição:** Como utilizador, quero poder consultar as lojas do aeroporto, de forma a saber quais existem dentro do aeroporto.

<b>Título:</b> US-FO-10 – Efetuar compra de bilhete para vários passageiros.	<b>SP:</b> 8
--	--------------

**Descrição:** Como utilizador, quero poder comprar um lugar para vários passageiros em um só bilhete, para que não seja forçado a comprar um bilhete para cada passageiro.

**Critérios de Aceitação:**

- O utilizador tem de estar autenticado;
- O utilizador tem de preencher todos os dados dos passageiros que pretende incluir na viagem.

<b>Título:</b> US-FO-11 – Efetuar a compra através de um determinado método de pagamento.	<b>SP:</b> 5
---	--------------

**Descrição:** Como utilizador, quero ter várias opções de pagamento ao efetuar uma compra, para que possa efetuar o pagamento através de outro método em caso de erro.

**Critérios de Aceitação:**

- O utilizador tem de estar autenticado;
- O utilizador tem de preencher todos os dados para a compra do voo.

<b>Título:</b> US-FO-12 – Contactar o suporte sobre perdidos e achados.	<b>SP:</b> 8
---	--------------

**Descrição:** Como utilizador, quero poder enviar um ticket ao suporte de perdidos e achados, de forma a contactar o mesmo sobre um item perdido.

**Critérios de Aceitação:**

- O utilizador tem de estar autenticado;
- O utilizador tem de escrever a mensagem a ser enviada para o suporte.

<b>Título:</b> US-BO-1 – Efetuar gestão dos itens de perdidos e achados.	<b>SP:</b> 8
--	--------------

**Descrição:** Como trabalhador, devo poder gerir os itens dos perdidos e achados para que possa fornecer um melhor suporte aos clientes.

**Critérios de Aceitação:**

- O trabalhador tem de estar autenticado;
- Tem de haver objetos perdidos.

<b>Título:</b> US-BO-2 – Contactar com os clientes sobre os perdidos e achados.	<b>SP:</b> 5
---	--------------

**Descrição:** Como trabalhador, devo poder enviar uma mensagem ao cliente que efetuou o ticket para que o mesmo seja ajudado.

**Critérios de Aceitação:**

- O trabalhador tem de estar autenticado;
- Tem de haver tickets enviados pelos clientes.

<b>Título:</b> US-BO-3 – Efetuar a gestão dos voos.	<b>SP:</b> 5
---	--------------

**Descrição:** Como trabalhador, quero poder gerir os voos do aeroporto para que posteriormente o cliente possa consultar a lista de voos.

**Critérios de Aceitação:**

- O trabalhador tem de estar autenticado;
- Tem de haver aviões registados;
- O trabalhador tem de inserir voos para que os possa alterar posteriormente;
- O trabalhador deve inserir uma promoção para o voo caso exista.

<b>Título:</b> US-BO-4 – Efetuar a gestão de aviões.	<b>SP:</b> 5
--	--------------

**Descrição:** Como trabalhador, quero poder gerir os aviões que vão estar no aeroporto para que possa associar o avião ao voo.

**Critérios de Aceitação:**

- O trabalhador tem de estar autenticado;
- O trabalhador tem de inserir aviões para que possa efetuar alterações posteriormente.

<b>Título:</b> US-BO-5 – Efetuar a inserção dos restaurantes.	<b>SP:</b> 3
---	--------------

**Descrição:** Como administrador, quero poder inserir restaurantes na base de dados para que o gerente dos mesmos possa gerir as ementas.

**Critérios de Aceitação:**

- O administrador tem de estar autenticado;
- O administrador tem de preencher todos os campos para que o restaurante seja inserido.

<b>Título:</b> US-BO-6 – Efetuar gestão dos gerentes dos restaurantes.	<b>SP:</b> 3
--	--------------

**Descrição:** Como administrador, quero poder gerir os gerentes de cada restaurante para que cada gerente possa gerir a ementa do seu restaurante.

**Critérios de Aceitação:**

- O administrador tem de estar autenticado;
- O restaurante tem de estar registado.

<b>Título:</b> US-BO-7 – Efetuar gestão dos restaurantes.	<b>SP:</b> 8
---	--------------

**Descrição:** Como gerente do restaurante, quero poder gerir a ementa do restaurante para que o cliente possa visualizar a ementa através do website/App Mobile.

**Critérios de Aceitação:**

- O gerente tem de estar autenticado;
- O restaurante tem de estar registado.

<b>Título:</b> US-BO-8 – Efetuar a gestão de clientes.	<b>SP:</b> 5
--	--------------

**Descrição:** Como trabalhador, quero poder gerir as informações dos clientes, para que possa fornecer suporte ao cliente se o mesmo necessitar.

**Critérios de Aceitação:**

- O trabalhador tem de estar autenticado;
- O cliente tem de estar registado.

<b>Título:</b> US-BO-9 – Sistema de Log servidor.	<b>SP:</b> 5
---	--------------

**Descrição:** Como administrador, quero poder consultar todas as atividades que foram realizadas na back-office para que possa ter um melhor controlo de todas as operações efetuadas.

**CrITÉrios de Aceitação:**

- O administrador tem de estar autenticado;
- Tem de haver operações antigas.

<b>Título:</b> US-BO-10 – Efetuar a gestão das lojas registadas na base de dados.	<b>SP:</b> 3
---	--------------

**Descrição:** Como administrador, quero poder registar e alterar as lojas presentes na base de dados para que o cliente possa consultá-las.

**CrITÉrios de Aceitação:**

- O administrador tem de estar autenticado;
- O administrador tem de preencher todos os campos obrigatórios para que o registo seja efetuado.

<b>Título:</b> US-BO-11 – Efetuar a gestão das companhias que frequentam o aeroporto.	<b>SP:</b> 5
---	--------------

**Descrição:** Como administrador, quero poder gerir as companhias que frequentam o aeroporto para que o cliente possa ter uma opção de voos mais vasta.

**CrITÉrios de Aceitação:**

- O administrador tem de estar autenticado;
- Para o registo ser efetuado todos os campos obrigatórios têm de estar preenchidos.

<b>Título:</b> US-BO-12 – Efetuar a gestão dos métodos de pagamento aceites.	<b>SP:</b> 5
--	--------------

**Descrição:** Como trabalhador, quero poder gerir os métodos de pagamento para que possa fornecer mais opções de pagamento aos clientes e desabilitar os métodos de pagamento que deixam de ser aceites.

**CrITÉrios de Aceitação:**

- O trabalhador tem de estar autenticado;
- O trabalhador tem de preencher todos os campos obrigatórios para que o registo seja efetuado;
- Para desabilitar um método o mesmo já tem de estar registado na base de dados.



**Título:** US-BO-13 – Efetuar a gestão dos trabalhadores.

**SP:5**

**Descrição:** Como administrador, quero poder gerir os trabalhadores do restaurante, para que os mesmos possam ajudar na gestão do aeroporto.

**CrITÉrios de Aceitação:**

- O administrador tem de estar autenticado;
- O administrador tem de preencher todos os campos obrigatórios para que o registo seja efetuado;

**Título:** US-BO-14 – Efetuar a gestão dos aeroportos de destino dos voos.

**SP:5**

**Descrição:** Como trabalhador, quero poder gerir os aeroportos de destino dos voos, para que os clientes possam selecionar o local de destino do voo.

**CrITÉrios de Aceitação:**

- O trabalhador tem de estar autenticado;

<b>Título:</b> US-AM-1 – Efetuar consulta de voos	<b>SP:</b> 5
---	--------------

**Descrição:** Como utilizador, quero ser capaz de consultar a lista de voos, de forma a poder escolher o que quero.

**Critérios de Aceitação:**

- O utilizador tem de inserir o destino e a origem do voo;
- O utilizador tem de escolher se são voos diretos ou com escalas.

<b>Título:</b> US-AM-2 – Efetuar consulta lista de restaurantes	<b>SP:</b> 5
---	--------------

**Descrição:** Como utilizador, quero poder ser capaz de consultar a lista de restaurantes registados no aeroporto, de forma a conseguir escolher qual restaurante ir comer.

<b>Título:</b> US-AM-3 – Efetuar a compra de bilhetes de avião	<b>SP:</b> 8
--	--------------

**Descrição:** Como utilizador, quero ser capaz de comprar um bilhete de avião, de forma a conseguir embarcar no mesmo.

**Critérios de Aceitação:**

- O utilizador tem de estar registado no sistema;
- O utilizador tem de estar autenticado;
- O utilizador tem de preencher todos os campos relativos ao voo selecionado.

<b>Título:</b> US-AM-4 – Conversão de bilhete para QR Code	<b>SP:</b> 13
--	---------------

**Descrição:** Como utilizador, quero que o meu bilhete possa ser convertido para QR Code para que o consiga aceder e mostrar mais facilmente.

**Critérios de Aceitação:**

- O utilizador tem de estar autenticado;
- O utilizador tem de ter o bilhete comprado.

<b>Título:</b> US-AM-5 – Contactar o suporte sobre perdidos e achados.	<b>SP:</b> 8
--	--------------

**Descrição:** Como utilizador, quero enviar um ticket para consultar os pedidos e achados, de forma a encontrar os meus materiais perdidos.

**Critérios de Aceitação:**

- O utilizador tem de estar registado no sistema;
- O utilizador tem de estar autenticado.

<b>Título:</b> US-AM-6 – Efetuar consulta das ementas dos restaurantes	<b>SP:</b> 5
--	--------------

**Descrição:** Como utilizador, quero poder consultar todas as ementas dos restaurantes do aeroporto, de modo a conseguir ver as ofertas de cada restaurante.

**Critérios de Aceitação:**

- O utilizador tem de escolher o restaurante primeiro.

<b>Título:</b> US-AM-7 – Efetuar alteração dos dados do bilhete	<b>SP:</b> 8
---	--------------

**Descrição:** Como utilizador, quero poder fazer alterações no bilhete de forma a alterar as suas datas ou de forma a cancelar o próprio bilhete.

**Critérios de Aceitação:**

- O utilizador tem de estar autenticado;
- O utilizador tem de ter comprado o bilhete anteriormente;
- Para que haja alteração de data tem de haver um voo na data que o utilizador necessita.

<b>Título:</b> US-AM-8 – Consultar os bilhetes comprados	<b>SP:</b> 5
--	--------------

**Descrição:** Como utilizador, quero ter acesso aos bilhetes de forma a ter acesso aos bilhetes a qualquer altura.

**Critérios de Aceitação:**

- O utilizador tem de estar registado no sistema;
- O utilizador tem de estar autenticado;
- O utilizador tem de ter efetuado a compra do bilhete.

<b>Título:</b> US-AM-9 – Receber o bilhete por email	<b>SP:</b> 8
--	--------------

**Descrição:** Como utilizador, quero receber o bilhete do voo no email para que o possa imprimir.

**Critérios de Aceitação:**

- O utilizador tem de estar autenticado;
- O utilizador tem de comprar o bilhete anteriormente.

<b>Título:</b> US-AM-10 – Efetuar consulta da lista de lojas	<b>SP:</b> 5
--	--------------

**Descrição:** Como utilizador, quero consultar a lista de lojas existentes no aeroporto de forma a saber quais as lojas existem no aeroporto.

<b>Título:</b> US-AM-11 – Efetuar compra do bilhete para mais que um passageiro	<b>SP:</b> 13
---	---------------

**Descrição:** Como utilizador, quero poder comprar um bilhete para mais do que uma pessoa de forma a poder comprar os bilhetes para a minha família.

**Critérios de Aceitação:**

- O utilizador tem de estar autenticado;
- O utilizador tem de preencher todos os dados do passageiro que pretende incluir na viagem.

<b>Título:</b> US-AM-12 – Efetuar compra do bilhete através de um método de pagamento	<b>SP:</b> 3
---	--------------

**Descrição:** Como utilizador quero ter várias opções de pagamento ao efetuar uma compra para que possa efetuar o pagamento através de outro método em caso de erro.

**Critérios de Aceitação:**

- O utilizador tem de estar autenticado;
- O utilizador tem de preencher todos os dados para a compra do voo.

## 10 DIAGRAMA DE ENTIDADE-RELACIONAMENTO

O DER consiste em várias entidades, estas sendo:

- o utilizador, administrador, cliente, gerente, funcionário e função do mesmo;
- o restaurante e respetivos itens;
- as lojas;
- o voo, os aviões, as companhias, os aeroportos, os passageiros e os métodos de pagamentos;
- os tickets de suporte com as respetivas mensagens e os itens perdidos.

Uma pessoa registada no sistema tem de ser **obrigatoriamente** um cliente, um administrador, um gerente ou um funcionário e nunca pode assumir vários perfis.

Decidiu-se **dividir** o DER em **duas imagens**, para uma melhor visualização. A 1ª (Figura 2) contém só as relações e a 2ª (Figura 3) contém só os atributos.

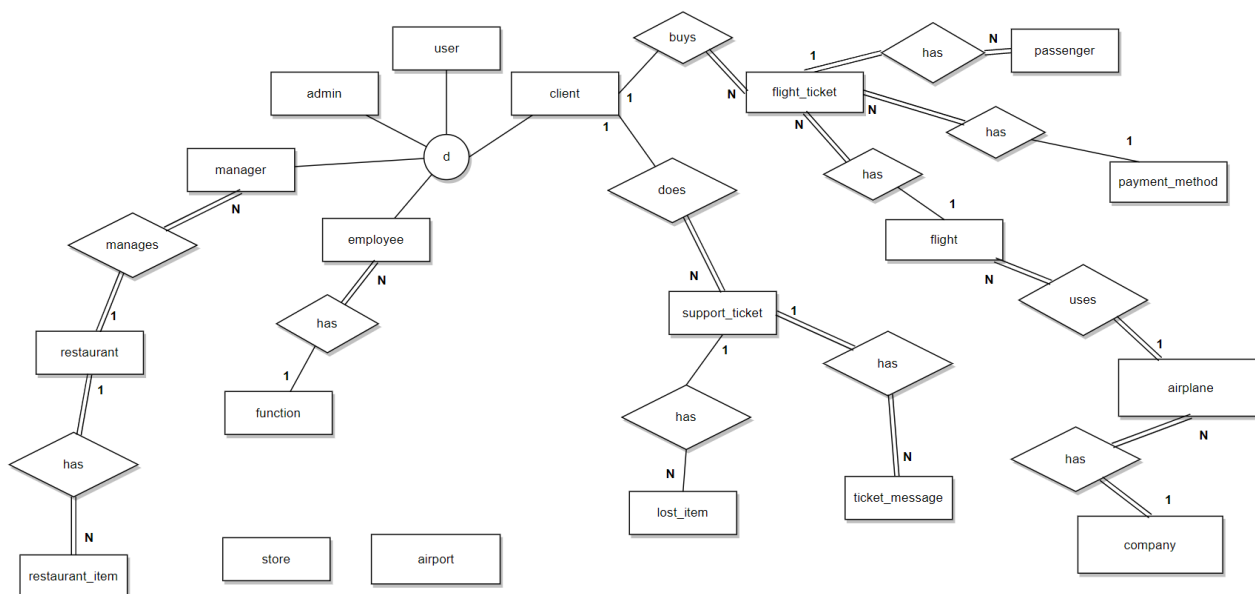


Figura 2 - Diagrama de Entidade-Relacionamento (Relações)

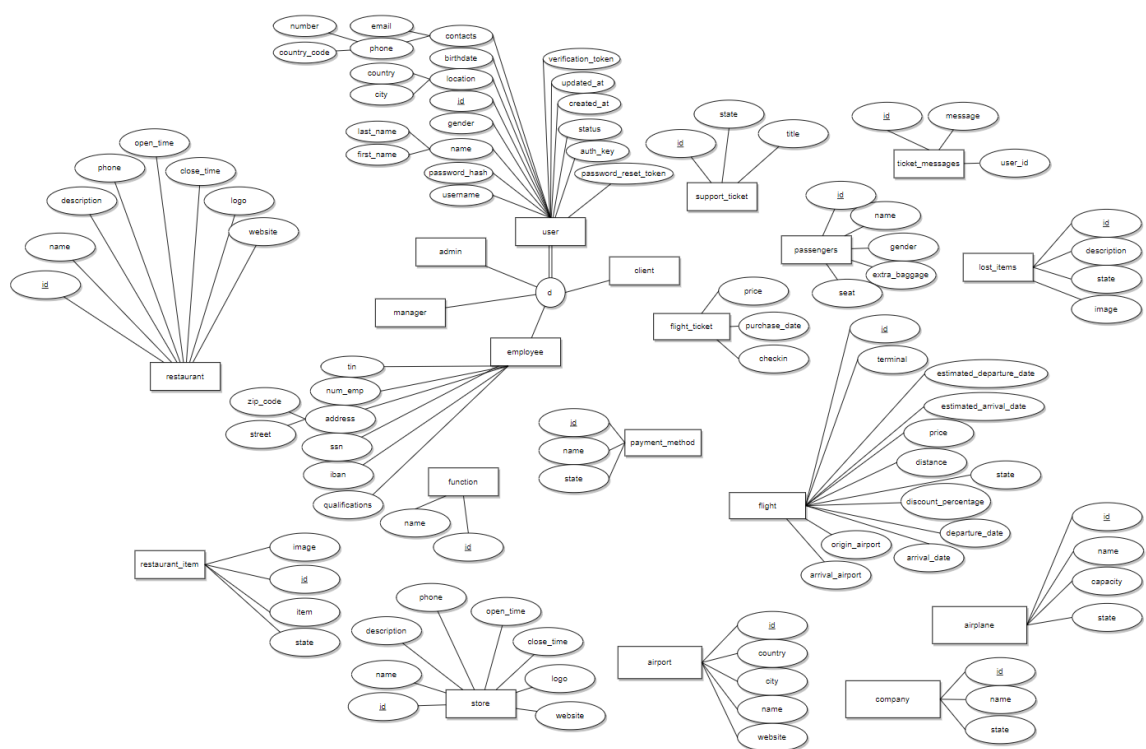


Figura 3 – Diagrama de Entidade-Relacionamento (Atributos)

# 11 DIAGRAMA DE CLASSES

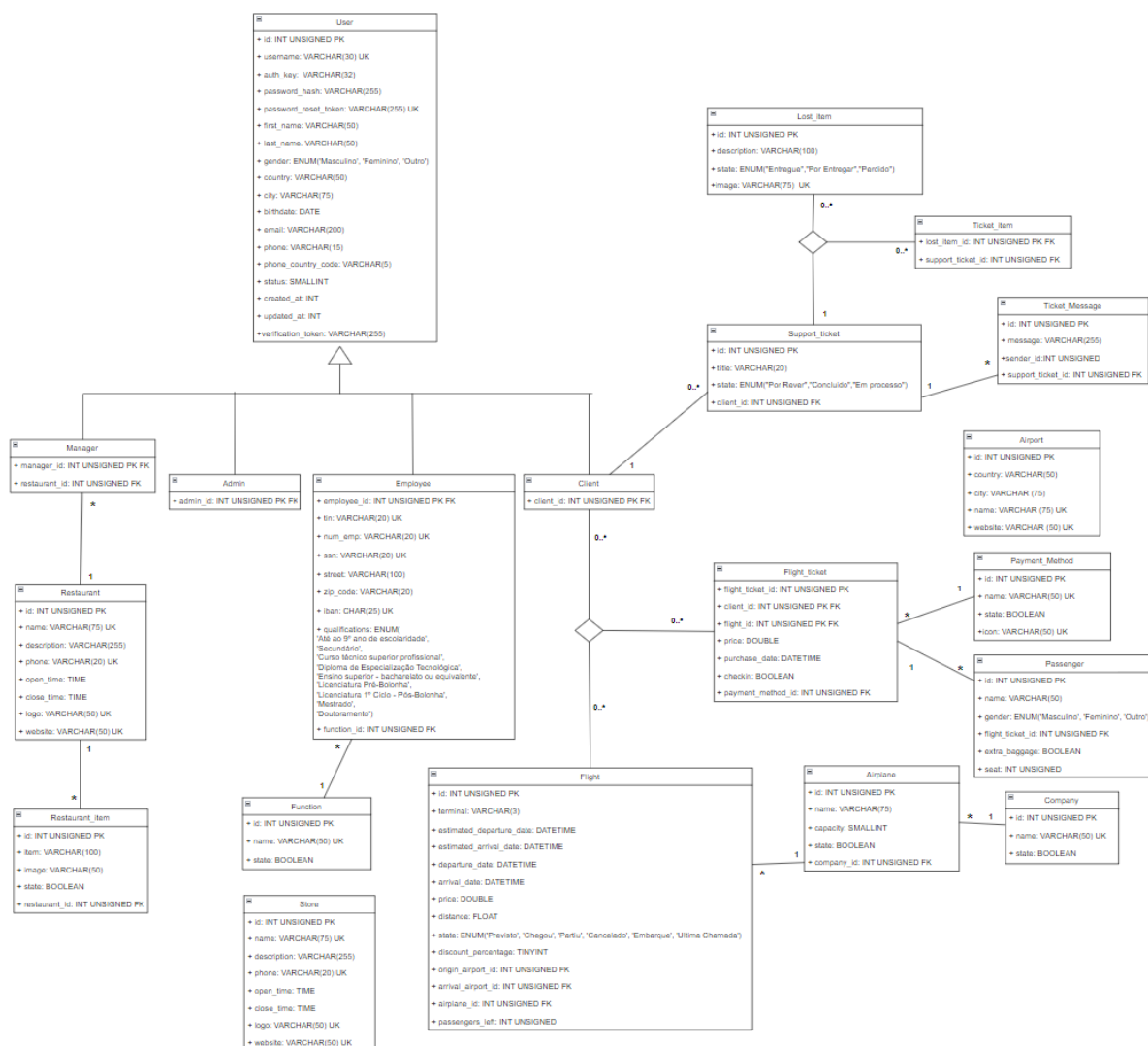


Figura 4 - Diagrama de Classes

## 12 WIREFRAMES/MOCKUPS

Todos os **wireframes** foram feitos **à mão**, enquanto, os **mockups** foram feitos utilizando o **Figma**, uma ferramenta de design. Assim, se necessitar de **ver** os **mockups** em **mais detalhe**, disponibilizamos os links, tanto para o website, como também, para a aplicação móvel.

Neste relatório **não estão presentes** os **mockups** para um dispositivo móvel na parte do website, no entanto, pelos links disponibilizados **é possível observar** tais **mockups**.

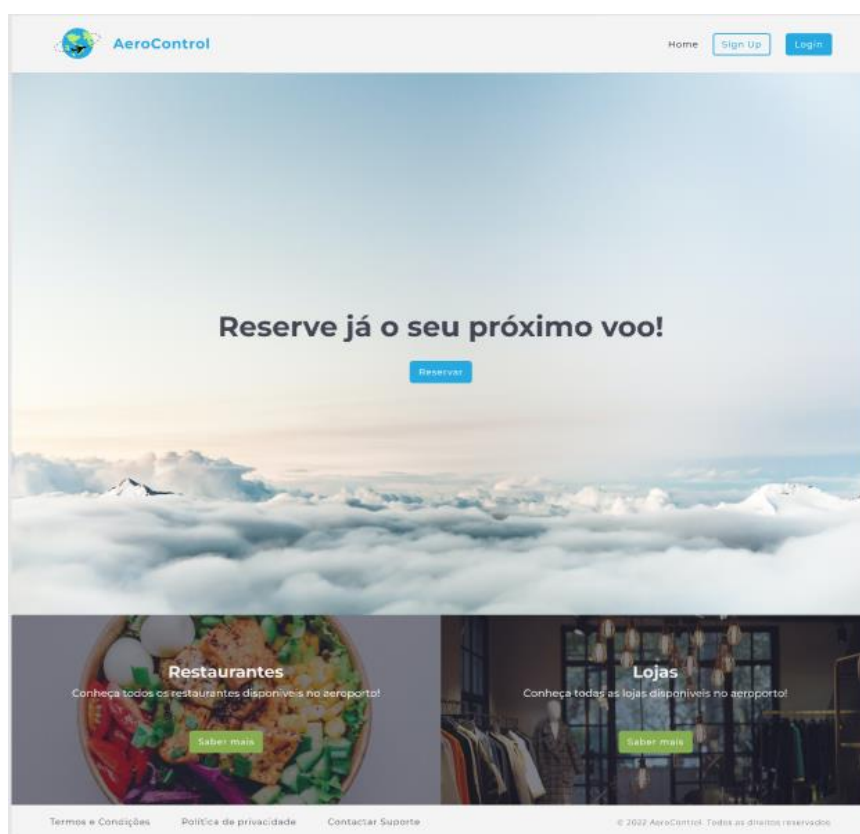
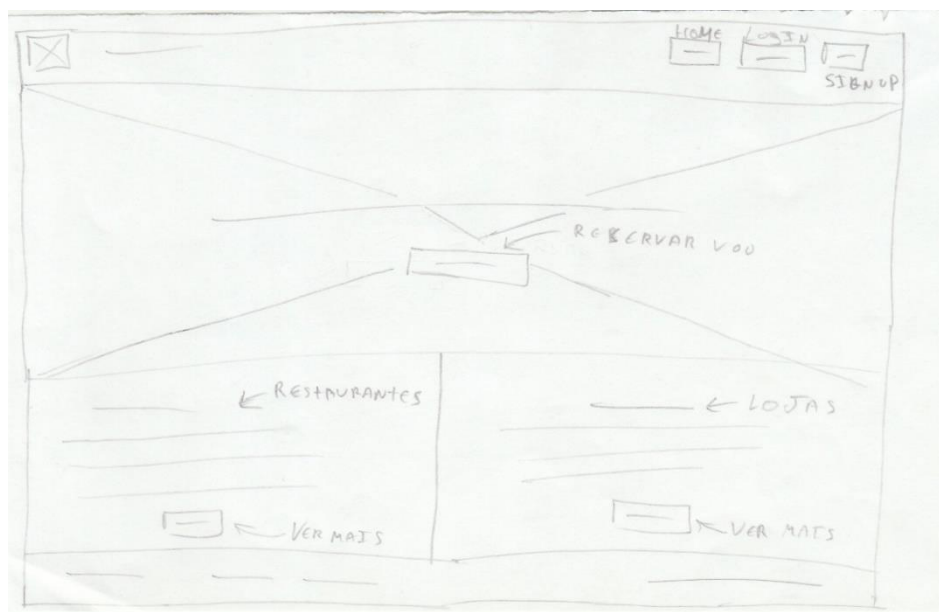
Mockups da aplicação móvel: <https://www.figma.com/file/tIJk0UlhP6F2CYhgmVQH6/Projeto-Acesso-M%C3%B3vel-a-Sistemas-de-Infoma%C3%A7%C3%A3o?node-id=0%3A1&t=z5v848c3YPyrcyG6-1>

Mockups do website: <https://www.figma.com/file/Z7MAm8hIRLVYTR1laByQ9Y/Projeto-Plataformas-Sistemas-Infoma%C3%A7%C3%A3o?node-id=1%3A54&t=WNT2F0fL5u2C1wUb-1><sup>o</sup>

Abaixo estão representados os wireframes/mockups do website e da aplicação móvel.



Tabela 32 - Wireframe/Mockup do ecrã principal - Website



Cofinanciado por:



Tabela 33 - Wireframes/Mockups do ecrã de abertura (Splash Screen) – Aplicação Móvel

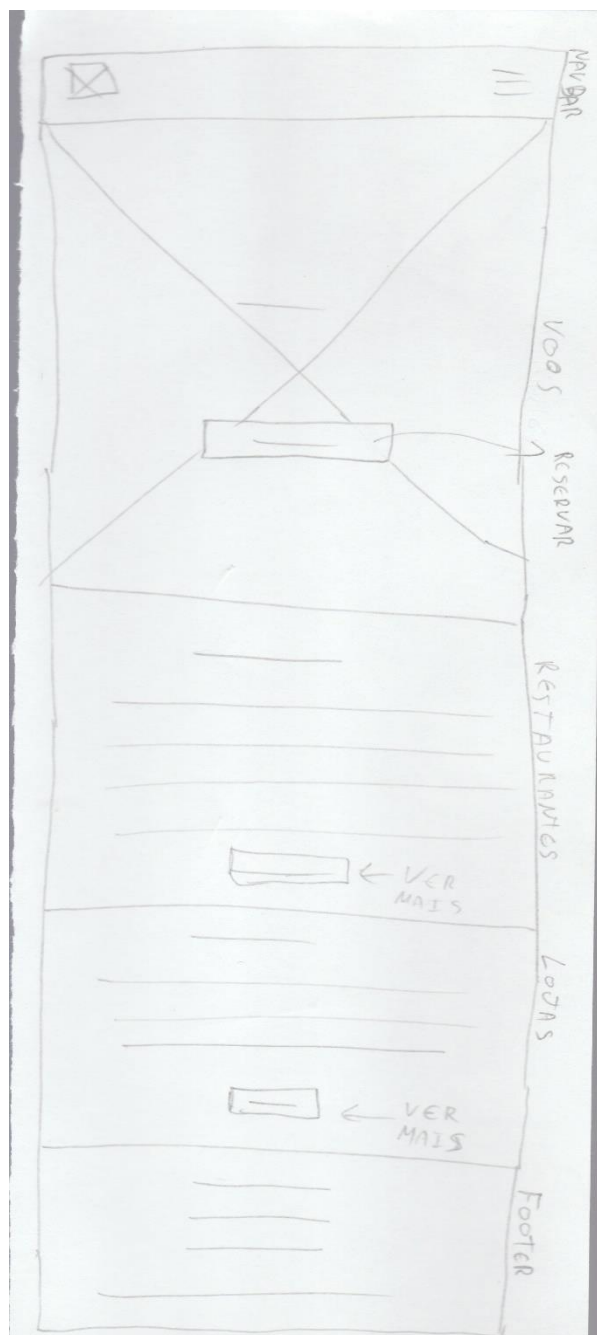


Tabela 34 - Wireframe/Mockup do ecrã de registo - Website

**DADOS DE ACESSO**

← USERNAME      ← PASSWORD

**DADOS PESSOAIS**

← PRIMEIRO NOME      ← ÚLTIMO NOME      ← GÉNERO

← DATA NASCIMENTO      ← PAÍS      ← CIDADE

**CONTACTOS**

← EMAIL      ← TELEMOVEL

← CRIAR CONTA/GUARDAR DADOS

**AeroControl** Home Sign Up Login

## Criar conta!

Crie a sua conta para começar a reservar voos e muito mais.

**Dados de acesso**

Username: something ... Password: something ...

**Dados pessoais**

Primeiro Nome: something ... Último Nome: something ... Género: something ...

Data de Nascimento: something ... País: something ... Cidade: something ...

**Contactos**

Email: something ... O email inserido é inválido

Contacto telefónico: Indicativo Número telemóvel

**Criar conta**

Termos e Condições Política de privacidade Contactar Suporte © 2022 AeroControl. Todos os direitos reservados.

Cofinanciado por:



Tabela 35 - Wireframe/Mockup do ecrã de registo - Aplicação Móvel

**CRIPAR COWTA**

**DADOS DE ACESSO**

← USERNAME

← PASSWORD

**DADOS PESSOAIS**

← PRIMEIRO NOME

← ÚLTIMO NOME

← GÉNERO

← DATA NASCIMENTO

← PAÍS

← CIDADE

**CONTACTOS**

← EMAIL

← TELEFONE

← INDICATIVO

NOVA COWTA

FOOTER

**AeroControl**

**Criar conta!**

**Dados de acesso**

Username

Password

**Dados pessoais**

Primeiro Nome

Último Nome

Género

Data Nascimento

País

Cidade

**Contactos**

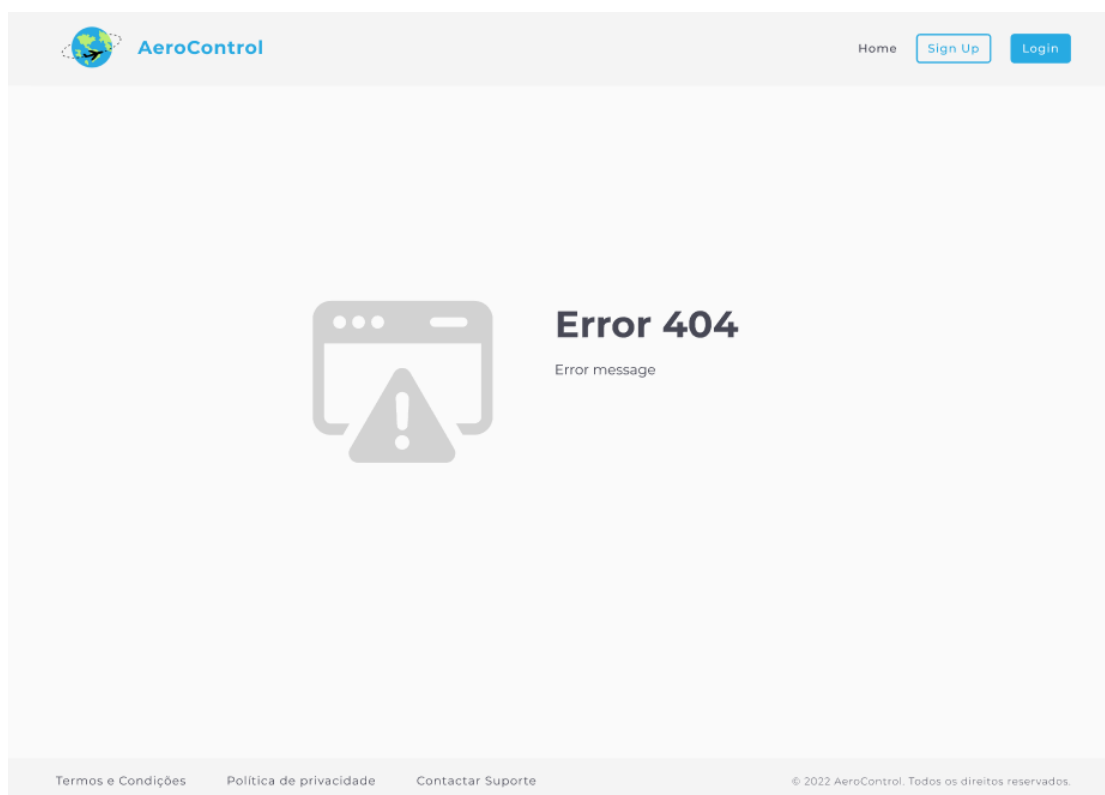
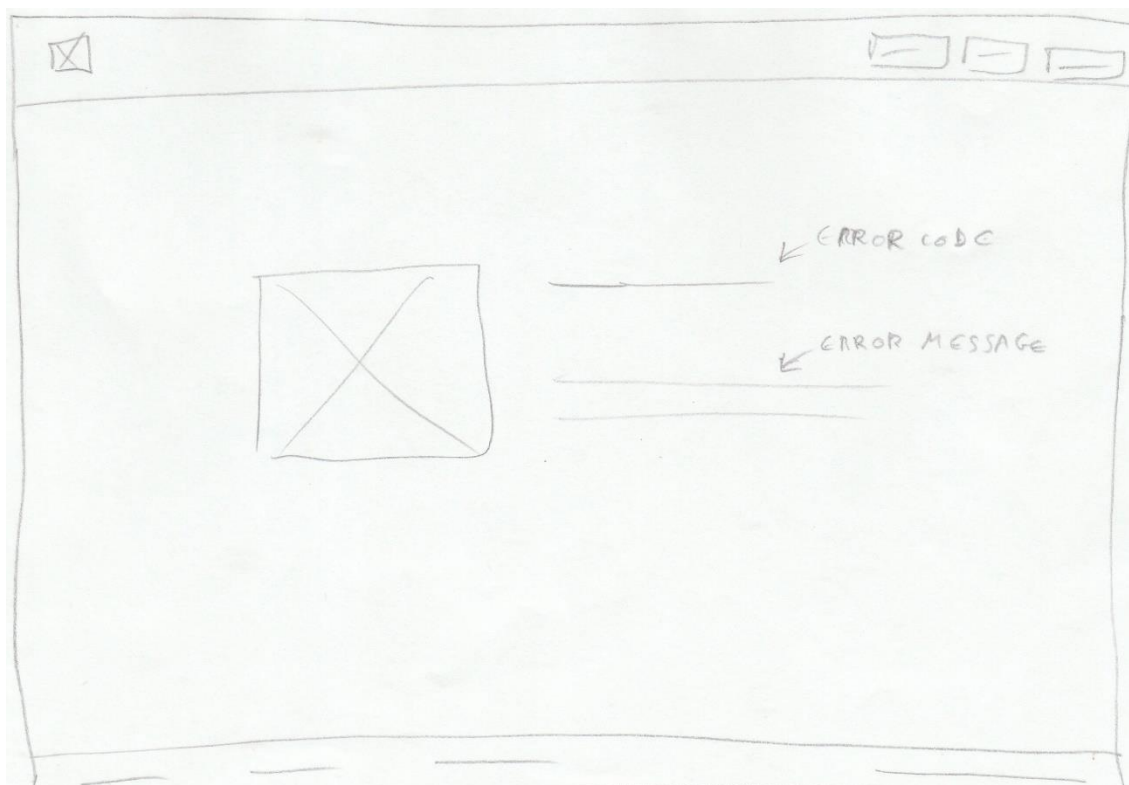
Email

Indicativo

Númrto telefone

**Criar conta**

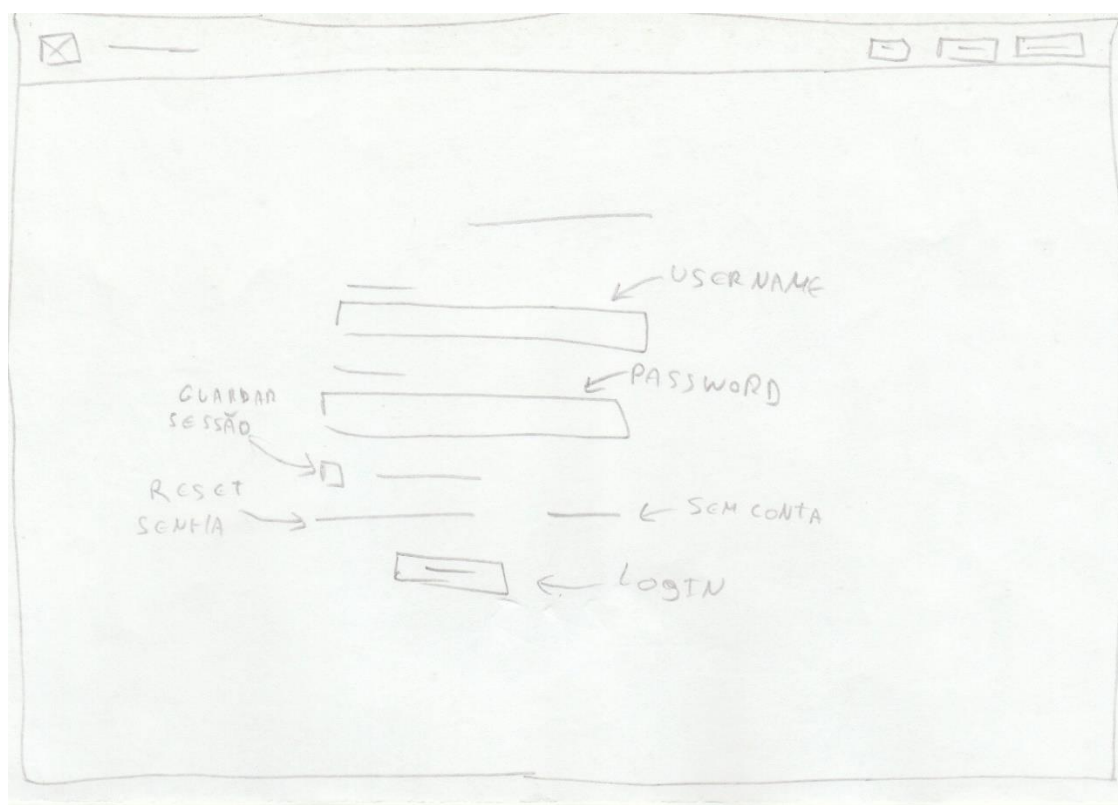
Tabela 36 - Wireframe/Mockup do ecrã da página de erro - Website



Cofinanciado por:



Tabela 37 – Wireframe/Mockup do ecrã de Login - Website



A digital mockup of the AeroControl login page. The header includes the AeroControl logo, a 'Home' link, and 'Sign Up' and 'Login' buttons. The main content area has a 'Login' heading, followed by 'Username:' and 'Password:' labels with corresponding input fields. Below the password field is a 'Guardar sessão' checkbox, a 'Esqueceu da password?' link, and a 'Não tem conta?' link. A blue 'Login' button is at the bottom. The footer contains links for 'Termos e Condições', 'Política de privacidade', and 'Contactar Suporte', along with a copyright notice for 2022 AeroControl.

Cofinanciado por:



Tabela 38 - Wireframe/Mockup do ecrã de Login - Aplicação Móvel

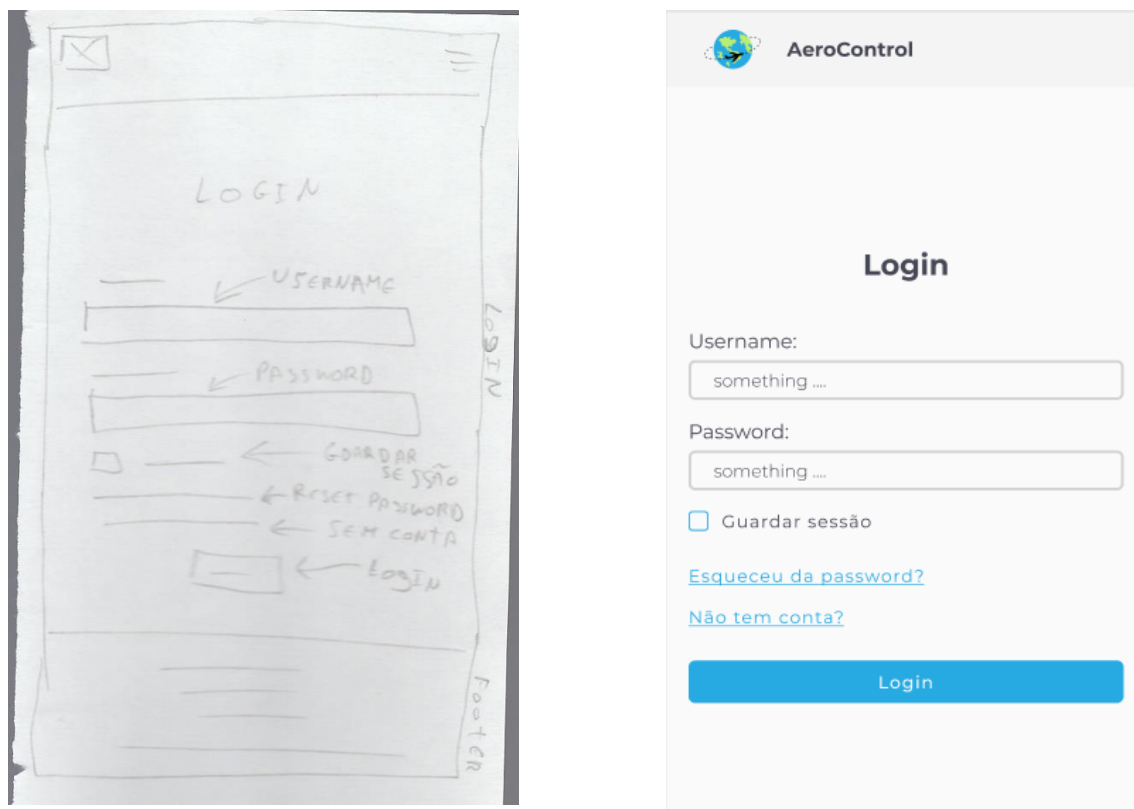


Tabela 39 - Mockup do ecrã da conta autenticado/não autenticado - Aplicação Móvel

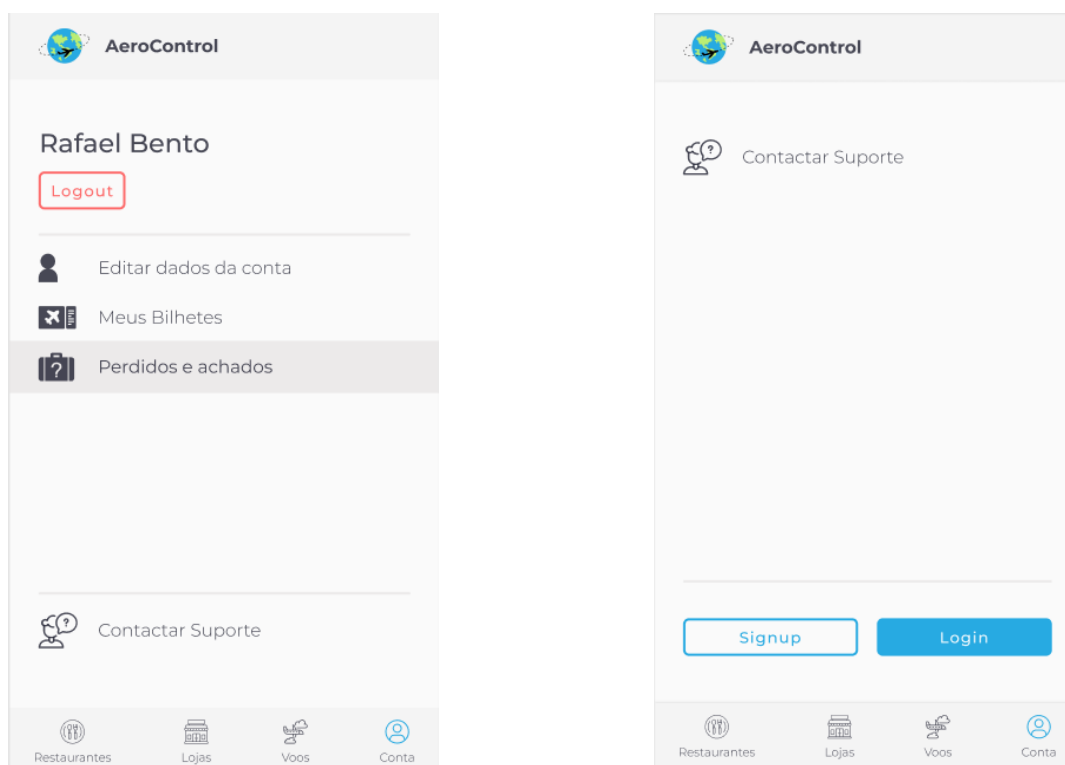
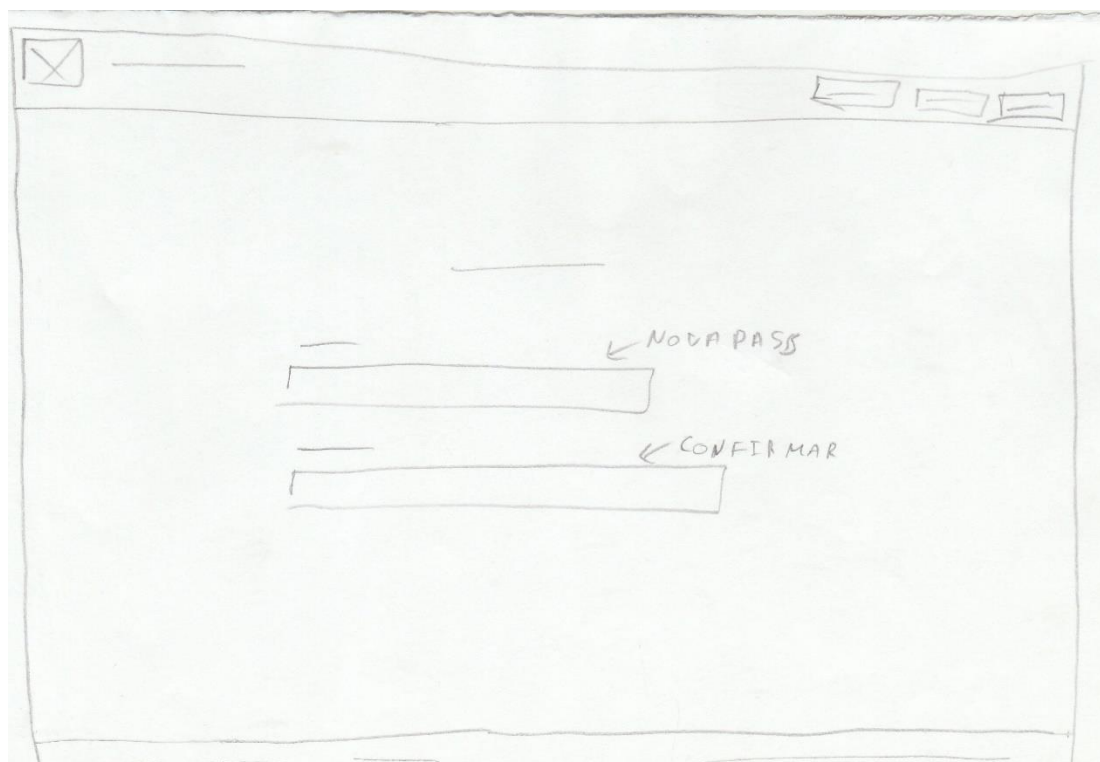




Tabela 40 - Wireframe/Mockup do ecrã de repor a palavra-passe - Website



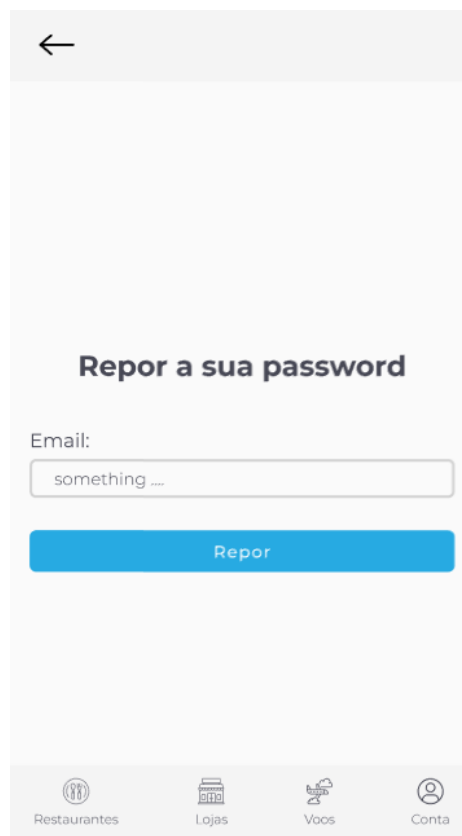
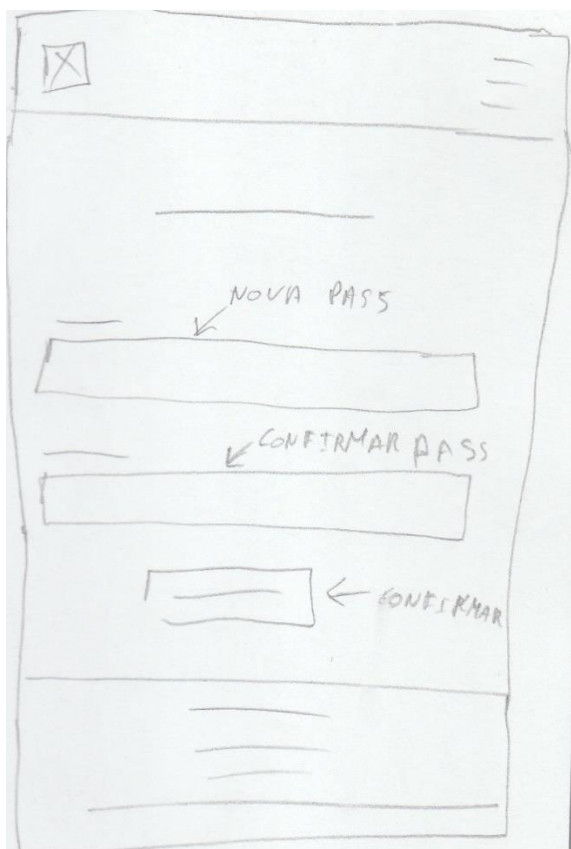
A screenshot of the AeroControl website's password reset form. The header features the AeroControl logo on the left and navigation links for 'Home', 'Sign Up', and 'Login' on the right. The main content area is titled 'Repor password' and contains two input fields: 'Nova password:' and 'Confirmar password:'. Both fields have placeholder text 'something ...'. Below the fields is a blue 'Repor' button. The footer contains links for 'Termos e Condições', 'Política de privacidade', and 'Contactar Suporte', along with a copyright notice: '© 2022 AeroControl. Todos os direitos reservados.'

Cofinanciado por:





Tabela 41 - Wireframe/Mockup do ecrã de repor a palavra-passe - Aplicação Móvel



Cofinanciado por:



Tabela 42 - Wireframe/Mockup do ecrã da pesquisa de voos - Website

The wireframe shows a flight search form with the following elements:

- Top Section:**
  - Buttons for **IDA**, **IDA/VOLTA**, and **ESCALA**.
  - Input fields for **ORIGEM** and **DESTINO**.
  - Input field for **DATA**.
  - Input field for **PASSAGEIROS**.
  - PROCURAR** button.
- Bottom Section:**
  - Buttons for **COMPANHIA** and **DESTINO/HORA**.
  - Buttons for **PREÇO** and **NEXT**.

The screenshot shows the AeroControl website with the following content:

**Header:** AeroControl logo, Home, Sign Up, Login buttons.

**Section: Reserve agora o seu voo**

**Form:**

- Buttons: **Só Ida**, **Ida e Volta**, **Sem Escalas**, **Com Escalas**.
- Origem: something...
- Destino: something...
- Data de Ida: something...
- Data de Volta: something...
- Nº Passageiros: something...
- Pesquisar** button.

**Section: 100 Resultados encontrados!**

TAP	21.00 → 13.00 Lisboa → Porto	Lugares disponíveis: 3 <b>13€</b> <b>Reservar</b>
TAP	21.00 → 13.00 Lisboa → Porto 21.00 → 13.00 Porto → Pequim	Lugares disponíveis: 3 <b>13€</b> <b>Reservar</b>

**Footer:** Termos e Condições, Política de privacidade, Contactar Suporte, © 2022 AeroControl. Todos os direitos reservados.

Cofinanciado por:



UNIÃO EUROPEIA  
Fundo Social Europeu

Tabela 43 - Wireframe/Mockups dos ecrãs da pesquisa de voos - Aplicação Móvel

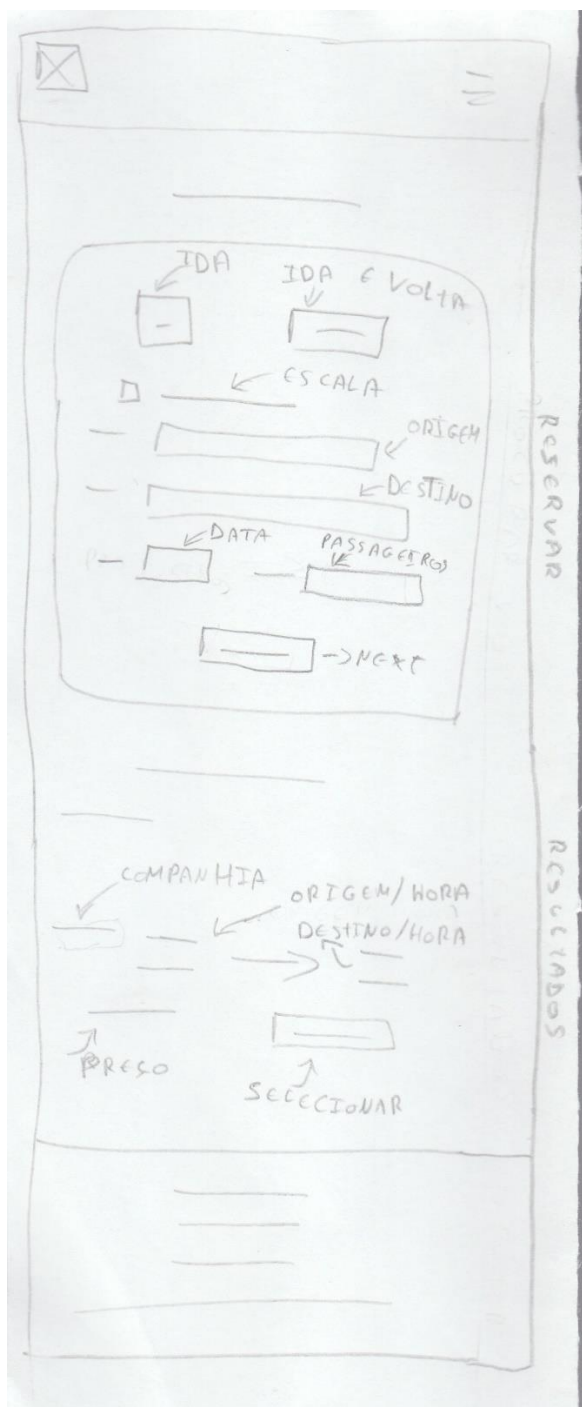


Tabela 44 - Wireframe/mockup do ecrã da compra de um bilhete de voo - Website

The wireframe shows a flight booking form with the following sections:

- Passenger Information:**
  - NAME (two input fields)
  - GÊNERO (Gender dropdown)
  - BAGAGEM (Baggage dropdown)
- Método de Pagamento (Payment Method):**
  - Tipo de Cartão (Card Type dropdown)
  - Nº do Cartão (Card Number input field)
  - NOME (Name input field)
  - APELIDO (Surname input field)
  - MES (Month dropdown)
  - ANO (Year dropdown)
- Confirmation:**
  - TERMINOS (Terms checkbox)
  - CONFIRMAR (Confirm button)

The screenshot shows the AeroControl website with the following elements:

- Header:**
  - AeroControl logo
  - Navigation links: Home, Voos, Restaurantes, Lojas
  - User profile: Pedro Norberto
  - Links: Ver conta, Logout
- Informações dos Passageiros (Passenger Information):**

	Nome	Género	Bagagem extra
Passageiro 1	<input type="text" value="something ..."/>	<input type="text" value="something ..."/>	<input checked="" type="checkbox"/>
Passageiro 2	<input type="text" value="something ..."/>	<input type="text" value="something ..."/>	<input checked="" type="checkbox"/>
Passageiro 3	<input type="text" value="something ..."/>	<input type="text" value="something ..."/>	<input checked="" type="checkbox"/>
- Pagamento (Payment):**
  - Payment methods: Cartão de crédito, Cartão de débito, MB Way, Multibanco, PayPal
  - Form fields:
    - Tipo de Cartão:
    - Nome:
    - Apelido:
    - Número do Cartão:
    - Código de Segurança:
    - Data de Validade (Mês/Ano):
  - Li e aceito integralmente os termos e condições. ☐
  - Confirmar button
- Footer:**
  - Links: Termos e Condições, Política de privacidade, Contactar Suporte
  - Copyright: © 2022 AeroControl. Todos os direitos reservados.

Cofinanciado por:



Tabela 45 - Wireframe/Mockups dos ecrãs da compra de um bilhete de voo - Aplicação Móvel

Hand-drawn wireframe of a mobile app screen for flight booking. The screen is divided into two main sections for passengers, labeled "PASSAGEIRO 1" and "PASSAGEIRO 2". Each passenger section contains fields for "NOME" (Name), "GÉNERO" (Gender), and "BAGAGEM EXTRA" (Extra Baggage). Below the passenger sections is a "Método de Pagamento" (Payment Method) section. This section includes fields for "TIPO DE CARTÃO" (Card Type), "NOME" (Name), "APELLIDO" (Surname), "Nº CARTÃO" (Card Number), "CÓDIGO SEGURANÇA" (Security Code), "MÊS" (Month), "ANO" (Year), and "VALIDADE" (Validity). There are also checkboxes for "TERMO" (Terms) and a "CONFIRMAR" (Confirm) button.

Digital mockup of the "Insira os dados dos passageiros" (Enter passenger data) screen. The screen has a back arrow at the top left. The title is "Insira os dados dos passageiros". Below the title, there are two sections for passengers: "Passageiro 1" and "Passageiro 2". Each section contains a "Nome:" label followed by a text input field, a "Género" label followed by a text input field, and a "Bagagem Extra" label followed by a checkbox. At the bottom of the screen is a blue button labeled "Avançar" (Advance).

Digital mockup of the "Escolha o metodo de Pagamento" (Choose payment method) screen. The screen has a back arrow at the top left. The title is "Escolha o metodo de Pagamento". Below the title, there are four payment method options: "Cartão de crédito" (Credit Card), "Cartão de débito" (Debit Card), "MbWay", and "Multibanco". Each option has a corresponding icon and logo. At the bottom of the screen is a blue button labeled "Confirmar" (Confirm).

Tabela 46 - Wireframe/Mockup do ecrã de editar os dados da conta - Website

**DADOS DE ACESSO**

← USERNAME      ← PASSWORD

\_\_\_\_\_  
\_\_\_\_\_

**DADOS PESSOAIS**

← PRIMEIRO NOME      ← ÚLTIMO NOME      ← GÉNERO

\_\_\_\_\_  
\_\_\_\_\_

← DATA NASCIMENTO      ← PAÍS      ← CIDADE

\_\_\_\_\_  
\_\_\_\_\_

**CONTACTOS**

← EMAIL      ← TELEMOVEL

\_\_\_\_\_  
\_\_\_\_\_

\_\_\_\_\_  
\_\_\_\_\_

← CRIAR CONTA/GUARDAR DADOS

**AeroControl** Home Voos Restaurantes Lojas

Pedro Norberto

Ver conta

Logout

## Editar dados da conta

**Dados de acesso**

Username: something ... Password: something ...

**Dados pessoais**

Primeiro Nome: something ... Último Nome: something ... Género: something ...

Data de Nascimento: something ... País: something ... Cidade: something ...

**Contactos**

Email: something ... O email inserido é inválido

Contacto telefónico: Indicativo Número telemóvel

**Guardar**

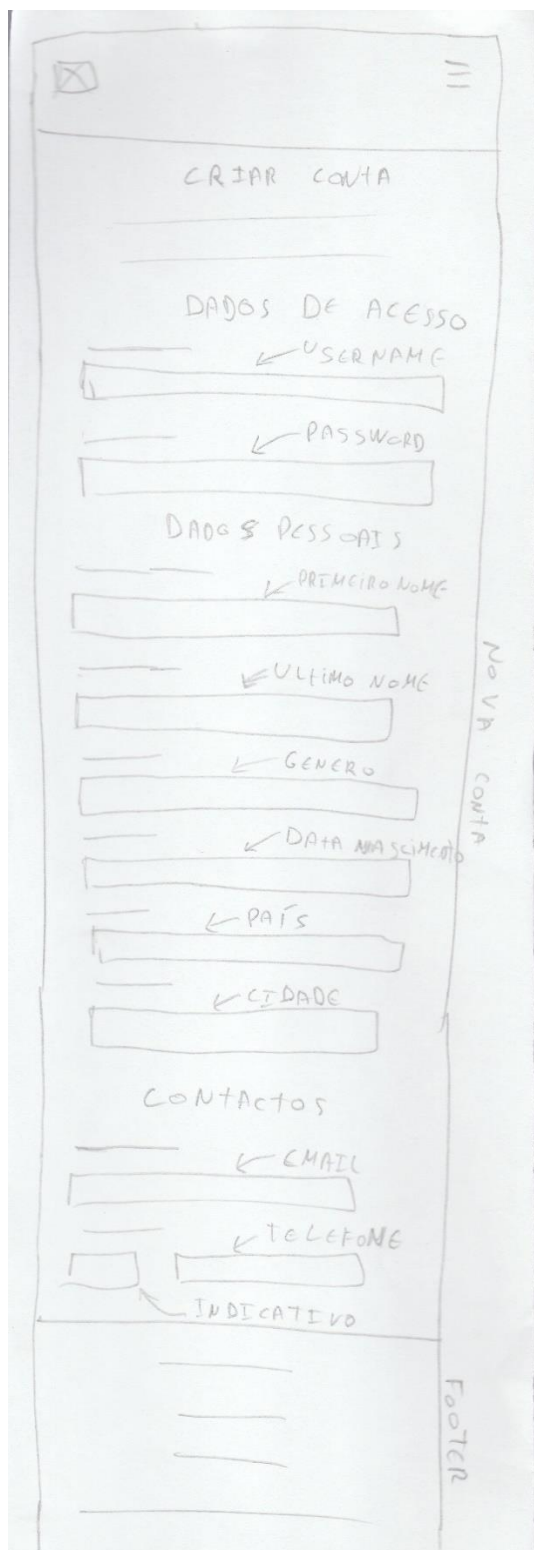
Termos e Condições Política de privacidade Contactar Suporte

© 2022 AeroControl. Todos os direitos reservados.

Cofinanciado por:



Tabela 47 - Wireframe/Mockups dos ecrãs de atualizar os dados da conta - Aplicação Móvel




**AeroControl**

**Editar dados da Conta**

[Dados de Acesso](#)
[Dados Pessoais](#)
[Contactos](#)

Fragment

Guardar

Username:

Password:

Primeiro Nome:

Último Nome:

Género:

Data de Nascimento:

País:

Cidade:

Email:

Contacto telefónico:



Tabela 48 - Wireframe/Mockup do ecrã da lista de bilhetes - Website

The wireframe illustrates a flight ticket management interface. It features a table with the following columns and labels:

- DATA** (with a sub-label **DATA PARTIDA**)
- ORIGEM** (with a sub-label **DATA CHEGADA**)
- DESTINO**
- ESTADO**
- CHECK IN** (with a sub-label **PRICE**)
- DISTANCIA**
- DATA COMPRA**
- NAME**
- GENDER**
- LUGAR**
- BAGAGEM** (with a sub-label **TERMINAL**)
- ALTERAR**
- CANCELAR**

The table contains two rows of placeholder data, each with arrows indicating the flow of information between fields.

The screenshot shows the AeroControl website's 'Meus Bilhetes' (My Tickets) section. The header includes the AeroControl logo, navigation links (Home, Voos, Restaurantes, Lojas), and a user profile for Pedro Norberto. The main content area displays two flight tickets:

- Ticket 1:**
  - Data: 07/11/2022
  - Origem: Lisboa
  - Destino: Madrid
  - Estado: Chegou
  - Hora Partida: 11:15 h
  - Hora Chegada: 13:15 h
  - Distância: 600.00 Km
  - Data de Compra: 04/10/2022
  - Passageiros: Marcio Soares (Masculino, S4), Pedro Leandro (Masculino, S5)
  - TAP
  - Terminal: T1
  - Price: 300€ 265€
  - Buttons: Alterar, Cancelar, Guardar, Check In
- Ticket 2:**
  - Data: 13/11/2022
  - Origem: Paris
  - Destino: Lisboa
  - Estado: Chegou
  - Buttons: Ver mais detalhes

The footer contains links for Terms and Conditions, Privacy Policy, Contact Support, and a copyright notice for AeroControl 2022.

Cofinanciado por:





Tabela 49 - Wireframes/Mockup do ecrã da lista de bilhetes - Aplicação Móvel



Tabela 50 - Wireframe/Mockup da informação do bilhete comprado - Aplicação Móvel

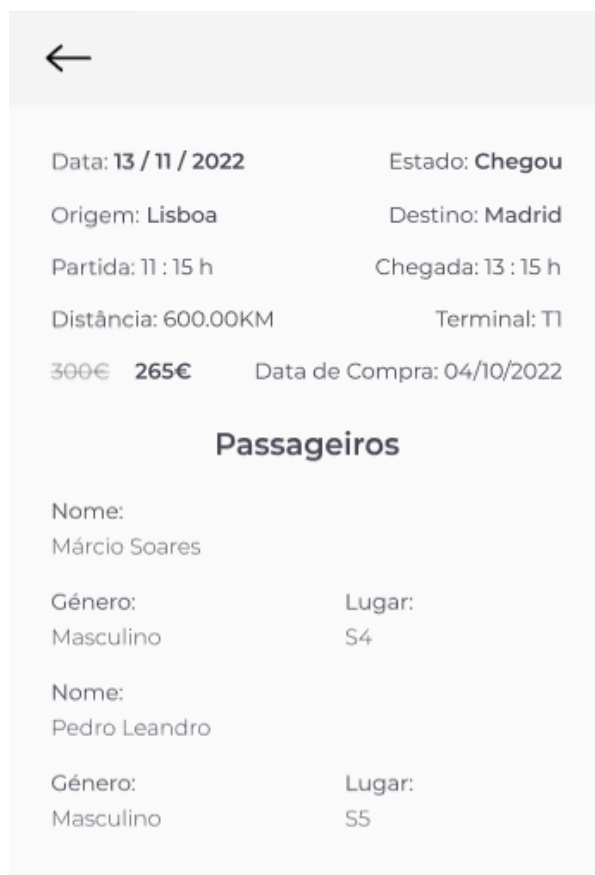
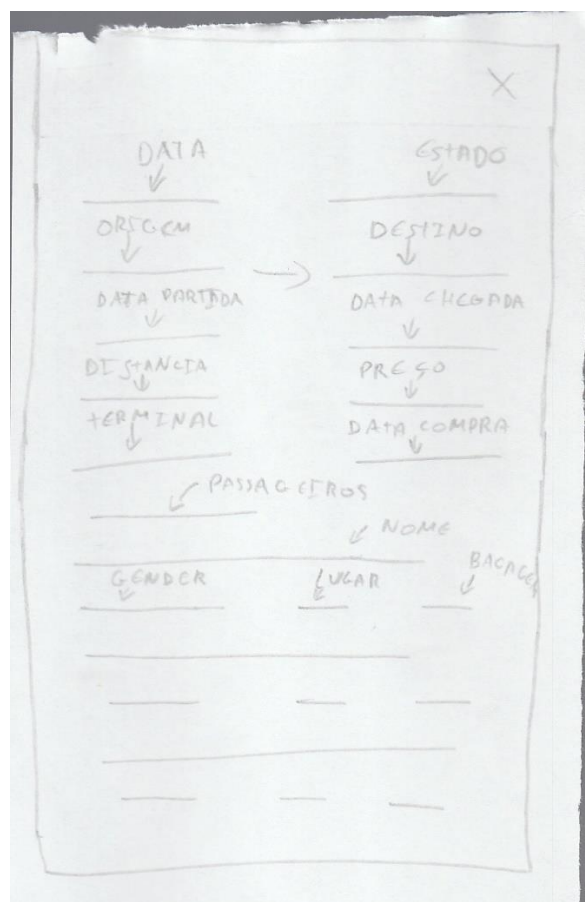
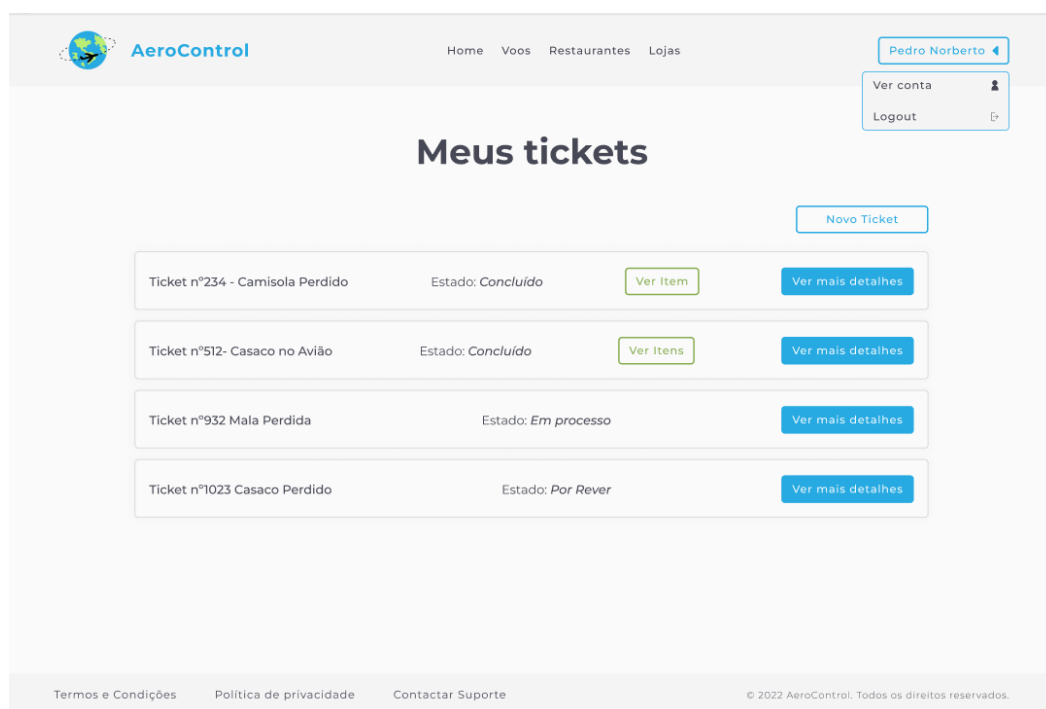
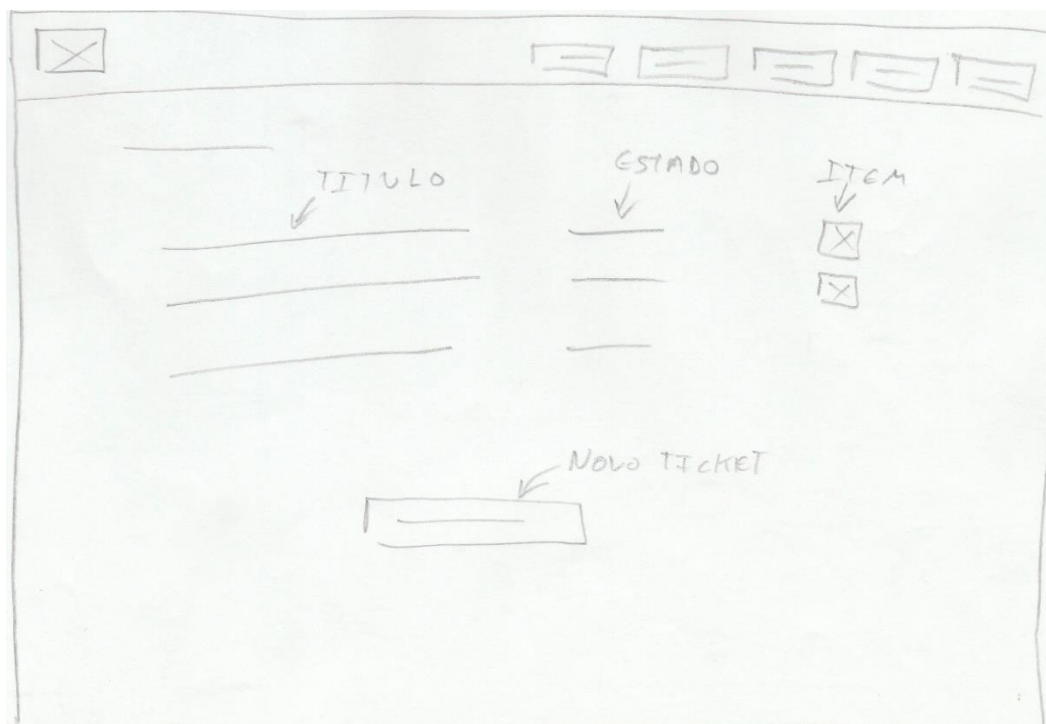


Tabela 51 - Wireframe/Mockup do ecrã da lista de tickets de suporte - Website



Cofinanciado por:



Tabela 52 - Wireframe/Mockup do ecrã da lista de tickets de suporte - Aplicação Móvel

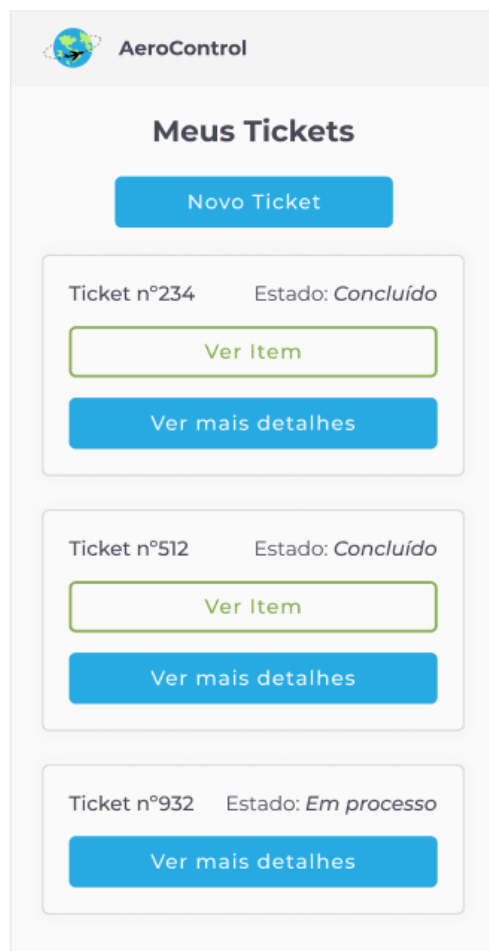
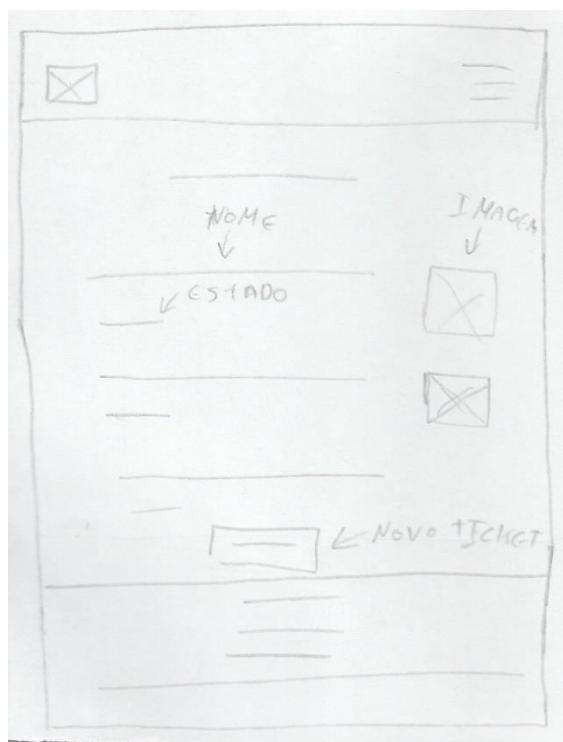
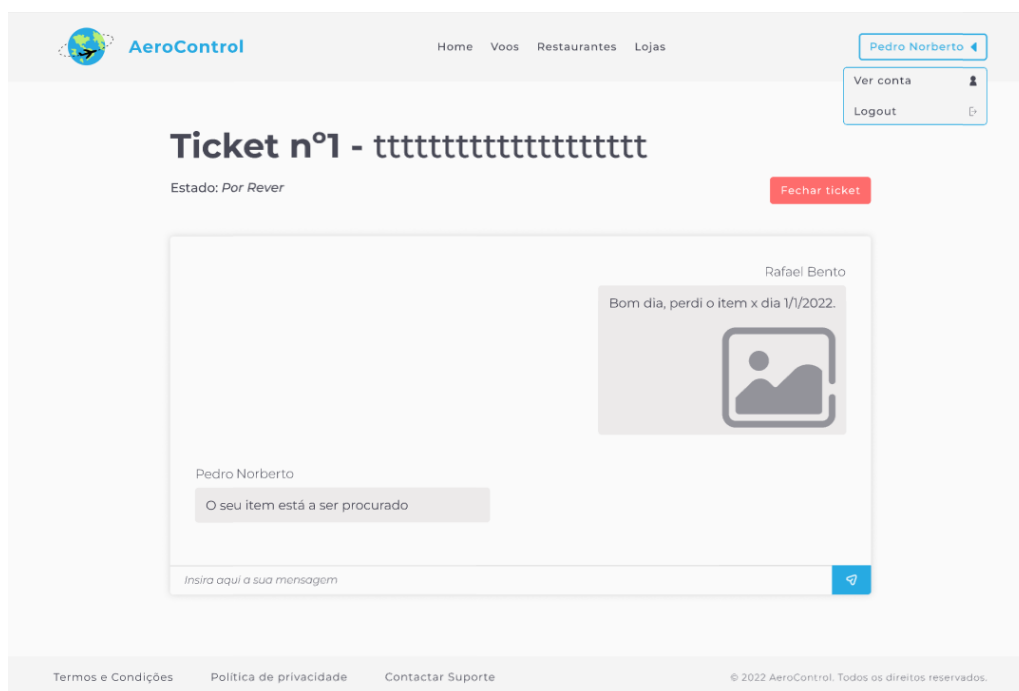
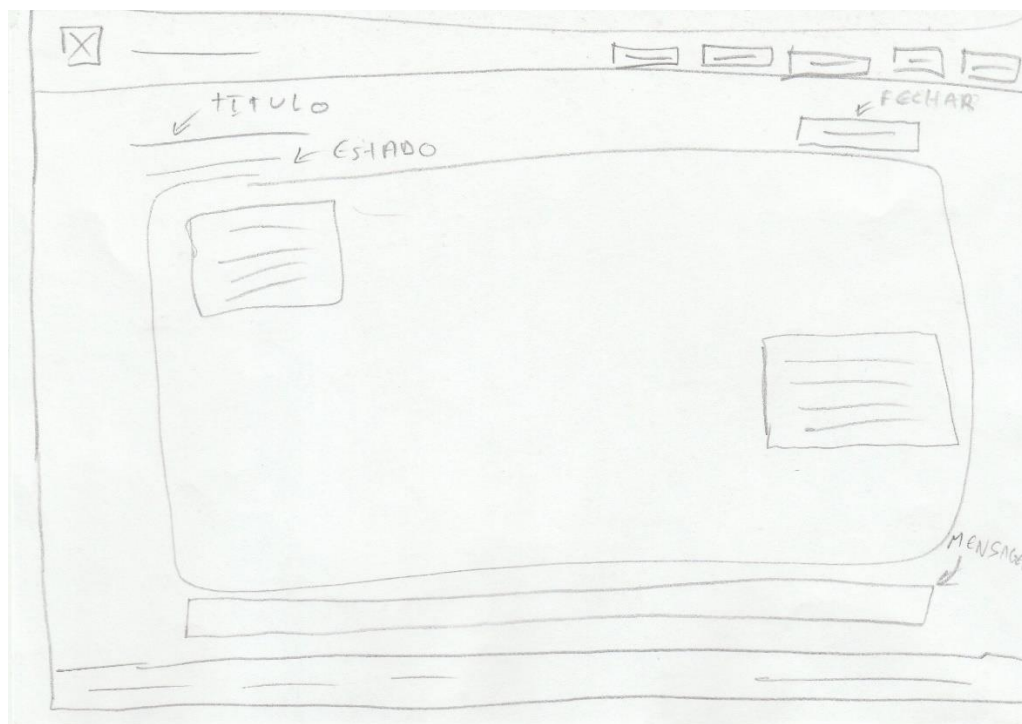


Tabela 53 - Wireframe/Mockup do ecrã do ticket de suporte - Website



Cofinanciado por:



Tabela 54 - Wireframe/Mockup do ecrã do ticket de suporte - Aplicação Móvel

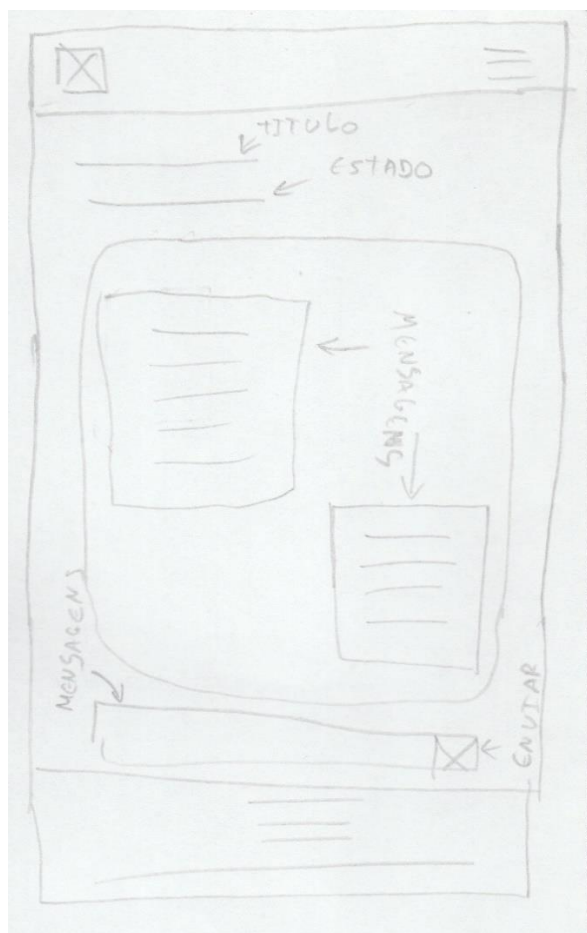
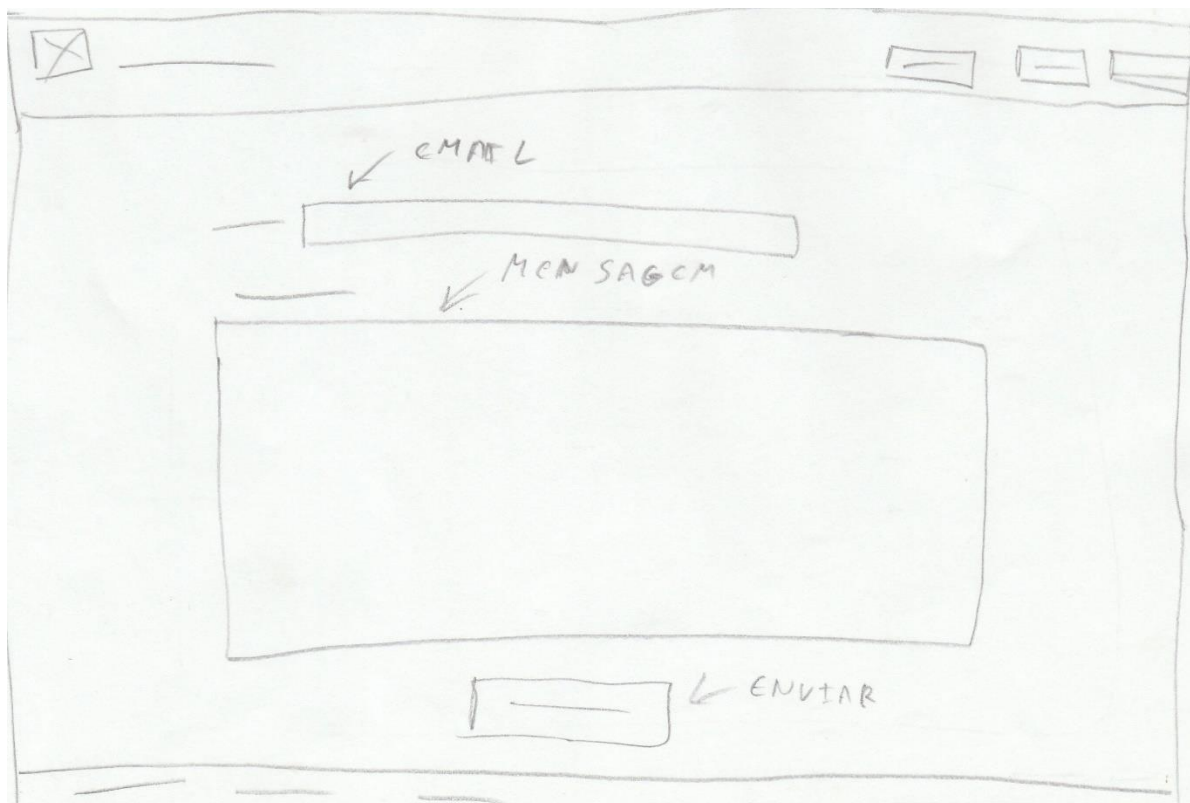



Tabela 55 - Wireframe/Mockup do ecrã de suporte através de email - Website




**AeroControl**

[Home](#)
[Sign Up](#)
[Login](#)

## Contactar Suporte

Email:

Mensagem:

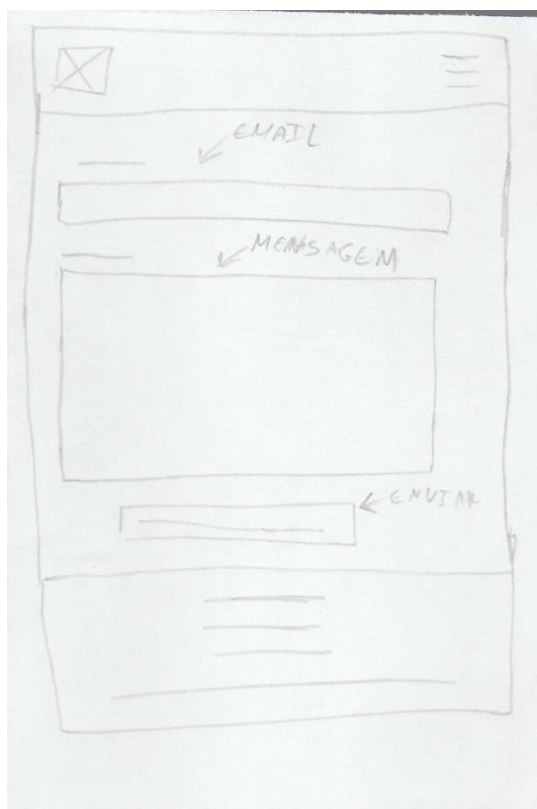
Bom dia, preciso de ajuda com a minha conta de utilizador!


Enviar

[Termos e Condições](#)
[Política de privacidade](#)
[Contactar Suporte](#)

© 2022 AeroControl. Todos os direitos reservados.

Tabela 56 - Wireframe/Mockup do ecrã de suporte através de email - Aplicação Móvel




**AeroControl**

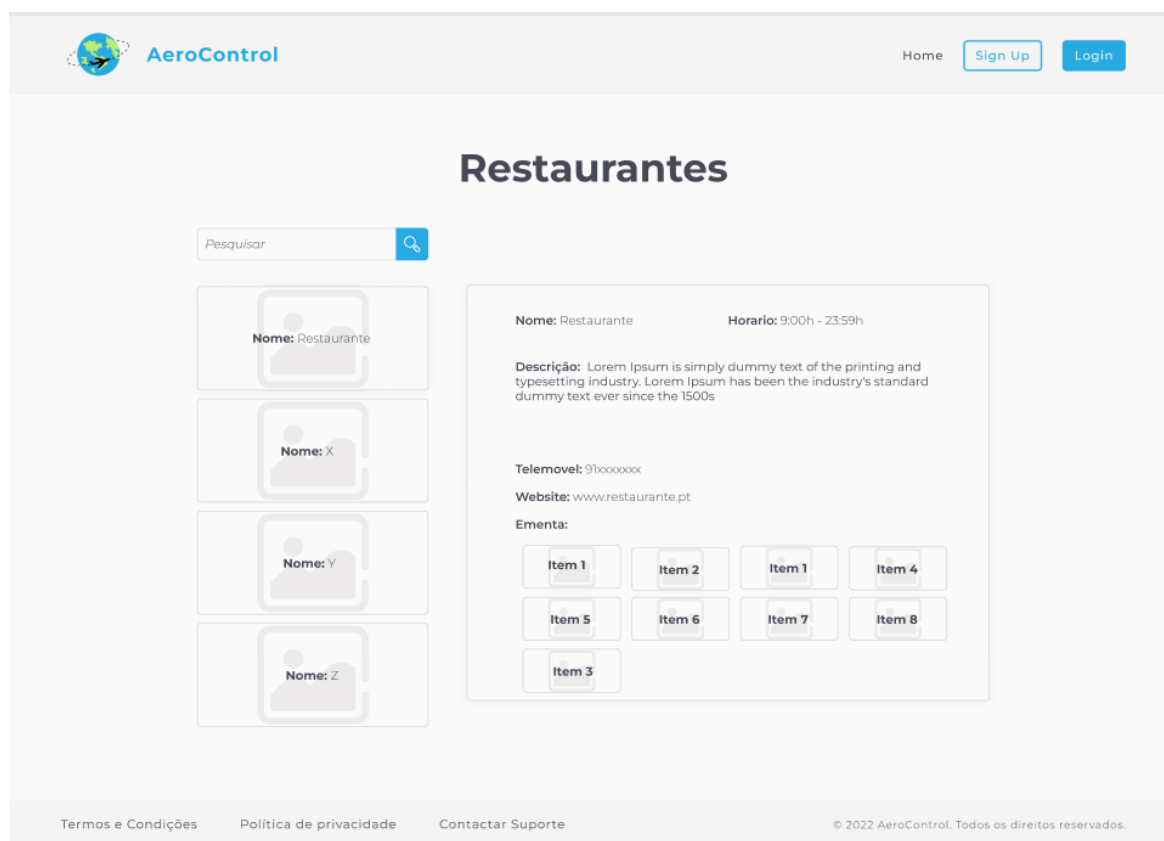
## Contactar Suporte

Email:

Mensagem:

Enviar

Tabela 57 - Wireframe/Mockup do ecrã da lista e informação dos restaurantes - Website



Cofinanciado por:



UNIÃO EUROPEIA  
Fundo Social Europeu



Tabela 58 - Wireframes/Mockups do ecrã da lista e da Informação dos restaurantes - Aplicação Móvel

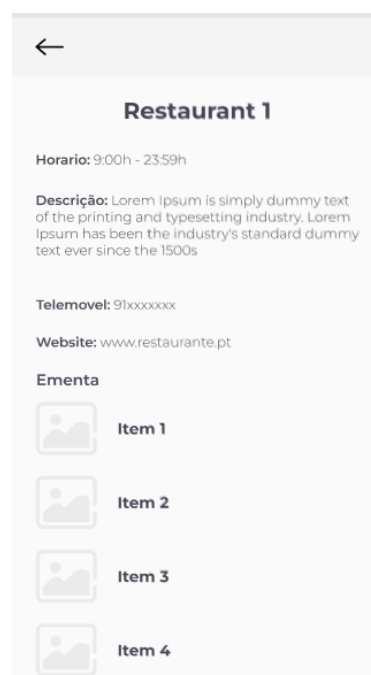
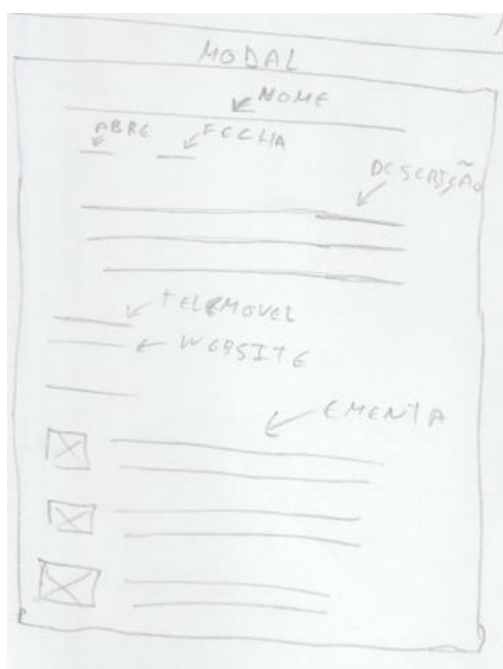
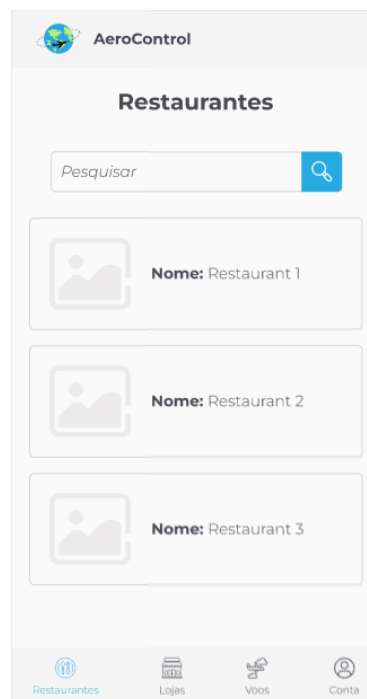
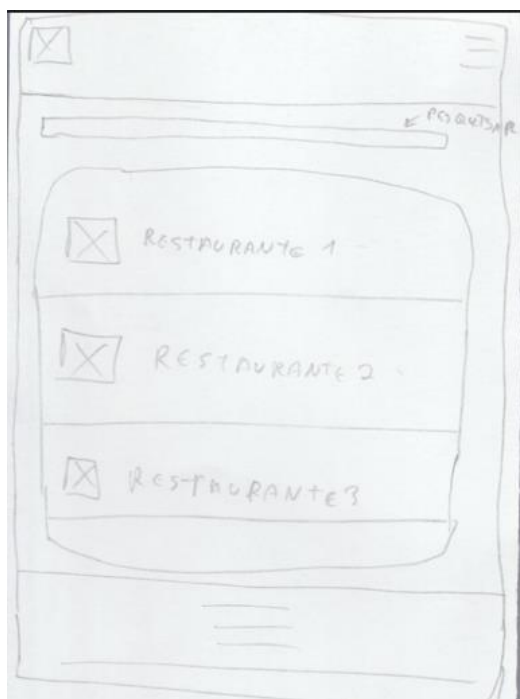
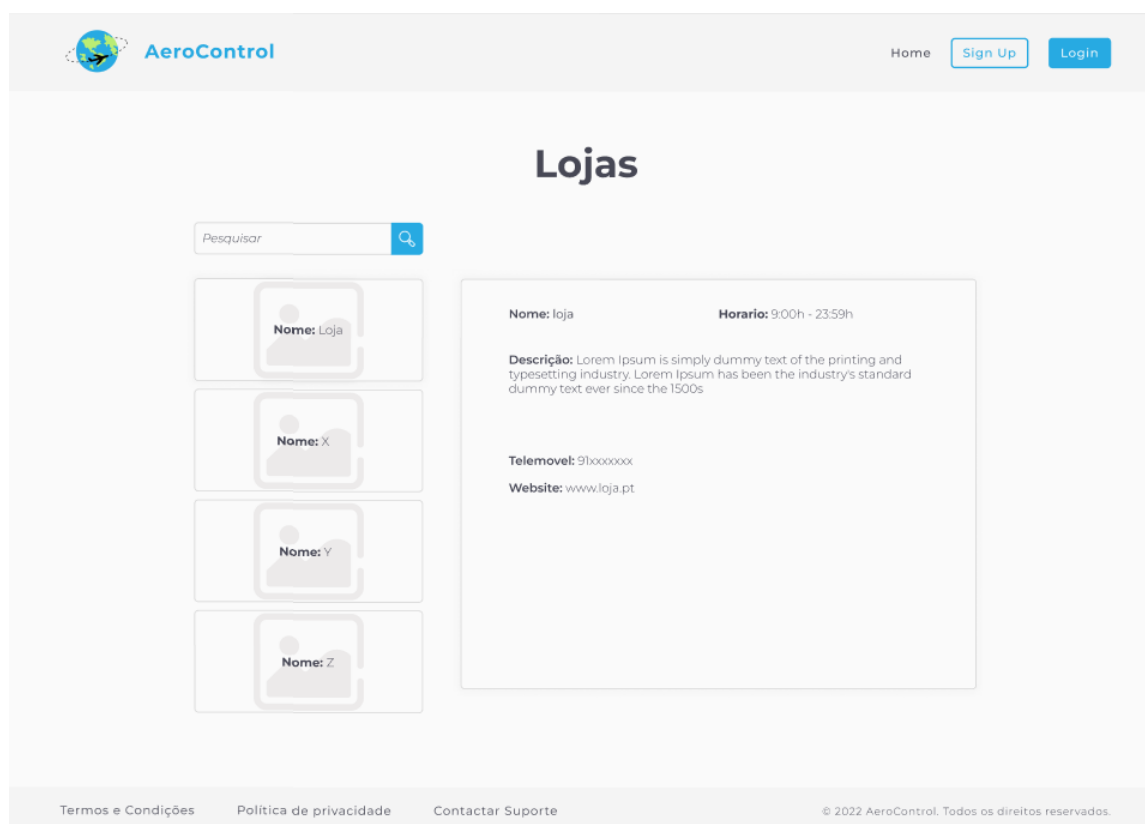
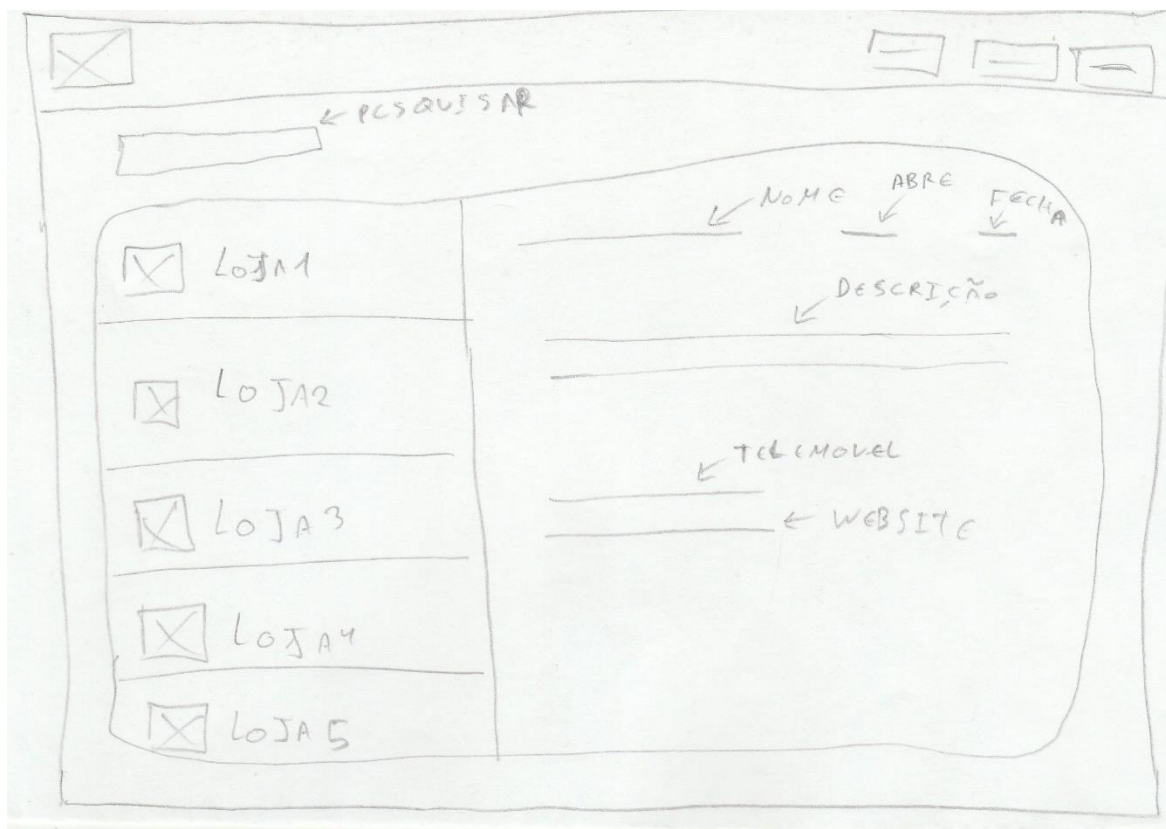


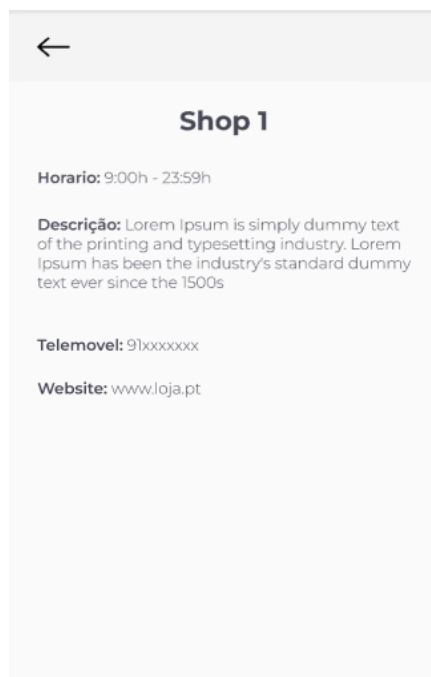
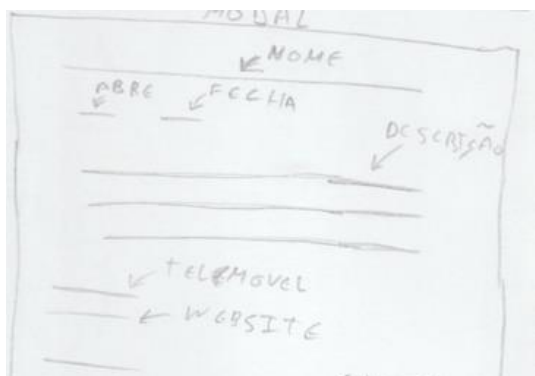
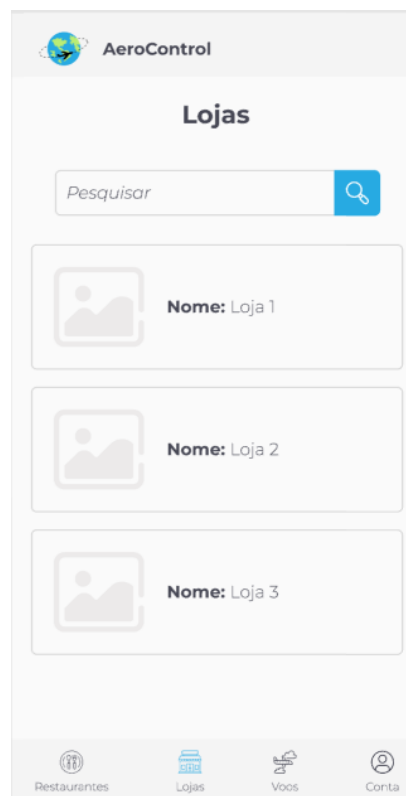
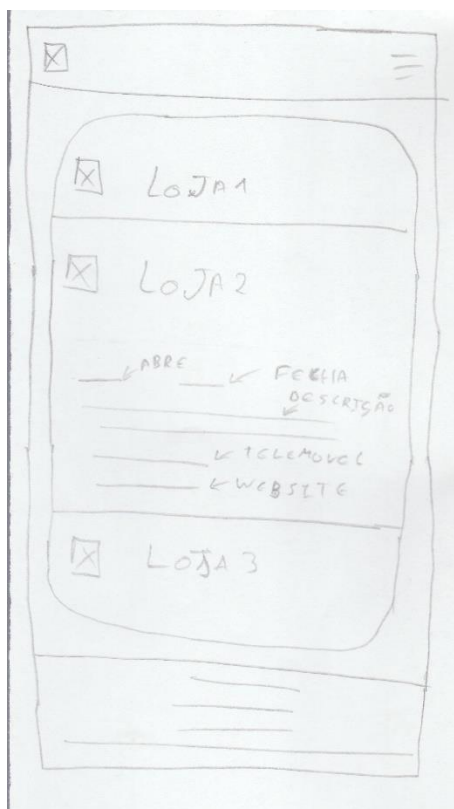
Tabela 59 - Wireframe/Mockup do ecrã da lista e informação das lojas - Website



Cofinanciado por:



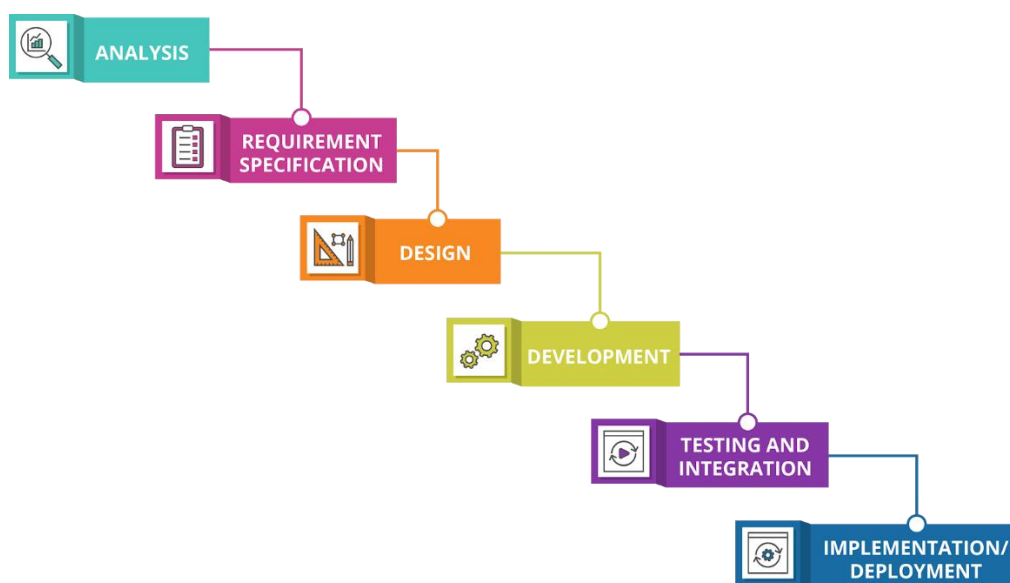
Tabela 60 - Wireframes/Mockups dos ecrãs da lista e das informações das lojas – Aplicação Móvel



## 13 IMPLEMENTAÇÃO

A nossa implementação do projeto foi baseada na Framework SDLC (Software Development Life Cycle), em que consiste em várias fases, estas sendo:

1. Planeamento, onde se planeia o projeto consoante a análise de requisitos;
2. Definição dos requisitos, onde se define e documenta claramente os requisitos num SRS (documento Software Requirement Specification);
3. Desenho da arquitetura, onde é desenvolvido um desenho da arquitetura com referência ao SRS (documento Software Requirement Specification), com a possibilidade de ser proposto mais do que uma abordagem de desenho, este desenho será documentado no DDS (Design Document Specification);
4. Desenvolvimento, onde se começa o desenvolvimento do projeto de acordo com o DDS (Design Document Specification);
5. Testes, onde os defeitos do projeto são reportados, rastreados, corrigidos e retestados;
6. Deployment, onde o projeto está pronto a ser lançado.



*Figura 5 - Framework SDLC (Software Development Life Cycle)*

## 14 FASE DE TESTES

Na fase de testes a ferramenta utilizada na UC de PLSI (Plataformas de Sistemas de Informação) foi a Codeception. Esta Framework foi concebida para PHP e fornece a construção de testes unitários, funcionais e de aceitação.

Apenas foram desenvolvidos testes para o website, estes que foram lecionados na UC e eram inclusive parte da avaliação, foram desenvolvidos após a realização do projeto.

Antes de começar a criação de testes começou-se por criar uma base de dados de testes, tal como recomendado pelo professor da unidade curricular, e após a criação da base, devido a esta não ter dados, optou-se por fazer a criação de “dumps” SQL (método disponibilizado pela Codeception), estes que seriam automaticamente executados antes da execução dos testes. Dito isto, começou-se a construção dos testes.

## 14.1 Testes Unitários

Começou-se por realizar os testes unitários ao website. Para a realização dos testes unitários optou-se inicialmente por fazer o teste CRUD aos modelos da base de dados, visto que estes testes são destinados apenas a uma classe. De seguida é possível observar nos alguns dos testes criados.

```
public function testCreate()
{
    $this->tester->haveRecord(Company::class, [
        'name' => 'company_test',
        'state' => Company::STATE_ACTIVE
    ]);
    $this->tester->seeRecord(Company::class, ['name' => 'company_test']);
}
```

Figura 6 - Teste unitário - Criar companhia

```
public function testRead()
{
    $this->tester->seeRecord(Company::class, ['name' => 'TAP Portugal']);
}
```

Figura 7 - Teste unitário - Ler companhia da base de dados

```
public function testUpdate()
{
    $company = $this->tester->grabRecord(Company::class, [
        'name' => 'TAP Portugal'
    ]);
    $company->name = 'New Company name';
    $this->assertTrue($company->save());
    $this->assertEquals( expected: 'New Company name', $company->name);
}
```

Figura 8 - Teste unitário - atualização de uma companhia

```

public function testDelete()
{
    $ticketItem = $this->tester->grabRecord(TicketItem::class, [
        'lost_item_id' => 1,
        'support_ticket_id' => 1
    ]);
    $ticketItem->delete();
    $this->tester->dontSeeRecord(TicketItem::class, [
        'lost_item_id' => 1,
        'support_ticket_id' => 1
    ]);
}

```

Figura 9 - Teste unitário - Apagar registo do item dos perdidos e achados associado ao ticket de suporte

Após a criação dos testes dos modelos da base de dados, foram também desenvolvidos alguns testes para os modelos dos formulários, para que fosse possível testar os mesmos de forma a não deixar o utilizador corromper o website. De seguida é possível observar um dos modelos de testes criados, o de autenticação.

Realizou-se vários testes ao mesmo, desde proteção contra campos vazios, campos errados, e totalmente corretos.

```

public function testLoginNoUser()
{
    $model = new LoginForm([
        'username' => 'not_existing_username',
        'password' => 'not_existing_password',
    ]);

    verify($model->login())->false();
    verify(Yii::$app->user->isGuest)->true();
}

```

Figura 10 - Teste unitário - Sem campos preenchidos

```

public function testLoginWrongPassword()
{
    $model = new LoginForm([
        'username' => 'rafael',
        'password' => 'wrong_password',
    ]);

    verify($model->login())->false();
    verify($model->errors)->arrayHasKey( key: 'password');
    verify(Yii::$app->user->isGuest)->true();
}

```

Figura 11 - Teste unitário - Autenticação com a password errada

```

public function testLoginCorrect()
{
    $model = new LoginForm([
        'username' => 'rafael',
        'password' => '12345678',
    ]);

    verify($model->login())->true();
    verify($model->errors)->arrayHasNotKey( key: 'password');
    verify(Yii::$app->user->isGuest)->false();
}

```

Figura 12 - Teste unitário - Autenticação com dados corretos



## 14.2 Testes Funcionais

Na construção de testes funcionais, sendo estes não diretamente apontados apenas a uma classe, mas sim a toda a Framework, ou seja, com as routes (caminhos) incluídos, optou-se pela realização de testes que testariam maioritariamente os formulários, porém incluindo os caminhos, verificação de erro dos campos dos formulários, etc., tal como é possível observar nos seguintes exemplos, estes que pertencem ao formulário do contactar o suporte.

Começou pela criação da função “before”, sendo esta a responsável por indicar que o utilizador estaria na página certa antes de começar a execução de cada teste.

```
public function _before(FunctionalTester $I)
{
    $I->amOnRoute('site/contact');
}
```

Figura 13 - Teste funcional - função “before”

De seguida, começou a construção dos testes funcionais.

Sem dados preenchidos:

```
public function checkContactSubmitNoData(FunctionalTester $I)
{
    $I->submitForm('#contact-form', []);
    $I->see('Contactar Suporte', 'h1');
    $I->seeValidationError('O nome não pode ser vazio.');
```

Figura 14 - Teste funcional - Submissão do formulário sem dados preenchidos

Submissão de dados com o formato de email incorreto:

```

public function checkContactSubmitNotCorrectEmail(FunctionalTester $I)
{
    $I->submitForm('#contact-form', [
        'ContactForm[name]' => 'Rafael',
        'ContactForm[email]' => 'rafael.email',
        'ContactForm[body]' => 'Mensagem',
    ]);
    $I->seeValidationError('O email tem de ser válido.');
```

Figura 15 - Teste funcional - Submissão do formulário com formato de email incorreto

Submissão com tudo correto e email enviado após a submissão:

```

public function checkContactSubmitCorrectData(FunctionalTester $I)
{
    $I->submitForm('#contact-form', [
        'ContactForm[name]' => 'Rafael',
        'ContactForm[email]' => 'email@email.com',
        'ContactForm[body]' => 'Esta é uma mensagem',
    ]);
    $I->seeEmailIsSent();
    $I->see('Obrigado por nos contactar. Responderemos assim que possivel!');
```

Figura 16 – Submissão do formulário com todos os dados devidamente preenchidos e corretos

Outros testes ainda que achamos importante realçar foram os testes efetuados na autenticação do back-office visto que este não permite o acesso de utilizadores e que o administrador tem diferentes permissões do funcionário. Dito isto, criou-se dois testes funcionais para o teste da autenticação:

```

public function loginUserAdmin(FunctionalTester $I)
{
    $I->submitForm('#login-form', [
        'LoginForm[username]' => 'rafael',
        'LoginForm[password]' => '12345678',
    ]);

    $I->see('Rafael Bento');
    $I->see('Aeroporto');
    $I->see('Utilizadores');
    $I->see('Perdidos e achados');
    $I->see('Métodos de pagamento');
    $I->see('Restaurantes');
    $I->see('Server log');
```

Figura 17 - Teste funcional - Autenticação de um administrador

```

public function loginUserEmployee(FunctionalTester $I)
{
    $I->submitForm('#login-form', [
        'LoginForm[username]' => 'pedro',
        'LoginForm[password]' => '12345678',
    ]);

    $I->see('Pedro Norberto');
    $I->see('Aeroporto');
    $I->see('Clientes');
    $I->see('Perdidos e achados');
    $I->see('Métodos de pagamento');
}

```

*Figura 18 - Teste funcional - Autenticação de um funcionário*

### 14.3 Testes de Aceitação

Nos testes de aceitação o grupo pensou criar um teste que cobriria a parte mais importante do projeto, ou seja, num software de um aeroporto, sem dúvidas a compra de um bilhete de Voo.

Sendo que estes testes não tinham sido lecionados em aula, pois deveriam ser pesquisados pelos estudantes, começámos a pesquisa para a realização dos mesmos através do website da própria Framework Codeception. Os testes de aceitação necessitam de um servidor, pois são executados através de um navegador, ao contrário dos restantes, portanto, optou-se por utilizar o “Chrome Driver”, disponibilizado no website da Codeception, após seguir todos os passos da instalação e da execução conseguimos colocar o teste a executar, porém este não tinha acesso à base de dados. Realizámos bastante pesquisa, porém devido a falta de tempo não conseguimos realizar a conexão à mesma pelo que não fomos capazes de concluir o teste de aceitação.

## 15 PROBLEMAS/DIFICULDADES E SOLUÇÕES

Ao longo do desenvolvimento do projeto ocorreram inúmeros problemas dos quais foi necessário efetuar pesquisas para resolvê-los. Assim, é possível observar os principais problemas obtidos em cada uma das componentes.

Cofinanciado por:



UNIÃO EUROPEIA  
Fundo Social Europeu

## 15.1 Aplicação Móvel

Um dos problemas, foi ao fazer um datePicker personalizado, o que o android oferece por predefinição não era o que pretendíamos, pois, o nosso objetivo é fornecer ao cliente a melhor interface possível e o que o android oferece tende a não cumprir este requisito caso o utilizador queira seleccionar uma data de anos atrás (data de nascimento). Através de pesquisa encontrámos uma solução, que seria, criar um estilo personalizado no próprio tema que se sobrepõe.

Outra dificuldade obtida foi a construção de caixas de seleção (comboBoxs), visto que, mais uma vez, queríamos que fosse semelhante a todos os restantes campos. Assim, voltou-se a fazer pesquisa e através do Material Design encontrámos a solução. esta foi utilizar o Material Design em conjunto com os nossos estilos personalizados e posteriormente utilizar um adaptador que preencheria a caixa de seleção.

Uma dificuldade que também nos custou bastante tempo foi a criação de bases de dados SQLite. Ao criar mais do que um ficheiro DBHelper, utilizámos a mesma base de dados, assim, existia o problema de que só o 1º é que corria o “create()”, porém, após pesquisar percebemos que não seria possível ter várias classes com a mesma base de dados, logo, optámos por dividir tudo em duas base de dados. Sendo estas uma relativa ao utilizador (informação de bilhetes de voo, tickets de suporte, etc.) e outra relativa às empresas que constam no aeroporto, ou seja, restaurantes (com a sua ementa) e lojas.

Outra dificuldade encontrada, foi a realização do QR code devido a fazer conflito com as versões de java e do gradle que estávamos a utilizar no projeto, portanto, optámos por deixar de parte a funcionalidade de gerar o código QR e avançar no projeto pois estávamos muito perto da entrega final e não era uma funcionalidade que considerássemos imprescindível.

## 15.2 API

Na construção da API apenas encontrámos dois problemas.

O primeiro problema, com o qual nos deparámos, foi a criação automática do CRUD, sendo este automaticamente criado na criação do controlador foi necessário remover estes métodos que já vinham predefinidos pelo que encontrámos uma função que o permitia, o método “actions” herdado da classe “ActiveController” que dispõe este método e fornece uma opção de os remover através de uma simples linha de código “`unset($actions['view']);`”.

O segundo e último problema, que descobrimos foi que após registar um método personalizado, apesar de ter sido registado com “POST” (por exemplo) no ficheiro “main”, permitia pedidos com o método “GET”. Assim, para a resolução deste problema, achou-se a solução através do método “verbs”, que herdara da classe “Controller”. Através deste, era possível definir apenas o método “POST” e não ambos, “POST” e “GET”.

## 15.3 Website

No desenvolvimento do Website, devido à dimensão do projeto, tivemos bastantes problemas dos quais foi necessário efetuar bastante pesquisa, sendo estes os principais:

- Envio de emails – A Yii2 já disponibiliza uma biblioteca que permite enviar emails, porém obtivemos inúmeros problemas e foi feita bastante pesquisa, pois o “Gmail” já não permite ativar uma permissão que em tempos era a solução, a permissão “Permitir aplicações menos seguras”. Esta foi desabilitada em maio de 2022. Após mais pesquisa encontrámos uma solução, sendo esta a criação de uma “palavra-passe de aplicação”, esta que forneceria um código e posteriormente serviria como uma password de email que seria utilizada pelo servidor de SMTP;
- Guardar imagens – Sendo as imagens algo imprescindível para o nosso projeto foi feita alguma pesquisa, inclusive pediu-se também ajuda ao docente da UC, que nos forneceu um guia que teria encontrado e através desse guia encontrámos a solução, a utilização das bibliotecas disponibilizadas pela Yii2, “FileHelper” e “UploadedFile”, a utilização das mesmas resolveria o problema de guardar as imagens no servidor, bastando apenas a inserção das mesmas num formulário, problema este que foi resolvido através do “FileInput”, disponibilizado pelo “Kartik”;
- Modelos para formulários – Em alguns dos formulários, principalmente no front-office não estavam diretamente ligados a modelos “ActiveRecord”, pelo que foi necessário efetuar a criação de modelos especificamente para cada um desses formulários que criámos, de forma que pudéssemos definir rules e métodos nos mesmos. Um exemplo dos modelos criados foi o “FlightForm”, modelo utilizado no formulário de quando se pesquisa um novo voo, quer através do website por parte do cliente ou através da API;
- Testes de aceitação – Tal como já mencionado no tópico dos testes, o teste de aceitação foi um problema no desenvolvimento do projeto, pois não conseguimos que o mesmo estabelecesse uma conexão com a base de dados, o que levou a que o mesmo não fosse realizado;
- Campos “dropdown” – Em inúmeros dos formulários que possuíamos necessitávamos de um campo onde o utilizador pudesse não escrever, mas sim escolher uma opção, isto tinha sido algo já realizado em fichas, porém achámos que não seria a forma mais correta pois necessitávamos de usar o mesmo menu em mais do que um formulário, sendo que não seria correto caso o menu alterasse ter de se alterar em todo o lado. Pelo que se optou pela criação de constantes no próprio modelo, de forma que posteriormente este fornecesse um par de chave-valor ao campo e fornecesse ao utilizador as opções disponíveis;
- Campos não necessários dos modelos – Por vezes, não era necessário todos os campos dos modelos, logo, obtivemos um problema, porque, não sabíamos como utilizar apenas parte destes campos, como por exemplo na criação de novos voos no back-office. No entanto, depois de pesquisar achou-

Cofinanciado por:



se uma solução fornecida pela Yii2, a criação de cenários, deste modo, o modelo fornecia apenas os campos relativos ao cenário atual do mesmo.

Cofinanciado por:



## 16 CONCLUSÃO

Para concluir este relatório, achamos que fizemos um bom projeto, ao longo do planeamento achámos que o projeto era um pouco ambicioso, no entanto organizámos bem o tempo e conseguimos fazer os requisitos com alguns dos extras que tínhamos planeado.

Concluimos que este projeto foi importante para a nossa vida futura, pois atualmente é cada vez mais normal trabalhar com web-services. Assim, os conhecimentos aprendidos ao longo das UCs de SIS e AMSI são uma mais-valia na nossa vida profissional.

Este projeto ajudou-nos também a preparar para a vida profissional, mais especificamente, no trabalho autónomo e na pesquisa de recursos, pois a quantidade de pesquisa que tivemos de fazer foi bastante visto que ambicionámos implementar vários extras.

Para implementar este projeto seria necessário ocorrer algumas alterações tanto na parte mobile como na parte web, como por exemplo adicionar a função de comprar bilhete com escalas e a tradução para vários idiomas, visto que um aeroporto não é possível possuir apenas o português já que é utilizado por pessoas de diversos lugares do mundo.