



*Escola Superior de Tecnologia e Gestão*

*Instituto Politécnico da Guarda*

# RELATÓRIO DE PROJETO CARPOOLING

VASCO MANUEL DE DEUS LELLO FORTUNA

RELATÓRIO PARA A OBTENÇÃO DO GRAU DE LICENCIADO

EM ENGENHARIA INFORMÁTICA

Novembro/2015



# Projeto de Informática

2014/2015

## Car Pooling

Orientador: Prof. Doutor José Fonseca

Realizado por: Vasco Manuel de Deus Lello Fortuna

# Resumo

Este trabalho consiste no processo de desenvolvimento de uma aplicação informática para gestão de boleias entre um grupo de colegas que trabalham na mesma empresa e partilham o mesmo percurso na deslocação de casa para o trabalho. Um exemplo que ilustra esta necessidade passa-se no IPG, onde dez docentes que moram na área de Viseu combinam entre si a partilha de boleias de casa e para o seu local de trabalho que é na Guarda.

Já existem diversas aplicações de boleias no mercado, no entanto elas somente satisfazem as necessidades individuais do utilizador, tais como pedir boleias a condutores e a gestão da sua conta de utilizador e boleias associadas. Estas aplicações não fazem a gestão da partilha de boleias para um grupo de utilizadores que pagam as boleias dando outras boleias, em vez de ser em dinheiro. Isto tem necessidades de organização próprias, como sendo a gestão de um grupo, a visualização das várias boleias entre os membros, gestão dos condutores e interfaces intuitivas para o grupo, permitindo realizar as operações mais comuns.

A aplicação proposta designa-se por CarPooling e foca-se num único mapa de boleias<sup>1</sup> que será visualizado pelo grupo inteiro. Os membros terão ferramentas específicas para a gestão de boleias dentro deste mapa. Estas funcionalidades incluirão gestão dos membros por parte de um administrador e funcionalidades específicas para a organização das boleias, tais como repetição de boleias para um período de tempo (procedimento muito utilizado) e escolha automática do condutor para que a partilha seja o mais justa possível.

A interface da aplicação CarPooling deve ser o mais simplificada e automatizada possível, adaptada as reais necessidades deste tipo de grupos de boleias, para facilitar o seu uso e diminuir o tempo que cada utilizador necessita para utilizar a aplicação.

Palavras-chave:

Programação web, MySQL, PHP, AJAX, boleias.

---

<sup>1</sup> Este mapa de boleias é uma representação gráfica, em estilo de um calendário, das boleias. Não confundir com um mapa de estradas.

# Abstract

This report consists on the process of development of a web app for managing carpools by a group of colleagues that work on the same corporation e share the same route from home to work. An example that illustrates this necessity exists on IPG, where 10 professors, which live on the area of Viseu, schedule carpools from home to their workplace, in Guarda, between themselves.

There is already a wide variety of carpooling apps in the market, however they only satisfy the individual needs of the user, such as asking for carpools to drivers and the management of their user's account and associated carpools. These apps do not make the management of shared carpools for a group of users that pay carpools by giving carpools instead of money. This has necessities of organization of its own, such as management of a group of users, visualization of several carpools between all members, management of the drivers and intuitive interface for the group, performing the most common tasks.

The application requested will focus on a single carpooling map that will be visualized by the entire group. The members will have specific tools for the management of carpools inside this map. These tools will include basic management of the members by an administrator and specific functionalities for the organization of carpools, for example, repetition of carpools for a given period of time (a very used procedure) and automatic choice of driver. so that the members can share carpools fairly.

The interface of the application should be as simplified and automated as possible, adapted to the real necessities of this type of groups of carpools, so its access can be facilitated and the time that each user spends on the app can be diminished.

Key-words:

Web programming, MySQL, PHP, AJAX, Carpooling.

## Conteúdo

1.	Introdução .....	8
1.1.	Contexto do projeto.....	8
1.2.	Objetivos da aplicação .....	10
1.3.	Estrutura do relatório .....	11
2.	Estado de arte .....	12
2.1.	Análise de aplicações de boleias correntes .....	12
3.	Metodologia e Análise de Requisitos .....	17
3.1.	Metodologia .....	17
3.2.	Planeamento.....	18
3.3.	Diagrama de contexto .....	19
3.4.	Diagrama de casos de uso.....	20
3.5.	Descrições de casos de uso.....	22
3.5.1.	Inserir boeia .....	22
3.5.2.	Eliminar boeia .....	24
3.6.	Diagramas de sequência .....	24
3.6.1.	Inserir boeia .....	25
3.6.2.	Eliminar boeia .....	26
3.7.	Diagrama de classes e modelo EER.....	26
3.8.	Semântica.....	29
3.8.1.	Utilizadores.....	29
3.8.2.	Boeias.....	32
3.8.3.	Passageiros.....	34
3.8.4.	Estatísticas .....	35
3.8.5.	Configurações .....	36
3.8.6.	Alterações.....	38
4.	Tecnologias utilizadas .....	39
4.1.	Tecnologias utilizadas.....	39
4.1.1.	HTML .....	39

4.1.2.	CSS .....	39
4.1.3.	PHP.....	40
4.1.4.	Javascript .....	40
4.1.5.	AJAX .....	41
4.1.6.	MySQL.....	41
4.1.7.	GitHub.....	42
5.	Desenvolvimento .....	43
5.1.	Diagrama de hierarquia .....	43
5.2.	Análise e implementação.....	44
5.2.1.	Visualização dos registos de boleias num único mapa de boleias.....	44
5.2.2.	Interface e interação eficiente da aplicação .....	46
5.3.	Avaliação da aplicação.....	48
6.	Conclusão.....	50
7.	Bibliografia .....	52
8.	Anexos.....	54

# Lista de Figuras

Figura 1 - Custo dos combustíveis entre 1960 e 2014 em Portugal .....	8
Figura 2 - Mapa de Gantt efetivo .....	18
Figura 3 - Diagrama de Contexto.....	19
Figura 4 - Diagrama de Casos de Uso .....	21
Figura 5- Diagrama de sequência "Inserir boleia" .....	25
Figura 6 - Diagrama de sequência "Eliminar boleia " .....	26
Figura 7 - Diagrama de classes .....	27
Figura 8 - Modelo EER criado em MySQL Workbench .....	28
Figura 9 - Diagrama de Hiararquia.....	43
Figura 10 - Interface da aplicação CarPooling .....	45
Figura 11 - Ordem de construção de tabelas pelo HTML.....	45
Figura 12 - Fluxograma do algoritmo de leitura de boleias .....	46
Figura 13 - Interface do mapa de boleias .....	48
Figura 14 - Diagrama de sequência do caso de uso "Alterar boleia" .....	59
Figura 15 - Diagrama de sequência do caso de uso "Entrar numa boleia" .....	60
Figura 16 - Diagrama de sequência do caso de uso "Repetir boleia" .....	60
Figura 17 - Diagrama de sequência do caso de uso "Sair de uma boleia" .....	60
Figura 18 - Avaliação do PageSpeed da aplicação para telemóvel (Parte 1) .....	61
Figura 19 - Avaliação do PageSpeed da aplicação para telemóvel (Parte 2) .....	62
Figura 20 - Avaliação do PageSpeed da aplicação para computador .....	63

# Lista de Tabelas

Tabela 1 – Funcionalidades de blablacar .....	13
Tabela 2 - Funcionalidades de boleia.net.....	14
Tabela 3 - Funcionalidades de pendura.pt.....	15
Tabela 4 - Comparação de todas as aplicações estudadas.....	16
Tabela 5 - Descrição do caso de uso "Inserir Boleia" .....	23
Tabela 6 – Descrição do caso de uso “Eliminar Boleia” .....	24
Tabela 7 - Dicionário de dados da tabela Utilizadores .....	30
Tabela 8 - Operações da tabela Utilizadores .....	31
Tabela 9 - Dicionário de dados da tabela Boleias .....	32
Tabela 10 - Operações da tabela Boleias .....	33
Tabela 11 - Dicionário de dados da tabela Passageiros .....	34
Tabela 12 - Operações da tabela Passageiros .....	34
Tabela 13 - Dicionário de dados da tabela Estatísticas.....	35
Tabela 14 - Operações da tabela Estatísticas. ....	35
Tabela 15 - Dicionário de dados das configurações .....	36
Tabela 16 - Operações da tabela Configurações .....	37
Tabela 17 - Dicionário de dados da tabela Alterações.....	38
Tabela 18 - Operações da tabela Alterações.....	38
Tabela 19 – Resultados dos testes de desempenho da aplicação para telemóvel .....	49
Tabela 20 – Resultados dos testes de experiência do utilizador da aplicação para telemóvel.....	49
Tabela 21 – Resultados dos testes de desempenho da aplicação para computador.....	49
Tabela 22 - Descrição do caso de uso "Alterar boleia" .....	55
Tabela 23 - Descrição do caso de uso "Entrar numa boleia" .....	56
Tabela 24 - Descrição do caso de uso "Repetir boleia" .....	57
Tabela 25 - Descrição do caso de uso "Sair de uma boleia" .....	58



# Glossário

IPG- Instituto Politécnico da Guarda

PHP- PHP Hypertext Preprocessor

AJAX- Asynchronous Javascript and XML

HTML- HyperText Markup Language

XML - Extensible Markup Language

CSS- Cascading Style Sheets

SGBD - Sistema de Gestão de Base de Dados

SQL - Structured Query Language

BD – Base de Dados

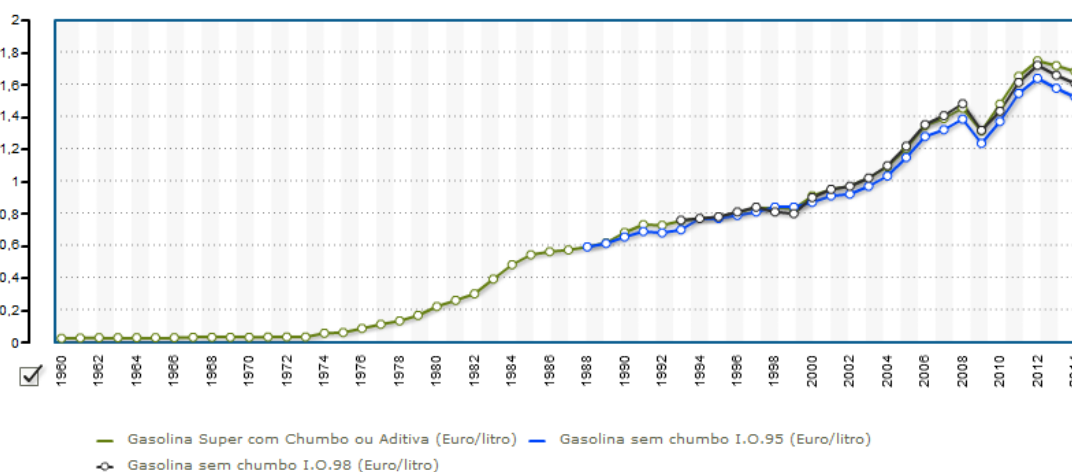
EER – Enhanced Entity-Relationship

## 1. Introdução

O presente relatório caracteriza o desenvolvimento do projeto, chamado CarPooling, realizado pelo aluno Vasco Manuel de Deus Lello Fortuna, no âmbito da unidade curricular Projeto de Informática, do 3º ano da Licenciatura em Engenharia Informática da Escola Superior de Tecnologia e Gestão do Instituto Politécnico da Guarda.

### 1.1.Contexto do projeto

O crescente aumento dos custos de combustíveis (uma subida de preço que de 0,02€/litro para 1,68€/litro entre 1960 e 2014 (Figura 1)) e mais recentemente portagens (portagens nas SCUT desde finais de 2011, para o caso da A25, por exemplo) tem fomentado uma crescente partilha de boleias regulares em viaturas nas deslocações entre cidades, nomeadamente entre professores, funcionários de fábricas e empresas, etc.



**Figura 1 - Custo dos combustíveis entre 1960 e 2014 em Portugal** (Fonte: <http://www.pordata.pt/Portugal/Pre%C3%A7os+m%C3%A9dios+de+venda+ao+p%C3%BAblico+dos+combust%C3%ADveis+l%C3%ADquidos+e+gasosos+E2%80%93+Continente-1265>)

Dado o potencial número de interessados nessas partilhas e as muitas tarefas de contacto, organização e gestão necessárias, existe uma cada vez maior procura de ferramentas informáticas direcionadas para este mercado.

Um desses grupos que faz partilhas de boleias entre os seus membros é constituído por um grupo de docentes do IPG. Atualmente, organizam e planeiam as boleias entre si, utilizando uma folha de cálculo partilhada no Google Spreadsheet. No entanto, esta utilização solução é um processo moroso e sujeito a erros, especialmente quando se pretendem organizar boleias a longo prazo, quando há alterações de última hora e é necessário avisar os colegas, quando há

## CAPÍTULO 1.INTRODUÇÃO

necessidade de distribuir os condutores pelas boleias,etc. Há também necessidade de melhorar a visualização otimizada das boleias semanais, a sua interação e os processos de configuração.

Existe uma grande variedade de ferramentas/aplicações online que fornecem o serviço de partilha de boleias entre utilizadores. No entanto, estas ferramentas não providenciam a troca de informações entre grupos de utilizadores, não permitem o planeamento fácil de múltiplas viagens quer a curto ou a longo prazo e obrigam o passageiro a pagar o condutor pela viagem, não permitindo a troca de boleias entre utilizadores.

Este projeto visa responder às necessidades do grupo, através de uma aplicação online que permita automatizar o planeamento e a organização de boleias e tornar a sua utilização rápida e eficaz por parte do utilizador. A utilização da aplicação vai se centralizar numa único mapa de boleias visível para todo o grupo. Este mapa consistirá num horário onde estão registadas todas as boleias e os seus atributos (hora, data, partida, etc). Existirá uma variedade de ferramentas e opções disponíveis aos utilizadores para conseguirem criar, aceitar, duplicar e customizar ofertas de boleias dentro deste mapa.

O design da aplicação terá de ser simples e eficiente, de modo ao utilizador conseguir ler o mapa de boleias rapidamente e sem esforço. Assim, o design focar-se-á numa vista semanal do mapa de boleias. Esta vista consistirá numa tabela preenchida com as horas do dia verticalmente e os dias da semana horizontalmente e cada espaço será preenchido consoante as boleias existentes. A cada utilizador será atribuído uma cor, que serão utilizadas para colorir cada boleia na tabela de modo a identificar rapidamente quem é o condutor.

Para maior usabilidade e acessibilidade, a aplicação será responsiva, ou seja, a aplicação web adaptará o seu formato consoante o tamanho do ecrã do dispositivo. Assim, pode ser visualizada a partir de qualquer dispositivo móvel sem perder eficiência do design.

### 1.2.Objetivos da aplicação

Tendo em conta os pontos referidos anteriormente, podemos estabelecer vários requisitos que a aplicação terá de cumprir:

1. Registrar, alterar e eliminar utilizadores. Esta é uma funcionalidade básica para permitir a gestão dos utilizadores.
2. Organizar pedidos e ofertas de boleias dentro do grupo. A aplicação tem de ter uma organização visual intuitiva ao utilizador, de modo a permitir conseguir localizar as boleias no tempo facilmente.
3. Contabilizar boleias efetuadas e recebidas. Este objetivo será aplicado em formato de estatísticas, de modo aos membros do grupo conseguirem ter maior controlo sobre as boleias que efetuam.
4. Duplicar mapas de boleias para semestres, anos. Conseguir copiar uma porção do mapa de boleia e colocá-lo noutra período de tempo.
5. Preencher mapas automaticamente de acordo com as boleias contabilizadas de cada membro. A aplicação deve conseguir criar boleias novas automaticamente de acordo com a relação de número de vezes que cada utilizador foi passageiro/condutor.
6. Enviar notificações por email acerca das próximas boleias ou alterações de boleias que afetam o utilizador.
7. Calcular a redução da pegada de carbono. Desta forma pode-se avaliar impacto das ações dos utilizadores da aplicação na luta contra o aquecimento global.
8. Exportar a BD das boleias para o Google Spreadsheet. Assim consegue-se ter um backup anual ou semestral das boleias para memória futura permitindo uma fácil consulta dos dados agregados num único mapa.
9. Criar cópias de segurança dos mapas de boleias.
10. Registrar as alterações mais relevantes à base de dados, para auditorias de segurança.
11. Interface eficiente e intuitiva. Nesta aplicação, é essencial fazer um grande número de ações num curto espaço de tempo de modo a que a aplicação se torne o mais útil possível.

### 1.3.Estrutura do relatório

Este relatório é constituído por 5 capítulos. No primeiro capítulo, é apresentado o problema do projeto e é feita uma breve introdução ao relatório. No segundo capítulo, é apresentado o estado de arte, contendo um estudo às aplicações existentes que se enquadram no tema do projeto. No terceiro capítulo é descrita a metodologia e a análise de requisitos. No quarto capítulo, as tecnologias utilizadas são descritas tal como alguns desafios que foram encontrados durante o desenvolvimento e a avaliação da aplicação. No quinto capítulo é mencionada a conclusão deste projeto e futuro desenvolvimento da aplicação.

## 2. Estado de arte

Este capítulo é dedicado à pesquisa de aplicações cujas funcionalidades se enquadram na área deste projeto, analisar os seus benefícios e suas limitações e se há alguma que se enquadre perfeitamente nos nossos objectivos. Não havendo pretende-se também obter informações e conhecimentos sobre aspetos que poderão ser benéficos para o desenvolvimento do projeto.

Todas as aplicações e informação relevantes a elas foram pesquisadas pelo autor. As aplicações foram escolhidas pelo número de ferramentas relevantes ao projeto e pela popularidade da aplicação. A pesquisa foi feita em motores de busca de websites e aplicações.

### 2.1. Análise de aplicações de boleias correntes

A análise das aplicações teve como pontos principais a interação dos vários utilizadores entre si através da própria aplicação e as funcionalidades disponíveis para os utilizadores. Foi estudado e observado, em cada aplicação, se era possível haver interação entre grupos de utilizadores e se existem funcionalidades que permitiam a criação e manipulação de boleias partilhadas entre vários utilizadores ao mesmo tempo.

As aplicações web estudadas focalizam-se numa interação singular do utilizador, ou seja, cada utilizador planeia e interage somente com as suas boleias, sem ter acesso direto ou facilitado às dos outros utilizadores. Apesar de existir alguma variedade nas funcionalidades em cada aplicação, todas elas focam nesta interação singular, como se poderá observar nas secções seguintes.

## Blablacar

Blablacar ([www.blablacar.pt](http://www.blablacar.pt)) é uma das aplicações web mais conhecidas para a partilha de boleias. A aplicação permite aos passageiros pesquisar por viagens, quer por ponto de partida quer por destino, previamente anunciadas pelos condutores. Os passageiros pagam ao condutor através da aplicação, na qual uma certa percentagem é revertida para a aplicação. Além disso, tanto o condutor como o carro são avaliados e comentados pelos passageiros. Estas avaliações servem como certificação da qualidade do condutor e da viagem aos futuros passageiros. Também contém ligação ao Facebook, de modo a ligar o perfil dos utilizadores ao Facebook.

Na tabela seguinte (Tabela 1), estão descritas as funcionalidades da aplicação Blablacar:

Funcionalidade	Sim	Não
Responsivo		X
Gratuito	X	
Pagamento em boleias		X
Open-source		X
Registo de utilizadores	X	
Duplicação de ofertas a longo prazo		X
Editar ofertas depois de anunciadas		X
Criação de mapas de boleias		X
Duplicação de pedidos		X
Reserva de viagens a longo prazo		X
Aplicação móvel	X	

Tabela 1 – Funcionalidades de blablacar

## Boleia.net

Boleia.net([www.boleia.net](http://www.boleia.net)) é uma plataforma de partilha de boleias. As boleias são anunciadas pelos condutores na aplicação, no entanto, a aplicação não gere a adesão dos passageiros às boleias. Para aderirem a uma boleia, os passageiros têm de contactar o condutor pessoalmente, através da informação que ele coloca na aplicação. O preço é colocado pelo condutor em cada boleia, sendo o pagamento feito pessoalmente entre os intervenientes. Os passageiros podem avaliar e comentar o condutor na aplicação. A aplicação utiliza as redes sociais como o Twitter e o Facebook para divulgar as boleias.

Na tabela seguinte (Tabela 2), estão descritas as funcionalidades da aplicação Boleia.net:

Funcionalidade	Sim	Não
Responsivo	X	
Gratuito	X	
Pagamento em boleias		X
Open-source		X
Registo de utilizadores	X	
Duplicação de ofertas a longo prazo		X
Editar ofertas depois de anunciadas		X
Criação de mapas de boleias		X
Duplicação de pedidos		X
Reserva de viagens a longo prazo		X
Aplicação móvel		X

Tabela 2 - Funcionalidades de boleia.net



## Pendura.pt

Pendura.pt ([www.pendura.pt](http://www.pendura.pt)) é uma plataforma mais simples que as anteriores. O utilizador (quer seja condutor ou passageiro) coloca um anúncio na aplicação juntamente com o seu contacto. O utilizador é posteriormente contactado pelos interessados que negociam os detalhes da boleia e do pagamento. Não existe nenhum tipo de avaliação dos condutores nem interação com as redes sociais.

Na tabela seguinte (Tabela 3), estão descritas as funcionalidades da aplicação Pendura.pt:

Funcionalidade	Sim	Não
Responsivo	X	
Gratuito	X	
Pagamento em boleias		X
Open-source		X
Registo de utilizadores		X
Duplicação de ofertas a longo prazo		X
Editar ofertas depois de anunciadas		X
Criação de mapas de boleias		X
Duplicação de pedidos		X
Reserva de viagens a longo prazo		X
Aplicação móvel		X

Tabela 3 - Funcionalidades de pendura.pt

## 2.2. Avaliação das aplicações

A tabela a seguir (Tabela 4) apresenta todas as funcionalidades das aplicações estudadas, comparando-as com a aplicação a ser desenvolvida, que se encontra na última coluna:

Funcionalidade	Blablacar	Boleianet	Pendura.pt	CarPooling
Responsivo	S	S	N	S
Gratuito	S	S	S	S
Pagamento em boleias	N	N	N	S
Open-source	N	N	N	S
Registo de utilizadores	S	S	N	S
Duplicação de ofertas a longo prazo	N	N	N	S
Editar ofertas depois de anunciadas	N	N	N	S
Criação de mapas de boleias	N	N	N	S
Duplicação de pedidos	N	N	N	S
Reserva de viagens a longo prazo	N	N	N	S
Aplicação móvel	N	N	S	N

**Tabela 4 - Comparação de todas as aplicações estudadas (S: Sim; N: Não)**

Apesar de constituírem sistemas de boleias, a interação destas aplicações é feita apenas entre o passageiro e o condutor. Esta interação vai contra o objetivo principal da aplicação que é a interação entre um grupo de amigos e a visualização de um único mapa de boleias entre eles todos. Dadas estas diferenças fundamentais, não se podem usar estas aplicações para o projeto proposto.

Apesar de também existirem grupos online a partilhar boleias entre si, estas partilhas são feitas em aplicações que não são especificadas para este propósito. Normalmente, estes grupos são criados em rede sociais (ex: Facebook). Dada a falta de uma aplicação que se direcione para o nosso nicho de mercado, o projeto será construído de raiz e não utilizará aplicações ou ferramentas existentes.

### 3. Metodologia e Análise de Requisitos

Neste capítulo é descrita a metodologia selecionada para o desenvolvimento da aplicação e a análise de requisitos do projeto.

#### 3.1.Metodologia

Durante todo o processo do projeto, será utilizada uma variante do desenvolvimento ágil: a programação extrema (XP). Esta metodologia foi selecionada de modo a obter uma maior interação com o orientador do projeto e, ao mesmo tempo, manter um ritmo de programação simples mas eficiente. O desenvolvimento XP caracteriza-se por etapas de desenvolvimento curtos, o que possibilita uma revisão frequente do projeto, de modo a aumentar produtividade e a introduzir pontos de referência, nos quais novos requerimentos podem ser adotados (1). Durante o desenvolvimento da aplicação, sempre que possível, o autor reuniu-se com o orientador semanalmente para rever o progresso da aplicação até aquele momento.

O XP focaliza-se em inicializar e construir o projeto com a solução mais simples e em adicionar funcionalidades extras mais tarde. Assim, ao dividir o processo de criação de software em várias iterações, minimiza os riscos de desenvolvimento de software. Cada iteração desta metodologia procura adicionar um conjunto de funcionalidades ao produto final e cada iteração contém as seguintes quatro fases de desenvolvimento:

1. Na fase de planeamento, discutiu-se e documentaram-se com o orientador todos os requisitos de software necessários para o produto final.
2. Na fase de design, foram desenhados e documentados os vários esquemas UML e as tecnologias necessárias para desenvolver a aplicação.
3. Na fase de codificação, desenvolveu-se o código para o projeto.
4. Na fase final de testes, testou-se a aplicação de modo a garantir que está de acordo com os requisitos de software e para garantir a sua coerência e usabilidade.

3.2.Planeamento

Durante a fase de planeamento, fez-se um plano da sequência de atividades a seguir durante o processo de desenvolvimento do projeto. No mapa de Gantt seguinte (Figura 2 - Mapa de Gantt efetivo), estão descritas as atividades planeadas e efetuadas. É de notar que o relatório foi sendo construído ao longo do processo de desenvolvimento e que a fase de testes se intercala com a fase de codificação porque estavam a ser corrigidos os erros que se encontravam nos testes.

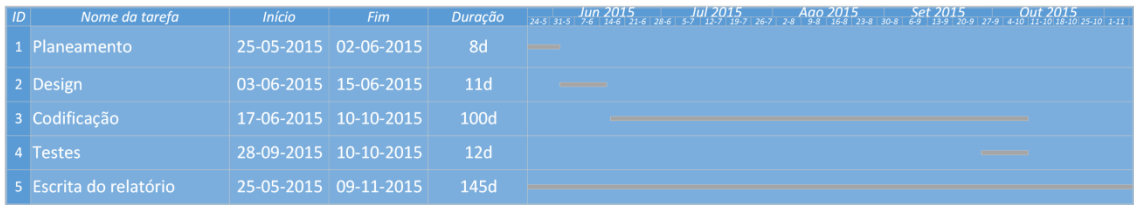


Figura 2 - Mapa de Gantt efetivo

### 3.3. Diagrama de contexto

O diagrama de contexto é composto por fluxos de dados que mostram as interfaces entre o sistema e as entidades externas (2). O seu principal objetivo é simplificar a interação de atores ou sistemas externos com a nossa aplicação. Neste caso, as únicas interações externas que temos são os utilizadores comuns e o administrador.

As ações dos utilizadores comuns estão focalizadas na interatividade com as boleias, como é demonstrado no diagrama abaixo (Figura 3). Cada utilizador vai ter acesso ao mapa de boleias e consegue utilizar a aplicação para inserir, alterar, eliminar e repetir as suas boleias no mapa de boleias. Cada utilizador também consegue colocar-se nas boleias de outros utilizadores como passageiros, bem como sair dessas boleias. Os utilizadores também vão ter a possibilidade de duplicar o mapa de boleias e sincronizar a base de dados com o Google Spreadsheet como foi mencionado anteriormente nas alíneas 4 e 8 da secção [1.2](#), respetivamente.

O administrador, além das permissões do utilizador comum, também poderá adicionar, alterar e eliminar utilizadores.

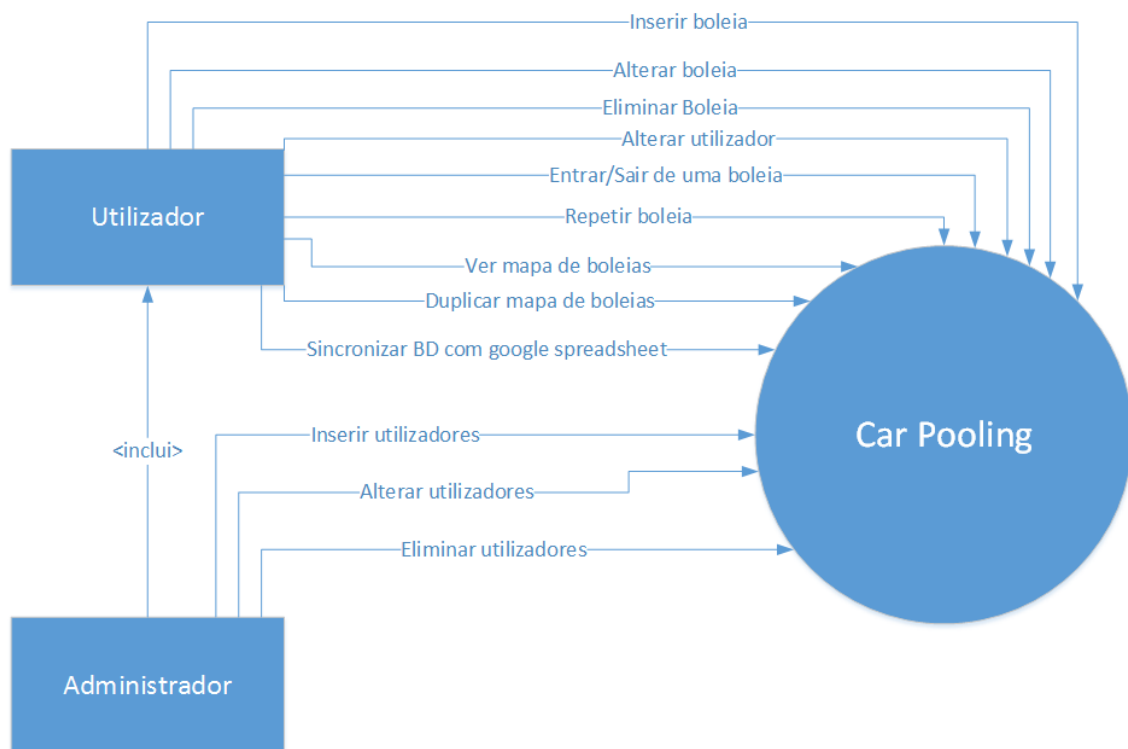


Figura 3 - Diagrama de Contexto

### 3.4.Diagrama de casos de uso

O diagrama de casos de uso descreve a funcionalidade proposta para um novo sistema que será projetado (3). Este diagrama mostra todas as funcionalidades do sistema e de que modo este interage com o utilizador. O diagrama de casos de uso para este projeto (Figura 2) tem em conta as interações descritas no diagrama de contexto anterior, mas também indica os casos de uso da própria aplicação.

O sistema irá contabilizar as boleias efetuadas e recebidas de cada utilizador, ou seja, sempre que ocorre uma entrada ou saída de um condutor ou passageiro de uma boleia, serão calculadas estatísticas para os utilizadores envolventes (de acordo com a alínea 3 do ponto [1.2](#) dos objetivos). O sistema também enviará emails, sempre que haja uma alteração de uma boleia, aos passageiros envolventes (alínea 6), criará cópias de segurança da base de dados regularmente (alínea 9) e registará as alterações dos utilizadores à base de dados (alínea 10).

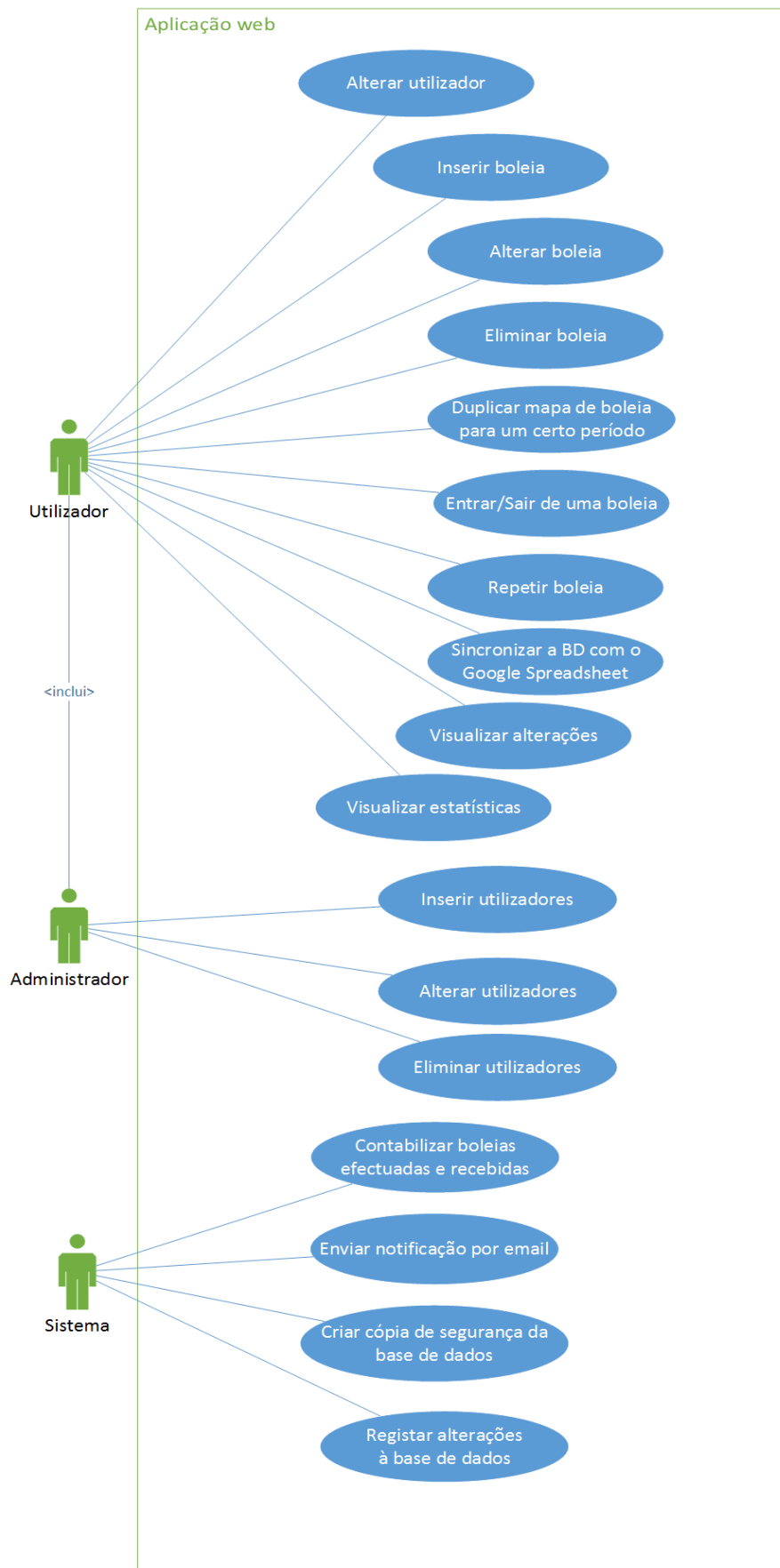


Figura 4 - Diagrama de Casos de Uso

### 3.5. Descrições de casos de uso

As descrições de caso de uso explicam detalhadamente como cada caso de uso irá funcionar e em que condições irá funcionar. As descrições apresentam detalhadamente passos sequenciais de como os casos de uso irão funcionar. Também apresentam cenários alternativos ao caso de uso e testes necessários para garantir a integridade do caso de uso. Nesta secção são apresentadas as descrições dos casos de uso mais relevantes do projeto, que são aquelas ligadas à gestão das boleias e dos utilizadores. O conjunto completo das descrições encontra-se no Anexo A.

#### 3.5.1. Inserir boeia

Este caso de uso é essencial ao bom funcionamento da aplicação. O cenário principal deste caso de uso requer o ator seleccionar o botão “Inserir Boleia” e preencher um formulário com os dados da boeia desejada.

O que faz este caso de uso diferente do habitual é a existência de um cenário alternativo onde o ator não precisa preencher nenhum formulário. Ao clicar num botão alternativo, localizado nos espaços vazios do mapa de boleias, o sistema coloca automaticamente uma boeia nesse espaço com as pré-configurações do ator (as configurações do ator são preenchidas durante o registo do utilizador na base de dados). Este processo todo está descrito na tabela abaixo (Tabela 5).



Nome	Inserir boleia
Objetivo	O ator insere uma boleia
Prioridade	Alta
Pré-condição	Login válido
Cenário principal	<ol style="list-style-type: none"> <li>1. O caso de uso começa quando o ator clica no botão “Inserir Boleia” ou quando seleciona um espaço vazio no mapa de boleias</li> <li>2. O sistema apresenta o formulário “Inserir boleia”</li> <li>3. O ator preenche os campos obrigatórios.</li> <li>4. O ator clica no botão “Ok”, confirmando os dados</li> <li>5. O sistema regista os dados</li> </ol>
Cenário alternativo	<p>O ator pode cancelar a operação a qualquer momento.</p> <ol style="list-style-type: none"> <li>1. a. No caso de selecionar um espaço vazio no mapa, a boleia é colocada automaticamente sem a necessidade de formulário. A boleia será inserida com os valores pré-configurados pelo utilizador.</li> <li>4. a. O ator não preenche todos os campos obrigatórios e aparece uma mensagem de erro.</li> <li>4. b. Se a sintaxe de algum campo estiver incorreta, mostra mensagem de erro.</li> </ol>
Pós-condição	Os dados estatísticos relevantes ao ator são atualizados.
Casos de teste	<ul style="list-style-type: none"> <li>• Verificar se o mapa de boleias é atualizado corretamente.</li> <li>• Verificar se as estatísticas são atualizadas corretamente.</li> <li>• Verificar se ao omitir campos obrigatórios, o sistema dá erro.</li> <li>• Verificar se os campos são preenchidos corretamente <ul style="list-style-type: none"> <li>○ Se os campos numéricos só contêm caracteres numéricos.</li> <li>○ Se os campos alfabéticos só contêm caracteres alfabéticos.</li> </ul> </li> </ul>

Tabela 5 - Descrição do caso de uso "Inserir Boleia"

### 3.5.2. Eliminar boleia

O que devemos notar deste caso de uso é o cenário alternativo. Se uma boleia fizer parte de uma repetição, o sistema faz uma pergunta adicional ao ator para eliminar todas as boleias ligadas a essa repetição.

Nome	Eliminar boleia
Objetivo	O ator elimina uma boleia
Prioridade	Baixa
Pré-condição	Login válido
Cenário principal	<ol style="list-style-type: none"> <li>1. O caso de uso começa quando o ator clica no botão “Eliminar” após selecionar uma boleia.</li> <li>2. O sistema pede a confirmação da eliminação.</li> <li>3. O ator clica no botão “Ok”, confirmando a eliminação.</li> <li>4. O sistema elimina a boleia.</li> </ol>
Cenário alternativo	<p>O ator pode cancelar a operação a qualquer momento.</p> <p>1.a Se a boleia pertencer a uma repetição, o sistema pergunta ao ator se ele quer eliminar as boleias associadas a essa repetição. Se ele confirmar, então no passo 4, são eliminadas todas as boleias ligadas a essa repetição.</p>
Pós-condição	Os dados estatísticos relevantes ao condutor e passageiros (se existirem) da boleia repetida são atualizados.
Casos de teste	<ul style="list-style-type: none"> <li>• Verificar se o mapa de boleias é atualizado corretamente.</li> <li>• Verificar se as estatísticas são atualizadas corretamente.</li> </ul>

Tabela 6 – Descrição do caso de uso “Eliminar Boleia”

### 3.6. Diagramas de sequência

Os diagramas de sequência mostram a interação entre o utilizador, a aplicação e as tabelas da base de dados por passos sequenciais. Os diagramas de sequência baseiam-se nas descrições de caso de uso e estabelecem a ponte entre as descrições e a base de dados. Cada seta representa uma ação entre os dois intervenientes ligados pela seta. Aqui são apresentados os diagramas de sequência mais relevantes do ao projeto. O conjunto completo dos diagramas de sequência encontram-se no Anexo B.

3.6.1. Inserir boleia

Para complementar a descrição de caso de uso Inserir Boleia (Tabela 5), é apresentado aqui o respetivo diagrama de sequência (Figura 5). Apresenta o cenário principal e o cenário alternativo do caso de uso, onde o utilizador seleciona um espaço vazio para inserir uma boleia.

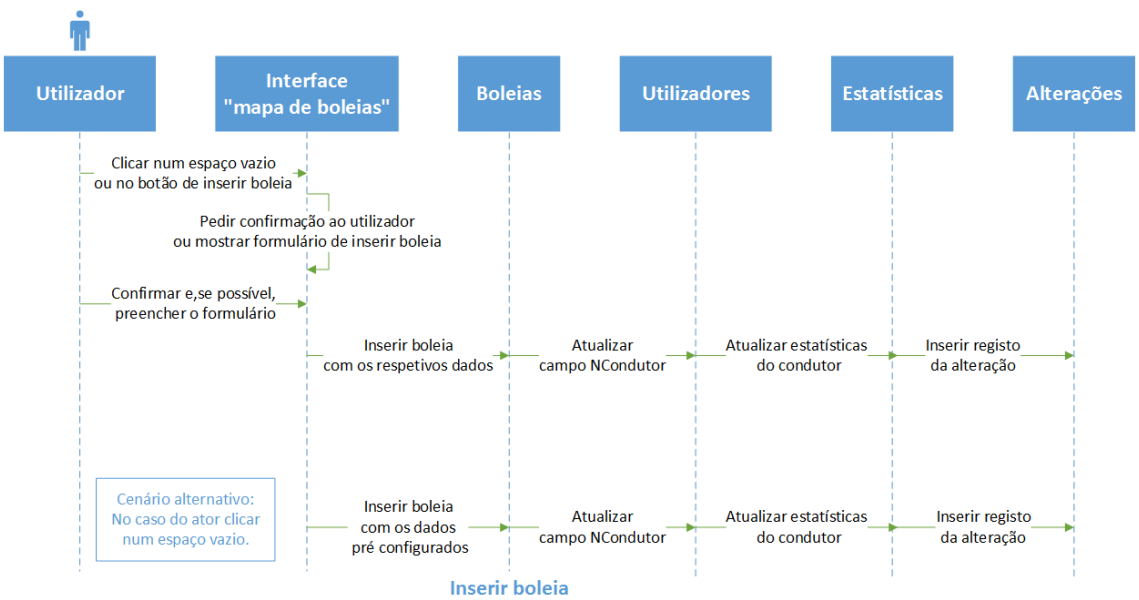


Figura 5- Diagrama de sequência "Inserir boleia"

### 3.6.2. Eliminar boleia

Para complementar a descrição do caso de uso Eliminar Boleia (Tabela 6), é apresentado o diagrama seguinte (Figura 6).

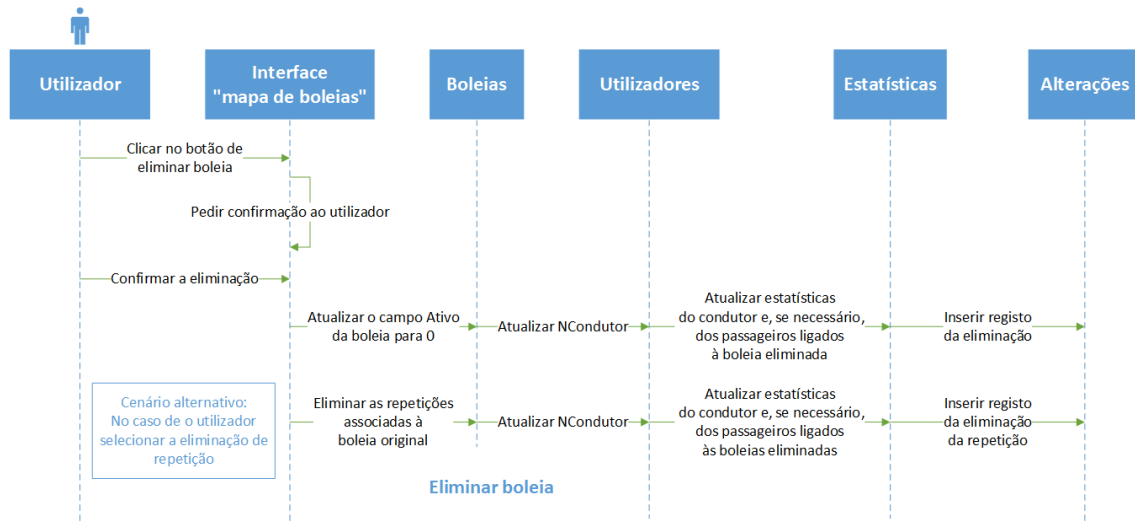


Figura 6 - Diagrama de sequência "Eliminar boleia"

Neste diagrama, é de notar o cenário alternativo onde o utilizador também seleciona a eliminação da repetição. Neste caso, todas as boleias associadas a esta repetição são eliminadas.

### 3.7. Diagrama de classes e modelo EER

Aqui é apresentado o diagrama de classes e respetivo o modelo EER (Enhanced Entity-Relationship model) desenvolvidos consoante os objetivos mencionados anteriormente. O diagrama de classes é constituído por seis classes, mas as principais são as dos utilizadores, passageiros e boleias (Figura 7). Estas irão guardar os registos dos utilizadores e das boleias, estabelecendo a interação entre elas.

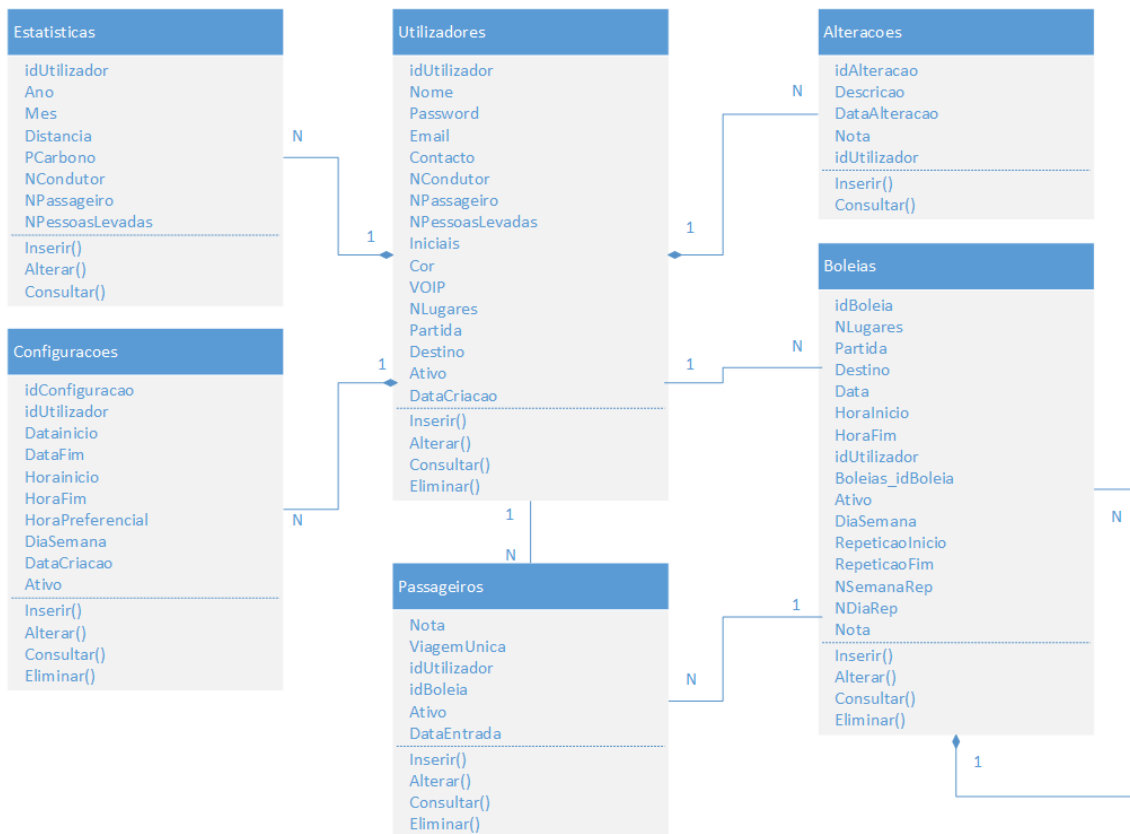


Figura 7 - Diagrama de classes

A classe dos utilizadores irá guardar os dados, estatísticas e as configurações de predefinição para a criação de boleias de cada utilizador (Partida, Destino, NLugares). À classe dos utilizadores estão ligadas as classes das alterações, estatísticas e configurações. A classe de alterações guarda registos das alterações mais importantes feitas por cada utilizador. A classe de estatísticas guarda estatísticas mensais de cada utilizador, utilizando uma chave composta que inclui os atributos ano e mes (que guardam o ano e o mês de uma estatística) e a chave estrangeira do utilizador a qual se aplica a estatística. Assim, garante-se que não existem registos duplicados para aquele mês e utilizador. A classe de configurações guarda as configurações necessárias para a automatização de escolha do condutor. Esta classe simplesmente guarda os dados para a criação de uma boleia caso o utilizador seja escolhido como condutor pela aplicação.

A classe de boleias irá guardar os dados essenciais à boleia e os dados relacionados à repetição de boleias. A classe de boleias está ligada à dos utilizadores para indicar o condutor da boleia e está ligada a si mesma para que, no caso de repetição de boleias, se consiga indicar quem é a boleia original da repetição. Também existe um atributo Nota no caso do utilizador querer colocar uma nota sobre a boleia para os outros utilizadores. É de notar que o atributo DiaSemana se encontra desnormalizado para acesso facilitado ao dia da semana em que a boleia se encontra.

### CAPÍTULO 3. METODOLOGIA E ANÁLISE DE REQUISITOS

A classe dos passageiros vai guardar uma chave composta. Esta chave é formada pela chave estrangeira dos utilizadores e das boleias para indicar quais são os passageiros de cada boeia. Tal como a classe de boleias, a classe passageiros tem uma atributo Nota.

A implementação do diagrama de classes no modelo relacional originou o EER apresentado na Figura 8.

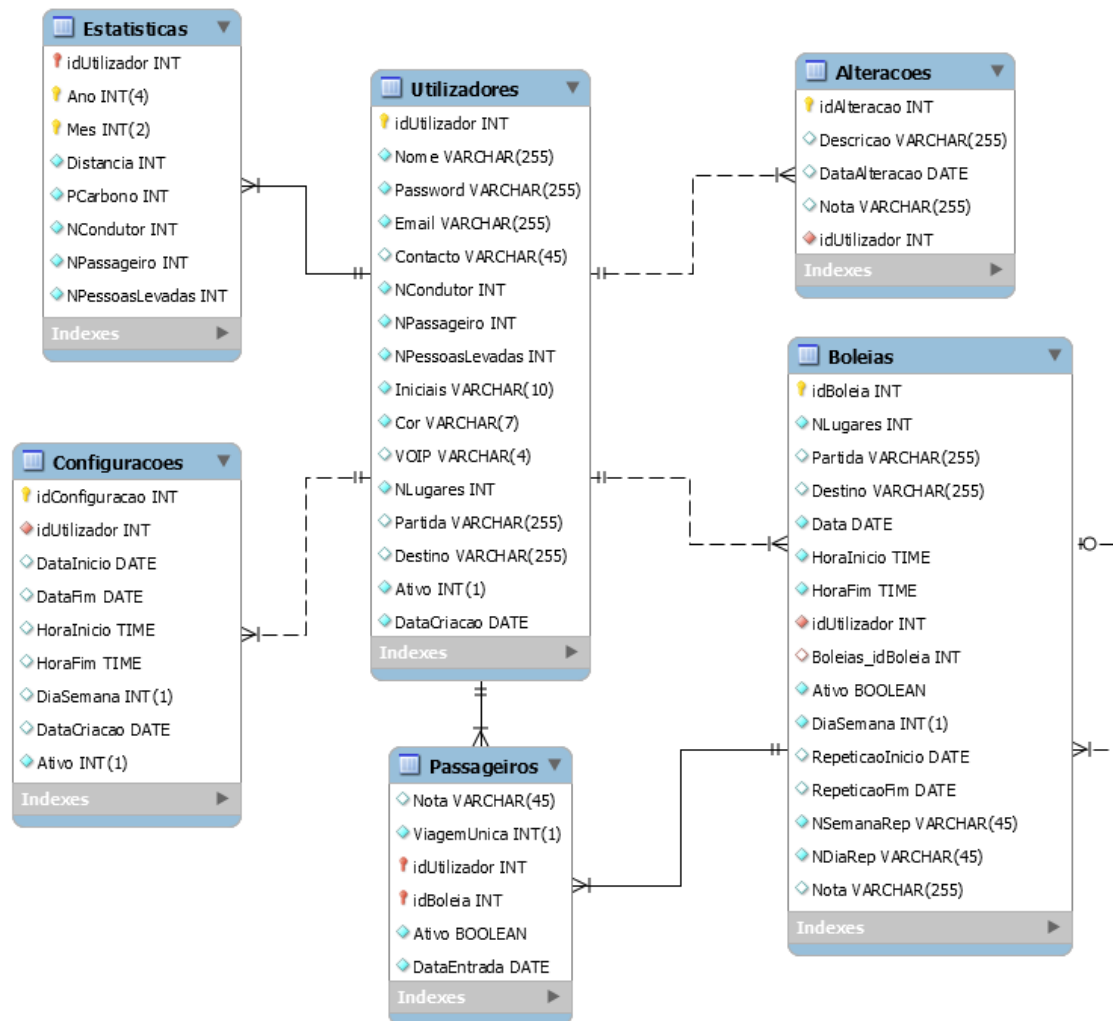


Figura 8 - Modelo EER criado em MySQL Workbench

### 3.8. Semântica

Nesta secção é apresentado o dicionário de dados e a tabela de operações às tabelas da base de dados do projeto.

#### 3.8.1. Utilizadores

Dicionário de dados:

Nome do campo	Tipo de dados	Descrição	Restrições
<b>IdUtilizador</b>	Int	(PK) Id do utilizador	PK
<b>Nome</b>	Varchar(255)	Nome do utilizador	Não nulo.
<b>Password</b>	Varchar(255)	Password do utilizador encriptada com md5	Não nulo.
<b>Email</b>	Varchar(255)	Email do utilizador	Não nulo. Único.
<b>Contacto</b>	Varchar(45)	Telemóvel/telefone do utilizador	Não nulo.
<b>NCondutor</b>	Int	Nº de vezes que o utilizador foi condutor	Não nulo.
<b>NPassageiro</b>	Int	Nº de vezes que o utilizador foi passageiro	Não nulo.
<b>NPessoasLevadas</b>	Int	Nº de passageiros que o utilizador levou	Não nulo.
<b>Iniciais</b>	Varchar(10)	Iniciais do nome do utilizador, de modo a identificar o utilizador, ocupando pouco espaço quando está a ser mostrado na página.	Não nulo.
<b>Cor</b>	Varchar(45)	Cor identificadora do utilizador. Esta cor é utilizada para pintar as boleias na aplicação consoante o condutor da boeia, de modo a identificar rapidamente o condutor.	Não nulo.
<b>VOIP</b>	Varchar(45)	VOIP do utilizador	
<b>NLugares</b>	Int	Nº de passageiros que o utilizador pode levar, por defeito.	Não nulo.
<b>Partida</b>	Varchar(255)	Este campo indica o lugar de onde,	

		habitualmente, o condutor inicia a viagem.	
<b>Destino</b>	Varchar(255)	Este campo indica o lugar onde, habitualmente, o condutor termina a viagem.	

Tabela 7 - Dicionário de dados da tabela Utilizadores



## Operações

Nome	Descrição
<b>Inserir()</b>	<p>Operação que permite inserir um utilizador</p> <ol style="list-style-type: none"> <li>1. Introduzir Nome, Password, Email, Contacto, Iniciais, Cor, VOIP, Nlugares, Partida e Destino</li> <li>2. O sistema gera o IdUtilizador</li> </ol>
<b>Alterar()</b>	<p>Operação que permite alterar um utilizador</p> <ol style="list-style-type: none"> <li>1. Selecionar um utilizador</li> <li>2. Alterar os campos necessários e que forem permitidos</li> <li>3. Atualizar o utilizador</li> </ol>
<b>Consultar()</b>	<p>Operação que permite consultar um utilizador</p> <ol style="list-style-type: none"> <li>1. Selecionar um utilizador</li> <li>2. O sistema mostra os detalhes desse utilizador</li> </ol>
<b>Eliminar()</b>	<p>Operação que permite eliminar um utilizador</p> <ol style="list-style-type: none"> <li>1. Selecionar um utilizador</li> <li>2. O sistema pede a confirmação da eliminação</li> <li>3. Eliminar o utilizador</li> </ol>

Tabela 8 - Operações da tabela Utilizadores

## 3.8.2. Boleias

Dicionário de dados:

Nome do campo	Tipo de dados	Descrição	Restrições
<b>IdBoleia</b>	Int	(PK) Id da boleia	PK
<b>NLugares</b>	Int	Nº de passageiros que podem aderir à boleia	Não nulo.
<b>Partida</b>	Varchar(255)	Partida da boleia	
<b>Destino</b>	Varchar(255)	Destino da boleia	
<b>Data</b>	Date	Em que dia a boleia se realizará	Não nulo.
<b>HoraInicio</b>	Time	Hora em que a boleia começa	Não nulo.
<b>HoraFim</b>	Time	Hora em que a boleia termina	Não nulo.
<b>IdUtilizador</b>	Int	(FK) Id do condutor da boleia. Relacionada com o idUtilizador da tabela utilizadores.	FK
<b>Boleias_IdBoleia</b>	Int	(FK) Id da boleia pai, no caso de repetição de boleias. Relacionada com o idBoleia da tabela boleias.	FK.
<b>Ativo</b>	Boolean	Indica se a boleia está ativa	Não nulo.
<b>DiaSemana</b>	Int(1)	Dia da semana da boleia	Não nulo.
<b>RepeticaoInicio</b>	Date	Se houver repetição de boleias, o dia em que ela começa	
<b>RepeticaoFim</b>	Date	Se houver repetição de boleias, o dia em que ela termina	
<b>NSemanaRep</b>	Varchar(45)	No caso de a repetição ser mensal, a semana em que se repete.	Não nulo.
<b>NDiaRep</b>	Varchar(45)	No caso de a repetição ser semanal ou mensal, o dia em que se repete.	Não nulo.
<b>Nota</b>	Varchar(255)	Nota adicional da boleia, preenchida pelo condutor	

Tabela 9 - Dicionário de dados da tabela Boleias

Operações:

Nome	Descrição
<b>Inserir()</b>	Operação que permite inserir uma boleia <ol style="list-style-type: none"> <li>1. Introduzir data, idutilizador, horainicio, horafim</li> <li>2. O sistema gera o IdBoleia</li> </ol>
<b>Alterar()</b>	Operação que permite alterar uma boleia <ol style="list-style-type: none"> <li>1. Selecionar uma boleia</li> <li>2. Alterar os campos necessários e que forem permitidos</li> <li>3. Atualizar a boleia</li> </ol>
<b>Consultar()</b>	Operação que permite consultar uma boleia <ol style="list-style-type: none"> <li>1. Selecionar uma boleia</li> <li>2. O sistema mostra os detalhes dessa boleia</li> </ol>
<b>Eliminar()</b>	Operação que permite eliminar uma boleia <ol style="list-style-type: none"> <li>1. Selecionar uma boleia</li> <li>2. O sistema pede a confirmação da eliminação</li> <li>3. Atualizar o campo Ativo para 0</li> </ol>

Tabela 10 - Operações da tabela Boleias

## 3.8.3. Passageiros

Dicionário de dados:

Nome do campo	Tipo de dados	Descrição	Restrições
<b>IdUtilizador</b>	Int	(PK/FK) Id do utilizador. Relacionada com o idUtilizador da tabela utilizadores.	PK/FK..
<b>IdBoleia</b>	Int	(PK/FK) Id da boleia. Relacionada com o idBoleia da tabela boleias.	PK/FK
<b>ViagemUnica</b>	Int(1)	Indica se é ida e/ou volta.	Não nulo
<b>Ativo</b>	Boolean	Indica se o passageiro está ativo	Não nulo.
<b>Nota</b>	Varchar(255)	Nota adicional, preenchida pelo passageiro	
<b>DataEntrada</b>	Date	Data que indica quando o passageiro entrou na boleia	Não nulo.

Tabela 11 - Dicionário de dados da tabela Passageiros

Operações:

Nome	Descrição
<b>Inserir()</b>	Operação que permite inserir um passageiro. <ol style="list-style-type: none"> <li>1. Introduzir idutilizador, idboleia, viagemunica, nota.</li> <li>2. O sistema gera o IdPassageiro.</li> </ol>
<b>Consultar()</b>	Operação que permite consultar um passageiro. <ol style="list-style-type: none"> <li>1. Selecionar um passageiro.</li> <li>2. O sistema mostra os detalhes desse passageiro.</li> </ol>
<b>Eliminar()</b>	Operação que permite eliminar um passageiro. <ol style="list-style-type: none"> <li>1. Selecionar um passageiro.</li> <li>2. O sistema pede a confirmação da eliminação.</li> <li>3. Atualizar campo Ativo para 0.</li> </ol>

Tabela 12 - Operações da tabela Passageiros

## 3.8.4. Estatísticas

Dicionário de dados:

Nome do campo	Tipo de dados	Descrição	Restrições
<b>IdUtilizador</b>	Int	(PK/FK) Id do utilizador. Relacionada com o idUtilizador da tabela utilizadores.	PK/FK.
<b>Ano</b>	Int(4)	(PK) Ano da estatística	PK.
<b>Mes</b>	Int(2)	(PK) Mês da estatística	PK.
<b>Distancia</b>	Int	Distância que o utilizador viajou durante o mês.	Não nulo.
<b>PCarbono</b>	Int	Pegada de carbono. Indica o dióxido de carbono que o utilizador salvou este mês.	Não nulo.
<b>NCondutor</b>	Int	Nº de vezes que o utilizador foi condutor durante o mês.	Não nulo.
<b>NPassageiro</b>	Int	Nº de vezes que o utilizador foi passageiro durante o mês.	Não nulo.
<b>NPessoasLevadas</b>	Int	Nº de pessoas que o utilizador levou este mês.	Não nulo.

Tabela 13 - Dicionário de dados da tabela Estatísticas.

Operações:

Nome	Descrição
<b>Inserir()</b>	Operação que permite inserir uma estatística. 1. Introduzir idutilizador, ano, mes, distancia, PCarbono, NCondutor, NPassageiro, NPessoasLevadas.
<b>Consultar()</b>	Operação que permite consultar uma estatística. 1. Selecionar uma estatística. 2. O sistema mostra os detalhes dessa estatística.
<b>Alterar()</b>	Operação que permite alterar uma estatística. 1. Selecionar uma estatística. 2. Alterar os campos permitidos. 3. Atualizar a estatística.

Tabela 14 - Operações da tabela Estatísticas.

## 3.8.5. Configurações

Dicionário de dados:

Nome do campo	Tipo de dados	Descrição	Restrições
<b>idConfiguracao</b>	Int	(PK) Id da configuração	PK.
<b>idUtilizador</b>	Int	(FK) Id do utilizador a qual se refere a configuração. Relacionada com o campo idUtilizador da tabela utilizadores.	FK.
<b>DataInicio</b>	Date	Data inicial, a partir da qual se pode aplicar a configuração.	
<b>DataFim</b>	Date	Data final, a partir da qual não se pode aplicar a configuração.	
<b>HoraInicio</b>	Time	Hora de início da boleia configurada.	
<b>HoraFim</b>	Time	Hora final da boleia configurada.	
<b>DiaSemana</b>	Int(1)	Dia da semana da boleia configurada.	
<b>DataCriacao</b>	Date	Data de criação da configuração.	
<b>Ativo</b>	Int(1).	Indica se a configuração está ativa ou não.	Não nulo.

Tabela 15 - Dicionário de dados das configurações

Operações:

Nome	Descrição
<b>Inserir()</b>	<p>Operação que permite inserir uma configuração.</p> <ol style="list-style-type: none"> <li>1. Introduzir idUtilizador, HoraInicio, HoraFim, DataInicio, DataFim, DiaSemana, DataCriacao.</li> <li>2. O sistema gera o idConfiguracao</li> </ol>
<b>Consultar()</b>	<p>Operação que permite consultar uma configuração.</p> <ol style="list-style-type: none"> <li>1. Selecionar uma configuração.</li> <li>2. O sistema mostra os detalhes dessa configuração.</li> </ol>
<b>Alterar()</b>	<p>Operação que permite alterar uma configuração.</p> <ol style="list-style-type: none"> <li>1. Selecionar uma configuração.</li> <li>2. Alterar os campos permitidos.</li> <li>3. Atualizar a configuração.</li> </ol>
<b>Eliminar()</b>	<p>Operação que permite eliminar uma configuração.</p> <ol style="list-style-type: none"> <li>1. Selecionar uma configuração.</li> <li>2. O sistema pede a confirmação da eliminação.</li> <li>3. Atualizar campo Ativo para 0.</li> </ol>

Tabela 16 - Operações da tabela Configurações

## 3.8.6. Alterações

Dicionário de dados:

Nome do campo	Tipo de dados	Descrição	Restrições
<b>idAlteracao</b>	Int	(PK) Id da configuração	PK.
<b>Descricao</b>	Varchar(255)	Descrição da alteração onde se encontram os detalhes do evento registado.	
<b>DataAlteracao</b>	Date	Data em que se efetuou a alteração.	
<b>Nota</b>	Varchar(255)	Nota do utilizador sobre a alteração.	
<b>IdUtilizador</b>	Int	(FK) Id do utilizador a qual se refere a alteração. Relacionada com o idUtilizador da tabela utilizadores.	Não nulo. FK.

Tabela 17 - Dicionário de dados da tabela Alterações

Operações:

Nome	Descrição
<b>Inserir()</b>	Operação que permite inserir uma alteração. <ol style="list-style-type: none"> <li>1. Introduzir Descricao, DataAlteracao, Nota, idUtilizador.</li> <li>2. O sistema gera o idAlteracao.</li> </ol>
<b>Consultar()</b>	Operação que permite consultar uma alteração. <ol style="list-style-type: none"> <li>1. Selecionar uma alteração.</li> <li>2. O sistema mostra os detalhes dessa alteração.</li> </ol>

Tabela 18 - Operações da tabela Alterações



## 4. Tecnologias utilizadas

Neste capítulo serão apresentadas as tecnologias utilizadas para o desenvolvimento da aplicação.

### 4.1.Tecnologias utilizadas

Nesta subsecção são apresentadas as tecnologias utilizadas para o desenvolvimento da aplicação.

#### 4.1.1. HTML

HTML (abreviação para a expressão inglesa HyperText Markup Language, que significa Linguagem de Marcação de Hipertexto) é uma linguagem de marcação utilizada para produzir páginas na Web. Documentos HTML podem ser interpretados por navegadores Web (4). HTML é o bloco de construção mais básico de uma página Web e é utilizado para criar e visualmente representar uma página web. Ele determina os conteúdos de uma página Web mas não a sua funcionalidade, ao contrário de linguagens como PHP e JavaScript que focam na interatividade do site com o utilizador em vez do seu conteúdo.

Esta linguagem tornou-se no padrão e base para criação de qualquer página Web. Como se decidiu desenvolver a aplicação desejada em página Web, esta linguagem tem de obrigatoriamente ser usada.

O autor também já tem experiência em usar esta linguagem não só de projetos anteriores realizados mas também de matérias estudadas nas unidades curriculares de programação para a Web, especificamente em Tecnologias de Internet e Programação para a Internet. Pelo que a utilização desta linguagem não será um desafio, no entanto ela será estudada e revista, de modo não só a garantir a integridade da aplicação desenvolvida mas também, possivelmente, a introduzir novas boas práticas de desenvolvimento e novos conceitos.

#### 4.1.2. CSS

O CSS (abreviação para a expressão inglesa Cascading Style Sheets, que significa Folhas de Estilos em Cascata) é uma linguagem de folhas de estilo utilizada para definir a apresentação de documentos escritos numa linguagem de marcação, como HTML ou XML. O seu principal benefício é fornecer a separação entre o formato e o conteúdo de um documento (5).

O CSS permite criar e alterar conjuntos de propriedades de estilo. Estas propriedades alteram o design e a visualização de qualquer elemento HTML, seguindo um conjunto de regras impostas

pelo programador. Assim, com esta linguagem, conseguem-se criar layouts e designs específicos de modo eficiente.

Especificamente, vai ser utilizado um template de CSS chamado Bootstrap. Este template está pré-configurado para alterar os elementos básicos de HTML não só para designs mais complexos, mas também para tornar páginas web responsivas. Esta responsividade é responsável por modificar a interface da página de modo a adaptar-se ao ecrã do dispositivo. Assim, a página consegue ser utilizada e visualizada de modo intuitivo e eficiente quer nos pequenos ecrãs dos telemóveis quer nos maiores ecrãs dos computadores.

### 4.1.3. PHP

PHP (PHP: Hypertext Preprocessor, originalmente Personal Home Page, significa pré-Processador de Hipertexto PHP) é uma linguagem interpretada livre, usada originalmente apenas para o desenvolvimento de aplicações presentes e interativas no lado do servidor, capazes de gerar o conteúdo dinâmico nas páginas Web (6). Há muitas outras linguagens de programação para a Web, tais como ASP.Net, Java, Python. No entanto, decidiu-se utilizar o PHP pelas seguintes razões:

1. A experiência do autor nesta linguagem de programação. O autor teve várias disciplinas dedicadas à programação na internet, incluindo Programação em PHP. Graças a estas disciplinas, o autor conseguiu uma introdução à linguagem e ao seu funcionamento.
2. A enorme quantidade de informação atualizada que existe sobre a linguagem. Sendo uma linguagem bastante conhecida e utilizada, PHP está constantemente a ser discutido e desenvolvido. Consequentemente, conseguimos encontrar bastante informação em livros, artigos e em documentação oficial em como trabalhar e utilizar a linguagem efetivamente.

### 4.1.4. Javascript

Javascript é considerada uma das três linguagens essenciais para o desenvolvimento Web (sendo as outras duas HTML e CSS). Javascript é a principal linguagem responsável pelo comportamento das páginas web, sendo capaz de tornar as páginas web dinâmicas e interativas para o utilizador.

Tendo em conta que parte dos objetivos desta aplicação tem a ver com a eficiência do interface e a rapidez de resposta por parte da aplicação (pontos 12 e 13 da secção [1.2](#)), conseguimos concluir que Javascript é uma ferramenta fulcral para o desenvolvimento desta aplicação. Adicionalmente, sendo uma das linguagens mais conhecidas para programação, além de

continuar em constante desenvolvimento, também se consegue encontrar com facilidade muita informação, exemplos, boas práticas e tutoriais sobre ela.

### 4.1.5. AJAX

O AJAX (Asynchronous JavaScript and XML, que significa Javascript e XML Assíncrono) não é uma ferramenta de programação, mas sim uma técnica, que utiliza as tecnologias Javascript e XML, para criar páginas web dinâmicas, assemelhando-se a aplicações Desktop.

Ao contrário das páginas web clássicas, o AJAX permite que as páginas web sejam carregadas assincronamente, trocando apenas pequenas porções de dados entre o servidor, sem que a página seja completamente recarregada (7). Deste modo, consegue-se melhorar a interação do utilizador (maior velocidade de resposta e menos redesenho da página) com a aplicação ao trocar só a informação necessária entre o servidor e o cliente.

Este processo de troca de informações ocorre da seguinte maneira: quando um evento especificado pelo programador ocorre, é criado um objeto XMLHttpRequest que é enviado para o servidor. Seguidamente, o servidor processa o pedido e envia os dados processados de volta para o navegador. Finalmente, estes dados são processados pelo Javascript, que só atualiza o conteúdo específico da página web.

O AJAX torna-se numa solução para aumentar a performance e resposta entre o cliente e o servidor, pelo que a interação do cliente com a aplicação será programada em AJAX.

### 4.1.6. MySQL

O MySQL é um sistema de gestão de base de dados (SGBD) que utiliza a linguagem SQL (Linguagem de Consulta Estruturada, do inglês Structured Query Language) como interface standard de acesso aos dados. É gratuito, possui facilidades para certas operações Web, como seja a paginação dos resultados das consultas, e é atualmente um dos bancos de dados mais populares nas aplicações Web, com mais de 10 milhões de instalações pelo mundo (8).

Para além do seu grande desempenho e estabilidade, o MySQL tem grande compatibilidade com o PHP, tendo este um módulo de interface próprio. Este módulo de interface, contendo funções nativas preparadas de raiz para lidar com conexões de base de dados e manipulação de dados, tem-se mantido regularmente atualizado e testado. Isto não só garante um nível de segurança mais alto comparando a outros SGBDs, mas também uma maior eficácia a programar, para além do acesso a uma grande quantidade de documentação disponível.

Todos os pontos anteriormente referidos, levaram ao autor escolher MySQL em relação às outras SGBDs gratuitos disponíveis, , como seja o PostgreSQL.

### 4.1.7. GitHub

O GitHub é um repositório web Git que oferece controlo de revisões distribuído e a funcionalidade de gestão de código-fonte do Git, com as suas próprias funcionalidades. As funções da plataforma GitHub, as aplicações Desktop, e o GitHub Enterprise tornam a tarefa de escrever código muito mais fácil para indivíduos e equipas (9).

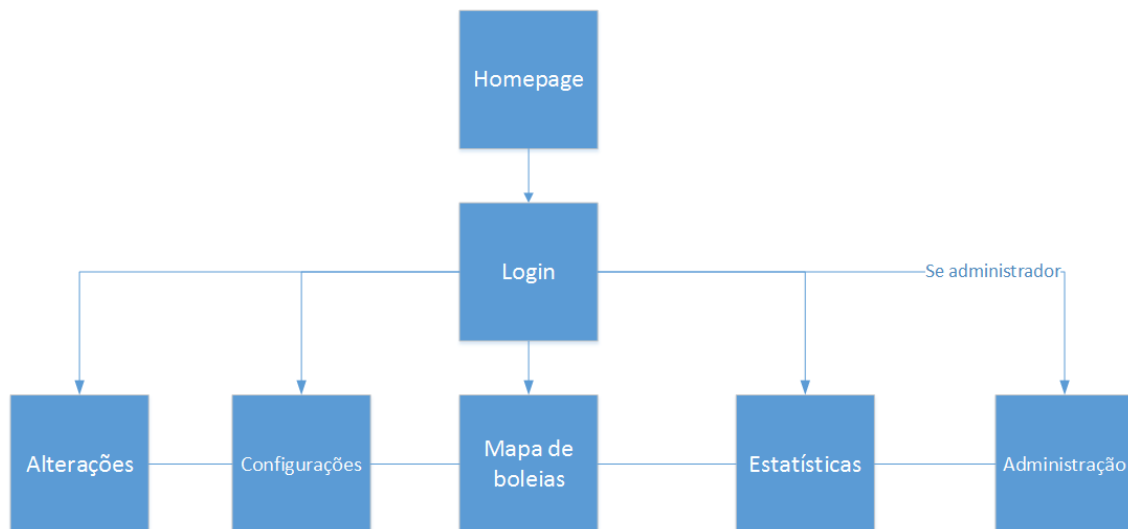
O GitHub vai ser utilizado como gestor de versões e como backup da aplicação em desenvolvimento. O GitHub foi escolhido pelo autor pela sua experiência com a aplicação e pela simplicidade que o GitHub oferece, apesar de haver outras soluções, como seja o SVN.

## 5. Desenvolvimento

Nesta secção são apresentados o diagrama de hierarquia da aplicação, os desafios de maior complexidade tiveram que ser superados com as respetivas soluções e a avaliação do produto desenvolvido.

### 5.1.Diagrama de hierarquia

O seguinte diagrama apresenta a hierarquia das várias interfaces de cada página. Ao iniciar a aplicação, o utilizador terá acesso à página inicial. A partir desta, o utilizador pode entrar na página de login que lhe dará acesso, depois do login efetuado corretamente, a uma rede de páginas que lhe dará uma série de funcionalidades e interfaces para manipular e visualizar o mapa de boleias.



**Figura 9 - Diagrama de Hierarquia**

Aqui são descritos os ecrans do diagrama de hierarquia:

1. **Homepage:** ecran inicial da aplicação. Contém o link para encaminhar o utilizador para a página de login e uma descrição do que é o carpooling e os objetivos da aplicação.
2. **Login:** ecran para o utilizador entrar na aplicação. Tem dois campos para o utilizador colocar o email e password correspondentes. Se o utilizador fizer o login com sucesso, é encaminhado para a página do mapa de boleias. Este ecran existe como medida de segurança para negar o acesso a qualquer utilizador não autorizado. Se o utilizador tenta aceder a outro ecran, exceto o ecran de Homepage, sem o login efetuado, é redireccionado para o ecran de login.
3. **Alterações:** ecran dos registos de alterações relevantes dos utilizadores. Contém os registos por parte de todos os membros da aplicação. Contém links para redireccionar o

utilizador para os outros ecrans, excepto o de login e o de administração se não for administrador.

4. Configurações: ecran de configuração da conta do utilizador. Aqui o utilizador pode alterar as configurações da sua conta. Contém os mesmos links de redireccionamento que o ecran de alterações.
5. Mapa de boleias: ecran do mapa de boleias. O utilizador só consegue aceder a esta página com o login efetuado. Contém o mapa de boleias, uma lista dos membros da aplicação e as funcionalidades para interagir com o mapa de boleias. Contém os mesmos links de redireccionamento que o ecran de alterações.
6. Estatísticas: contém as estatísticas mensais de cada utilizador. Neste ecran, o utilizador pode ver mais detalhadamente as estatísticas de cada utilizador. Contém os mesmos links de redireccionamento que o ecran de alterações.
7. Administração: contém as funcionalidades do administrador. Este ecran só pode ser acedido com o login do administrador, contendo as funcionalidades para o administrador que incluem adicionar, alterar e eliminar utilizadores. Estas funcionalidades só foram dadas ao administrador para controlar quem utiliza a aplicação. Contém os mesmos links de redireccionamento que o ecran de alterações.

### 5.2.Análise e implementação

Durante a análise e implementação da aplicação, foram superados vários desafios, dos quais se destacam dois. Esta secção será utilizada para apresentar e descrever os pontos principais destes desafios e as respetivas soluções.

#### 5.2.1. Visualização dos registos de boleias num único mapa de boleias

Problema – Visualização gráfica e intuitiva do mapa das boleias:

Existem registos de boleias com data, hora inicial, hora final e outros dados associados a elas. A aplicação tem de conseguir ler os registos das boleias para um determinado período de tempo (neste caso, vamos considerar uma semana) e colocá-los de forma organizada numa única página Web, de modo a que o utilizador consiga, facilmente, identificar as boleias existentes. Os pontos mais exigentes neste desafio são:

1. A criação de um mapa intuitivo utilizando elementos e estrutura HTML.
2. Posição e tamanho da boeia no mapa consoante a sua data, hora inicial e hora final.
3. A adaptação do mapa a boleias que partilhem o mesmo período de tempo (boleias sobrepostas).

Solução:

Em resposta à criação de um mapa intuitivo utilizando elementos e estrutura HTML (ponto 1 do problema), decidiu-se fazer um mapa de boleias com o aspeto visual de um horário semanal, ou seja, uma tabela em que as colunas indicam o dia da semana (segunda a sexta) e as linhas indicam as horas em intervalos de meia hora (ex: 8:00-8:30). A interface pode ser observada na Figura 10.

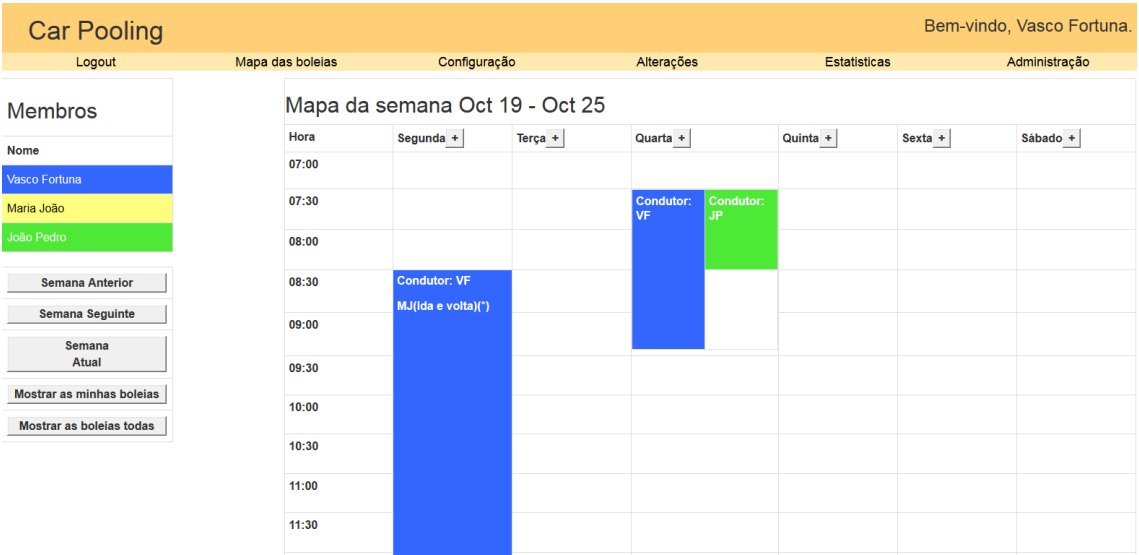


Figura 10 - Interface da aplicação CarPooling

Em termos de implementação, há alguns desafios para superar. Para calcular a posição de cada boleia na estrutura HTML, temos de ter em conta que o HTML lê as tabelas linha a linha, ou seja, ele lê cada célula da tabela (vamos chamar célula à junção de cada coluna com cada linha da tabela) da esquerda para a direita e de cima para baixo, como podemos ver na Figura 11. Isto é um ponto importante porque, como se decidiu colocar as horas em cada linha, significa que temos de averiguar a cada hora do horário se existem boleias à medida que a tabela é lida.

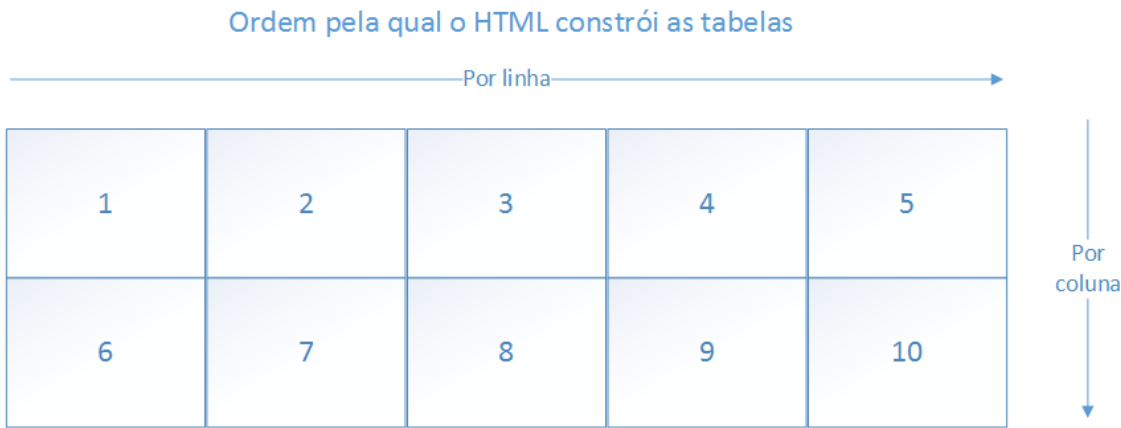
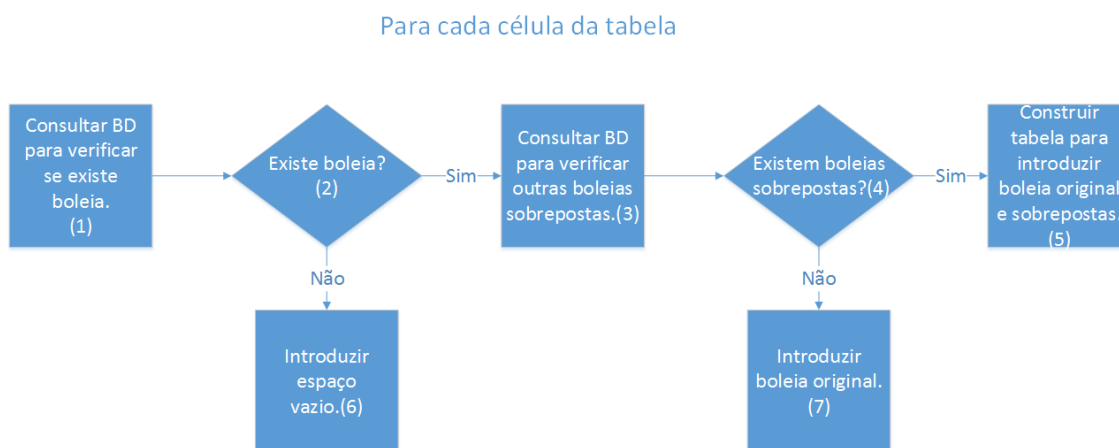


Figura 11 - Ordem de construção de tabelas pelo HTML

Tendo isto em conta, para solucionar a posição e tamanho da boleia no mapa consoante a sua data, hora inicial e hora final (ponto 2 do problema), a aplicação vai seguir o seguinte algoritmo (Figura 12). Começa-se por fazer uma consulta à base de dados a cada célula que ainda não esteja preenchida para averiguar se existem boleias com a hora inicial e dia da semana correspondente a essa célula (item 1 da Figura 12). No caso de existir, a aplicação vai calcular o tamanho da célula necessário para corresponder a boleia à sua hora final.

No entanto, a solução anterior não consegue responder a boleias que partilhem o mesmo período de tempo porque, devido à estrutura HTML, não se consegue nem dividir uma célula de uma tabela HTML nem alinhar boleias sobrepostas na mesma célula sem elementos adicionais.

Para solucionar este problema, em referência à adaptação do mapa a boleias sobrepostas (ponto 3 do problema), decidiu-se fazer uma consulta adicional à base de dados a cada célula da tabela para descobrir se existem elementos adicionais (itens 2 e 3). Se esta consulta não devolver resultados, a célula é preenchida normalmente sem elementos adicionados (item 7). Se devolver, será criada uma tabela adicional dentro dessa célula (item 5). Esta tabela adicional terá uma coluna para cada boleia nesse período de tempo, em que cada boleia sobreposta será posicionada nessa coluna consoante a sua diferença de tempo em relação à hora inicial da boleia encontrada na primeira consulta.



**Figura 12 - Fluxograma do algoritmo de leitura de boleias**

### 5.2.2. Interface e interação eficiente da aplicação

**Problema – Interação fácil e intuitiva com o mapa das boleias, nomeadamente nas operações mais comuns:**

Um dos objetivos do trabalho proposto é desenvolver um interface para a aplicação que seja o mais eficiente possível, de modo a que o utilizador consiga fazer um grande número de ações num curto espaço de tempo e sem esforço. Isto significa que a interface tem de ser o mais



simples e automatizada possível. Tendo em conta estes aspetos, podemos listar os pontos essenciais deste problema:

1. Tornar o processo de inserir/eliminar/alterar/repetir boleias e inserir/retirar passageiros, fácil e rápido (fazer ações com o menor número de clicks).
2. Tornar a interface intuitiva, de modo ao utilizador perceber como a aplicação funciona sem a necessidade de um tutorial ou instruções.

Solução:

O interface do mapa de boleias é o ponto focal deste desafio, já que é onde o utilizador vai passar a maioria do tempo dentro da aplicação. Tendo em conta este ponto, decidiu-se colocar os casos de uso mais relevantes e mais usados (inserir/alterar/eliminar/repetir boleia, entrar/sair de uma boleia) nesta interface.

De modo a tornar o processo de inserir boleias rápido e fácil (ponto 1 do problema), decidiu-se tornar cada célula do mapa de boleias não preenchido num botão. Este botão foi feito com uma função Javascript que é acionado quando o utilizador clica num elemento HTML ligado a essa função. O botão permite ao utilizador colocar uma boleia, nessa célula, com todos os seus dados automaticamente. Para o caso de todas as células estarem preenchidas nesse dia, foi colocado um botão ao lado dos dias da semana, onde o utilizador simplesmente insere a hora da boleia.

Seguindo estas ideias e de modo a tornar a interface intuitiva (ponto 2 do problema), foi feito o mesmo para as células preenchidas, tornando estas células igualmente em botões. No caso do utilizador clicar numa célula preenchida cuja boleia não é sua, simplesmente aparece uma caixa com as informações da boleia e com um botão que permite entrar ou sair da boleia caso seja passageiro ou não. No caso de ser condutor, aparece uma caixa semelhante, mas com elementos de texto caso o utilizador queira alterar os dados da boleia, repetir a boleia ou eliminá-la.

Todas estas decisões fazem com todas as funcionalidades essenciais não só se encontrem no mapa de boleias, mas também com que a interface simule um calendário interativo. Assim, o utilizador consegue facilmente identificar como utilizar a aplicação corretamente, tal como podemos observar na imagem a seguir Figura 13.

Mapa da semana Nov 16 - Nov 22

Hora	Segunda +	Terça +	Quarta +	Quinta +	Sexta +	Sábado +
09:00	MJ/JP		JP			
09:30						
10:00						
10:30						
11:00						
11:30						
12:00						
12:30	MJ					
13:00						
13:30						
14:00						
14:30						
15:00						

**MJ/JP** Condutor: Maria João

Partida:

Destino:

Lugares: 4

Vagas: 3

**Passageiro 1: João Pedro**

**Ida e volta.**

**JP** Condutor: João Pedro

Partida:

Destino:

Lugares: 4

Hora início: 9 :00

Hora fim: 9 :30

Vagas: 4

**MJ** Condutor: Maria João

Partida:

Destino:

Lugares: 4

Vagas: 4

**Quer inserir boleia?**

Figura 13 - Interface do mapa de boleias (À esquerda, em cima: uma boleia selecionada por um membro passageiro; À esquerda, em baixo: uma boleia selecionada por um membro não-passageiro; Ao centro: uma boleia selecionada pelo condutor; À direita: um espaço vazio selecionado;)

### 5.3.Avaliação da aplicação

A quarta e última fase da programação extrema é a fase de testes. Nesta secção, iremos falar dos testes que foram feitos à aplicação desde o início do seu desenvolvimento até ao seu estado de desenvolvimento.

Durante a sua programação, a aplicação foi testada continuamente para garantir a sua integridade e segurança. Para manter a integridade da aplicação, testou-se a interação entre os vários casos de uso e a validação dos dados colocados pelo utilizador, de acordo com as descrições de caso de uso que se encontram no Anexo A.

Para uma melhor avaliação da aplicação, foi utilizada uma aplicação web desenvolvida pela Google chamada Pagespeed Insights. Esta aplicação avalia o desempenho da aplicação quer em telemóvel quer em computador. Esta avaliação inclui a experiência da aplicação para o utilizador e o desempenho do carregamento da aplicação no browser.

Para telemóvel, o Google Insights encontrou um erro de bloqueio de Javascript (Anexo C1). Mas devido à estrutura do código desenvolvido estar focalizado para computador, não foi possível corrigir este erro até à entrega do relatório, pelo que o mesmo se encontra para correção

futura. Em termos de desempenho, aplicação só encontrou alguns avisos menores que incluem a compactação do CSS e HTML em 6% e a utilização da cache do navegador. O valor da compactação é mínimo e não se justifica fazer. Além disso, não se consegue tirar partido da cache devido aos visuais dinâmicos da aplicação, especialmente do mapa de boleias. No entanto, a aplicação passou nos testes de redução de Javascript, redução de tempo de resposta ao servidor e priorização ao conteúdo visível da aplicação. Estes testes são sumarizados na Tabela 19:

Avisos.	Testes passados.
Compactação de CSS e HTML.	Redução de Javascript.
Utilização da cache do navegador.	Redução de tempo de resposta ao servidor.
	Priorização ao conteúdo visível da aplicação.

**Tabela 19 – Resultados dos testes de desempenho da aplicação para telemóvel.**

A aplicação, para telemóvel, conseguiu pontuação máxima na experiência de utilizador (Anexo C2). Os testes incluíram configuração da viewport, dimensionamento do conteúdo em função da janela atual, dimensionamento adequado de elementos tácteis e sem a utilização de plugins. Estes testes são sumarizados na Tabela 20:

Testes passados.
Configuração da viewport.
Dimensionamento do conteúdo em função da janela atual.
Dimensionamento adequado de elementos tácteis.
Não utiliza plugins.

**Tabela 20 – Resultados dos testes de experiência do utilizador da aplicação para telemóvel.**

Para computador, a aplicação foi avaliada, em termos de desempenho, com uma pontuação excelente contendo só alguns avisos (Anexo C3). Os avisos incluíram também, compactação de CSS/HTML de 6% e utilização da cache do navegador. Mas, tal como no caso do telemóvel, não se justifica fazer compactação com um valor tão pequeno e nesta aplicação, não se consegue utilizar a cache eficientemente devido aos visuais dinâmicos da aplicação, especialmente do mapa de boleias. Estes testes são sumarizados na Tabela 21:

Avisos.	Testes passados.
Compactação de CSS/HTML.	Redução de Javascript.
Utilização da cache do navegador.	Redução de tempo de resposta ao servidor.
	Priorização ao conteúdo visível da aplicação.

**Tabela 21 – Resultados dos testes de desempenho da aplicação para computador.**

## 6. Conclusão

Este projeto foi uma oportunidade para alargar os meus conhecimentos em programação para a Web e Web design que foram aprendidos ao longo do curso, aliada à oportunidade de poder aplicar esses conhecimentos num projeto que visa responder a necessidades reais.

Foi proposta uma aplicação Web para a organização de boleias para um grupo de docentes do IPG com funcionalidades específicas, como a automatização da escolha de condutor, estatísticas dos passageiros e um mapa de boleias único. No entanto, só foi possível atingir alguns desses objetivos que incluíram aqueles com prioridade mais alta:

- Registrar, alterar e eliminar utilizadores.
- Organizar pedidos e ofertas de boleias dentro do grupo.
- Contabilizar boleias efetuadas e recebidas.
- Enviar notificações por email acerca das próximas boleias ou alterações de boleias que afetam o utilizador.
- Registrar as alterações mais relevantes à base de dados.
- Interface eficiente e intuitiva.

O projeto era muito ambicioso, nomeadamente no que diz respeito à interação com o utilizador, que não utiliza os controlos padrão das páginas web, portanto foram desenvolvidas interações próprias para a aplicação, especificamente a interação da grelha do mapa de boleias. Foi consumido muito tempo a implementar corretamente a visualização das boleias em formato de calendário no mapa e a criar novos controlos (pop-ups associados a cada boleia que gerenciam as funcionalidades ao utilizador) para tornar a utilização deste mapa intuitiva. Devido a estes motivos, não foi possível completar todos os objetivos previstos até à finalização deste relatório. Os objetivos que se encontram por finalizar são:

- Duplicar mapas de boleias para semestres, anos.
- Preencher mapas automaticamente de acordo com as boleias contabilizadas de cada membro.
- Calcular a redução da pegada de carbono.
- Sincronizar a BD das boleias com o Google Spreadsheet.
- Criar cópias de segurança dos mapas de boleias.

Apesar de não se encontrar totalmente terminada, a aplicação está funcional e encontra-se disponível online no site [carpooling-vascof.rhcloud.com](http://carpooling-vascof.rhcloud.com) com as funcionalidades anteriormente

## CAPÍTULO 6. CONCLUSÃO

mencionadas. Como a aplicação ainda se encontra em desenvolvimento, as funcionalidades disponibilizadas no site poderão variar no futuro.

Para trabalho futuro, prevê-se a conclusão dos objetivos em falta e a expansão da aplicação para um mercado global, ou seja, tornar a aplicação disponível e operacional para qualquer utilizador e não só para um grupo específico de pessoas. Com algumas alterações à aplicação, é possível adaptá-la a um meio familiar, onde famílias podem organizar boleias e eventos. Estas alterações incluirão funcionalidades para colocar crianças em cada boleia, a relação familiar entre cada utilizador (pai, mãe, tios, primos,...) e ferramentas específicas ao ambiente familiar (ex: determinar quem leva as crianças à escola).

## 7. Bibliografia

1. Extreme Programing. *USFCS*. [Online] [Citação: 15 de Junho de 2015.] <http://www.cs.usfca.edu/~parrt/course/601/lectures/xp.html>.
2. Regional ITS Architecture Guidance Document. *FHWA Operations*. [Online] [Citação: 13 de Julho de 2015.] <http://ops.fhwa.dot.gov/publications/regitsarchguide/5defineint.htm>.
3. Wikipedia - Use Case Diagram. *Wikipedia*. [Online] [Citação: 14 de Julho de 2015.] [https://en.wikipedia.org/wiki/Use\\_Case\\_Diagram](https://en.wikipedia.org/wiki/Use_Case_Diagram).
4. Mozilla Developer Network. *HTML / MDN*. [Online] [Citação: 14 de Julho de 2015.] <https://developer.mozilla.org/en-US/docs/Web/HTML>.
5. W3 - CSS. *W3*. [Online] [Citação: Julho de 2015.] <http://www.w3.org/TR/2011/REC-CSS2-20110607/about.html>.
6. PHP: Prefácio - Manual. *PHP*. [Online] [Citação: 16 de Julho de 2015.] [http://php.net/manual/pt\\_BR/preface.php](http://php.net/manual/pt_BR/preface.php).
7. W3Schools - AJAX. *W3Schools*. [Online] [Citação: Julho de 2015.] [http://www.w3schools.com/ajax/ajax\\_intro.asp](http://www.w3schools.com/ajax/ajax_intro.asp).
8. MySQL::Why MySQL? *MySQL*. [Online] [Citação: 15 de Julho de 2015.] <http://www.mysql.com/why-mysql/>.
9. GitHub | Crunchbase. *Crunchbase*. [Online] [Citação: 14 de Julho de 2015.] <https://www.crunchbase.com/organization/github>.
10. **DUCKETT, JON**. *HTML & CSS*. Indianapolis : John Wiley & Sons, Inc, 2011.
11. **Snyder, Chris, Myer, Thomas e Southwell, Michael**. *Pro PHP Security*. New York : Springer Science+Business Media, LLC., 2010.
12. Bootstrap. *Bootstrap*. [Online] [Citação: Junho de 2015.] <http://getbootstrap.com>.
13. Biblioteca Digital do IPG. *Biblioteca Digital do IPG*. [Online] [Citação: 2015.] <http://bdigital.ipg.pt/dspace/>.
14. Calculating color contrast. *24Ways*. [Online] [Citação: Julho de 2015.] <http://24ways.org/2010/calculating-color-contrast/>.
15. Moqups. *Moqups*. [Online] [Citação: 14 de Junho de 2015.] [moqups.com](http://moqups.com).

## CAPÍTULO 7.BIBLIOGRAFIA

16. Palleton - The Color Scheme Designer. *Palleton - The Color Scheme Designer*. [Online] [Citação: 14 de Junho de 2015.] <http://paletton.com/>.
17. *Stack Overflow*. [Online] [Citação: 2015.] <http://stackoverflow.com/>.
18. Usage | ColorSchemeDesigner. *ColorSchemeDesigner*. [Online] [Citação: Julho de 2015.] <http://www.colorschemedesigner.com/blog/usage/>.
19. *W3Schools Online Web Tutorials*. [Online] [Citação: 2015.] <http://www.w3schools.com/>.

## **8. Anexos**



## • Anexo A

Nome	Alterar boleia
Objetivo	O ator altera uma boleia
Prioridade	Média
Pré-condição	Login válido
Cenário principal	<ol style="list-style-type: none"> <li>1. O Caso de Uso começa quando o ator clica no botão “Alterar” após selecionar uma boleia.</li> <li>2. O sistema apresenta o formulário “Alterar boleia”</li> <li>3. O ator preenche os campos obrigatórios.</li> <li>4. O ator clica no botão “Ok”, confirmando os dados</li> <li>5. O sistema regista os dados</li> </ol>
Cenário alternativo	<p>O ator pode cancelar a operação a qualquer momento.</p> <ol style="list-style-type: none"> <li>3. a. O ator não preenche todos os campos obrigatórios e aparece uma mensagem de erro.</li> <li>4. b. Se a sintaxe de algum campo estiver incorreta, mostra mensagem de erro.</li> <li>4 c. Se a boleia faz parte de uma repetição, o sistema apresenta uma mensagem para alterar todas as boleias da repetição.</li> </ol>
Pós-condição	É enviado um email para todos os passageiros da boleia a informar da alteração efetuada.
Casos de teste	<ul style="list-style-type: none"> <li>• Verificar se o mapa de boleias é atualizado corretamente.</li> <li>• Verificar se as estatísticas são atualizadas corretamente.</li> <li>• Verificar se os campos são preenchidos corretamente <ul style="list-style-type: none"> <li>○ Se os campos numéricos só contêm caracteres numéricos.</li> <li>○ Se os campos alfabéticos só contêm caracteres alfabéticos.</li> </ul> </li> </ul>

Tabela 22 - Descrição do caso de uso "Alterar boleia"

Nome	Entrar numa boleia
Objetivo	O ator entra numa boleia
Prioridade	Alta
Pré-condição	Login válido
Cenário principal	<ol style="list-style-type: none"> <li>1. O Caso de Uso começa quando o ator clica no botão “Entrar” após selecionar uma boleia.</li> <li>2. O sistema apresenta o formulário “Inserir passageiro”</li> <li>3. O ator preenche os campos obrigatórios.</li> <li>4. O ator clica no botão “Ok”, confirmando os dados</li> <li>5. O sistema regista os dados</li> </ol>
Cenário alternativo	<p>O ator pode cancelar a operação a qualquer momento.</p> <p>4. a. O ator não preenche todos os campos obrigatórios e aparece uma mensagem de erro.</p>
Pós-condição	Os dados estatísticos relevantes ao ator são atualizados.
Casos de teste	<ul style="list-style-type: none"> <li>• Verificar se o mapa de boleias é atualizado corretamente.</li> <li>• Verificar se as estatísticas são atualizadas corretamente.</li> </ul>

Tabela 23 - Descrição do caso de uso "Entrar numa boleia"

Nome	Repetir boleia
Objetivo	O ator repete uma boleia
Prioridade	Média
Pré-condição	Login válido
Cenário principal	<ol style="list-style-type: none"> <li>1. O Caso de Uso começa quando o ator clica no botão “Repetir” após selecionar uma boleia.</li> <li>2. O sistema apresenta o formulário “Repetir boleia”</li> <li>3. O ator preenche os campos obrigatórios.</li> <li>4. O ator clica no botão “Ok”, confirmando os dados</li> <li>5. O sistema insere novas boleias consoante os campos preenchidos</li> </ol>
Cenário alternativo	<p>O ator pode cancelar a operação a qualquer momento.</p> <p>4. a. O ator não preenche os campos corretamente e aparece mensagem de erro.</p>
Pós-condição	Os dados estatísticos relevantes ao condutor e passageiros (se existirem) da boleia repetida são atualizados.
Casos de teste	<ul style="list-style-type: none"> <li>• Verificar se o mapa de boleias é atualizado corretamente.</li> <li>• Verificar se as estatísticas são atualizadas corretamente.</li> </ul>

Tabela 24 - Descrição do caso de uso "Repetir boleia"

Nome	Sair de uma boleia
Objetivo	O ator sai uma boleia
Prioridade	Média
Pré-condição	Login válido
Cenário principal	<ol style="list-style-type: none"> <li>1. O Caso de Uso começa quando o ator clica no botão “Sair” após selecionar uma boleia.</li> <li>2. O sistema pede a confirmação da saída.</li> <li>3. O ator clica no botão “Ok”, confirmando a saída.</li> <li>4. O sistema desativa o passageiro.</li> </ol>
Cenário alternativo	O ator pode cancelar a operação a qualquer momento.
Pós-condição	Os dados estatísticos relevantes ao condutor e passageiros (se existirem) da boleia repetida são atualizados.
Casos de teste	<ul style="list-style-type: none"> <li>• Verificar se o mapa de boleias é atualizado corretamente.</li> <li>• Verificar se as estatísticas são atualizadas corretamente.</li> </ul>

Tabela 25 - Descrição do caso de uso "Sair de uma boleia"

• Anexo B

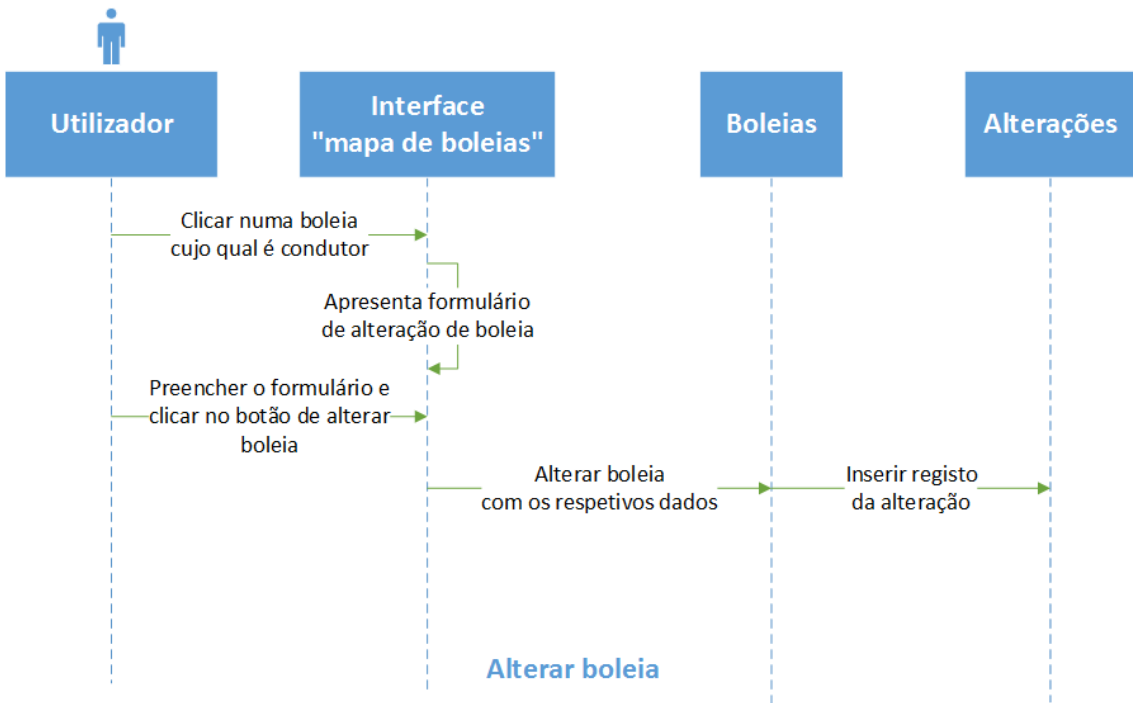


Figura 14 - Diagrama de sequência do caso de uso "Alterar boleia"

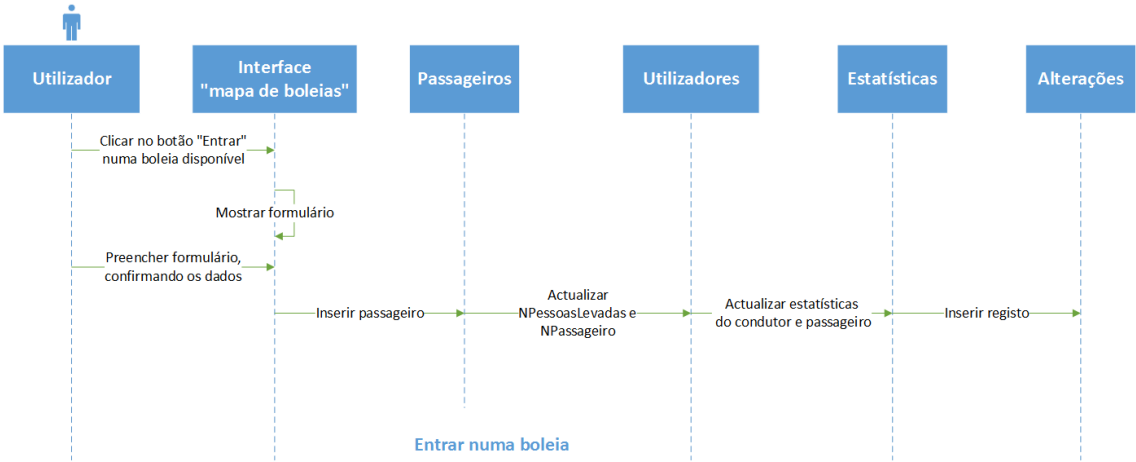


Figura 15 - Diagrama de sequência do caso de uso "Entrar numa boleia "

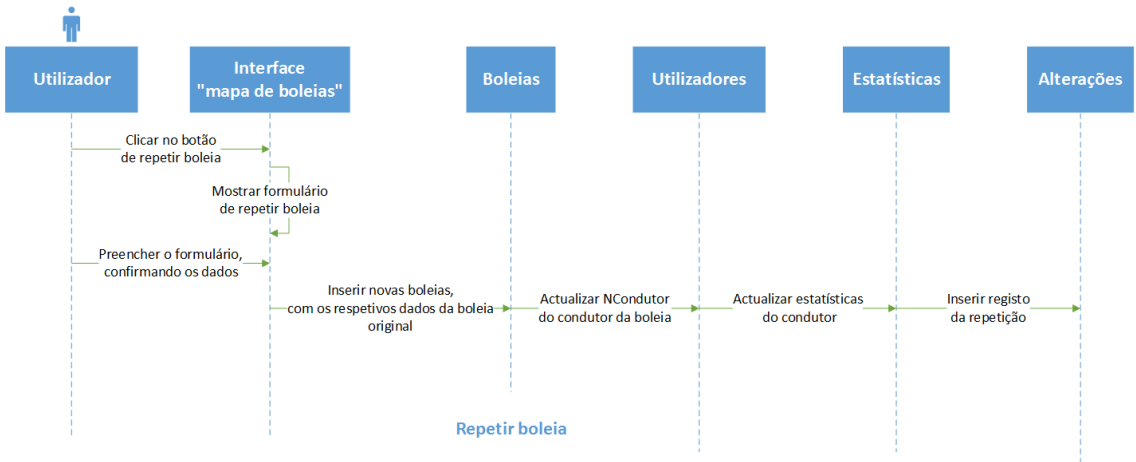


Figura 16 - Diagrama de sequência do caso de uso "Repetir boleia"



Figura 17 - Diagrama de sequência do caso de uso "Sair de uma boleia"

## • Anexo C1

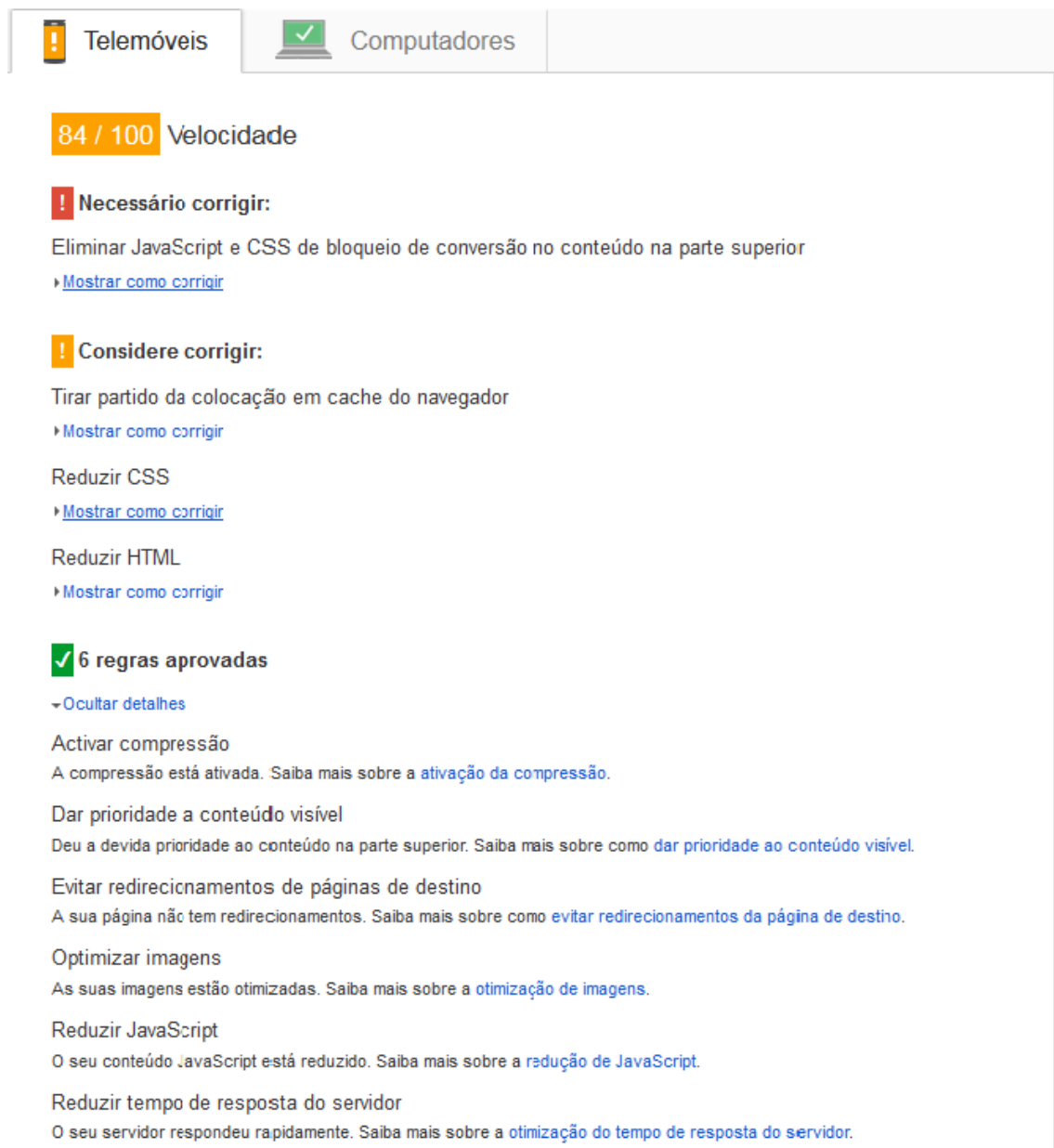


Figura 18 - Avaliação do PageSpeed da aplicação para telemóvel (Parte 1)

## • Anexo C2

100 / 100 Experiência do utilizador

✓ **Parabéns! Não foram encontrados problemas.**

### Configurar a viewport

A sua página especifica uma viewport que corresponde ao tamanho do dispositivo, o que lhe permite ser devidamente processada em todos os dispositivos. Saiba mais sobre como [configurar viewports](#).

### Dimensionar conteúdo em função da janela atual

O conteúdo da sua página cabe dentro da janela atual. Saiba mais sobre o [redimensionamento de conteúdo para a janela atual](#).

### Dimensionar elementos táteis adequadamente

A dimensão de todos os links/botões da sua página é suficientemente grande para permitir que um utilizador toque facilmente num ecrã tátil. Saiba mais sobre o [dimensionamento adequado de elementos táteis](#).

### Evitar anúncios intercalares de instalação de aplicação que ocultem conteúdo (BETA)

A sua página não parece ter anúncios intercalares de instalação de aplicação que ocultem um volume significativo de conteúdo. Saiba mais sobre a importância de [evitar a utilização de anúncios intercalares de instalação de aplicação](#).

### Evitar plug-ins

Aparentemente a sua página não utiliza plug-ins, o que pode evitar que determinados conteúdos sejam utilizados em várias plataformas. Saiba mais sobre a importância de [evitar plug-ins](#).

### Utilizar tamanhos de tipo de letra legíveis

O texto na sua página é legível. Saiba mais sobre [a utilização de tamanhos de tipo de letra legíveis](#).

Figura 19 - Avaliação do PageSpeed da aplicação para telemóvel (Parte 2)



## • Anexo C3

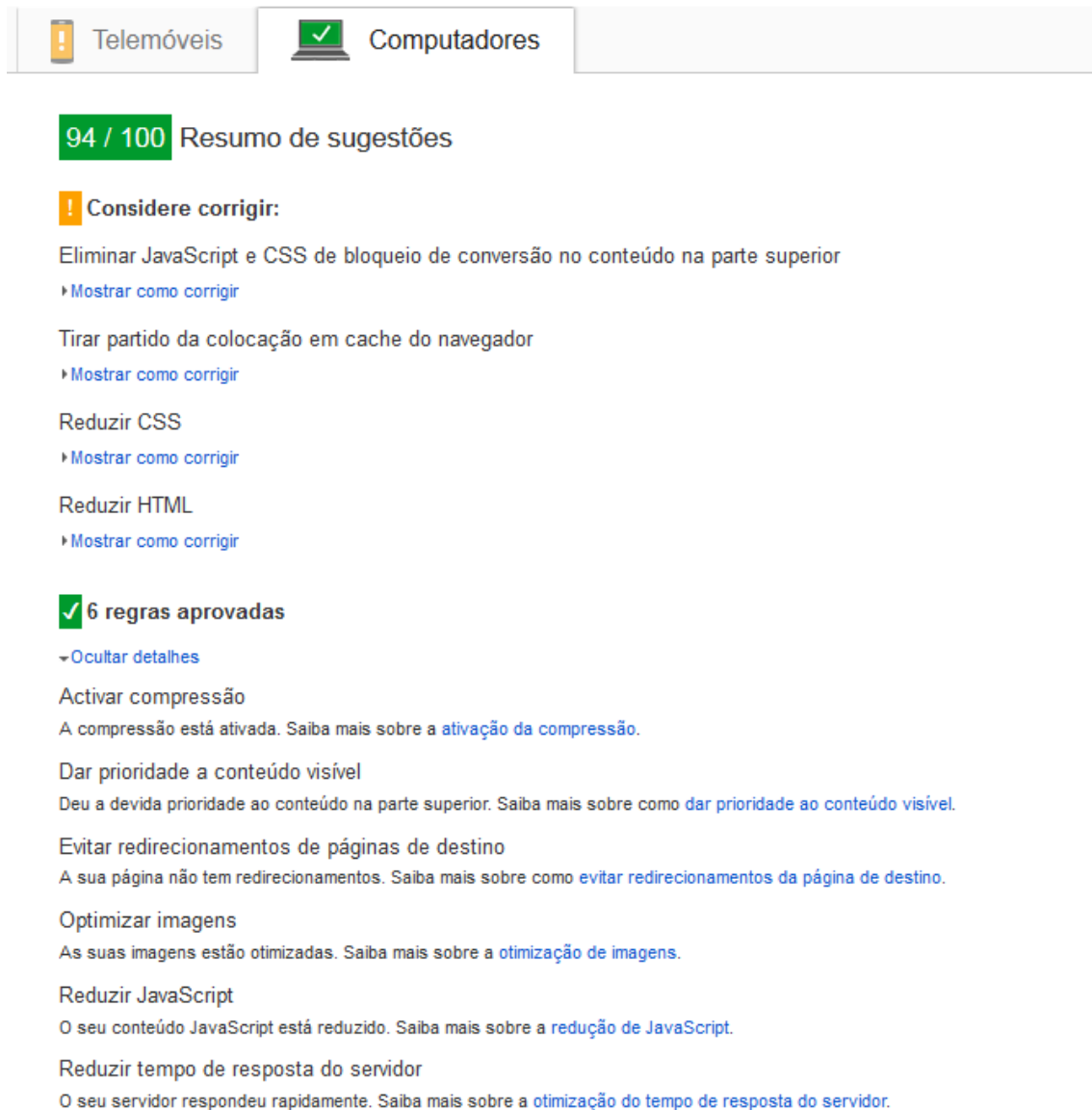


Figura 20 - Avaliação do PageSpeed da aplicação para computador