



**Universidade do Minho**

Escola de Engenharia

Mestrado Integrado em Engenharia Informática

## **Unidade Curricular de Laboratórios de Informática IV**

Ano Letivo de 2021/2022

### **MinhoPark - Fase de Implementação**

Bernardo Saraiva(93189)

Daniel Azevedo(93324)

José Gonçalves(93204)

Pedro Araújo(70699)

Rui Moreira(93232)

22 de junho de 2022

# LI4

Data de Receção	
Responsável	
Avaliação	
Observações	

## MinhoPark - Fase de Implementação

Bernardo Saraiva(93189)  
Daniel Azevedo(93324)  
José Gonçalves(93204)  
Pedro Araújo(70699)  
Rui Moreira(93232)

22 de junho de 2022

## Resumo

O MinhoPark é uma aplicação mobile de pesquisa de parques na região do Minho que visa dar a conhecer aos seus utilizadores vários tipos de parques existentes numa dada localização. Através da navegação pela aplicação, o utilizador consegue não só identificar parques que se encontrem perto do mesmo, mas também customizar as preferências e preservar os seus parques favoritos para visitas futuras, fomentando assim um estilo de vida mais saudável e ecológico.

Este trabalho reflecte a implementação do projecto proposto e planeado pelo grupo 27, tendo sido seguidas as especificações técnicas e funcionais apresentadas, bem como as interfaces, camada lógica e sistema de dados sugeridos. As informações associadas ao projecto foram disseminadas para a equipa de implementação através de uma apresentação oral e relatório de especificações que, na sua generalidade, a equipa considerou organizado e explicito, tendo transmitido um parecer de apreciação positivo.

O presente documento explora detalhadamente a fase de implementação da aplicação. A criação da base de dados que comporta toda a informação relativa aos parques, e que portanto alimenta a solução foram criadas recorrendo ao gestor de base de dados MySQL. Tratando-se de uma aplicação mobile, a equipa de desenvolvimento recorreu ao uso do Android Studio IDE para recriar os diversos mock-ups em ambientes de navegação, tendo o MinhoPark sido implementado em Java.

A aplicação MinhoPark contempla todas as especificações e requisitos, tendo sido desenvolvida de acordo com o cronograma proposto, encontrando-se apta para beta-testing e posterior disponibilização uso público.

**Área de Aplicação:** Desenvolvimento *mobile*, Desenvolvimento de Software, Base de Dados, Gestão de Projectos

**Palavras-Chave:** Parques, Natureza, Aplicação Mobile

# Índice

<b>1</b>	<b>Introdução</b>	<b>1</b>
1.1	Enquadramento geral . . . . .	1
1.2	Caracterização do trabalho . . . . .	1
1.3	Recursos . . . . .	5
1.4	Diagrama de GANTT . . . . .	5
1.5	Sumário . . . . .	7
<b>2</b>	<b>Desenvolvimento</b>	<b>8</b>
2.1	Estratégia . . . . .	8
2.2	Identificação e caracterização da aplicação e dos seus serviço . . . . .	8
2.3	Arquitectura . . . . .	9
2.3.1	Model . . . . .	9
2.3.2	Sistema de Dados . . . . .	11
2.3.3	Interface do utilizador . . . . .	12
<b>3</b>	<b>Conclusões e Trabalho Futuro</b>	<b>18</b>
	<b>Lista de Siglas e Acrónimos</b>	<b>20</b>
	<b>Anexos</b>	<b>21</b>
	Anexo 1 . . . . .	21
	Anexo 2 . . . . .	23

# Lista de Figuras

1.1	Diagrama de <i>GANTT</i>	6
2.1	Arquitectura da aplicação	9
2.2	Nova base de Dados da aplicação MinhoPark	11
2.3	Menu da aplicação MinhoPark	12
2.4	Página de preferências da aplicação MinhoPark	13
2.5	Página de procura da aplicação MinhoPark	14
2.6	Página de procura da aplicação MinhoPark	15
2.7	Página de procura da aplicação MinhoPark	16
2.8	Página de favoritos da aplicação MinhoPark	17
3.1	Grelha de avaliação ao Projecto MinhoPark	22
3.2	Use Case 1	23
3.3	Use Case 2	24
3.4	Use Case 3	24
3.5	Use Case 4	25
3.6	Use Case 5	25
3.7	Use Case 6	25
3.8	Use Case 7	25

## Lista de Tabelas

1.1	Requisitos Funcionais estabelecidos pelo Grupo 27 para o utilizador . . . . .	2
1.2	Requisitos Funcionais suplementares para o utilizador identificados pela equipa de desenvolvimento face às interfaces propostas . . . . .	3
1.3	Requisitos Não Funcionais estabelecidos para a aplicação. . . . .	4
1.4	Requisitos Não Funcionais estabelecidos para a aplicação. . . . .	4

# 1 Introdução

O presente capítulo pretende explorar a motivação e objectivos da aplicação MinhoPark. São ainda discutidos os recursos necessários à implementação do projecto, bem como o cronograma de execução do mesmo.

## 1.1 Enquadramento geral

Através da análise ao relatório de especificação recebido, considera-se que o intuito do projecto MinhoPark passa pela criação de uma aplicação mobile que tem como objectivo primordial tornar-se a principal ferramenta de pesquisa de parques da região do Minho.

Com base na contextualização, motivação e justificação presentes no relatório recebido, é possível perceber que a aplicação não se destina apenas aos amantes da natureza, apesar de incluir funcionalidades que atendem as necessidades de um utilizador para planear ou realizar um percurso, mas sim à população no geral, desde habitantes locais a turistas. Destaca-se assim a necessidade de uma aplicação cujo a interface seja de fácil utilização e uso para diversas comunidades e faixas etárias.

Esta aplicação tem também como grande objectivo disseminar e dar visibilidade à grande diversidade de parques existentes na região de modo a que o investimento realizado na reabilitação das zonas urbanas e florestais seja usufruído pelas comunidades. Deste modo, a aplicação MinhoPark demonstra uma preocupação quer na rentabilização dos investimentos realizados pelos Municípios, quer na fomentação de hábitos de vida saudável. Considera-se fundamental fomentar o contacto com a natureza, bem como promover a educação ecológica e a preservação do meio ambiente no geral, o que enaltece a importância da existência de espaços verdes em locais urbanos e das florestas. Assim a plataforma identifica diversos tipos de parque, nomeadamente parques urbanos, parques de lazer, parques infantis e reservas naturais, entre outros. Maior detalhe sobre os requisitos funcionais e não funcionais idealizados e implementados são apresentados no Capítulo 2.

## 1.2 Caracterização do trabalho

Sendo o MinhoPark uma aplicação que visa fomentar ao seus utilizadores o maior usufruto de espaços verdes, apresenta-se nesta secção os requisitos propostos e discutindo-se o modo que seriam estruturados durante a sua implementação.

Através da análise do relatório de apresentação e especificações providenciado pelo Grupo 27 foram extraídos os requisitos funcionais e não funcionais do sistema, que estão associados a um ID e apresentados nas Tabelas 1.1, 1.2, 1.3 e 1.4:

## Requisitos Funcionais do Utilizador

Requisitos Funcionais do Utilizador	
RF_U1	Como utilizador, ao iniciar a aplicação devo poder escolher ver as preferências, a lista de favoritos ou a procura
RF_U2	Como utilizador, ao clicar em preferências, devo poder escolher se quero utilizar a minha localização actual ou se quero utilizar outra localização
RF_U3	Como utilizador, ao clicar em preferências, devo poder escolher o tipo de parque que quero visualizar
RF_U4	Como utilizador, ao clicar em preferências, devo poder escolher se quero visualizar o top 5, 10 ou 15 dos parques disponíveis de acordo com a minha localização
RF_U5	Como utilizador, após ter escolhido as minhas preferências, devo poder visualizar a lista de parques actualizada tendo em conta as mesmas na interface procura
RF_U6	Como utilizador, ao clicar em favoritos, devo poder visualizar uma lista com os parque que marquei como preferidos

Tabela 1.1: Requisitos Funcionais estabelecidos pelo Grupo 27 para o utilizador

Da interpretação dos requisitos funcionais apresentados, a equipa de desenvolvimento visa implementar uma aplicação que permita de forma fácil e intuitiva o acesso: **i)** a uma interface de procura de parques com base na localização; **ii)** uma página de preferências que permite ao utilizador filtrar o tipo de parques que lhe são listados na interface de procura; e **iii)** uma página que permite guardar e navegar pela lista de parques previamente definidos como os favoritos do utilizador.

Mais concretamente a procura de parques com base na localização, **i)**, deve poder ser feita com base na localização actual do utilizador, o que implica um requisito não funcional associado à necessidade de acesso à geolocalização do mesmo, ou pesquisa no mapa de outras localizações. Relativamente ao ponto **ii)**, a mesma permite definir as predefinições de pesquisa, assim como a selecção cumulativa dos tipos de parques disponíveis, efectuando uma filtragem através dos parques existentes na base de dados. Finalmente o ponto **iii)** lista os parques definidos como favoritos pelo utilizador durante a sua navegação na interface de procura (i).

Após apreciação dos mock-ups de UI propostos no relatório de especificações, a equipa de desenvolvimento identificou e definiu requisitos funcionais suplementares que reflectiam os objectivos e funcionalidades do projecto MinhoPark.



Requisitos Funcionais Suplementares	
RF_U7	Como utilizador, ao seleccionar a interface de procura devo poder visualizar um mapa com pins de parques e a lista do nome dos mesmo
RF_U8	Como utilizador, ao seleccionar a interface de procura devo poder zoom in e zoom out no mapa
RF_U9	Como utilizador, ao seleccionar parques na interface de procura devo poder ver informação específica do mesmo, nomeadamente o tipo e nome do parque, distancia a que se encontra do utilizador, rating, horário de abertura e fecho, morada e se este encontra operacional
RF_U10	Como utilizador, ao seleccionar um parque na interface de procura devo poder seleccionar o modo de deslocação até ao mesmo (a pé, de bicicleta, de carro), e receber as instruções do trajecto até ao parque escolhido
RF_U11	Como utilizador, ao seleccionar o modo de deslocação "de carro" na interface de procura devo poder obter um trajecto com ou sem portagens, conforme estiver definido previamente na interface de preferências
RF_U12	Como utilizador, devo poder remover parques da minha lista de favoritos tanto na interface de favoritos como na interface de procura

Tabela 1.2: Requisitos Funcionais suplementares para o utilizador identificados pela equipa de desenvolvimento face às interfaces propostas

São apresentados nas Tabelas 1.3, 1.4 os requisitos não funcionais propostos para suportar o funcionamento da aplicação.

## Requisitos Não Funcionais

Requisitos Não Funcionais	
RNF_1	O programa deve ser rápido o suficiente para executar uma procura por parâmetros
RNF_2	O programa deve ser seguro para todos os utilizadores, sem comprometer alguma espécie de dados
RNF_3	O programa deve ser possível de utilizar sem erros ou bugs frequentes
RNF_4	Devem existir requerimentos operacionais que definem como o sistema deve ser usado
RNF_5	Uma linguagem de programação deve ser específica pelos requerimentos operacionais
RNF_6	Os recursos no ambiente de desenvolvimento devem ser sempre utilizados se forem mencionados como necessários para o funcionamento do sistema.
RNF_7	O ambiente de operações do sistema deve ser delimitado por certos parâmetros
RNF_8	O sistema deve possuir uma lista de requerimentos que determina se o sistema será aprovado por um regulador um agente que verifica as várias condições do sistema
RNF_9	O sistema deve sempre, sem exceção, operar dentro dos parâmetros legais das suas funcionalidades, nunca quebrando nenhuma lei de forma alguma
RNF_10	O sistema deve ser aceitável para uso do público geral e outros potenciais utilizadores, mantendo sempre um carácter ético

Tabela 1.3: Requisitos Não Funcionais estabelecidos para a aplicação.

Suplementarmente aos requisitos apresentados, e de acordo com as funcionalidades supramencionadas considera-se ainda como requisitos não funcionais os seguintes:

Requisitos Não Funcionais Suplementares	
RNF_11	O programa necessita de acesso à internet de modo a poder tirar partido da API do Google Maps
RNF_12	Com o objetivo de serem apresentados os parques que se encontram perto do utilizador, o programa deve ter acesso à geolocalização

Tabela 1.4: Requisitos Não Funcionais estabelecidos para a aplicação.

A apreciação da equipa de desenvolvimento aos requisitos propostos no relatório de especificação, é apresentada no Anexo 3 sendo a mesma que apesar de suficientemente claros carecem de detalhe e correlação com as interfaces propostas.

A aplicação proposta é também caracterizada pela necessidade de uma base de dados que contempla as

diversas informações referentes aos parques e respectivos atributos. A equipa de desenvolvimento considera que o diagrama de dados proposto, que contempla três tabelas, não era adequada às necessidades do projecto pelo que procedeu com a implementação de um novo sistema de dados. A justificação e novo diagrama são discutidos na Secção 2.3.2

## 1.3 Recursos

Alinhado com os diversos recursos recomendados pela equipa de planeamento (Grupo 27), no que se refere aos recursos tecnológicos necessários à implementação da aplicação MinhoPark, tanto na vertente de implementação de funcionalidades, como na vertente de agilização do processo de desenvolvimento pela equipa de trabalho em formato remoto, são apresentados os recursos utilizados:

- **Discord:** um serviço VoIP que possibilita a comunicação através de texto e voz. Permite a criação de um servidor com diversos canais de modo a possibilitar uma maior organização do mesmo. Recorreu-se à utilização deste software devido à familiaridade da equipa de desenvolvimento. Permitiu a organização e a delineação de tarefas e mostrou-se vital quando qualquer dificuldade ou duvida surgia devido à possibilidade da partilha de ecrã entre os membros;
- **Git:** ferramenta de controlo de versões utilizada em ambiente de desenvolvimento de software;
- **Visual Paradigm:** software de modulação UML. De modo a poder tirar partido dos diversos diagramas UML disponibilizados pelo o Grupo 27, recorreu-se ao uso deste software;
- **Linguagem de Programação:** Java, devido à popularidade e versatilidade proporcionada pela linguagem, assim como pela sua forte presença no mercado de desenvolvimento mobile;
- **Android Studio:** IDE primordial para desenvolvimento mobile para a plataforma Android;
- **MySQL:** software de gestão de base de dados que recorre à utilização da linguagem SQL, assim como as ferramentas associadas ao mesmo, nomeadamente Workbench e SQL Server;
- **Google Maps API:** API disponibilizada pela Google para a integração dos seus mapas no desenvolvimento de aplicações android.

## 1.4 Diagrama de GANTT

O diagrama de GANTT apresentada na Figura 1.1 reflecte o planeamento e gestão temporal levada a cabo pela equipa de desenvolvimento (Grupo 26) no decorrer da Unidade Curricular Laboratórios de Informática IV. Assim, contempla não só as actividades e tarefas associadas à primeira fase no qual a equipa propôs o projecto gUMloso, bem como as tarefas relativas à realização da segunda fase de avaliação e implementação do projecto MinhoPark.

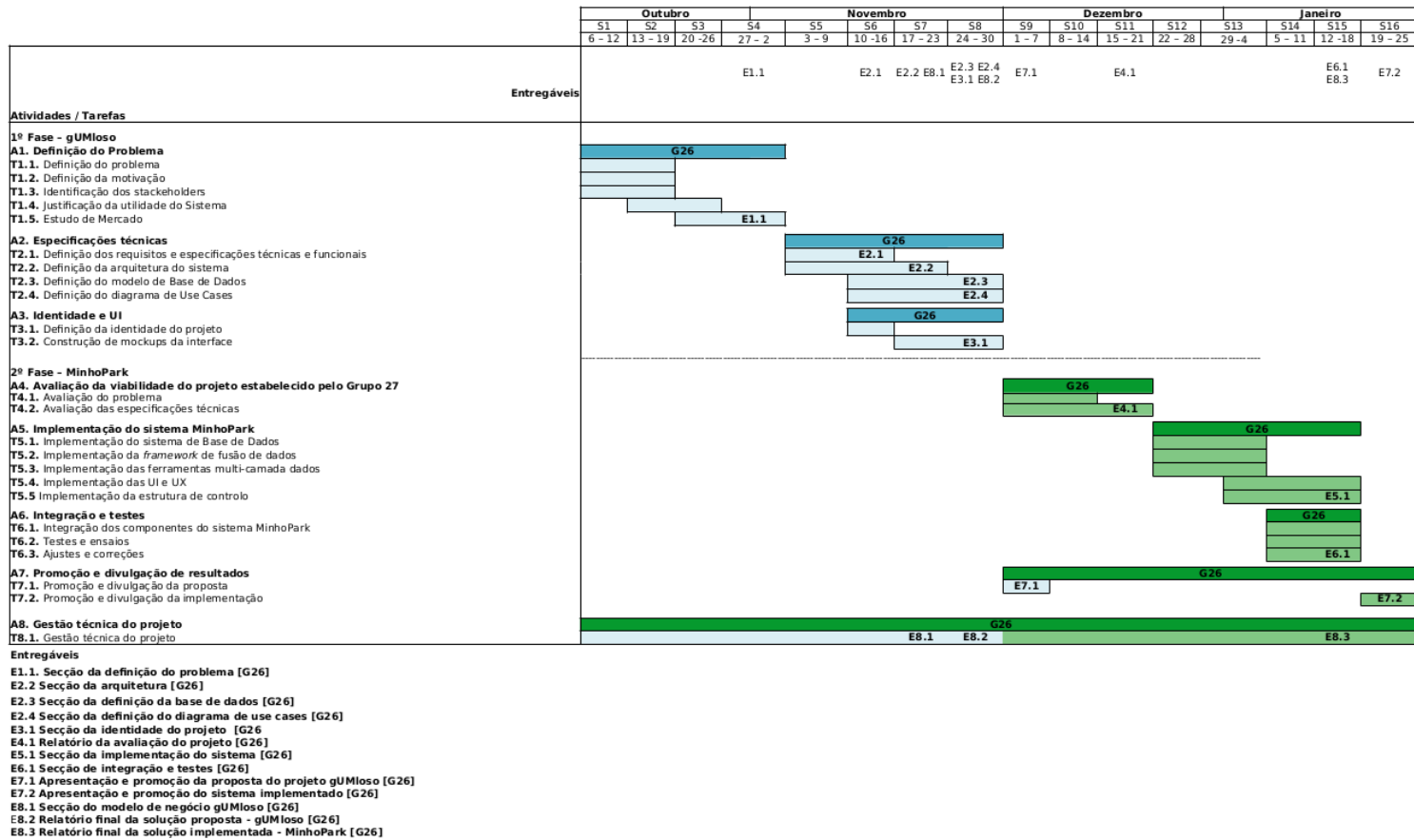


Figura 1.1: Diagrama de GANTT

## **1.5 Sumário**

Mediante o enquadramento apresentado no presente Capítulo 1, onde são descritos os principais objectivos, recursos e cronograma de execução, é discutida no Capítulo 2 a implementação da aplicação MinhoPark. Em particular, é discutida a arquitectura do sistema e modelo de base de dados que o sustenta, e são apresentados os requisitos do sistema e efectuado o levantamento do seu cumprimento. O Capítulo 3 discute os pontos fundamentais atingidos e trabalho futuro.

## 2 Desenvolvimento

O presente Capítulo 2, como mencionado anteriormente, pretende descrever e demonstrar detalhadamente todo o processo de desenvolvimento da aplicação MinhoPark.

### 2.1 Estratégia

A estratégia de implementação da aplicação MinhoPark, de uma forma alinhada com as tarefas e sequência temporal apresentados no diagrama de GANTT, passou pelo desenvolvimento sequencial e iterativo dos componentes chave, nomeadamente sistema de dados, camada lógica e as interfaces do utilizador.

Em maior detalhe, após a avaliação do relatório de especificações que proporcionou um conhecimento abrangente das motivações, objectivos e requisitos do projecto, a equipa procedeu da seguinte forma:

- Implementação do sistema de base de dados;
- Povoamento da base de dados com parques de diversas localizações na região do Minho;
- Criação das interfaces do utilizador;
- Estabelecimento das conexões entre as mesmas;
- Implementação dos requisitos funcionais nas UI;
- Teste funcionais, avaliação dos use cases e da UX.

O processo de desenvolvimento decorreu de forma iterativa garantindo que a inclusão de novos componentes e novas funcionalidades não só não comprometeu o correto funcionamento da aplicação como também compre na sua plenitude os requisitos propostos no relatório de especificações.

As seguintes Secções exploram detalhadamente o processo de implementação dos itens supramencionado discutindo as escolhas tecnológicas efectuas e desafios associados.

### 2.2 Identificação e caracterização da aplicação e dos seus serviços

A aplicação desenvolvida permite ao utilizador gerir as preferências, pesquisar parques filtrados pelas suas preferências, ver mais informações e solicitar direcções para um dos parques e ainda gerir os parques favoritos. Sendo assim, a aplicação desenvolvida implementa todos os serviços referidos na especificação recebida.

## 2.3 Arquitectura

Nesta secção é abordada a forma como foi estruturada e implementada a aplicação, tal como sugerido no relatório de especificação, toda a aplicação foi montada seguindo a filosofia de desenvolvimento de software MVC.

A partir da base de dados, implementada através do auxílio de MySQL Workbench e SQLServer, são gerados as várias lógicas de negócio da aplicação MinhoPark. Através das interações do utilizador, a model é manipulada. Após ser realizada e tratada a intervenção do utilizador, esta é exibida usando a respectiva view.

Deste modo, qualquer que seja a interacção que utilizador tenha com o sistema, esta terá de passar pela base de dados, model e view.

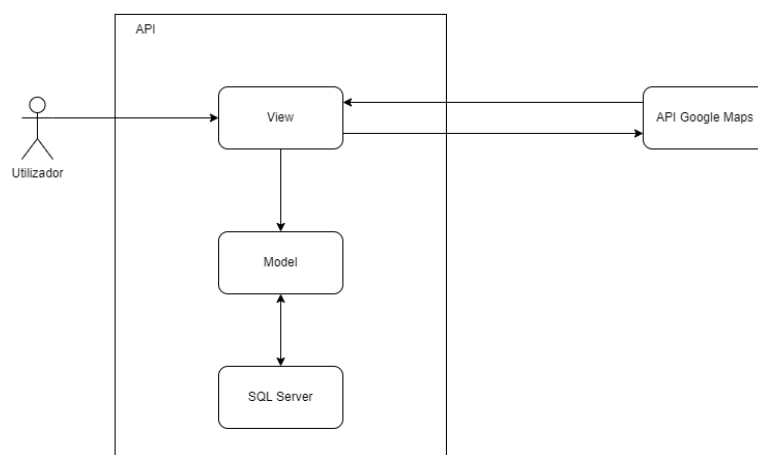


Figura 2.1: Arquitectura da aplicação

A arquitectura de alto nível do sistema é apresentada na Figura 2.1. São de seguidas discutidos individualmente cada um dos componentes fundamentais da mesma.

### 2.3.1 Model

Um dos componentes desenvolvidos foi o Model, que consiste em manipular e englobar as estruturas básicas dos dados. Deste modo, o presente componente foi dividido em dois sub-sistemas, sendo eles o SSUtilizadores e o SSParques.

#### SSUtilizadores

O subsistema de utilizadores permite efetuar todas as operações relacionadas com um utilizador. Este subsistema é centrado na classe SSUtilizadorFacade, sendo que esta classe contém um Utilizador que concentra a informação relativa:

- às suas coordenadas
- à sua lista de favoritos
- às suas preferências, recorrendo para tal à classe Preferências, sendo estas definidas através da opção presente no menu principal.

As preferências e favoritos são guardados num ficheiro em memória local no telemóvel, para que seja possível aceder a estes em outros menus. Nas preferências, o utilizador tem a opção de usar a localização actual do seu dispositivo ou uma outra localização introduzindo as coordenadas.

A opção de usar a localização actual utilizou-se a classe MyLocalizacaoListener, que obtém a latitude e longitude do dispositivo. No outro caso de introduzir outra localização com novas coordenadas, efectuou-se ainda uma validação destas mesmas coordenadas. Se não forem válidas, a localização fica definida como anteriormente sem alterações.

A classe SSUtilizadorFacade, apesar de ter sido implementada, torna-se obsoleta na medida de que apenas se possui um único utilizador, de acordo com o pretendido no enunciado. Deste modo, esta função constitui uma funcionalidade para trabalho futuro, onde se pretende que cada utilizador possa ter uma conta, passando neste caso a ser guardadas as suas preferências na Base de Dados.

## SSParques

O subsistema de parques é principalmente definido pelo SSParquesFacade que implementa a interface IParques. Através desta classe é gerida a classe Parque que tem como variáveis o parqueID, nome, imagem, endereço, coordenadas, número de Críticas, *rating*, uma Lista de Horário, e um Estado Operacional. Este Estado Operacional apenas tem um *boolean* sobre se o parque se encontra operacional. Cada Horário corresponde apenas a um dia ou pode corresponder a todos os dias se a variável dia corresponder à *String* "todos", tem um id, um *LocalTime* que corresponde à hora de abertura e outro correspondente à hora de encerramento.

A classe SSParquesFacade, apesar de inicialmente ter sido idealizada, pelo grupo 27, de modo a que a sua implementação se concretizasse através de uma lista de parques, foi decidido alterar a representação da informação deixando-a a cargo da base de dados exclusivamente, e possuindo apenas classes DAO nesta camada de forma a possibilitar a conexão entre a camada lógica e a camada de dados.

A interface IParques possui como métodos o getParque e a pesquisa, que implementam 2 das funcionalidades cruciais da aplicação de forma a que os parques consigam ser corretamente apresentados. Relativamente à pesquisa, esta processa-se inicialmente através da instância ParquesDAO que executa uma query na base de dados de forma a filtrar os parques de acordo com a preferência determinada pelo utilizador, sendo devolvido um *Set<Parques>* ordenado de acordo com a distância do utilizador ao parque, possibilitando assim seleccionar o número de parques máximo a apresentar ao utilizador de forma inequívoca, seleccionando sempre os N parques mais próximos.



### 2.3.2 Sistema de Dados

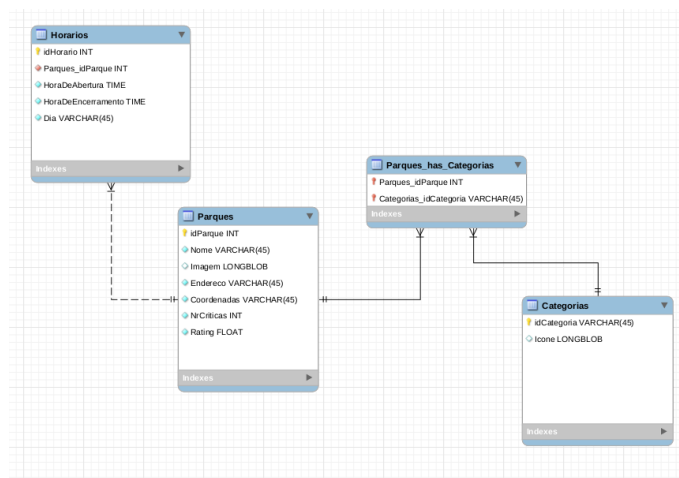


Figura 2.2: Nova base de Dados da aplicação MinhoPark

Relativamente à camada de dados, existem algumas diferenças entre a base de dados desenvolvida e utilizada comparativamente com a que foi apresentada no relatório de especificação. Visualizando o Anexo 3, Figura 3.1, é possível perceber que o sistema de dados é, precisamente, o campo onde a avaliação foi menos positiva, face à identificação de alguns problemas com a implementação proposta. Deste modo, prosseguiu-se com o desenvolvimento de um novo sistema de dados, tendo no entanto presente o sistema idealizado pelo Grupo 27, como é possível observar na Figura 2.2. Assim sendo as mudanças efectuadas focaram:

- Correção das multiplicidades, uma vez que estas se encontravam trocadas;
- Na tabela Horários, adição dos campos:
  - i) Parque\_ID - De modo a que seja possível associar um horário a um parque
  - ii) Dia - Para que o horário possa conter os dias em que o parque em questão se encontra aberto;
- Na tabela Parque:
  - i) Remoção dos parâmetros referentes ao horário e categorias, tendo estas propriedades sido passadas cada uma para a sua respectiva tabela
  - ii) Modificação do parâmetro imagem, tendo este deixado de ser considerado um INT de modo a tornar possível a inserção de imagens em formato binário;
- Na tabela de Categorias:
  - i) Remoção da descrição da categoria, não existindo a necessidade da existência da mesma para a identificação das várias categorias
  - ii) Modificação do ícone, uma vez que o mesmo não poderia ser representado por um INT;
  - iii) Alteração da primary key da tabela passando a ser o tipo de categoria.
- Criação da tabela associativa entre Parques e Categorias de modo a que a comunicação seja efectuada correctamente.

A implementação em si seguiu o modelo MVC, tal como recomendado no relatório de especificação, tendo o Controller tudo o que está relacionado com a base de dados. Foram criadas classes DAO e uma classe de comunicação com a base de dados.

A povoação da base de dados foi efectuada nas classes acima mencionadas, uma vez que estas permitem a leitura e gravação dos ficheiros em memória, como exigido no relatório de especificação.

### 2.3.3 Interface do utilizador

A presente secção explora as interfaces do utilizador implementadas e a navegação pelas mesmas. Destaca-se desde já que a equipa de desenvolvimento seguiu criteriosamente os mock-ups propostos no documento de especificações.

Assim é apresentada na Figura 2.3 a interface do menu principal da aplicação MinhoPark. Conforme definido pelo RF\_U1 o utilizador tem acesso na página inicial à possibilidade de definição das suas preferências de procura, procura efectiva de parques e acesso à sua lista de favoritos.



Figura 2.3: Menu da aplicação MinhoPark

Caso o utilizador pretenda começar a sua User Journey por definir as preferências de procura, isto é, seleccionando opção central no menu principal, será encaminhado para a interface apresentada na Figura 2.4. Nesta interface o utilizador define:

- se pretende procurar parques nas suas proximidades com base na sua localização, ou com recurso

a coordenadas GPS, cumprindo assim o RF\_U2;

- quantos parques pretende ver na lista apresentada aquando da procura, tendo em conta a sua proximidade e de acordo com o RF\_U4;
- o tipo de parque que quer contemplar na sua procura, estando a lista disponível em conformidade com os mock-ups apresentados e reflectindo o RF\_U3;
- a possibilidade do trajecto recomendado para deslocação para um parque via carro evitar ou não o uso de portagens, conforma apresentado o requisito RF\_11.



Figura 2.4: Página de preferências da aplicação MinhoPark

A navegação nesta interface reflecte o Use Case 1 apresentado em Anexo 3.

As definições definidas nesta interface de preferências determinam o tipo de conteúdo apresentado na interface de procura, conforme descrito no RF\_U5. Através da primeira opção do menu principal o utilizador acede à página de procura ilustrada na Figura 2.5.

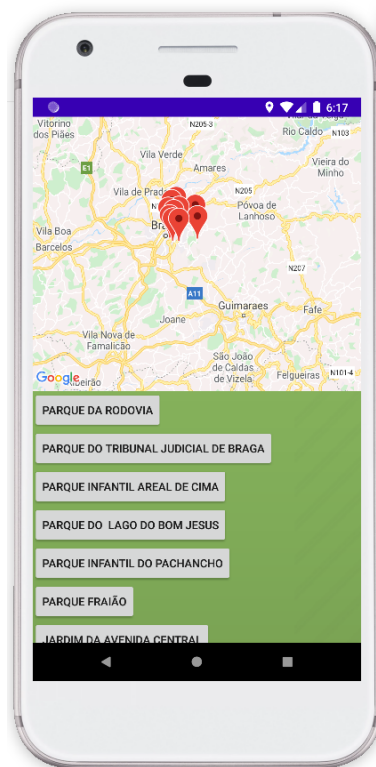


Figura 2.5: Página de procura da aplicação MinhoPark

Conforme o estipulado nos RF\_U7 e FR\_U8, que reflectem as mock-ups propostos e o Use Case 7 (presente no Anexo 3), o utilizador tem acesso a um mapa dinâmico com os pins dos diversos parques contemplados na base de dados, podendo Zoom in e Zoom out e arrastar o mapa para melhor visualização da localização dos parques. Em vista de meio ecrã, são ainda listados os parques (em número pré-definido) que se encontram no mapa (filtrados em número da mesma forma).

Se o utilizador clicar num dos parques disponíveis, informações complementares associadas ao mesmo irão surgir conforme apresentado na Figura 2.6, o que reflecte o estipulado no RF\_U9 assim como no Use Case 3 (no Anexo 3).

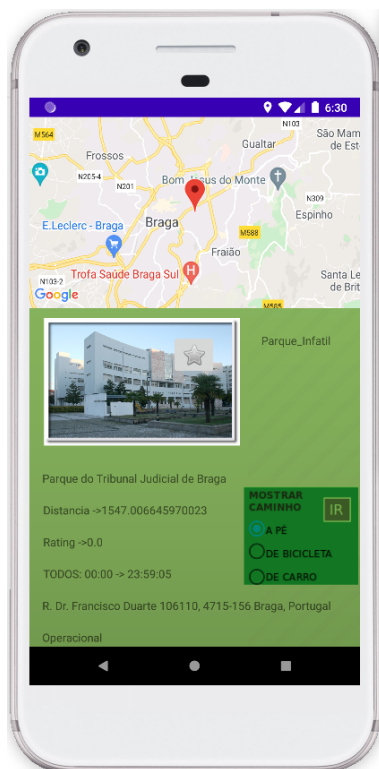


Figura 2.6: Página de procura da aplicação MinhoPark

É também nesta página que o utilizador tem a possibilidade de definir o modo de transporte que este pretende utilizar para efectuar o seu trajecto, conforme o indicado no RF\_U10 e no Use Case 2 (presente no Anexo 3).

Destaca-se, novamente que o trajecto sugerido no caso de transporte através do uso de carro terá associado a existência ou não de portagens conforme o definido nas preferências(RF\_U11). O estilo de trajecto é apresentado conforme ilustrado na Figura 2.7. Adicionalmente, no alinhamento de todo o esforço para fácil e intuitiva UX, a equipa de projecto decidiu contemplar uma funcionalidade extra pela integração da transição para o Google Maps uma vez que esta está mais fortemente disseminada.

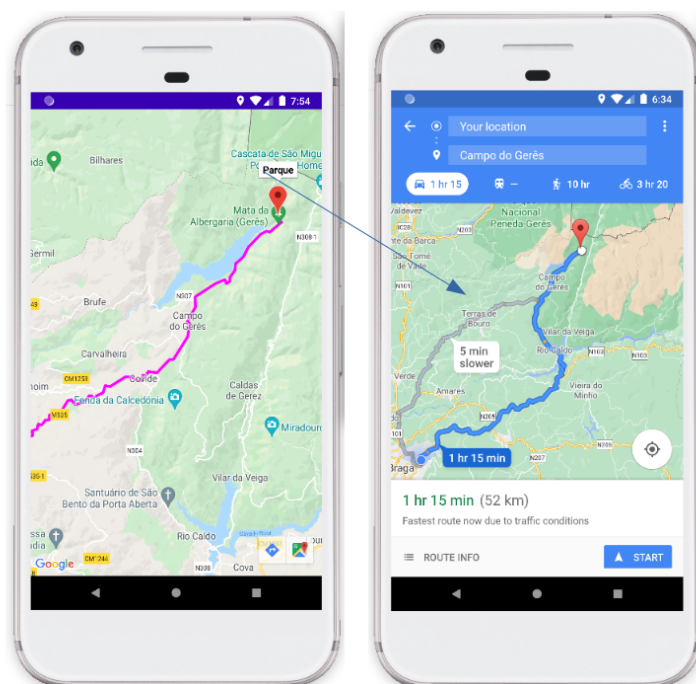


Figura 2.7: Página de procura da aplicação MinhoPark

Discutindo a última opção do menu principal, a página do Favoritos ilustrada na Figura 2.8 apresenta ao utilizador todos os parques que este definiu como seus favoritos a quando da procura, cumprindo assim o RF\_U6 e os Use Cases 4 e 6 presentes no Anexo 3.



Figura 2.8: Página de favoritos da aplicação MinhoPark

É dada ao utilizador a possibilidade de gestão da sua lista de favoritos, podendo remover parques da mesma tanto nesta interface de procura (RF\_U12 e Use Case 5).

### 3 Conclusões e Trabalho Futuro

O presente trabalho explorou a implementação da aplicação MinhoPark. Esta visa proporcionar aos seus utilizadores um maior conhecimento e possibilidade de usufruto de parques urbanos e florestais na região do Minho. Através de uma interface intuitiva para um público abrangente, o menu principal permite a exploração de parques num mapa dinâmico, a definição das preferências do utilizador e a identificação dos favoritos.

A implementação da aplicação recorreu a diversas ferramentas de desenvolvimento de software sendo as principais o Android Studio para construção das interfaces em si e das suas múltiplas funcionalidades, e o MySQL para a criação do sistema de dados que a alimenta. Pelo carácter novo destas ferramentas para a equipa de desenvolvimento, destaca-se como principal desafio da implementação a exploração das funcionalidades do Android Studio, assim como o seu modo de operação.

Numa fase inicial, a equipa de desenvolvimento processou o conteúdo proveniente do relatório de especificações do projecto com o intuito de identificar os principais requisitos da aplicação. Da análise das interfaces propostas foram identificados novos requisitos funcionais, e portanto não funcionais, que foram apresentados no presente documento. Em resposta às reais necessidades de implementação os esquema de base de dados sofreu alterações, cujo a motivação e descrição são de igual forma documentadas.

A equipa considera que a aplicação MinhoPark se encontra completamente funcional, cumprindo na sua plenitude os requisitos e objectivos traçados. Foi implementada suplementarmente a possibilidade de transição da aplicação MinhoPark para o sistema de navegação Google Maps, o que se considera trazer valor acrescido para a experiência de utilização no mercado.

Antes das disponibilização da aplicação ao público, trabalho futuro consideraria a inclusão abrangente de todos os parques da região Minho. Ainda durante o processo de desenvolvimento a equipa identificou novas funcionalidades que poderão ser contempladas para trabalho/versões futuras. Destaca-se a inclusão de um sistema de registo e login dos utilizadores, de forma a evitar guardar informação em memória e potenciais perdas de dados, e a possibilidade de integrar pesquisa pelo nome do parque. Estas alterações teriam implicações no modelo de base de dados, aumentando assim a sua complexidade. Estas são apenas apreciações da equipa de desenvolvimento não tendo as mesmas sido contempladas na documentação fornecida.

Globalmente, a equipa de desenvolvimento encontra-se muito satisfeita com a aplicação implementada, tanto de um ponto de vista tecnológico, uma vez que a mesma inclui todas as funcionalidades e reflecte a proposta do documento de especificação, mas também de um ponto de vista social e ecológico, na medida em que se considera ter sido desenvolvida uma aplicação que poderia ter real impacto na forma como a população interage com o meio ambiente.



# Referências

Connolly, T. M., Carolyn, E. B. (2002). Data-base Systems: A Practical Approach to Design, Implementation, and Management, Addison-Wesley. Reading, Massachusetts.

Belo, O. (2021). Bases de Dados Relacionais: Implementação com MySQL, FCA – Editora de Informática. Braga, Portugal.

# Lista de Siglas e Acrónimos

API	Application Programming Interface
CED	Comentário da Equipa de Desenvolvimento
ID	Identificação
IDE	Integrated Development Environment
MVC	Model-View-Controller
RF	Requisitos Funcionais
RNF	Requisitos Não Funcionais
SQL	Structured Query Language
UI	User Interfaces
UML	Unified Modeling Language
UX	User Experience
VoIP	Voice over Internet Protocol

# **Anexos**

## **Anexo 1**

**Avaliação da Fundamentação e Especificação do Sistema de Software**

<b>Grupo</b>	<b>26</b>
--------------	-----------

**Relatório Recebido**

<b>Grupo Nr:</b>	27
<b>Título:</b>	MinhoPark

**Grelha de Avaliação**

Elemento	Descrição	Pontuação (0..5)	Justificação
<b>A1</b>	<b>Definição e Fundamentação</b> <i>O sistema foi bem contextualizado? Foram apresentados elementos suficientes para se conhecer a motivação, objetivos, medidas de sucesso e a utilidade do sistema? O plano de desenvolvimento foi bem organizado e os recursos necessários identificados e explicados?</i>	<b>4</b>	O sistema encontra-se bem contextualizado, separando de forma orgânica a contextualização, a motivação e os objetivos. No entanto, quanto à identidade do projeto esta encontra-se demasiado próxima da justificação do sistema e distante do pretendido, uma vez que não apresenta as características necessárias para identificar a aplicação. No que toca às medidas de sucesso, a maioria são expostas como forma de requisitos. A maquete do sistema é apresentada de forma muito teórica, representando apenas um modelo MVC, não acrescentado nenhuma informação sobre o sistema a implementar. Quanto à identificação dos recursos necessários, esta encontra-se corretamente definida e as sugestões de tecnologias possíveis de ser usadas fornecem uma boa ajuda para a futura implementação. Por fim, o diagrama de GANTT encontra-se bem definido.
<b>A2</b>	<b>Levantamento e Análise de Requisitos</b> <i>O processo de requisitos foi bem desenvolvido? Foi revelada a forma como foi realizado o seu levantamento? Os requisitos estão claros, bem organizados e detalhados?</i>	<b>3</b>	Os requisitos contemplados não são suficientemente ricos em detalhe. É exigido como requisito a segurança do utilizador, no entanto não é explicado se o utilizador terá ou não de criar a conta na aplicação, sendo que não se percebe objetivamente o propósito que este requisito pretende atingir. Um dos requisitos passa pela capacidade de adicionar parques aos favoritos, no entanto não é explicado o modo como os mesmos ficam associados ao utilizador.
<b>A3</b>	<b>Especificação UML</b> <i>A especificação UML está organizada e completa? Acompanha de perto os requisitos previamente estabelecidos? Os principais casos de uso estão bem identificados e detalhados, bem como os atores envolvidos? Os objectos do sistema a desenvolver e os seus relacionamentos estão bem descritos?</i>	<b>5</b>	Apenas aspectos positivos se tem a apontar a este capítulo. É apresentada uma grande diversidade de diagramas UML, sendo todos eles ricos em detalhe e correção.
<b>A4</b>	<b>Sistema de Dados</b> <i>Os sistemas de dados estão apresentados de forma clara e sustentada? Os modelos apresentados foram devidamente enquadrados na especificação UML realizada?</i>	<b>2</b>	De acordo com o conhecimento adquirido para a elaboração da base de dados do Grupo 26, acredita-se que as relações entre as várias tabelas não se encontram corretamente definidas, uma vez que as multiplicidades se encontram trocadas. Para além disto, uma das principais características da aplicação, mencionada múltiplas vezes até este ponto - a possibilidade de selecionar parques como favoritos - encontra-se completamente ausente do sistema de dados apresentado, bem como as imagens que se encontram representadas apenas por um INT e as críticas que não são mencionadas nas tabelas (apenas a quantidade das mesmas).
<b>A5</b>	<b>Interfaces</b> <i>Os interfaces (mockups) do sistema são claros e foram devidamente explicados? É possível identificar a sua utilização no processo de especificação UML que foi realizado?</i>	<b>3</b>	Refere-se múltiplas vezes que a aplicação deve ser de fácil utilização para utilizadores com uma menor literacia tecnológica, no entanto os mockups apresentados revelam-se algo obsoletos assemelhando-se a aplicações criadas nos primórdios do desenvolvimento mobile. Deste modo, considera-se que possam elevar a dificuldade de utilização da aplicação devido às poucas semelhanças com aplicações atuais.
<b>A6</b>	<b>Implementação</b> <i>Com base na especificação realizada e na documentação disponibilizada é possível fazer a implementação do sistema requerido?</i>	<b>4</b>	A ideia revela-se interessante, cuidadosamente pensada e encontra-se bem suportada por diagramas UML, assim será possível implementar a aplicação apenas com a quantidade de documentação fornecida. No entanto, em certos requisitos e, especialmente, no modelo de dados faltam alguns parâmetros por especificar de modo a criar um sistema robusto. Como tal, a implementação segundo a documentação recebida não poderá ser seguida à risca de modo a serem colocados em prática todos os objetivos a qual esta se propõem a atingir, pois requer explicações mais detalhadas.

Total	21.00
Média	3.50

Figura 3.1: Grelha de avaliação ao Projecto MinhoPark

## Anexo 2

Use Case		
Use case:	Editar preferências	
Ator:	Utilizador	
Descrição:	O utilizador altera as preferências de pesquisa	
Cenários:	4	
Pré-condição:	True	
Pós-condição:	As preferências são atualizadas.	
	Utilizador	Sistema
Cenário normal	2. O utilizador edita as preferências (localização, tipo de parque, extensão da lista e opção de portagens).	1. O sistema mostra a página de edição.  3. O sistema atualiza as preferências.
Cenário de Exceção (1) [Utilizador insere coordenadas inválidas.] (passo 2)		2.1. O sistema informa o utilizador que as coordenadas são inválidas.

Figura 3.2: Use Case 1

Use Case		
Use case:	Solicitar direções para um parque	
Ator:	Utilizador	
Descrição:	O sistema mostra ao utilizador um percurso até ao parque	
Cenários:	5	
Pré-condição:	O utilizador está na página de informações do parque	
Pós-condição:	O sistema mostra o percurso até ao parque, podendo orientar o utilizador na sua deslocação	
	Utilizador	Sistema
Cenário normal	<p>1. «include» Mostrar mais informações sobre um dos parques.</p> <p>5. O utilizador escolhe a opção que pretende utilizar.</p>	<p>1. «include» Mostrar mais informações sobre um dos parques.</p> <p>2. O sistema obtém as coordenadas do utilizador.</p> <p>3. O sistema calcula os vários percursos até ao parque para os diferentes meios de deslocação.</p> <p>4. O sistema mostra as diferentes opções de deslocação ao utilizador (A PÉ/DE CARRO/DE TRANSPORTES PÚBLICOS) num mapa interativo onde o utilizador pode observar as diferenças entre os diferentes percursos.</p> <p>6. O sistema mostra o percurso no mapa, atualizando-o à medida que o utilizador se desloca.</p>
Cenário de Exceção (1) [Utilizador não escolhe nenhuma das opções e carrega no botão "VOLTAR":] (passo 5)		5.1. O sistema volta ao menu das informações sobre o parque.

Figura 3.3: Use Case 2

Use Case		
Use case:	Ver mais informações sobre um dos parques	
Ator:	Utilizador	
Descrição:	O utilizador obtém mais informações sobre um parque específico	
Cenários:	1	
Pré-condição:	Existe lista que responda às preferências do utilizador	
Pós-condição:	O utilizador obteve informação sobre um dos parques	
	Utilizador	Sistema
Cenário normal	1. O utilizador escolhe da lista o parque sobre o qual pretende obter mais informações	2. O sistema mostra as informações (nome, horário, 1 imagem, média e número de <i>ratings</i> , categorias de parque e endereço).

Figura 3.4: Use Case 3

Use Case		
Use case:	Adicionar parque à lista de favoritos	
Ator:	Utilizador	
Descrição:	Permite ao utilizador de guardar um parque na lista de favoritos	
Cenários:	2	
Pré-condição:	Parque a adicionar não está na lista de favoritos	
Pós-condição:	Existe um novo parque na lista de favoritos	
	Utilizador	Sistema
Cenário normal		1. O sistema verifica que o parque não pertence à lista de favoritos. 2. O sistema guarda o parque na lista de favoritos.

Figura 3.5: Use Case 4

Use Case		
Use case:	Retirar parque da lista de favoritos	
Ator:	Utilizador	
Descrição:	Permite ao utilizador remover o parque da lista de favoritos	
Cenários:	3	
Pré-condição:	O parque a remover está na lista de favoritos	
Pós-condição:	O parque não está na lista de favoritos	
	Utilizador	Sistema
Cenário normal		1. O sistema remove o parque da lista de favoritos

Figura 3.6: Use Case 5

Use Case		
Use case:	Aceder à lista de favoritos	
Ator:	Utilizador	
Descrição:	Permite ao utilizador aceder à lista de favoritos	
Cenários:	3	
Pré-condição:	True	
Pós-condição:	O sistema mostra a lista de favoritos	
	Utilizador	Sistema
Cenário normal		1. O sistema mostra a lista de favoritos.

Figura 3.7: Use Case 6

Use Case		
Use case:	Procura de parques com filtragem de preferências	
Ator:	Utilizador	
Descrição:	Permite ao utilizador pesquisar por parques utilizando uma filtragem de preferências	
Cenários:	4	
Pré-condição:	True	
Pós-condição:	O utilizador recebe uma lista de parques relevantes	
	Utilizador	Sistema
Cenário normal		1. O sistema procura por parques de acordo com as preferências escolhidas 2. O sistema mostra uma lista de parques relevantes, assim como um mapa com todos eles indicados.

Figura 3.8: Use Case 7