**SMART PARK**



**Projeto de Desenvolvimento de Software**

**Licenciatura em Engenharia de Sistemas Informáticos**

**2022/2023**

**Alunos**

Rosário Silva – Nº 21138

Mário Pinto – Nº 23506

Rui Pinto – Nº 18450

Rúben Costa – Nº 23508

Renato Silva – Nº 23511

**Orientação**

Profº Nuno Rodrigues

Índice

[**1** **Introdução** 4](#_Toc129449508)

[**2** **Organização de Grupo** 6](#_Toc129449509)

[**2.1** **Regulamento Interno** 6](#_Toc129449510)

[**2.2** **Cronograma** 7](#_Toc129449511)

[**3** **Proposta de Sistema** 8](#_Toc129449512)

[**3.1** **Regras de Negócios** 8](#_Toc129449513)

[**3.2** **Requisitos Não Funcionais** 9](#_Toc129449514)

[**4** **Levantamento de Requisitos** 10](#_Toc129449515)

[**4.1** **Diagrama de Use-Case** 10](#_Toc129449516)

[**4.1.1** **Administrador** 10](#_Toc129449517)

[**4.1.2** **Utilizador** 11](#_Toc129449518)

[**4.2** **Diagrama de Entidade-Relação** 12](#_Toc129449519)

[**4.3** **Diagrama de Classe** 13](#_Toc129449520)

[**4.4** **Diagrama de Sequência** 14](#_Toc129449521)

[**4.4.1** **Login** 14](#_Toc129449522)

[**4.4.2** **Reserva** 15](#_Toc129449523)

[**4.4.3** **Registar** 15](#_Toc129449524)

[**5** **Prototipagem** 16](#_Toc129449525)

[**5.1** **Software Utilizado** 16](#_Toc129449526)

[**5.2** **Mockups** 16](#_Toc129449527)

[**5.2.1** **Login** 16](#_Toc129449528)

[**5.2.2** **Registar** 17](#_Toc129449529)

[**5.2.3** **Página Inicial** 18](#_Toc129449530)

[**5.2.4** **Dashboard Admin** 19](#_Toc129449531)

[**5.2.5** **Dashboard User** 21](#_Toc129449532)

[**5.3** **Resultado obtido** 24](#_Toc129449533)

[**6** **Conclusão** 25](#_Toc129449534)

[**7** **Bibliografia** 26](#_Toc129449535)

Índice de Ilustrações

[Figura 1- Cronograma do projeto 7](#_Toc129449487)

[Figura 2 - Use-Case - Administrador 10](#_Toc129449488)

[Figura 3 – Use-Case - Utilizador 11](#_Toc129449489)

[Figura 4 - Entidade-Relação 12](#_Toc129449490)

[Figura 5 - Classe 13](#_Toc129449491)

[Figura 6 - Sequência - Login 14](#_Toc129449492)

[Figura 7 - Sequência - Reserva 15](#_Toc129449493)

[Figura 8 - Sequência - Registar 15](#_Toc129449494)

[Figura 9 - Mockup Login 16](#_Toc129449495)

[Figura 10 - Mockup Registar 17](#_Toc129449496)

[Figura 11 - Mockup Registar Veículos 17](#_Toc129449497)

[Figura 12 - Mockup Página Inicial 18](#_Toc129449498)

[Figura 13 - DashBoard Administrador 19](#_Toc129449499)

[Figura 14 - DashBoard Administrador (Alertas) 19](#_Toc129449500)

[Figura 15 - DashBoard Administrador (Reservas) 20](#_Toc129449501)

[Figura 16 - DashBoard Administrador (Utilizadores) 20](#_Toc129449502)

[Figura 17 - DashBoard Administrador (Veículos) 21](#_Toc129449503)

[Figura 18 - DashBoard Utilizador (A minha conta) 21](#_Toc129449504)

[Figura 19 - DashBoard Utilizador (Lugares Disponíveis) 22](#_Toc129449505)

[Figura 20 - DashBoard Utilizador (Reserva Lugar) 22](#_Toc129449506)

[Figura 21 - DashBoard Utilizador (Veículos) 23](#_Toc129449507)

# **Introdução**

No âmbito da UC de Projeto de Desenvolvimento de Software, pretende-se desenvolver um projeto que visa usar os meios tecnológicos da atualidade para tornar um dos setores do nosso campus mais eficiente, tanto em termos funcionais como económicos.

O projeto tem como objetivo melhorar e inovar os parques de estacionamento do IPCA. Passará por arranjar uma solução para os problemas em estacionar dos alunos, professores e todas as pessoas pertencentes ao IPCA.

Decidimos desenvolver todo um projeto em que fosse possível existir uma visualização das vagas nos parques, tal como acontece nos parques de estacionamento dos centros comerciais e supermercados.

Devido ao resultado de um questionário feito no IPCA juntamente com a nossa experiência em estacionar no IPCA, denotamos que os parques estavam a ser mal geridos. A administração pretende criar um sistema que fosse possível existir uma visualização das vagas nos parques, de forma a reduzir o tempo de procura de estacionamento e evitar o atraso às aulas.

Isto gera diversos problemas, para isso tem de ser criadas as condições necessárias, tais como:

* Sensores para entradas e saídas dos parques
* Câmara para identificar a matrícula
* Barreiras para um melhor controlo dos carros que entram no parque
* Aprovação e colaboração do IPCA.

Dito isto, iriamos oferecer aos alunos uma aplicação móvel ou via website que daria acesso as vagas existentes dentro de cada parque assim, no caso de não haver nenhum lugar disponível o aluno não gastaria combustível a procurar. Para registarmos as entradas e saídas dos carros utilizaríamos um sensor em cada zona dos parques em todo o campus IPCA. Na entrada de cada parque será lida as matrículas dos carros. o carro será introduzido na base de dados para saber em que zona do parque o carro entrou, juntamente com as horas de entrada. Na saída de cada parque teremos outro sensor onde registaria na base de dados outro registo de saída do parque com a matrícula e a respetiva hora do acontecimento.

O nosso objetivo neste projeto consistirá em criar uma divisão de todo o parque do IPCA em 3, onde cada parque, preferencialmente o mais perto de cada polo, terá o nome de EST, ESG, ESD. Em cada parque indicará o número de vagas disponíveis do mesmo e vai sempre sugerir ao utilizador o parque mais perto do polo que ele frequenta, mas não o irá proibir de estacionar em qualquer um dos outros parques.

* Evitar atrasos às aulas
* Reduzir o transito no Campus
* Obter uma melhor gerência dos parques do Campus

# **Organização de Grupo**

## **Regulamento Interno**

**Artigo 1º - Âmbito do Documento**

Esta parte do documento será utilizada para partilhar a constituição e funcionamento do nosso grupo na elaboração do projeto proposto pelo Professor Nuno Rodrigues na Unidade Curricular de Projeto de Desenvolvimento de Software. Poderá também ser observado os deveres e obrigações de todos os elementos deste grupo. Esta documentação poderá estar sujeita a alterações ao longo das diversas fases de entrega, pelo que deve ser verificada pontualmente.

**Artigo 2º - Constituição do Grupo**

O nosso grupo é constituído por cinco elementos, nomeadamente: Rosário Silva; Mário Pinto; Rui Pinto; Rúben Costa e Renato Silva. Os nossos contactos são, respetivamente: [a21138@alunos.ipca.pt](mailto:a21138@alunos.ipca.pt) ; [a23506@alunos.ipca.pt](mailto:a23506@alunos.ipca.pt) ; [a18450@alunos.ipca.pt](mailto:a18450@alunos.ipca.pt); [a23508@alunos.ipca.pt](mailto:a23508@alunos.ipca.pt) e [a23511@alunos.ipca.pt](mailto:a23511@alunos.ipca.pt)

O nosso orientador é o Professor Nuno Rodrigues que, de forma regular, tem auxiliado na construção e organização do projeto.

**Artigo 3º - Cargos e regularidade da mudança dos mesmos**

O Product Owner é responsável pela adequação da implementação das funcionalidades com a especificação do sistema.

O Scrum Master é responsável pela definição e estruturação das sprints.

O Development team member é responsável pela implementação de tarefas das sprints.

**Artigo 4º - Reuniões**

As reuniões dão lugar nas aulas de Projeto de Desenvolvimento de Software, ou seja, duas vezes por semana.

Desta forma é possível garantir a presença e disponibilidade de todos os elementos, tal como a presença do nosso Orientador.

É de frisar que o ponto inframencionado poderá não ser cumprido mediante compromissos de cariz pessoal ou escolar. Nesse caso as reuniões serão marcadas para uma data a definir através dos nossos canais de comunicação direta.

## **Cronograma**

De forma a exacerbar a organização interna, foi efetuado um cronograma com as datas relevantes para o melhoramento contínuo do nosso projeto. Este setor poderá, ao longo do tempo, ser alterado mediante as necessidades e prazos de entrega do nosso trabalho contínuo.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Seg** |  | **Ter** |  | **Qua** |  | **Qui** |  | **Sex** |  | **Sab** |  | **Dom** |  |
| **Fev'**  **23** | 13 |  | 14 |  | 15 |  | 16 |  | 17 | R1 | 18 |  | 19 |  |
| 20 | R2 | 21 |  | 22 |  | 23 |  | 24 | R3 | 25 |  | 26 |  |
| 27 | R4 | 28 |  |  |  |  |  |  |  |  |  |  |  |
| **Mar'**  **23** |  |  |  |  | 1 |  | 2 |  | 3 | R5 | 4 |  | 5 |  |
| 6 | R6 | 7 |  | 8 |  | 9 |  | 10 | R7 | 11 |  | 12 | Entrega |
| 13 | R8 | 14 |  | 15 |  | 16 |  | 17 | R9 | 18 |  | 19 |  |
| 20 | R10 | 21 |  | 22 | Entrega | 23 |  | 24 | R11 | 25 |  | 26 |  |
| 27 | R12 | 28 |  | 29 |  | 30 |  | 31 | R13 |  |  |  |  |
| **Abr'**  **23** |  |  |  |  |  |  |  |  |  |  | 1 | Entrega | 2 |  |
| 3 | R14 | 4 |  | 5 |  | 6 |  | 7 | R15 | 8 |  | 9 |  |
| 10 | R16 | 11 | Entrega | 12 |  | 13 |  | 14 | R17 | 15 |  | 16 |  |
| 17 | R18 | 18 |  | 19 |  | 20 |  | 21 | Entrega | 22 |  | 23 |  |
| 24 | R19 | 25 |  | 26 |  | 27 |  | 28 | R20 | 29 |  | 30 |  |
| **Mai'**  **23** | 1 | Entrega | 2 |  | 3 |  | 4 |  | 5 | R21 | 6 |  | 7 |  |
| 8 | R22 | 9 |  | 10 |  | 11 | Entrega | 12 | R23 | 13 |  | 14 |  |
| 15 | R24 | 16 |  | 17 |  | 18 |  | 19 | R25 | 20 |  | 21 | Entrega |
| 22 | R26 | 23 |  | 24 |  | 25 |  | 26 | R27 | 27 |  | 28 |  |
| 29 | R28 | 30 |  | 31 | Entrega |  |  |  |  |  |  |  |  |

Figura - Cronograma do projeto

# **Proposta de Sistema**

Abaixo poderá ser observado o ponto da situação atual, em termos das Regras de Negócios e Requisitos Não Funcionais que foram prontamente levantados pela nossa equipa para obter, desta forma, a fundação daquilo que é esperado, uma vez que o projeto esteja na fase de lançamento.

## **Regras de Negócios**

As Regras de Negócios (RN) são a fonte de muitas decisões de negócios. Pode mudar ao longo do tempo e pode ter impacto em mais do que no caso de uso. Portanto, é importante identificá-los e rastreá-los.

* **RN#01: Reservas –**
* **RN#02: Estatísticas –**
* **RN#03: Multas –**

## **Requisitos Não Funcionais**

Por outro lado, os Requisitos Não Funcionais definem o que é que o sistema fará mais concretamente. São premissas, restrições técnicas e necessidades que não podem ser atendidas através de funcionalidades. Estes Requisitos Não Funcionais (RNF) inerentes ao projeto, estão associados à qualidade e segurança da aplicação que garante o funcionamento otimizado de todo o sistema.

* **RNF#01: A palavra-passe tem de ser encriptada –** De forma a garantir a segurança de todos os atores do sistema, será necessário proceder à encriptação das palavras-passe inseridas na base de dados.
* **RNF#02: O site pode ser acessível com todos os browsers –** Com o aumento da diversidade de browsers utilizados pelo público, é um fator importante garantir a acessibilidade e eficiência em qualquer browser.
* **RNF#03: O site tem de ser interativo e fácil uso –** Dado que os utilizadores do sistema poderão não ter todos a mesma experiência a nível tecnológico, o site deverá ser simples e fácil de manusear.
* **RNF#04: A página em termos de performance tem de aguentar com um número de 50 pessoas –** Em horas de ponta, será uma necessidade ter um sistema capaz de aguentar com vários utilizadores em simultâneo, para que todos consigam ter uma experiência sem falhas a qualquer momento.

# **Levantamento de Requisitos**

## **Diagrama de Use-Case**

Este diagrama foi concebido para demonstrar os casos de uso presentes no nosso projeto. Este, encontra-se dividido em dois sistemas, que se encontrão interligados entre si.

### **Administrador**

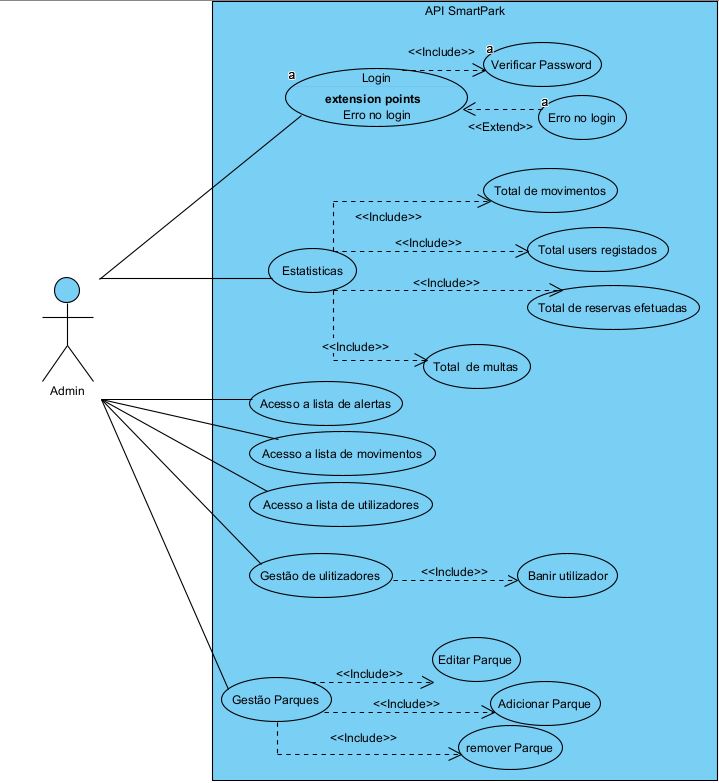


Figura - Use-Case - Administrador

### **Utilizador**

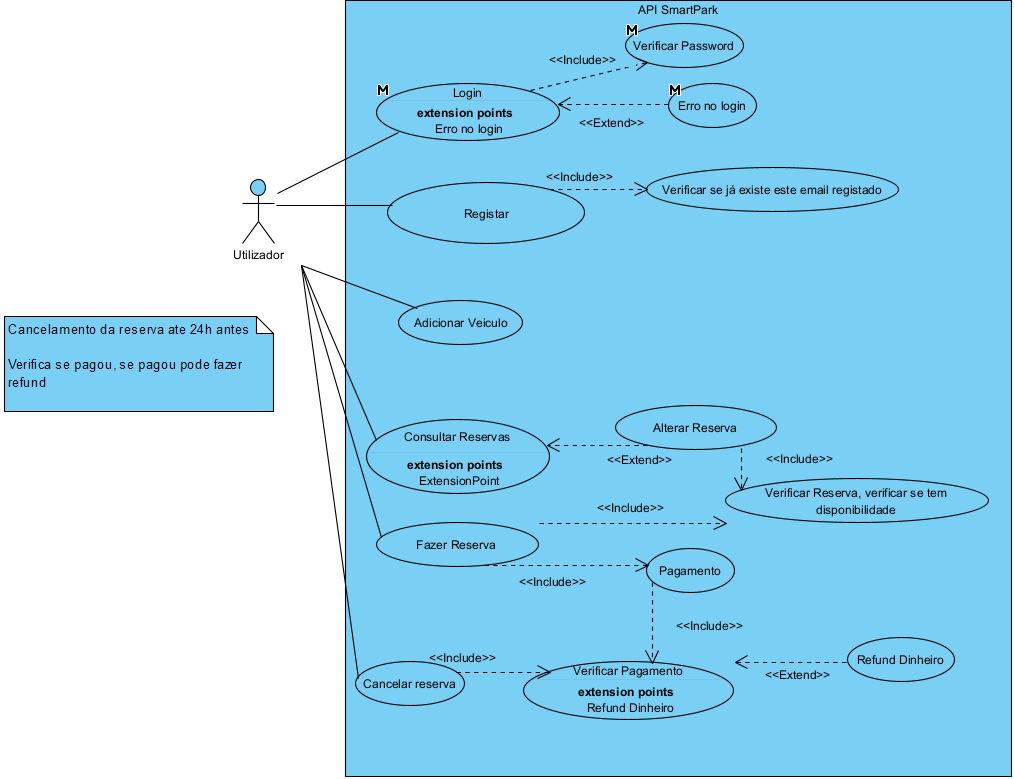


Figura – Use-Case - Utilizador

## **Diagrama de Entidade-Relação**

O diagrama Entidade-Relação foi construído já com a concessão da base de dados em mente para que todo o projeto seja alvo de análise posterior.

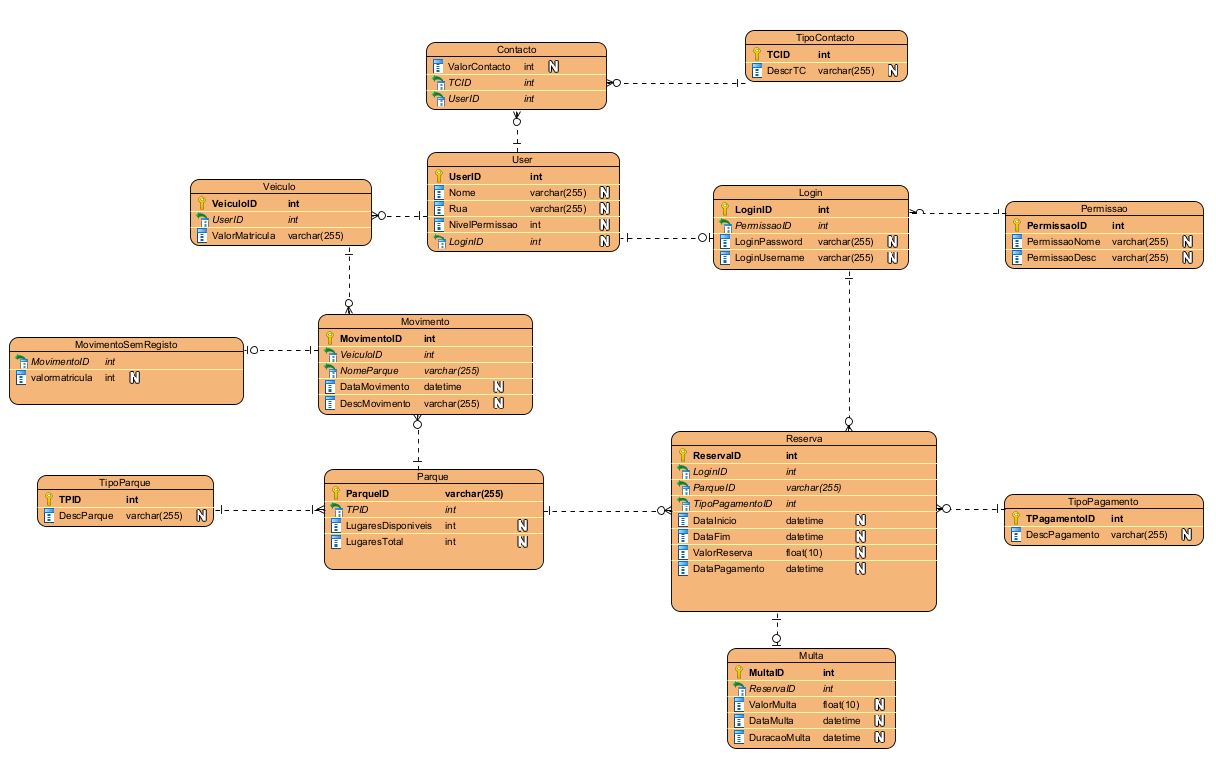


Figura - Entidade-Relação

## **Diagrama de Classe**

Os diagramas de classe são um tipo de diagrama que consiste na descrição de um conjunto de objetos que têm os mesmos atributos, operações, relacionamentos e semântica. Esta abordagem é orientada por objetos que leva a arquiteturas a serem mais estáveis.

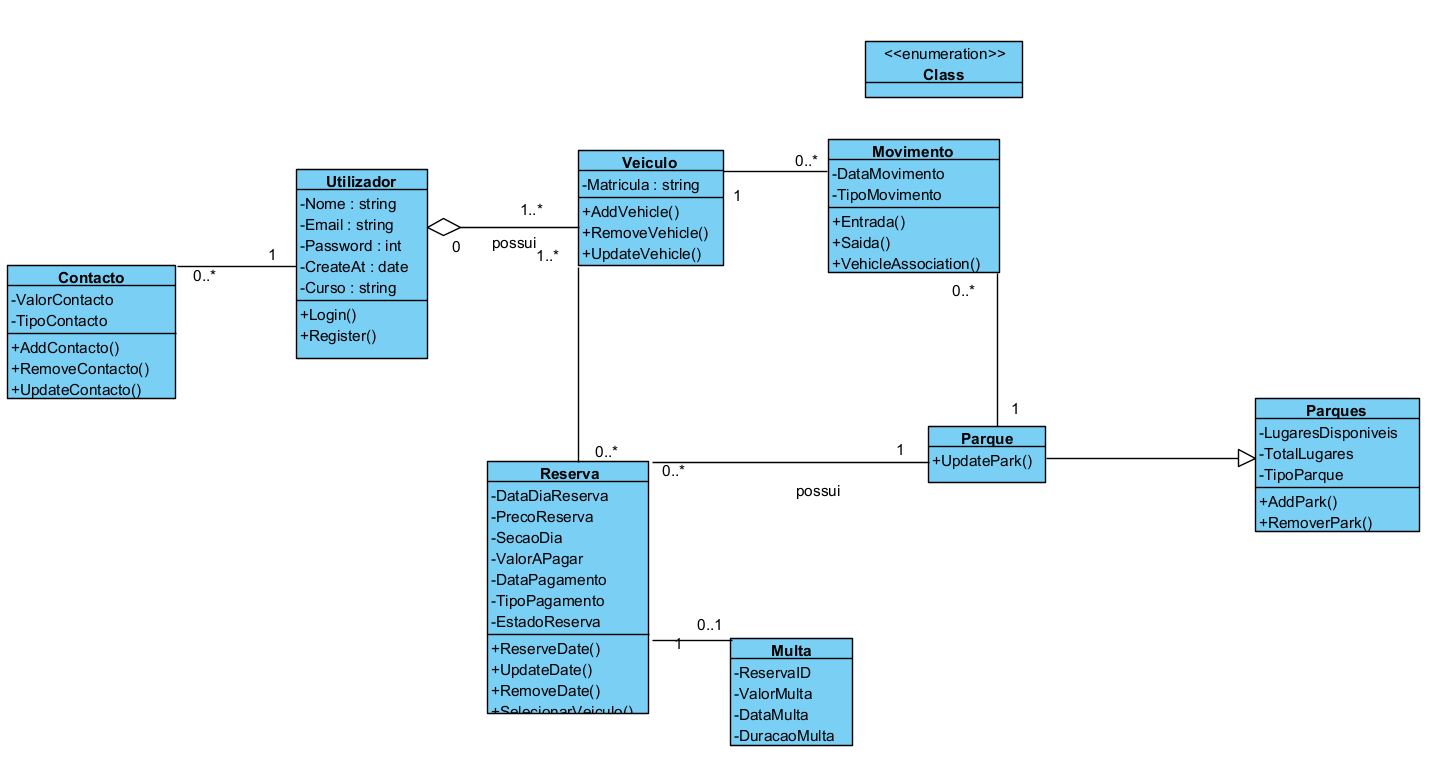


Figura - Classe

## **Diagrama de Sequência**

Os diagramas de sequência são um tipo de diagrama que mostra a interação entre diferentes entidades no nosso sistema. Ele é usado para modelar a sequência de mensagens trocadas entre as entidades no nosso sistema, bem como a ordem em que essas mensagens são trocadas.

### **Login**

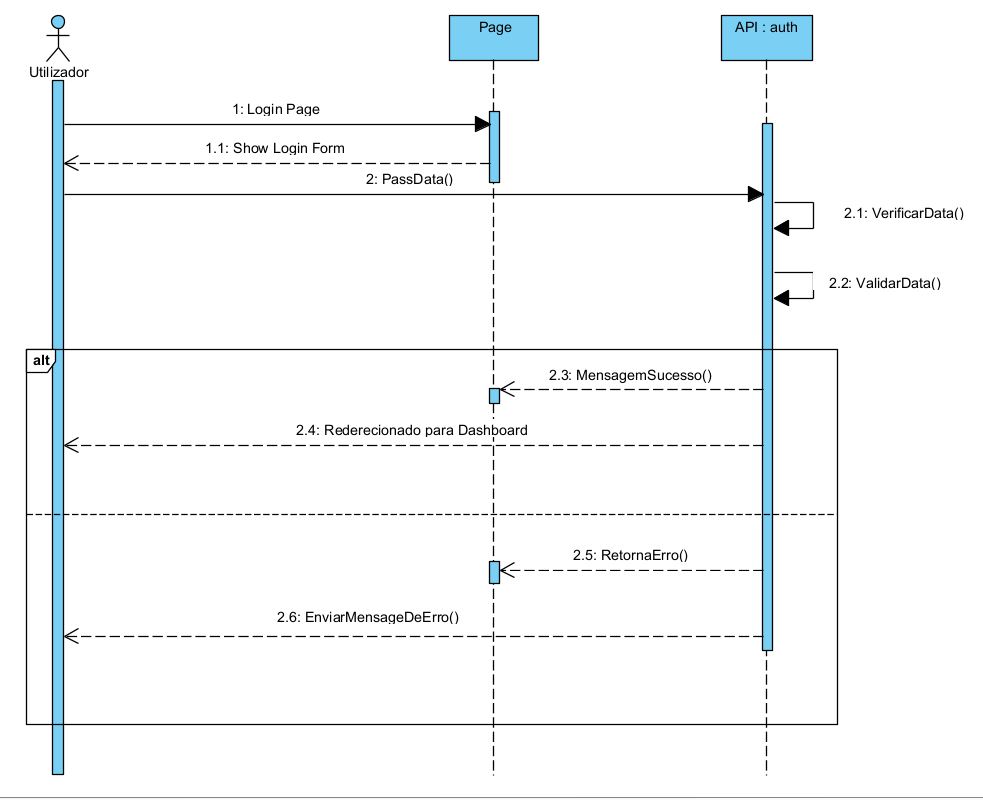


Figura - Sequência - Login

### **Reserva**

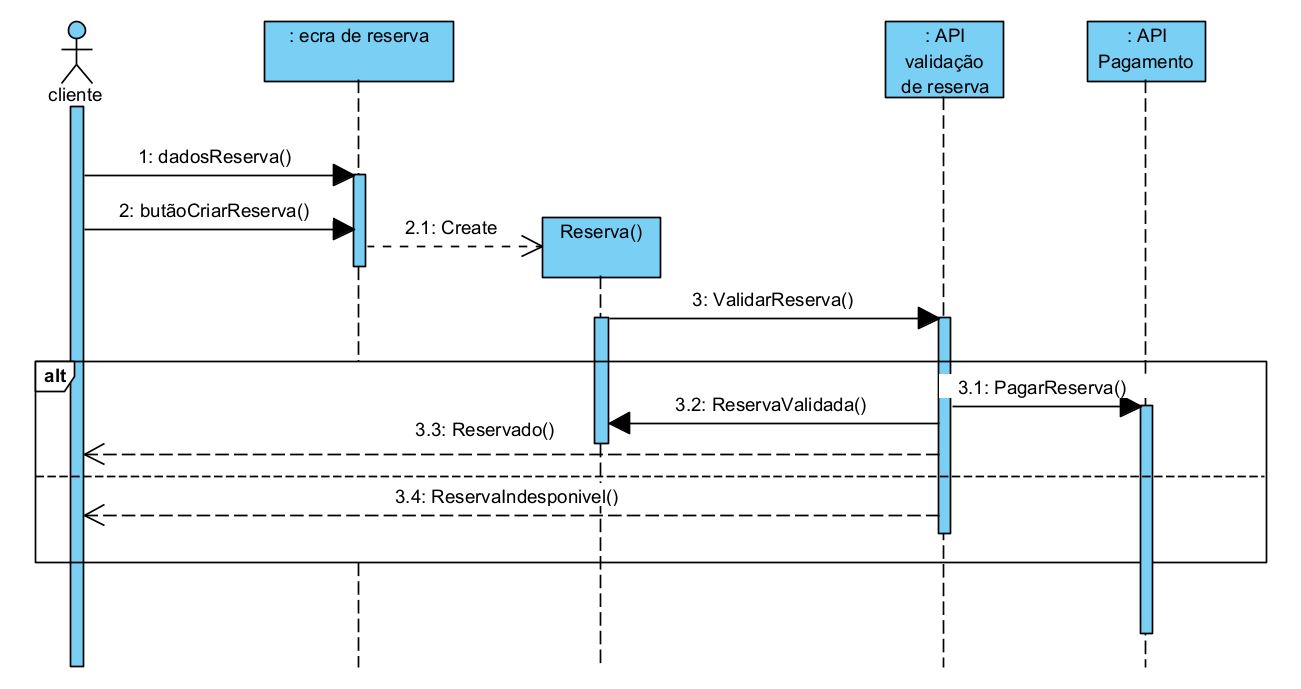


Figura - Sequência - Reserva

### **Registar**

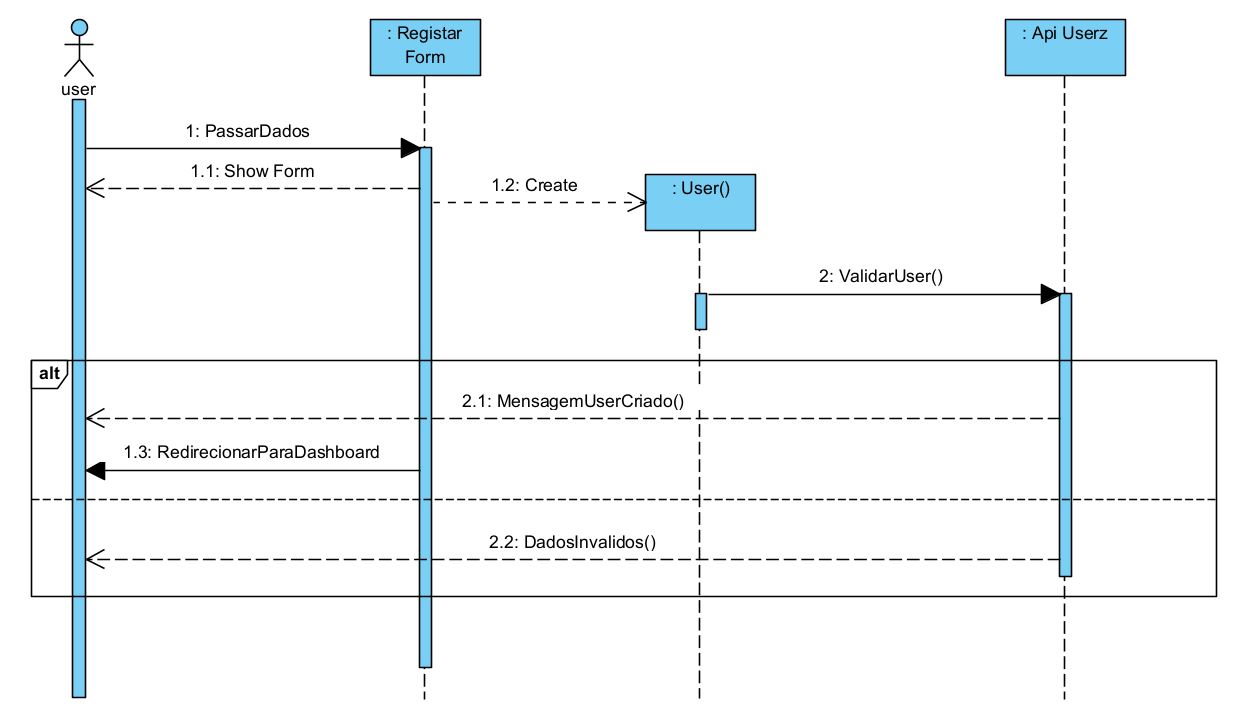


Figura - Sequência - Registar

# **Prototipagem**

Tendo em vista o plano final do projeto, já foram efetuados alguns protótipos daquilo que o nosso projeto visa atingir.

## **Software Utilizado**

Para o constante progresso deste projeto e, de forma a elaborar uma prova de conceito, foi utilizado um software, Figma que nos permitiu estabelecer o que foi planeado através dos diversos conceitos previamente discutidos.

## **Mockups**

### **Login**

Uma imagem com texto

Descrição gerada automaticamente

Figura - Mockup Login

### **Registar**

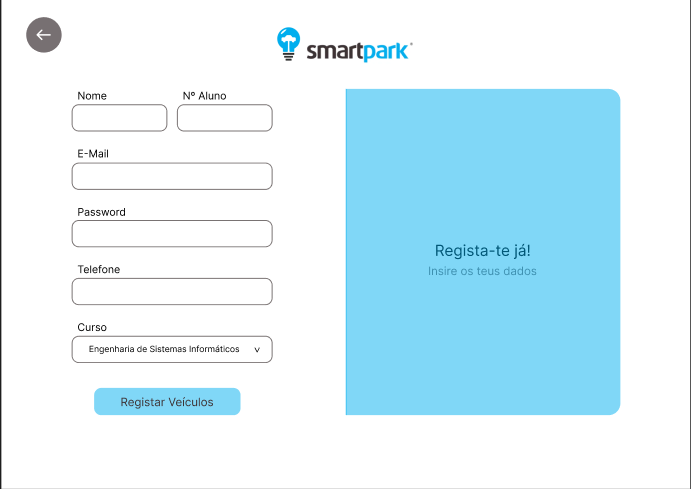


Figura - Mockup Registar

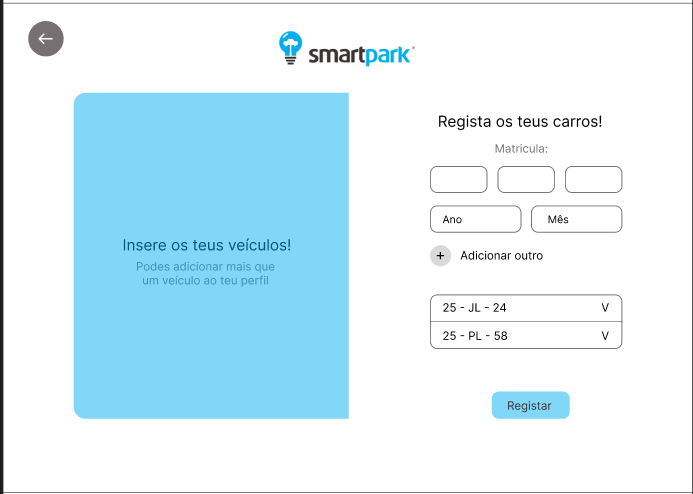


Figura - Mockup Registar Veículos

### **Página Inicial**

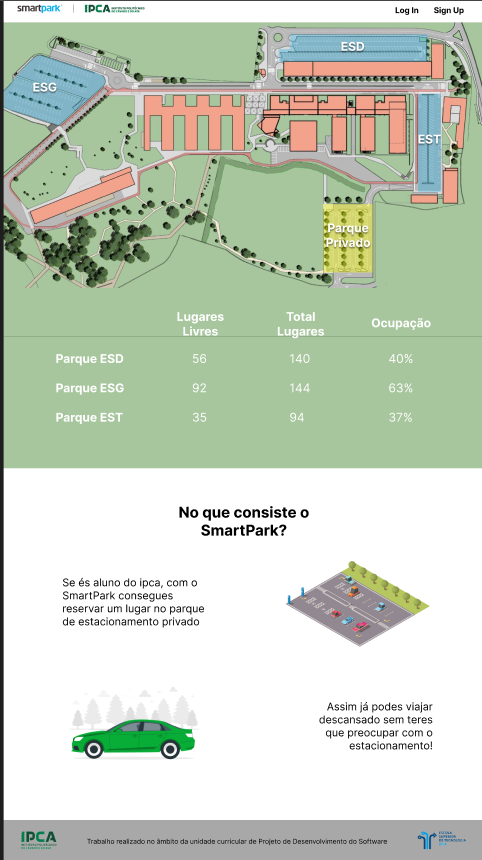


Figura - Mockup Página Inicial

### **Dashboard Admin**

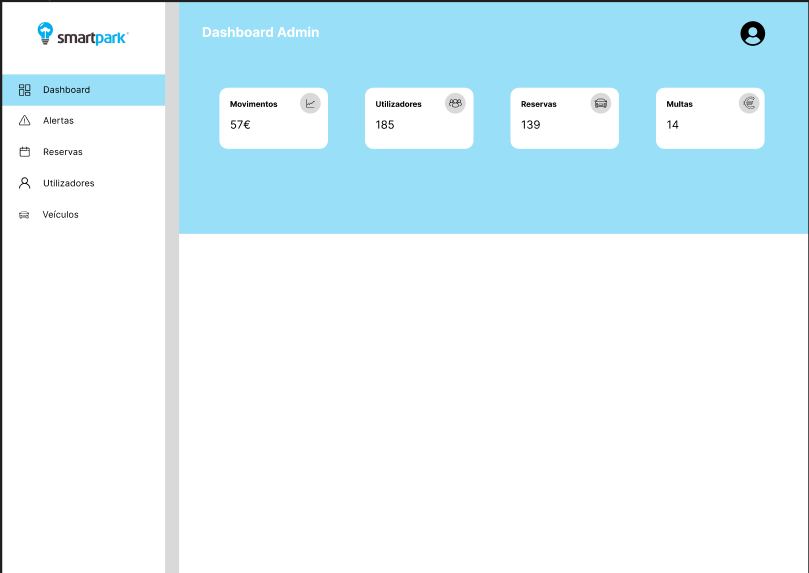


Figura - DashBoard Administrador

#### **Dashboard Admin (Alertas)**

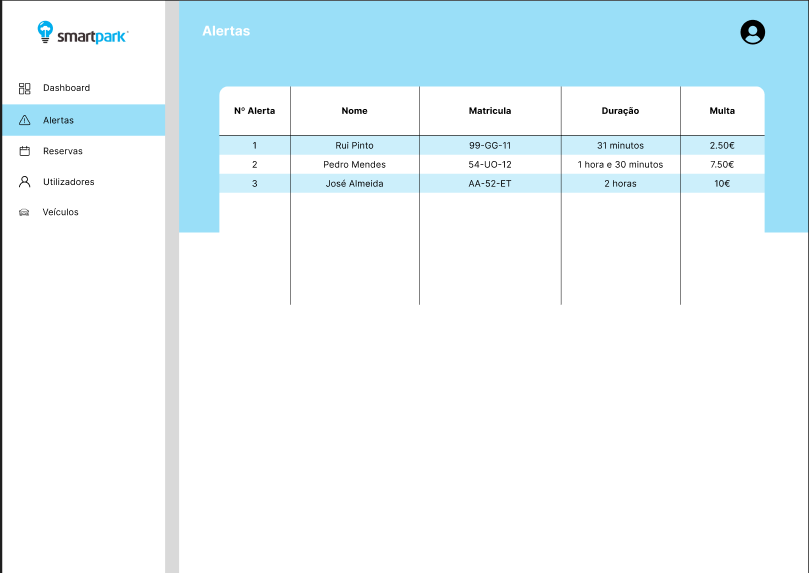


Figura - DashBoard Administrador (Alertas)

#### **Dashboard Admin (Reservas)**

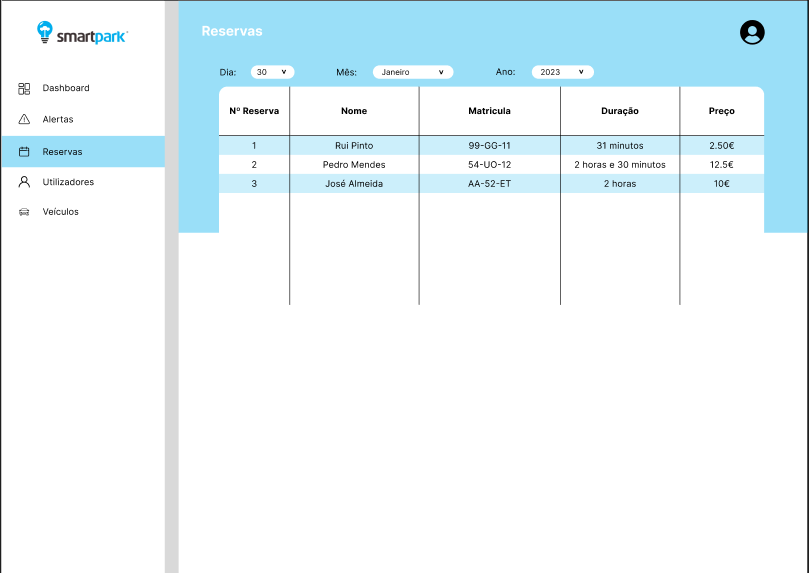


Figura - DashBoard Administrador (Reservas)

#### **Dashboard Admin (Utilizadores)**

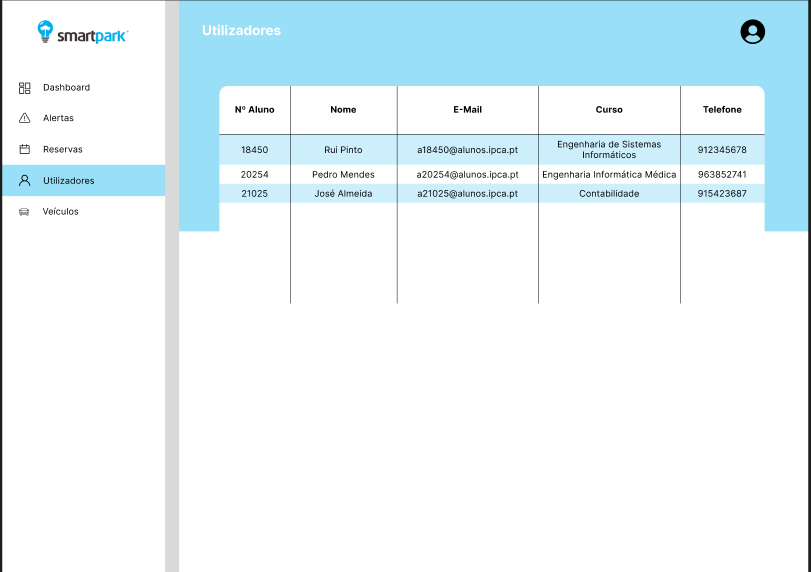


Figura - DashBoard Administrador (Utilizadores)

#### **Dashboard Admin (Veículos)**

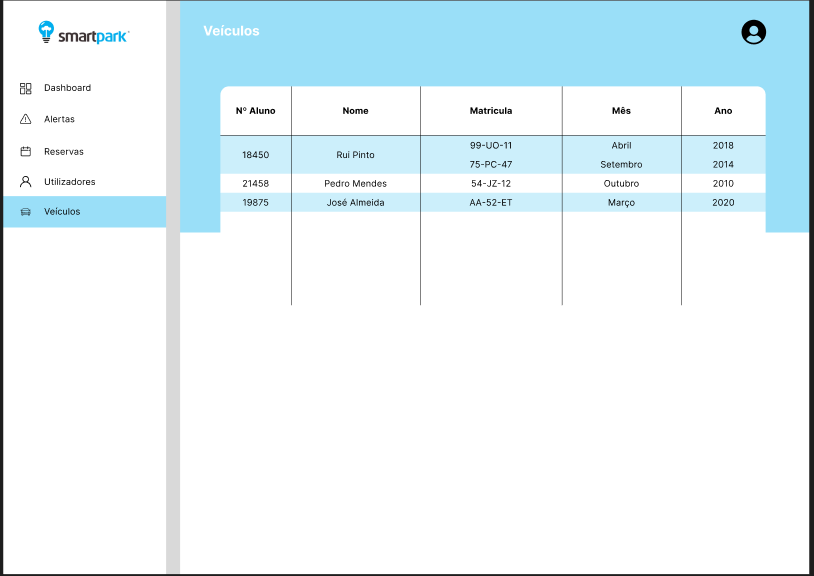


Figura - DashBoard Administrador (Veículos)

### **Dashboard User**

#### **Dashboard User (A minha conta)**

Uma imagem com texto

Descrição gerada automaticamente

Figura - DashBoard Utilizador (A minha conta)

#### **Dashboard User (Lugares Disponíveis)**

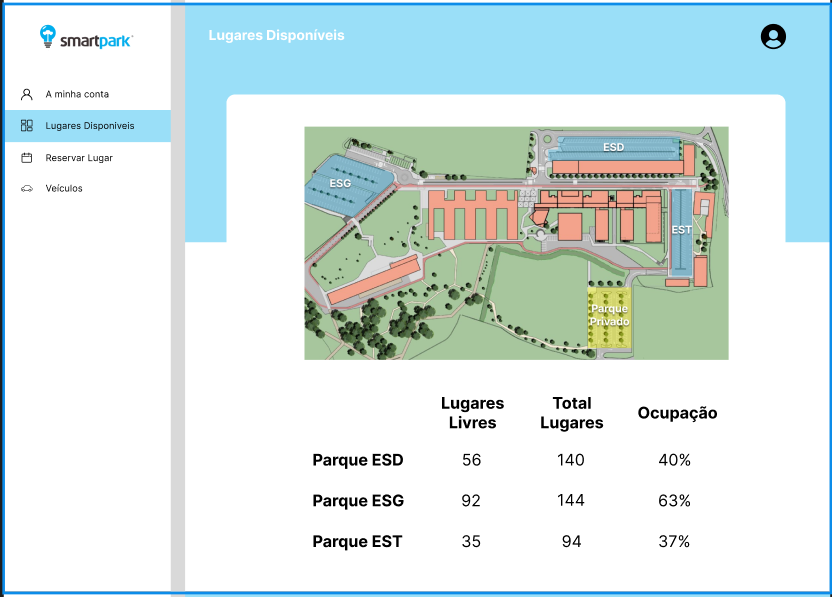


Figura - DashBoard Utilizador (Lugares Disponíveis)

#### **Dashboard User (Reserva Lugar)**

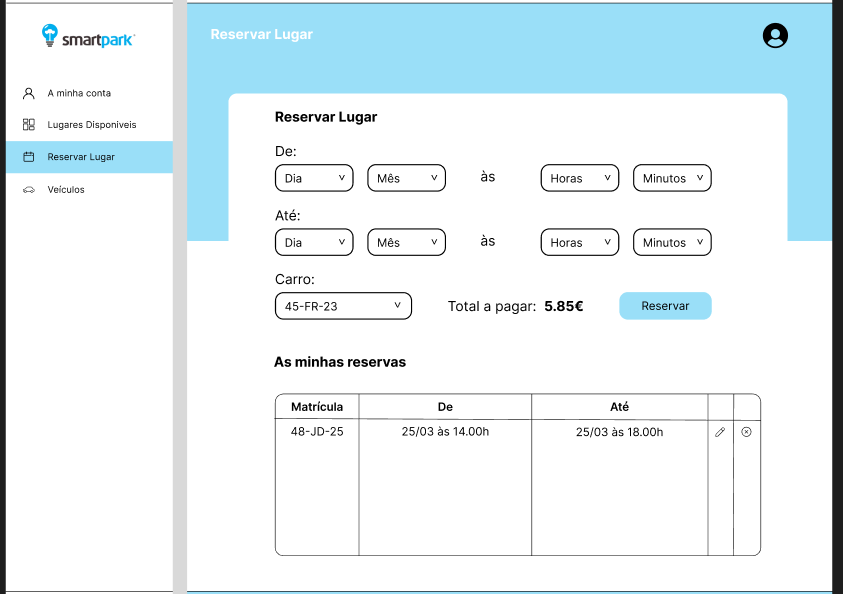


Figura - DashBoard Utilizador (Reserva Lugar)

#### **Dashboard User (Veículos)**

Uma imagem com mesa

Descrição gerada automaticamente

Figura - DashBoard Utilizador (Veículos)

## **Resultado obtido**

# **Conclusão**

# **Bibliografia**