

# EletricNET: Report

Trabalho realizado por:

*Bruno Lopes [68264]*

*Tiago Albuquerque [112901]*

*Abel Teixeira [113655]*

## Índice:

<b>1 Introdução</b>	<b>2</b>
1.1 Visão geral do projeto	2
<b>2 Conceito do produto</b>	<b>3</b>
2.1 Personas	3
2.2 Scenarios	4
2.3 Epics	4
<b>3 Domain Model</b>	<b>5</b>
<b>4 Arquitetura</b>	<b>6</b>
4.1 Requisitos e restrições	6
4.2 Vista da Arquitetura	6
4.3 Deployment Architecture	6
<b>5 API para desenvolvedores</b>	<b>7</b>
<b>6 Referências e Recursos</b>	<b>8</b>

# 1 Introdução

## 1.1 Visão geral do projeto

A **EletricNET** é uma empresa que tem como objetivo melhorar a experiência de carregamento de veículos elétricos (EV).

Numa altura em que o número de veículos elétricos cresce rapidamente, a experiência de carregamento continua a ser algo fragmentada e até confusa, podendo ser pouco eficiente. Atualmente, convivemos com a impossibilidade de efetuar reservas em postos de carregamento dificultando, assim, os clientes que pretendem carregar os seus veículos. A nossa plataforma pretende resolver esse problema oferecendo uma solução simples, acessível e integrada para condutores vulgares e operadores.

Posto isto, será desenvolvida uma **aplicação web** que permita:

- Aos **condutores vulgares**:
  - Procurar por postos de carregamento no mapa;
  - Reservar slots em horários específicos;
  - Visualizar consumos e históricos de sessões;
  - Efetuar pagamentos e contabilizar os custos finais.
- Aos **operadores/gerentes**:
  - Gerir a disponibilidade dos postos;
  - Colocar estações em manutenção;
  - Consultar estatísticas de utilização;
  - Definir períodos de desconto, para otimizar a procura e evitar filas.

O desenvolvimento será baseado nas personas e epics (com user-stories) previamente definidas, integrando também práticas relativas à qualidade de software como Continuous Integration (CI), e Continuous Delivery (CD), conforme os objetivos da cadeia.

## 2 Conceito do produto

De forma a garantir que a aplicação respondesse às necessidades reais dos seus utilizadores, foram definidas **personas**, baseadas em perfis típicos dos utilizadores da nossa plataforma (condutores de veículos elétricos e operadores de estações de carregamento).

Através da análise dos objetivos e contextos associados a cada persona, foram construídos cenários, que ajudaram a identificar os principais fluxos funcionais da aplicação. Com base nesses cenários, foram extraídas as **user stories** e agrupadas em **epics**, de modo a orientar o desenvolvimento do produto de forma estruturada e centrada no utilizador.

### 2.1 Personas

#### Persona 1 - Conductor:

- **Nome:** Joana Martins
- **Idade:** 34
- **Profissão:** Designer
- **Contexto:** Usa carro elétrico no dia-a-dia. Gosta de planejar tudo com antecedência.
- **Objetivos:** Reservar carregamentos perto do trabalho ou de casa para evitar esperas.

#### Persona 2 - Operador da estação:

- **Nome:** Carlos Almeida
- **Idade:** 45
- **Profissão:** Técnico de manutenção
- **Contexto:** Gere as estações da cidade. Quer evitar avarias inesperadas.
- **Objetivos:** Atualizar disponibilidade, monitorizar o uso diário.

#### Persona 3 - Conductor estudante:

- **Nome:** André Silva
- **Idade:** 20
- **Profissão:** Estudante universitário
- **Contexto:** Usa o seu carro elétrico para ir de casa até à universidade e para outras viagens curtas.
- **Objetivos:** Ver o histórico de consumos e controlar gastos.

#### Persona 4 - Gerente (Gestora de operações):

- **Nome:** Marta Lopes
- **Idade:** 38
- **Profissão:** Coordenadora de operações
- **Contexto:** Supervisiona várias estações
- **Objetivos:** Averiguar partes do dia que refiram sobrecarga nos carregamentos para lançar descontos e incentivar carregamentos fora das horas de ponta.

## 2.2 Cenários

### Cenário 1:

A Joana (condutora), antes de sair do trabalho, tenciona reservar um posto de carregamento. Usa o mapa disponível para ver os postos disponíveis perto de casa e encontra um com dois slots livres. Reserva um para as 19h e recebe uma notificação de confirmação. Quando chega, faz o check-in, carrega o carro e vê o consumo final na sua área pessoal.

Após o carregamento, faz o pagamento diretamente na app.

### Cenário 2:

O Carlos (operador da estação) recebe uma chamada a reportar um problema num posto. Accede à área de gestão, marca o posto como "em manutenção" e escreve uma análise técnica a identificar a indisponibilidade.

### Cenário 3:

O André (condutor estudante) entra na plataforma e vai à área relativa aos seus custos. Lá, vê um gráfico simples com o total de carregamentos feitos, por mês, e os respetivos custos.

### Cenário 4:

A Marta (gerente, gestora de operações) analisa os dados relativos ao uso das estações da região e percebe que os postos estão frequentemente sobrecarregados entre as 18h e as 20h, mas quase vazios de madrugada.

No painel de configurações da aplicação, define um desconto de 25% para reservas feitas entre as 00h e as 06h, válido durante todo o próximo mês. Espera com isso distribuir melhor a procura e reduzir filas nos horários de maior afluência.

## 2.3 Epics

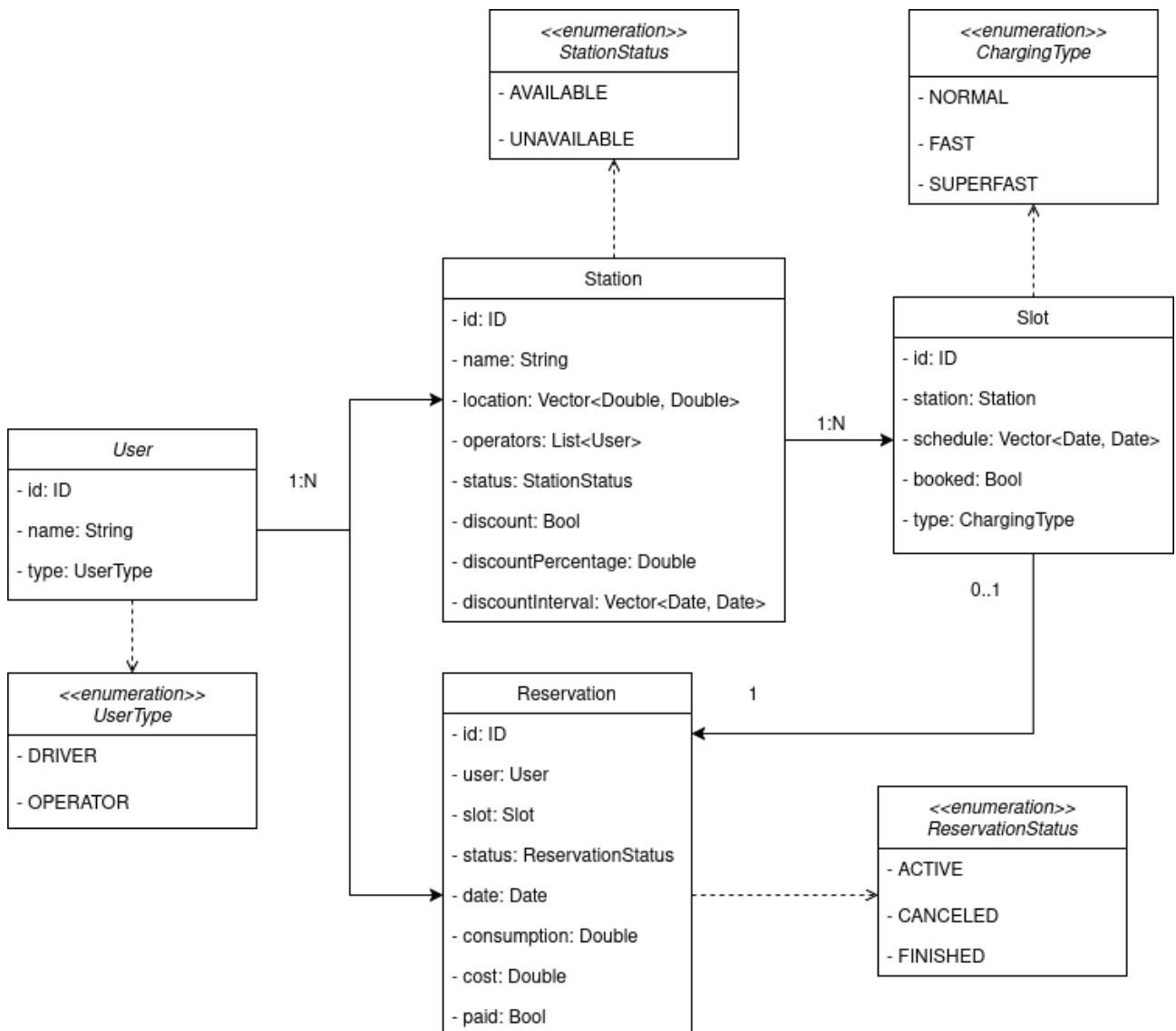
- **Procura por Estações** - Permitir aos utilizadores ver, num mapa, os postos de carregamento disponíveis;
  - **US1<sup>1</sup>**: Como Joana (condutora), quero visualizar no mapa os postos de carregamento próximos para escolher o mais conveniente.
  - **US2**: Como André (estudante), quero filtrar postos por tipo de conector para encontrar um compatível com o meu carro.
  - **US3**: Como Joana, quero ver a disponibilidade em tempo real para evitar deslocar-me a postos ocupados.
  
- **Reserva de Slots** - Permitir que os utilizadores reservem slots em horários específicos;
  - **US4**: Como Joana, quero reservar um slot num horário específico para garantir que tenho vaga ao chegar.
  - **US5**: Como André, quero receber uma notificação de confirmação após reserva para ter certeza de que foi efetuada.
  - **US6**: Como Joana, quero cancelar uma reserva até 1 hora antes para evitar penalizações.
  
- **Sessão de Carregamento** - Suportar o início de uma sessão de carregamento e registar o consumo feito;
  - **US7**: Como Joana, quero iniciar uma sessão de carregamento.
  - **US8**: Como André, quero ver em tempo real o consumo (kWh) e custo acumulado para controlar gastos.
  - **US9**: Como Carlos (operador), quero ser notificado se uma sessão exceder o tempo previsto para gerir melhor a disponibilidade.
  
- **Dashboard Pessoal** - Apresentar aos utilizadores um histórico das carregamentos realizadas, com consumos e custos mensais.
  - **US10**: Como André, quero ver um gráfico com meus gastos mensais para gerir melhor meu orçamento.
  
- **Gestão de Estações** - Permitir aos operadores configurar disponibilidades e manutenções, mas também efetuarem consultas estatísticas a nível de uso.
  - **US12**: Como Carlos, quero marcar um posto como "em manutenção" para evitar que utilizadores tentem reservá-lo.
  - **US13**: Como Marta (gestora), quero ver estatísticas de uso por estação para identificar problemas de sobrecarga.

---

<sup>1</sup> US\* - User-stories numeradas

- **Gestão de Descontos e Horários Off-Peak** - Possibilitar a criação de descontos a aplicar em horários de menor procura para, assim, reduzir as filas e as sobrecargas em determinadas partes do dia.
  - **US14:** Como Marta, quero definir descontos para horários noturnos para reduzir a sobrecarga nos momentos de afluência..
  - **US15:** Como André, quero receber notificações de promoções para poupar dinheiro em carregamentos.

### 3 Domain Model



## 4 Arquitetura

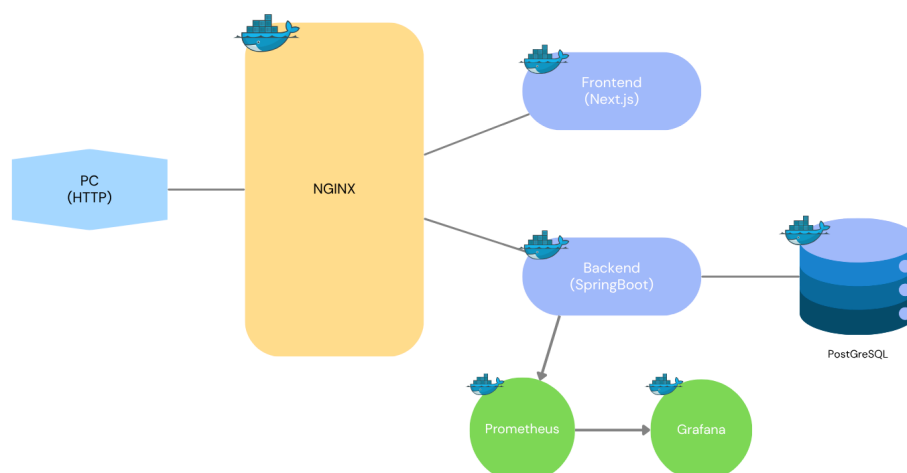
### 4.1 Requisitos funcionais e restrições

- **Disponibilidade:**
  - A aplicação deve estar acessível 24/7 (exceto em manutenção programada).
- **Escalabilidade:**
  - Suportar picos de uso (k6 testing performance).
- **Segurança:**
  - Autenticação de utilizadores (JWT/OAuth2).
  - *Solução:* NGINX.
- **Performance:**
  - Tempo de resposta < 2s para operações críticas (ex.: reservas).
  - *Solução:* Cache de estações disponíveis (Redis ?).

#### Requisitos Não-Funcionais

- **Confiabilidade:**
  - Backup diário da base de dados (PostgreSQL).
- **Monitorização:**
  - Logs centralizados (Prometheus/Grafana).
- **Portabilidade:**
  - Frontend compatível com browsers modernos (React + Vite).

### 4.2 Vista da Arquitetura



## 5 API para desarrolladores

<b>slot-controller</b>		^
PUT	/api/slots/dto/{id}	▼
POST	/api/slots/dto	▼
GET	/api/slots	▼
GET	/api/slots/{id}	▼
GET	/api/slots/stats	▼
GET	/api/slots/station/{stationId}	▼
GET	/api/slots/chargers	▼
GET	/api/slots/available	▼
DELETE	/api/slots/delete/{id}	▼
<b>reservation-controller</b>		^
PUT	/api/reservations/{id}/cancel	▼
POST	/api/reservations/create	▼
GET	/api/reservations/{id}	▼
GET	/api/reservations/slot/{slotId}/active	▼
GET	/api/reservations/revenue	▼
GET	/api/reservations/myStats	▼
GET	/api/reservations/myReservations	▼
GET	/api/reservations/all	▼
<b>station-controller</b>		^
GET	/api/stations	▼
POST	/api/stations	▼
POST	/api/stations/{id}/discount	▼
GET	/api/stations/{id}	▼
<b>auth-controller</b>		^
POST	/api/auth/register	▼
POST	/api/auth/login	▼
<b>user-controller</b>		^
GET	/api/users	▼
GET	/api/users/total-users	▼



## 6 Referências e Recursos

### Recursos do Projeto

Resource:	URL/location:
Git repository	- <a href="https://github.com/Brupez/EletricNET">https://github.com/Brupez/EletricNET</a>
Video demo	-
QA dashboard (online)	- <a href="https://sonarcloud.io/project/overview?id=Brupez_EletricNET">https://sonarcloud.io/project/overview?id=Brupez_EletricNET</a> ( <a href="https://sonarcloud.io/project/overview?id=Brupez_EletricNET">https://sonarcloud.io/project/overview?id=Brupez_EletricNET</a> )
CI/CD pipeline	- <a href="https://github.com/Brupez/EletricNET/actions/workflows/build.yml">https://github.com/Brupez/EletricNET/actions/workflows/build.yml</a> - <a href="https://github.com/Brupez/EletricNET/actions/workflows/deploy.yml">https://github.com/Brupez/EletricNET/actions/workflows/deploy.yml</a>
Deployment ready to use	- <a href="http://deti-tqs-05.ua.pt">http://deti-tqs-05.ua.pt</a>