
Relatório de Especificação: Desenho

Projeto: ***CityParking***

Preparado por: Grupo 03
Bruno Caseiro (nº 88804)
Guilherme Sousa (nº 80000)
Pedro Alves (nº 88861)
Rita Amante (nº 89264)

Data de preparação: Aveiro, 16 novembro de 2019

Circulação: Docentes e Discentes de AMS.

Controlo de versões

Quando?	Responsável	Alterações significativas
20/11/19	88804	Introdução; Objetivos e restrições da arquitetura; Tecnologias e implementação.
20/11/19	89264	Vistas de arquitetura.
21/11/19	88861	Modelo de comportamento.
22/11/19	80000	Protótipos dos produtos.

Índice

1	Introdução.....	5
2	Objetivos e restrições da arquitetura.....	6
2.1	Objetivos da arquitetura face aos requisitos do domínio	6
2.2	Atributos de qualidade com impacto na arquitetura.....	6
2.3	Decisões tomadas e justificação	6
3	Vistas de arquitetura.....	7
3.1	Arquitetura lógica da solução.....	7
3.2	Arquitetura operacional	9
4	Modelo de comportamento.....	10
4.1	Interação entre módulos	10
4.2	Operações de sistema	10
4.3	Integrações com sistemas externos	11
5	Tecnologias e implementação.....	13
6	Protótipos dos produtos	14

Índice de figuras

Figura 1: Arquitetura lógica.	7
Figura 2: Arquitetura de deployment.	9
Figura 3: Diagrama de sequência a comunicação dos módulos.	10
Figura 4: Diagrama de sequência de autenticação.	10
Figura 5: Diagrama de sequência de procura de lugares por GPS.	11
Figura 6: Diagrama de interação entre o CityParking e a MBWay.	11
Figura 7: Diagrama de sequência de pagamento com MBWay.	12
Figura 8: Página de Início de Sessão.	14
Figura 9: Página de criação de conta.	14
Figura 10: Página de perfil do utilizador.	15
Figura 11: Página inicial de pesquisa do parque.	15

1 Introdução

O presente relatório tem por finalidade apresentar a arquitetura do projeto *CityParking*, expondo as tecnologias, linguagens e frameworks utilizados em detalhe. Apresentamos o porquê das decisões e opções tomadas e também o percurso do desenvolvimento do projeto.

2 Objetivos e restrições da arquitetura

2.1 Objetivos da arquitetura face aos requisitos do domínio

Existem alguns requisitos importantes que reduziram as nossas opções enquanto escolhemos a arquitetura ideal para o *CityParking*:

- O sistema deve ser compatível com os browsers das versões mais recentes de Android e iOS;
- Os requisitos de hardware são mínimos, sendo apenas necessário conseguir correr o browser do sistema operativo;
- O sistema MBWay, PayPal, Visa e MasterCard estão integrados no *CityParking* como métodos de pagamento;
- Os dados do utilizador devem estar bem protegidos, atendendo à Política de Proteção de Dados.

2.2 Atributos de qualidade com impacto na arquitetura

De seguida, serão enumerados alguns atributos de qualidade relativos à arquitetura do sistema:

1. A versão do Android deve ser 4.1 ou superior, no caso do iOS a versão deve ser 9.0 ou superior;
2. A interface deve ser simples e intuitiva para todas as faixas etárias (a partir dos 18 anos);
3. O sistema deve estar ativo 24/7, especialmente durante o dia, e manter a persistência em caso de falha no servidor;
4. A App Web deve suportar os browsers mais populares: Safari, Chrome, Samsung Browser;
5. A interface deve ser adaptável aos vários dispositivos e ecrãs.

2.3 Decisões tomadas e justificação

- **Compatibilidade com o API do Google Maps:** o *CityParking* depende bastante do Google Maps, tendo como página principal o mapa com os lugares de estacionamento disponíveis;
- **Integração com o MBWay:** os pagamentos serão feitos através da plataforma MBWay;
- **Integração com o PayPal:** os pagamentos serão feitos através da plataforma PayPal;
- **Integração com o VISA:** os pagamentos serão feitos através da plataforma Visa;
- **Integração com o MasterCard:** os pagamentos serão feitos através da plataforma MasterCard.

3 Vistas de arquitetura

3.1 Arquitetura lógica da solução

Para o bom funcionamento do sistema tem de haver uma estrutura capaz de recolher, transmitir, armazenar e gerir a informação. Para tal, são necessários vários elementos para o funcionamento do mesmo.

Na Arquitetura em Camadas a camada mais alta comunica com a camada mais baixa e assim sucessivamente. Logo, cada camada é apenas dependente da camada imediatamente abaixo.

Para este projeto foi implementada uma arquitetura em camadas, abaixo ilustrada.

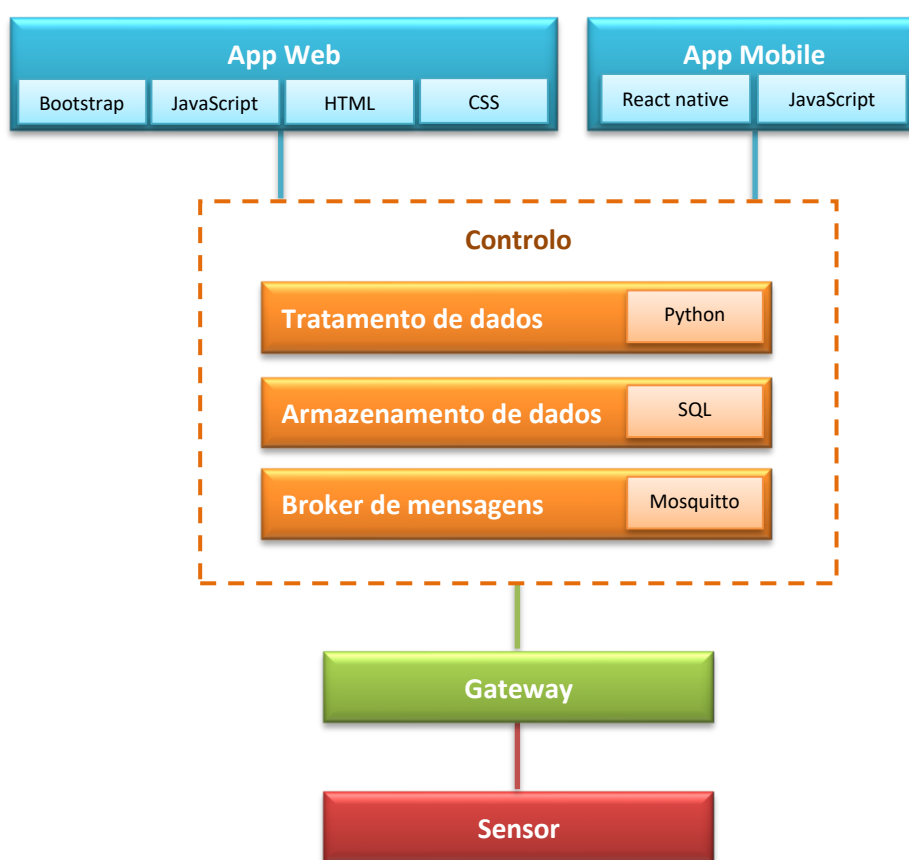


Figura 1: Arquitetura lógica.

De forma a compreender cada camada, será feita uma breve explicação:

1. Camada *Sensor*: camada que representa os sensores instalados nos parques;
2. Camada *Gateway*: camada que armazena informação dos vários sensores dos parques de estacionamento;
3. Camada *Controlo* abrange vários elementos:
 - 3.1. *Servidor*: cérebro do sistema e é através dele que todas as operações são controladas, sendo ele o responsável por toda a informação existente na base de dados de forma a analisá-la e processá-la;

- 3.2. *Base de dados*: reúne toda a informação de funcionamento do sistema;
- 3.3. *Broker de mensagens*: intermediário entre máquinas e os protocolos;
- 4. Camada *App*:
 - 4.1. *App Web*: aplicação que disponibiliza uma interface gráfica para visualização/ informação do sistema que engloba:
 - 4.1.1. *Gestão de clientes* - módulo que possui a informação sobre os clientes: parques de que são proprietários, tarifas, horários e localização;
 - 4.1.2. *Gestão dos utilizadores* - módulo que possui a informação sobre os utilizadores: nome, email, palavra-passe, número de contribuinte, telemóvel e morada;
 - 4.1.3. *Gestão de pagamentos* - módulo que possui a informação sobre todos os pagamentos e faz a interface com os métodos de pagamentos disponibilizados;
 - 4.1.4. *Gestão de negócios* - módulo que possui a informação que permite fazer a estatística: número de alugueres por dia, lucro, despesas, entre outros;
 - 4.2. *App Mobile*: aplicação que, a partir de um smartphone, permite ao utilizador realizar todas as ações necessárias para encontrar um lugar de estacionamento.

3.2 Arquitetura operacional

A arquitetura operacional tem como finalidade mostrar uma perspectiva de como a solução está apresentada de modo a responder aos requisitos especificados.

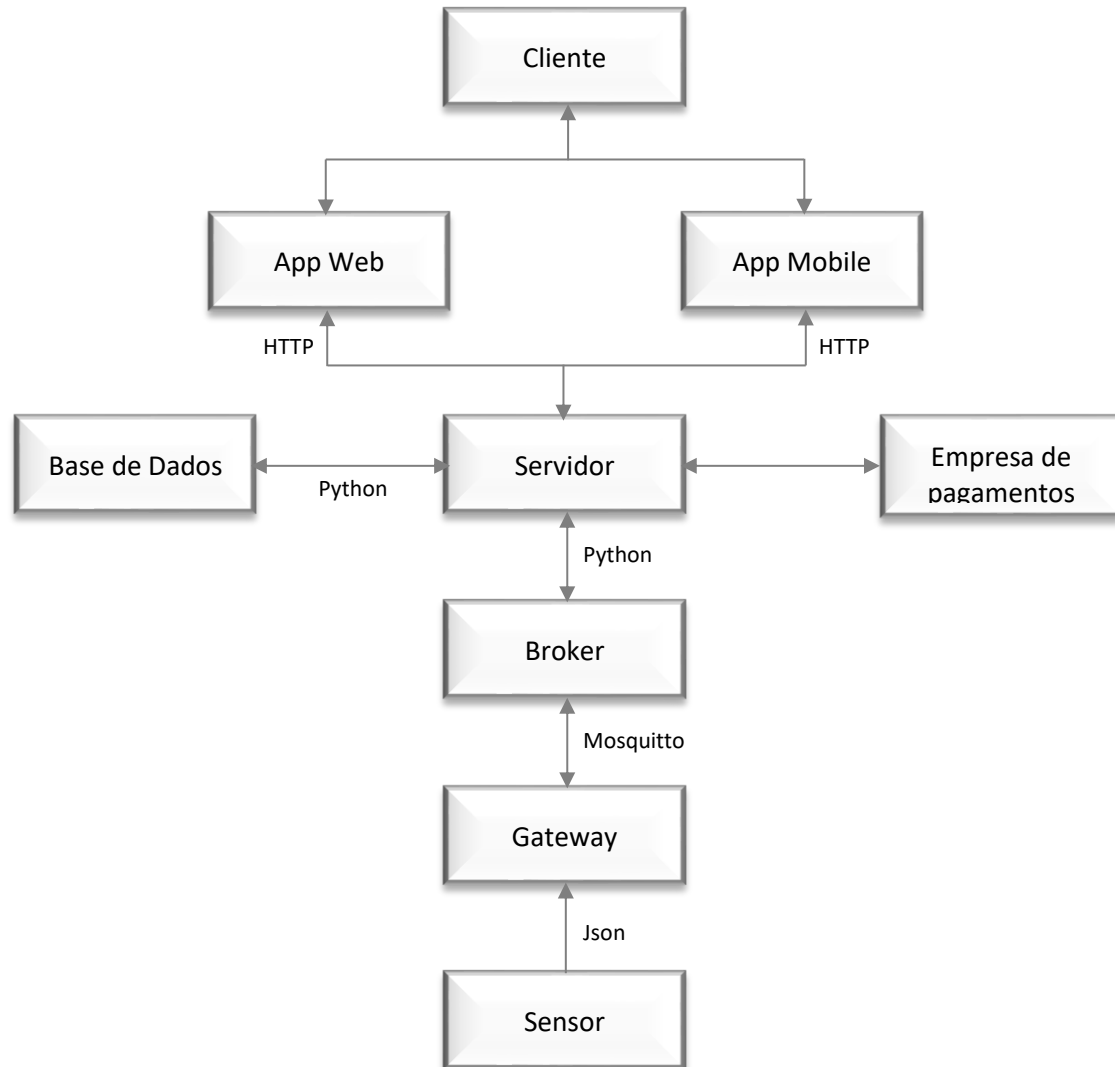


Figura 2: Arquitetura de deployment.

4 Modelo de comportamento

4.1 Interação entre módulos

Na figura a baixo pode-se ver de forma genérica através de um diagrama de sequência a comunicação dos módulos mencionados em 3.1.

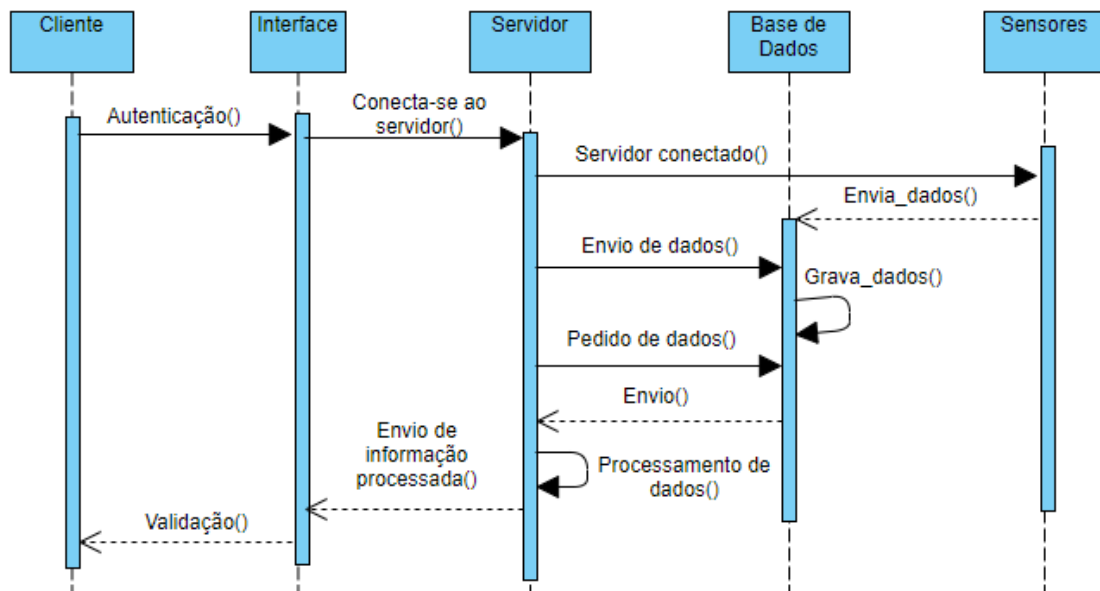


Figura 3: Diagrama de sequência a comunicação dos módulos.

4.2 Operações de sistema

Nesta iteração foi implementada a parte de autenticação do *CityParking* e mostragem dos parques disponíveis através de GPS. Nesta secção vão ser apresentados os diagramas de sequência correspondentes. Os restantes casos de uso serão implementados numa fase posterior.

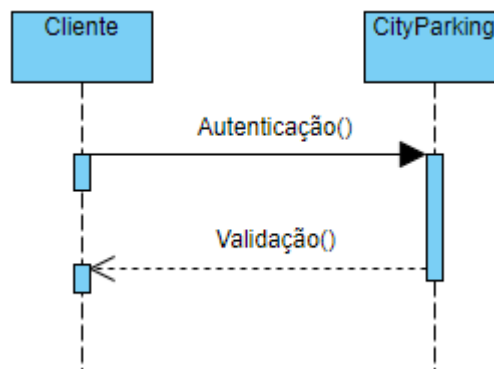


Figura 4: Diagrama de sequência de autenticação.

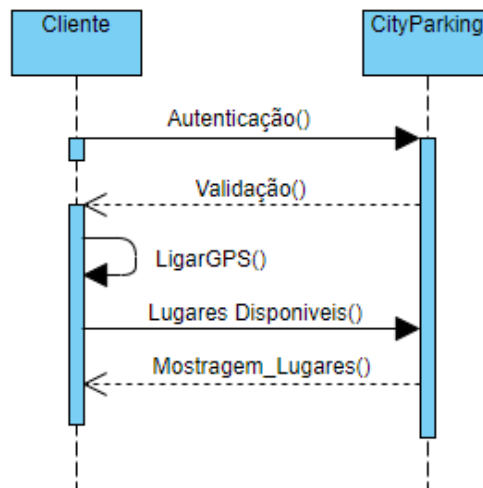


Figura 5: Diagrama de sequência de procura de lugares por GPS.

4.3 Integrações com sistemas externos

A equipa optou, nesta fase, por utilizar a plataforma MBway como forma de pagamento do serviço, dado a sua atual popularidade e fácil utilização.

De seguida, será demonstrada a interação entre o nosso sistema e o referido sistema de software.

Para o processo inicial:

1. Criar uma conta MBWay para a empresa *CityParking*
 - a) Definição de dados da empresa;
 - b) Informação bancária da empresa;
2. Entrar na conta MBWay criada e escolher um botão;
3. Adicionar o nome do Parque de Estacionamento, lugar e valor;
4. Copiar o código HTML gerado para criar o botão;
5. Colar o código no site *CityParking* para a conclusão do processo.

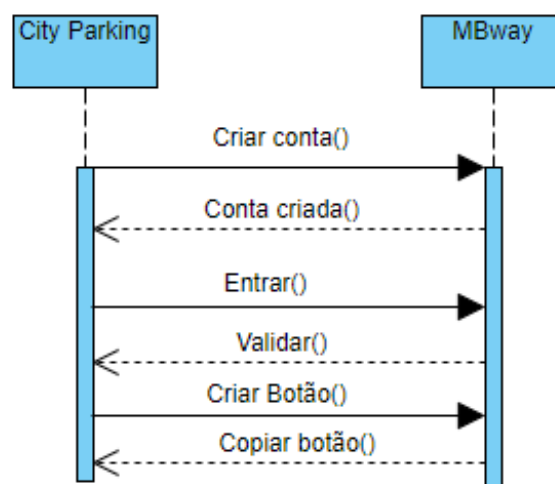


Figura 6: Diagrama de interação entre o CityParking e a MBWay.

Neste momento, os lugares de estacionamento serão possíveis ser pagos através de MBWay, os utilizadores deverão seguir o seguinte fluxo:

1. Visitam a aplicação *CityParking* e escolhem o parque desejado;
2. Efetuam o pagamento através de MBWay;
3. Iniciam sessão na conta MBWay;
4. Confirmam compra do título.

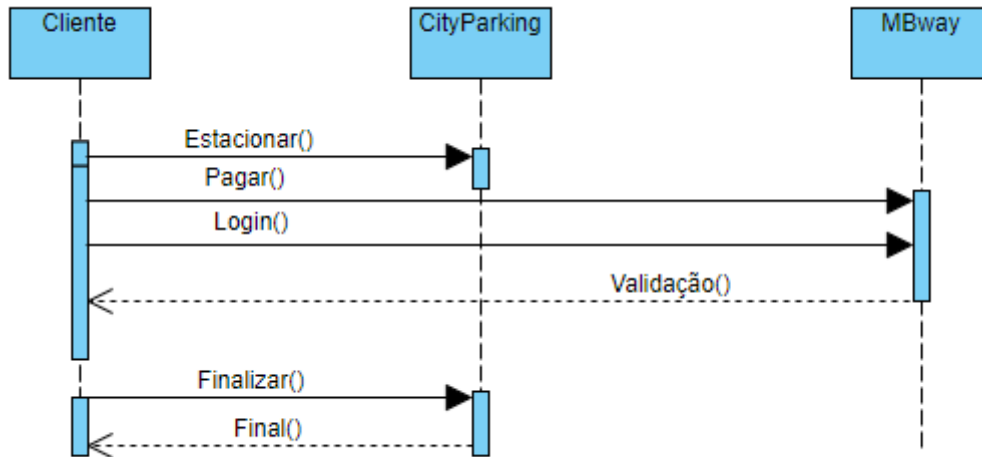


Figura 7: Diagrama de sequência de pagamento com MBWay.

5 Tecnologias e implementação

O *CityParking* será uma “Progressive Web Application”, uma aplicação destinada a vários sistemas operativos que executa através do browser. Esta App Web será construída com as linguagens HTML, CSS e JavaScript.

A aplicação será construída através do “React Native”, uma biblioteca JavaScript para User Interfaces dinâmicos, sem a necessidade de atualizar a página. O “React Native” é adaptável aos vários dispositivos, simples e rápido.

É usado também o API do Google Maps, que permite personalizar o mapa com o nosso próprio conteúdo e exibi-lo na nossa página. A implementação é feita através do JavaScript.

Para facilitar o pagamento, será implementado o API oficial do MBWay.

De forma a colaborar mutuamente de forma simplificada entre a equipa, é usado o sistema de controlo de versões GitHub.

Os sensores de proximidade serão também implementados nos parques de estacionamento de forma a detetar a disponibilidade de cada lugar. A informação recolhida por estes dispositivos será apresentada no mapa do *CityParking*, dando a possibilidade do utilizador verificar se o lugar está livre.

6 Protótipos dos produtos

As figuras abaixo apresentadas são algumas capturas realizadas sobre o protótipo. Para ver a implementação do protótipo recomenda-se a leitura de [‘README’](#).

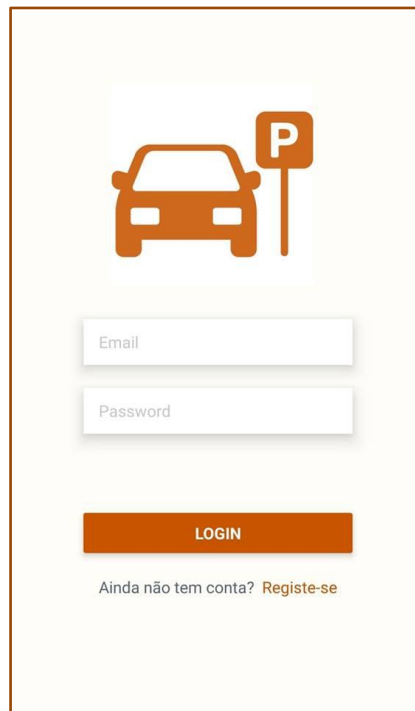
A screenshot of a login screen prototype. At the top, there is an orange icon of a car parked next to a 'P' parking sign. Below the icon, there are two white input fields with orange borders, labeled 'Email' and 'Password'. Underneath these fields is a solid orange button with the text 'LOGIN' in white. At the bottom, there is a link that says 'Ainda não tem conta? Registe-se'.

Figura 8: Página de Início de Sessão.

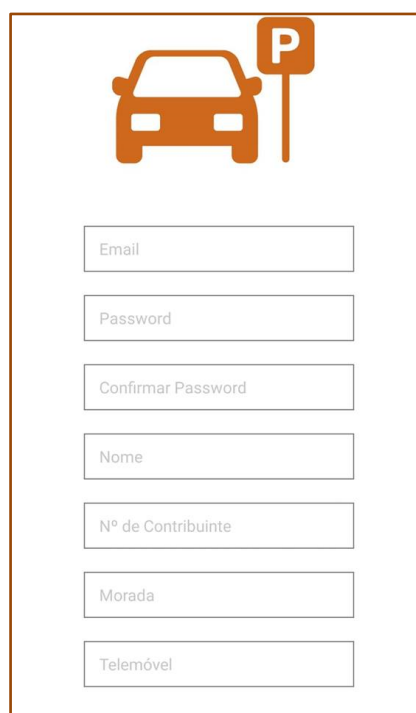
A screenshot of a registration screen prototype. At the top, there is an orange icon of a car parked next to a 'P' parking sign. Below the icon, there are seven white input fields with orange borders, stacked vertically. The labels for these fields are 'Email', 'Password', 'Confirmar Password', 'Nome', 'Nº de Contribuinte', 'Morada', and 'Telemóvel'.

Figura 9: Página de criação de conta.

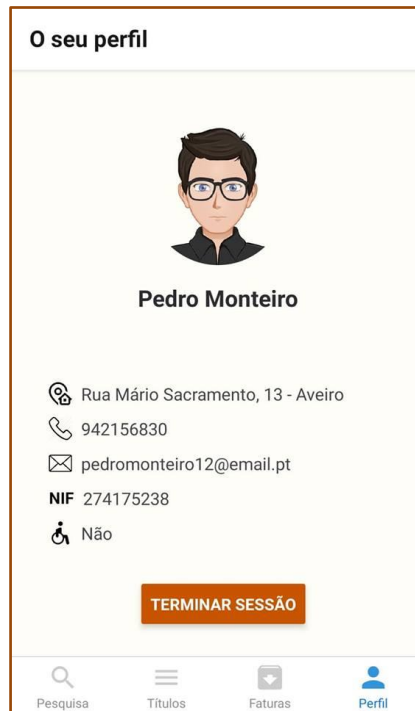


Figura 10: Página de perfil do utilizador.

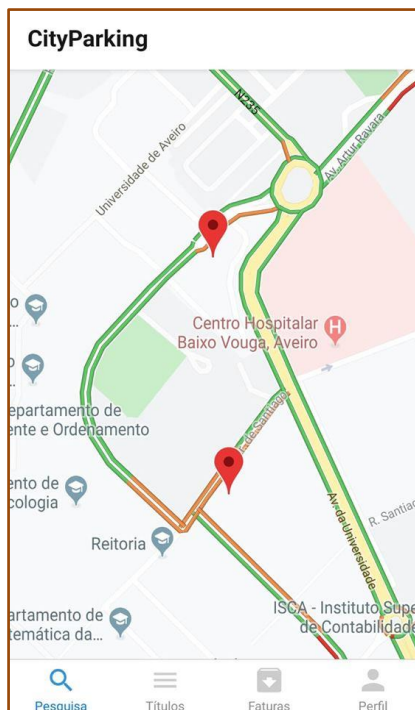


Figura 11: Página inicial de pesquisa do parque.