



**Licenciatura em Engenharia Informática e Computadores**  
**Projeto e Seminário - Semestre de Verão 2018/2019**

**Proposta de Projeto**

**Monitorização de qualidade de serviço em  
comunicações ferroviárias**

João Vaz, Nº 41920  
41920@alunos.isel.pt  
961833140

Luis Vasconcelos, Nº41556  
41556@alunos.isel.pt  
910365046

**Orientado por:**

Nuno Cota, ncota@deetc.isel.pt  
Ana Beire, anaritabeire@solvit.pt

# 1. Introdução

A monitorização permanente da qualidade de serviço em redes de comunicações móveis ferroviárias é fundamental para a deteção de alterações que coloquem em causa o serviço de comunicações de apoio à exploração. Nesse sentido, foram desenvolvidas unidades embarcadas em comboios (sondas) que realizam de forma autónoma e permanente a monitorização, ativa e passiva, aos sistemas de comunicações móveis.

O projeto proposto consiste no desenvolvimento de uma solução constituída por repositório de dados e aplicação Web que permita a análise e o tratamento de informação proveniente das sondas de forma a permitir ao utilizador uma fácil e objetiva observação dos dados e administração do sistema.

## 2. Solução Proposta

A solução proposta passa pelo desenvolvimento de uma Web API para acesso aos dados reportados pelo conjunto de sondas existente e apresentação de resultados de medidas e testes aos utilizadores, bem como possibilitar a administração das mesmas. A informação proveniente das sondas é já atualmente armazenada numa base de dados.

A informação resultante dos testes deverá ser apresentada de forma geo-referenciada, sobre mapas e através de gráficos. Para disponibilização do serviço de mapas será utilizada a API Leaflet [1].

De forma a que o cliente tenha acesso aos dados e informações terá de estar registado no sistema de informação, pelo que a solução deverá igualmente incluir toda a componente de gestão de utilizadores e perfis.

## 3. Arquitetura do Sistema

Para a realização desta aplicação é necessário desenvolver os componentes da arquitetura, apresentados na Figura 1.

São considerados os seguintes componentes principais:

- **Componente Cliente**

A componente cliente é acedida pelo utilizador através do *browser* e vai ser responsável pela apresentação e gestão dos dados. Vai ser desenvolvida utilizando a linguagem de programação JavaScript (Node.js) [2], a *framework* AngularJS [3] e a ferramenta de desenvolvimento Visual Studio Code [4].

Esta componente vai disponibilizar a informação já devidamente tratada usando a informação oferecida pela componente servidora.

- **Componente Servidora**

A componente servidora constitui o meio utilizado pela aplicação cliente para obter, publicar e atualizar dados. Estas operações são realizadas através da interação com a base de dados relacional.

Quanto à Web API vai ser desenvolvida utilizando a linguagem de programação kotlin [5]. A utilização da framework Spring MVC [6] permite simplificar e auxiliar o desenvolvimento da API.

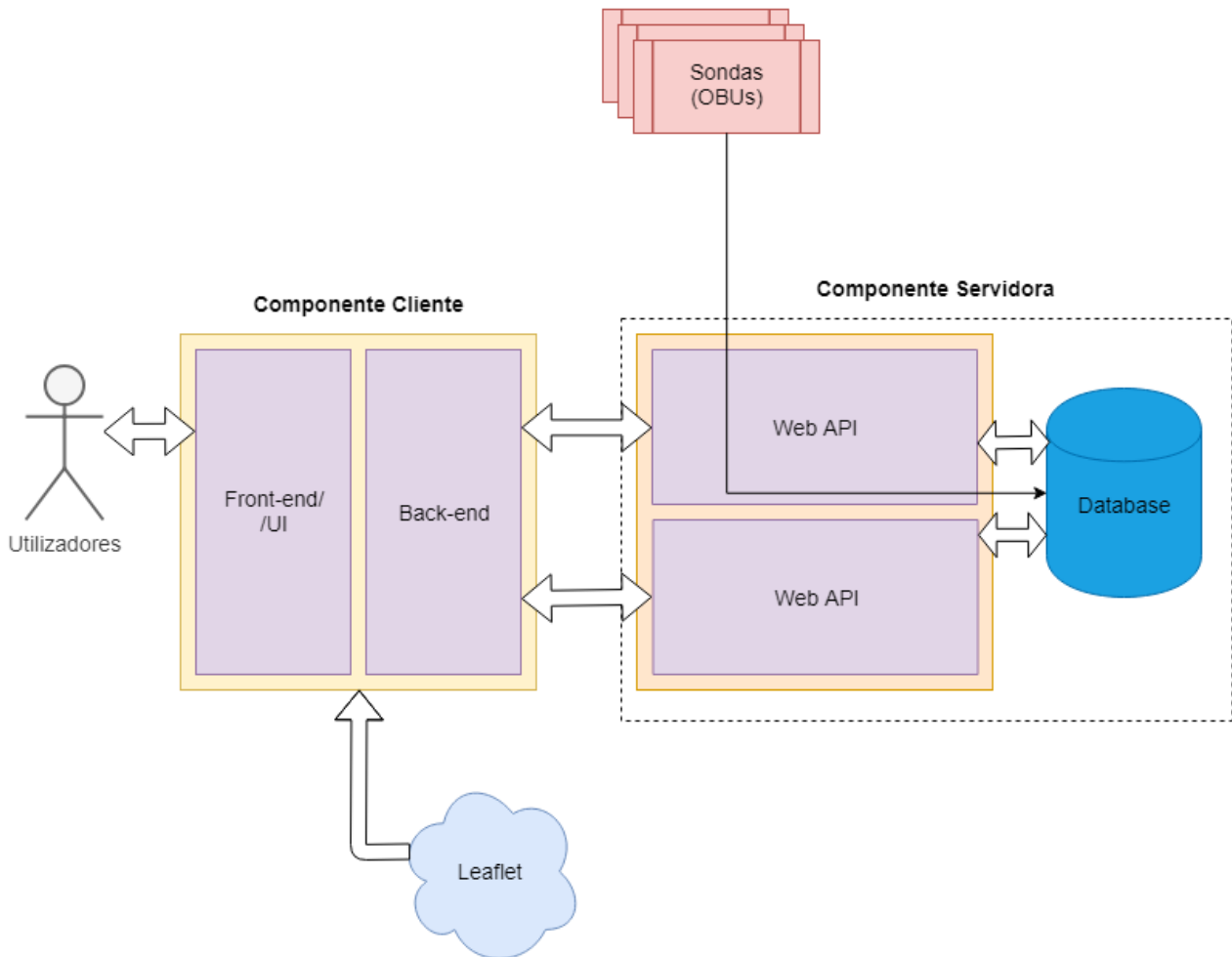


Figura 1 - Arquitetura do Sistema.

## 4. - Requisitos Funcionais

Os requisitos funcionais dividem-se em requisitos obrigatórios e requisitos opcionais.

### 4.1. Requisitos Obrigatórios

Os requisitos funcionais obrigatórios são os seguintes:

- Aplicação Web Single-Page com integração de mapas
- Gestão de OBUs (*On Board Unit*):
  - Gestão de grupos
  - Gestão de hardware
  - Gestão de configurações
  - Gestão de planos de teste

- Gestão de ligações
- Apresentação dos dados:
  - Mapas
  - Gráficos
  - Logs
- Gestão de utilizadores
  - Registo de utilizadores
  - Autenticação de utilizadores
  - Gestão de perfis de utilizadores
- Pesquisa de dados por nome, data, local, etc.

## 4.2. Requisitos Opcionais

Os requisitos funcionais opcionais são os seguintes:

- Monitorização de sondas em tempo real

## 5. Planeamento

Em termos de planeamento, considera-se que o projeto envolverá duas grandes atividades:

- Definição de requisitos e especificações – levantamento de todos os requisitos funcionais e não funcionais e a definição das especificações técnicas do projeto;
- Desenvolvimento – tarefas de desenvolvimento dos diversos componentes do projeto, incluindo a apresentação dos diferentes relatórios e apresentações.

No Anexo A encontra-se o diagrama de Gant proposto para as atividades e tarefas consideradas, permitindo verificar o planeamento temporal das tarefas.

## Referências

- [1] <https://leafletjs.com/>, Leaflet
- [2] <https://nodejs.org/en/>, JavaScript Node.js
- [3] <https://angularjs.org>, AngularJS
- [4] <https://code.visualstudio.com>, Visual Studio Code
- [5] <https://kotlinlang.org/>, Kotlin
- [6] <https://spring.io/guides/gs/serving-web-content/>, Spring MVC

## Anexo A - Planificação temporal das atividades e tarefas associadas ao projeto.

[illegible]