

**Instituto Superior de Engenharia do Porto**  
Departamento de Engenharia Electrotécnica  
Rua Dr. António Bernardino de Almeida, 431, 4249-015 Porto

# **Stand Virtual de Automóveis**

Mestrado em Engenharia Eletrotécnica e de Computadores

**Diogo Gonçalo Lima de Freitas**

Supervisor: Prof. Maria Benedita Campos Neves Malheiro

Ano Letivo: 2021-2022



---

# Conteúdo

---

<b>Conteúdo</b>	<b>i</b>
<b>Lista de Figuras</b>	<b>iii</b>
<b>Lista de Tabelas</b>	<b>v</b>
<b>Glossário</b>	<b>vii</b>
<b>1 Introdução</b>	<b>1</b>
1.1 Contexto . . . . .	1
1.2 Motivação . . . . .	1
1.3 Objetivos . . . . .	1
1.4 Estrutura do Relatório . . . . .	2
<b>2 Análise de Requisitos</b>	<b>3</b>
2.1 Problema . . . . .	3
2.2 Especificação de Requisitos . . . . .	3
2.2.1 Atores . . . . .	3
2.2.2 Casos de Uso . . . . .	4
2.3 Sumário . . . . .	4
<b>3 Desenvolvimento</b>	<b>5</b>
3.1 Tipo de Aplicações . . . . .	5
3.2 Tecnologias . . . . .	5
3.2.1 Servidor . . . . .	5
3.2.2 Cliente . . . . .	6
3.3 Arquitetura . . . . .	6
3.4 Modelo de dados . . . . .	7
3.4.1 Esquema de Dados . . . . .	7
3.5 Mapa de Navegação . . . . .	8
3.6 Testes . . . . .	8

3.6.1	Funcionais . . . . .	9
3.6.2	Desempenho . . . . .	9
3.6.3	Carga . . . . .	10
3.7	Sumário . . . . .	10
<b>4</b>	<b>Conclusão</b>	<b>11</b>
4.1	Resultados . . . . .	11
4.2	Desenvolvimentos Futuros . . . . .	11
	<b>Bibliografia</b>	<b>13</b>
<b>A</b>	<b>Protocol Messages</b>	<b>15</b>

---

# Lista de Figuras

---

2.1	Casos de Uso . . . . .	4
3.1	Arquitetura da aplicação <i>web</i> . . . . .	6
3.2	Esquema de Dados . . . . .	7
3.3	Mapa de Navegação . . . . .	8



---

## Lista de Tabelas

---

3.1	Funcionalidades de Utilizador . . . . .	9
3.2	Funcionalidades de Administrador . . . . .	9
3.3	Resultado dos testes de desempenho . . . . .	10
3.4	Resultado dos testes de desempenho . . . . .	10





---

# Glossário

---

**CSS** Cascading Style Sheets. 6

**DEE** Departamento de Engenharia Eletrotécnica. 1

**HTML** HyperText Markup Language. 6

**ISEP** Instituto Superior de Engenharia do Porto. 1

**MEEC** Mestrado de Engenharia Eletrotécnica e Computadores. 1

**RINTE** Redes Inteligentes e Serviços. 1

**SGBD** Sistema de Gerenciamento de Banco de Dados. 6

**SQL** Structured Query Language. 6



## Capítulo 1

---

# Introdução

---

### 1.1 Contexto

No âmbito da unidade curricular de Redes Inteligentes e Serviços (RINTE) do Mestrado de Engenharia Eletrotécnica e Computadores (MEEC), do Departamento de Engenharia Eletrotécnica (DEE), do Instituto Superior de Engenharia do Porto (ISEP), foi proposto um projeto que consiste na construção de uma aplicação *web*, orientada à venda online de automóveis, mais conhecido como um standvirtual de automóveis.

### 1.2 Motivação

Com este projeto, deseja-se aprender mais sobre o desenvolvimento de aplicações *web*, desde como funciona todo o processo de estruturação da aplicação desde a raiz, conhecer novas tecnologias e ferramentas que sejam úteis no desenvolvimento *web*, entre outras coisas.

### 1.3 Objetivos

Os objetivos propostos para este projeto são:

- Construir uma aplicação *web* acessível, de forma que o utilizador possa interagir com as funcionalidades disponibilizadas, tanto a consulta dos anúncios como a aquisição dos mesmos;
- Implementar um servidor para a gestão das operações e informações inseridas na aplicação *web*;

## 1.4 Estrutura do Relatório

Este relatório está dividido em 4 capítulos, sendo o primeiro, uma introdução ao projeto proposto. O capítulo seguinte tem como objetivo apresentar os requisitos do problema proposto, enumerando os atores e os respectivos casos de uso. No terceiro capítulo vai ser descrito a arquitetura do projeto e as tecnologias utilizadas no mesmo. Por fim, no último capítulo, vão ser apresentados todos os resultados e os possíveis desenvolvimentos futuros.

## Capítulo 2

---

# Análise de Requisitos

---

### 2.1 Problema

Com o avanço da tecnologia, as empresas têm, cada vez mais, a necessidade de criar um serviço de vendas online, de forma a aumentar a acessibilidade dos produtos aos seus clientes, podendo assim, lucrar mais. Partindo deste princípio, o resultado deste projeto tem como objetivo uma aplicação *web* pronta a implementar, sendo, neste caso em específico, um stand virtual de automóveis.

### 2.2 Especificação de Requisitos

Para a implementação da aplicação *web*, é necessário cumprir os seguintes requisitos:

- Autenticação, Registo e Gestão de Utilizadores
- Consulta e Gestão de Anúncios
- Operação de compra e o respetivo registo das compras

#### 2.2.1 Atores

Os atores são todos os tipos de utilizadores que podem usufruir dos serviços disponibilizados pela aplicação *web*. Neste projeto vão haver 3 tipos de atores:

- Utilizadores não-registados,
- Utilizadores registados,
- Administrador

### 2.2.2 Casos de Uso

Os casos de uso são as ações/operações que cada tipo de ator pode realizar enquanto utilizador da aplicação *web*. Como referido, os casos de uso foram divididos por cada tipo de ator (acima enumerado), como está demonstrado na Figura 2.1.

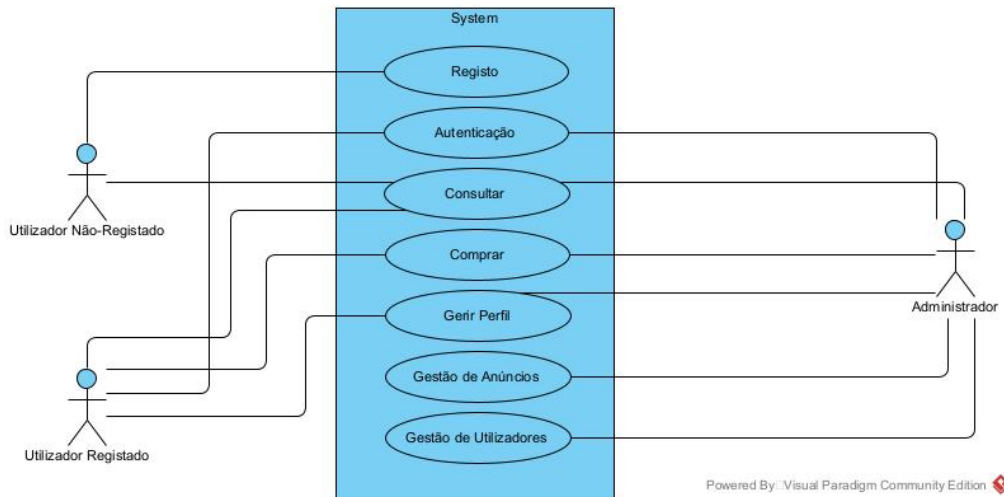


Figura 2.1: Casos de Uso

## 2.3 Sumário

Após a apresentação dos atores presentes na realização/utilização da aplicação *web* e das respetivas funcionalidades disponibilizadas para cada um destes, vai ser explicado o todo o desenvolvimento realizado para a construção desta aplicação *web*.

## Capítulo 3

---

# Desenvolvimento

---

### 3.1 Tipo de Aplicações

No desenvolvimento do projeto, foram utilizados as 3 componentes básicos de uma aplicação *web*:

- Interface gráfica (*Front-end*): Possibilita a interação do utilizador, para que lhe seja possível usufruir das funcionalidades disponibilizadas pela aplicação *web*
- Servidor (*Back-end*): Responsável pela lógica da aplicação *web*, desde a troca de informação com o *front-end* até ao envio e gestão dos dados na base de dados.
- Base de dados: Permite o armazenamento dos dados inseridos na aplicação *web*, de forma a estarem disponíveis posteriormente para serem acedidos pelo servidor.

### 3.2 Tecnologias

Neste projeto foram utilizadas várias tecnologias para satisfazer as componentes enumeradas anteriormente. Estas tecnologias estão divididas em 2 categorias: Servidor e Cliente.

#### 3.2.1 Servidor

Na parte do servidor, foram utilizadas 2 tecnologias: Flask e MySQL.

- Flask [1]: O Flask é uma *framework* de Python [2] utilizada para desenvolvimento de aplicações *web*. O Flask é considerada uma *micro-framework*, por causa da quantidade reduzida de configurações necessárias para executar a aplicação *web*, fornecendo imensa liberdade ao programador para desenvolver a aplicação *web*. Este tipo de *framework* são utilizados em casos de desenvolvimento de pequenas aplicações *web* onde as necessidades não são as mesmas que uma estrutura de nível corporativo que economizaria tempo de manutenção e dinheiro.
- MySQL [3]: O MySQL é um Sistema de Gerenciamento de Banco de Dados (SGBD), que utiliza a linguagem Structured Query Language (SQL) como interface. É atualmente um dos sistemas de gerenciamento de bancos de dados mais populares da *Oracle Corporation*, com mais de 10 milhões de instalações pelo mundo. [4]

### 3.2.2 Cliente

Na parte do cliente, para além da linguagem HyperText Markup Language (HTML), foi utilizado o Bootstrap [5], com o objetivo de agilizar o processo de construção da interface gráfica.

- Bootstrap [5]: Bootstrap é um *framework web* com código-fonte aberto para desenvolvimento de componentes de interface e *front-end* para sites e aplicações *web* usando HTML, Cascading Style Sheets (CSS) e *JavaScript*, baseado em modelos de design para a tipografia, melhorando a experiência do utilizador em um site amigável e responsivo. [6]

## 3.3 Arquitetura

A arquitetura do projeto consiste numa aplicação *web* que tem como base um servidor *back-end* gerado pela tecnologia Flask [1], que está interligado a uma base de dados relacional, criada com a tecnologia MySQL [3]. Os resultados vão ser representados numa interface gráfica, com o auxílio da *framework* Bootstrap [5]. Essa arquitetura está representada na Figura 3.1.

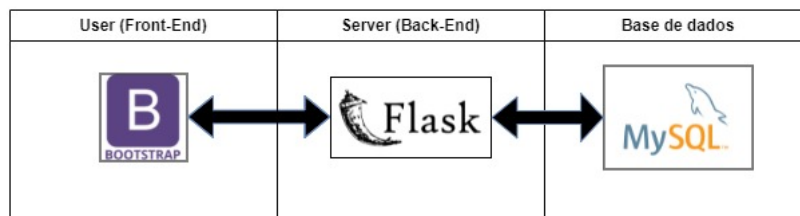


Figura 3.1: Arquitetura da aplicação *web*



## 3.4 Modelo de dados

Para este projeto, foi criado uma base de dados relacional para o armazenamento dos dados dos veículos adicionados e de todos os dados necessários para o funcionamento da aplicação *web*, como as credenciais dos utilizadores e as compras feitas na aplicação.

### 3.4.1 Esquema de Dados

A base de dados é constituída por 8 tabelas, havendo uma tabela principal “veículos”, onde são armazenadas as informações principais sobre cada anúncio inserido na aplicação, como pode ser visto na Figura 3.2.

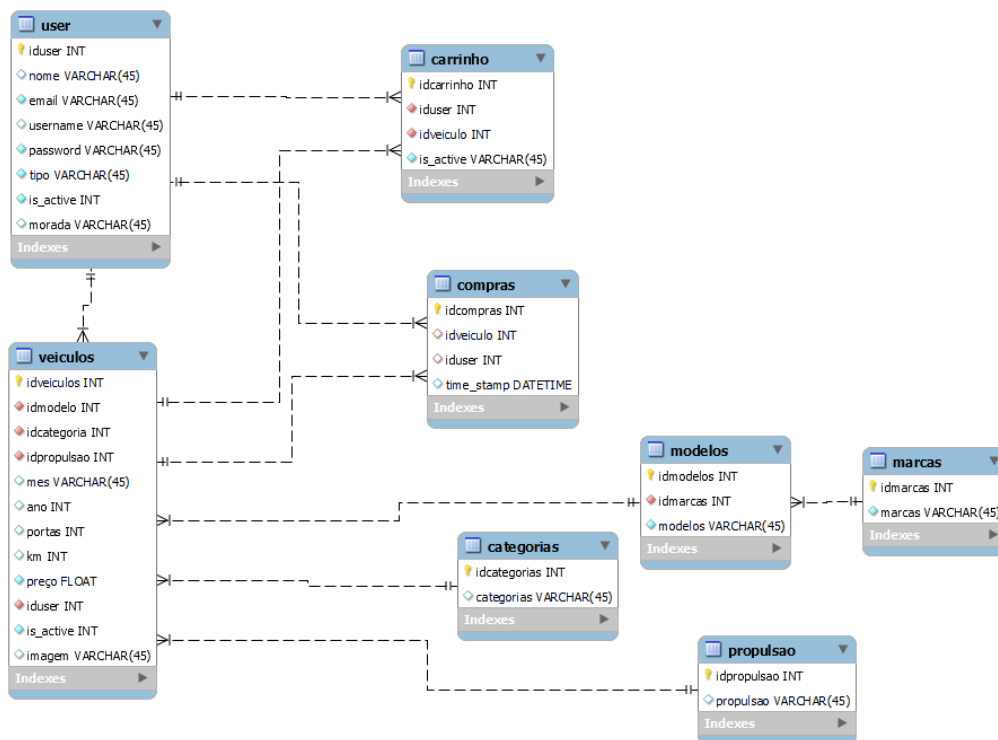


Figura 3.2: Esquema de Dados

A tabela “veículos” é a tabela principal da base de dados, onde vão estar organizados todos os dados de cada anúncio inserido na aplicação. Nesta tabela são organizados dois tipos de dados em relação aos anúncios: os dados comuns e os dados não comuns. Alguns dos dados inseridos nos anúncios diferem de veículo para veículo, sendo praticamente impossíveis de prever, como o mês e ano do registo, a quilometragem e o preço de cada um. Estes dados serão chamados de dados não comuns e vão ser armazenados diretamente na tabela principal.

Os dados comuns vão ser apresentados na tabela principal, através de ids, que correspondem á linha onde estes se encontram na respetiva tabela independente. Estas tabelas independentes foram criadas para armazenar os dados que são, de alguma forma, previsíveis em cada anúncio. Esta organização dos dados comuns em várias tabelas separadas, torna a base de dados mais eficiente, evitando armazenamento desnecessário com a repetição de informação adquirida.

### 3.5 Mapa de Navegação

Para a existência da interação com o utilizador, deverá ser implementada uma interface gráfica em que este possa consultar todos os anúncios publicados, realizar as suas compras e ter acesso a várias outras funcionalidades disponibilizadas pela aplicação *web*. Estas funcionalidades estarão distribuídas por diferentes páginas, como está apresentado na Figura 3.3.

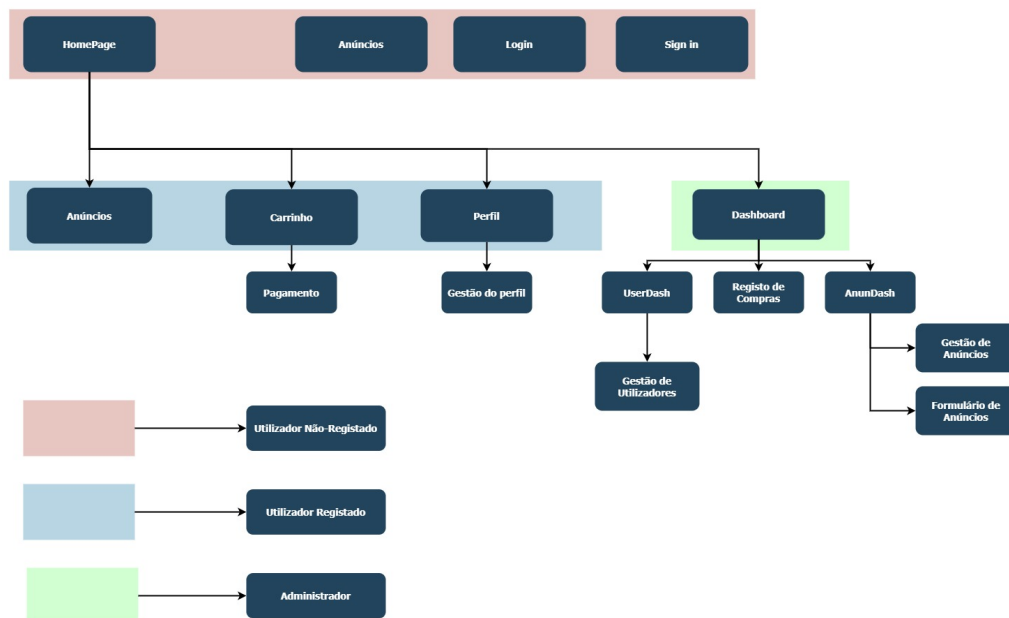


Figura 3.3: Mapa de Navegação

### 3.6 Testes

De forma a testar e confirmar o funcionamento de todas as funcionalidades e operação realizadas na/pela aplicação *web*, foram feitos vários tipos de testes, sendo eles: Funcionais, de Desempenho, de Carga e de Usabilidade. Os testes realizados foram feitos numa máquina com um processador Intel(R) Core(TM) i7-10750H CPU @ 2.60GHz, com memória RAM de 16,0 GB e um HDD de 1 TB.

### 3.6.1 Funcionais

Os testes funcionais são testes que avaliam as funcionalidades disponibilizadas pela aplicação *web*. Estes vão estar divididos em 2 grupos principais: Funcionalidades de Utilizador e Funcionalidades de Administrador.

- As **funcionalidades de utilizador** são as funcionalidades básicas disponibilizadas pela aplicações *web*, aos utilizadores. Este teste vai ser dividido em 2 tipos: Utilizadores Não-Registados e Utilizadores Registados (incluindo administradores). Essas funcionalidades estão presentes na Tabela 3.1.

	Registo	Autenticação	Consultar	Comprar	Gerir perfil
Utilizador	✓	✓	✓	✓	✓

Tabela 3.1: Funcionalidades de Utilizador

- As **funcionalidades de administrador** são funcionalidades que apenas estão disponíveis a utilizadores do tipo "*admin*", utilizadores com permissão extra em relação aos outros. Estas funcionalidades extras são acrescentadas às funcionalidades normais de um utilizador básico. Essas funcionalidades estão presentes na Tabela 3.2.

	Adicionar Anúncios	Gerir Anúncios	Gerir Utilizadores	Registo de Compras
Administrador	✓	✓	✓	✓

Tabela 3.2: Funcionalidades de Administrador

### 3.6.2 Desempenho

Os testes de desempenho avaliam a quantidade de informação transmitida e o tempo necessário para essa transmissão. Foram realizadas 10 tentativas de cada teste, e a respectiva média dos resultados. Os resultados destes testes vão ser divididos em 2 partes: Funcionalidades de Utilizadores e de Administradores. Os resultados estão apresentados na Tabela 3.3 e na Tabela 3.4.

- **Funcionalidades de utilizador:**

	$\Delta b$ (kiB)	$\Delta t$ (ms)
Autenticação	$14,626 \pm 0,000$	$7 \pm 1,2$
Registo	$3,734 \pm 0,000$	$5,5 \pm 0,7$
Consultar Anúncios*	$17,714 \pm 0,000$	$9,9 \pm 1,1$
Compra	$3,942 \pm 0,000$	$12,8 \pm 1,4$
Perfil	$6,435 \pm 0,000$	$12,9 \pm 5,4$

Tabela 3.3: Resultado dos testes de desempenho

\*- Foi carregado o mesmo número de elementos (10), podendo ter diferentes quantidades de dados

- **Funcionalidades de administrador:**

	$\Delta b$ (kiB)	$\Delta t$ (ms)
Adicionar Anúncios	$12,104 \pm 0,000$	$18,5 \pm 4,6$
Dashboard Utilizadores*	$11,926 \pm 0,000$	$9,8 \pm 2,6$
Dashboard Anúncios	$26,285 \pm 0,000$	$10,7 \pm 2,5$
Registo de Compras	$10,599 \pm 0,000$	$12,2 \pm 1,8$

Tabela 3.4: Resultado dos testes de desempenho

\*- Foi carregado o mesmo número de elementos (10), podendo ter diferentes quantidades de dados

### 3.6.3 Carga

Os teste de carga avaliam a capacidade da aplicação *web*, sendo feito vários testes com vários utilizadores virtuais em simultâneo. De forma a obter apenas resultados relevantes, estes testes são feitos apenas às operações que exigem uma maior transferência de dados, para que seja testada uma possível sobrecarga na aplicação *web*. Por razões técnicas, não foi possível realizar os testes de carga.

## 3.7 Sumário

Após a apresentação das várias etapas de desenvolvimento da aplicação (Casos de uso, Modelo Conceptual, Modelo de Dados, Mapa de Navegação e os testes efetuados), vão ser apresentadas as respetivas conclusões, mencionando os resultados do projeto em geral e os aspetos a melhor no futuro.

## Capítulo 4

---

# Conclusão

---

### 4.1 Resultados

Após a conclusão deste projeto, os resultados provaram-se positivos, pois todos os objetivos propostos inicialmente foram alcançados com sucesso. Para além disso, este projeto veio-se a revelar muito proveitoso, pois apresentou uma versão básica mas rica sobre o desenvolvimento de aplicações *web*, tendo sido uma experiência bastante enriquecedora ao descobrir várias tecnologias novas e formas de estruturar e desenvolver uma aplicação *web* de raiz.

### 4.2 Desenvolvimentos Futuros

Nesta secção, vão ser apresentado alguns tópicos que podem ser melhorados e/ou adicionados ao projeto no futuro, de forma a melhor o desempenho da aplicação web. Os tópicos registados para uma possível melhoria são:

- Possibilidade de upload de múltiplas imagens para cada anúncio ;
- Implementação da operação de pagamento;
- Aumentar/Completar os dados disponibilizados pela base de dados



---

# Bibliografia

---

- [1] “Flask.” <https://flask.palletsprojects.com/en/2.0.x/>, 2010. Consultado a : 2022-01-09. [citado na p. 6]
- [2] “Python.” <https://www.python.org/>, 2021. Consultado a : 2022-01-09. [citado na p. 6]
- [3] “Mysql.” <https://www.mysql.com/>, 2022. Consultado a : 2022-01-09. [citado na p. 6]
- [4] “Mysql.” <https://pt.wikipedia.org/wiki/MySQL>), 2022. Consultado a : 2022-01-13. [citado na p. 6]
- [5] “Bootstrap.” <https://getbootstrap.com/docs/5.1/getting-started/introduction/>, 2022. Consultado a : 2022-01-09. [citado na p. 6]
- [6] “Bootstrap.” [https://pt.wikipedia.org/wiki/Bootstrap\\_\(framework\\_frontend\)](https://pt.wikipedia.org/wiki/Bootstrap_(framework_frontend))), 2022. Consultado a : 2022-01-09. [citado na p. 6]





## **Anexo A**

---

# **Protocol Messages**

---

