



Universidade do Minho  
Mestrado em Cibersegurança

Aplicações e Serviços de Computação  
em Nuvem Fase-1  
Grupo nº33

Carlos Daniel Silva Fernandes  
(PG59783)

Afonso Miguel da Silva Ribeiro  
(PG60232)

Pedro Augusto Ennes de Martino Camargo  
(PG59791)

Renato Pereira Garcia  
(PG61542)

Luís Filipe Pinheiro Silva  
(PG59790)

2 de novembro de 2025



# Conteúdo

<b>1</b>	<b>Contextualização</b>	<b>3</b>
1.1	Visão Geral da Aplicação . . . . .	3
1.1.1	Funcionalidades Principais . . . . .	3
1.2	Principais Tecnologias . . . . .	3
1.3	Arquitetura . . . . .	4
1.3.1	Infraestrutura e Componentes . . . . .	4
1.4	APIs fornecidas pela aplicação . . . . .	4
1.4.1	API REST . . . . .	4
1.4.2	APIs Externas . . . . .	4
<b>2</b>	<b>Análise de Riscos</b>	<b>5</b>
2.1	Ponto Único de Falha . . . . .	5
2.2	Gargalos de Desempenho . . . . .	5



# Capítulo 1

## Contextualização

### 1.1 Visão Geral da Aplicação

Numa primeira fase deste projeto, é feito o estudo da aplicação de tracking de voos airtrail. Neste relatório estão presentes detalhes sobre a arquitetura da aplicação, as suas funcionalidades e APIs, bem como uma reflexão e discussão sobre potenciais pontos de falha da aplicação, gargalos de desempenho e possíveis dificuldades na instalação.

O principal objetivo da aplicação **AirTrail** é fornecer um sistema completo de **tracking de viagens aéreas**, permitindo que o utilizador registe e acompanhe todas as viagens realizadas num determinado período de tempo.

#### 1.1.1 Funcionalidades Principais

- **Adição Manual de Viagens:** O utilizador pode adicionar uma viagem manualmente através de formulários disponibilizados pela própria aplicação.
- **Importação de Dados:** O **AirTrail** permite importar ficheiros provenientes de outras plataformas de gestão de voos, facilitando a migração e centralização de dados.
- **Exportação de Dados:** O utilizador pode exportar os seus registos de voo, permitindo criar cópias de segurança (*backups*) ou transferir os dados para outras aplicações compatíveis.
- **Estatísticas de Voo:** A aplicação gera automaticamente estatísticas e métricas baseadas nas viagens registadas, fornecendo ao utilizador uma visão geral do seu histórico de voos.

### 1.2 Principais Tecnologias

- **Base de Dados:** *PostgreSQL*
- **Web Server:** Desenvolvido com *SvelteKit* e *TypeScript*



- **Backend:** Integrado no próprio frontend através do *SvelteKit*
- **Acesso à Base de Dados:** *Kysely* (SQL Builder)
- **Schema e Migrações:** Geridas com *Prisma*

## 1.3 Arquitetura

A arquitetura adotada é do tipo **cliente-servidor**, em que o cliente interage com o servidor web que, por sua vez, comunica com a base de dados.

### 1.3.1 Infraestrutura e Componentes

A infraestrutura é composta por dois containers Docker que comunicam entre si através de uma *bridge network* criada automaticamente pelo *docker compose*:

NETWORK ID	NAME	DRIVER	SCOPE
b16b80e06117	airtrail_default	bridge	local

- **Web Server:** Container *airtrail*
- **Base de Dados:** Container *airtrail\_db*

## 1.4 APIs fornecidas pela aplicação

### 1.4.1 API REST

A **API REST** da aplicação **AirTrail** fornece um conjunto de endpoints para gestão das viagens registadas por cada utilizador. Abaixo estão listadas as principais rotas:

GET /flight/list	-> Lista todos os voos do utilizador.
GET /flight/get/[id]	-> Detalhes de um voo específico.
POST /flight/save	-> Cria ou atualiza um voo existente.
POST /flight/delete	-> Remove um voo pelo seu ID.

### 1.4.2 APIs Externas

A aplicação **AirTrail** faz uso de serviços externos para obter e enriquecer informações sobre voos e aeronaves.

- **Adsbdb:** Serviço utilizado para realizar *fetch* de informações associadas a *flight numbers*.
- **AeroDataBox:** Alternativa à Adsbdb, permitindo que o utilizador opte por uma integração premium.



## Capítulo 2

# Análise de Riscos

### 2.1 Ponto Único de Falha

O principal risco na arquitetura da aplicação **AirTrail** é o **Web Server**. Caso o servidor falhe por qualquer motivo, todos os clientes conectados perdem imediatamente o acesso à aplicação.

Algumas causas possíveis para uma falha no servidor incluem:

- Falta de energia elétrica no local de hospedagem;
- Ciberataques direcionados ao servidor web;
- Erros humanos durante manutenção ou atualização do sistema.

Para mitigar este risco, recomenda-se a implementação de mecanismos de **redundância**, **replicação** e **failover**, garantindo alta disponibilidade do serviço.

### 2.2 Gargalos de Desempenho

À medida que o número de utilizadores cresce, o servidor pode tornar-se um gargalo, resultando em tempos de resposta mais lentos ou até falhas. Como a aplicação **AirTrail** utiliza o **SvelteKit** com **Server-Side Rendering (SSR)**, cada requisição ao servidor requer processamento adicional, o que aumenta a carga computacional.

Com apenas uma instância de servidor web, o sistema não possui escalabilidade horizontal. Assim, em situações de aumento de tráfego, o servidor pode atingir o limite da sua capacidade, afetando o desempenho global da aplicação.