

# 1. Objetivo do Projeto

O objetivo deste projeto é a criação de uma API que irá receber valores de diversos sensores, armazenar esses dados em um banco de dados e permitir a consulta desses valores. Além disso, uma aplicação no front para apresentar gráficos e tabela dos valores médios desses sensores.

## 2. Arquitetura e Stack

### 2.1. Arquitetura

O projeto foi desenvolvido em duas frentes: o back e o front. No back foi feito os serviços de registro e consulta de eventos e equipamentos ligados ao banco de dados.

No front foi consumida a API feita no back para consultar os valores médios de cada equipamento em um determinado período de tempo.

### 2.2. Stack Utilizada

- Back implementado em JavaScript utilizando sequelize
- Linguagem do Banco de dados utilizado é o sqLite
- Bibliotecas usadas no back:
  - cors
  - csvtojson
  - dotenv
  - express
  - express-fileupload
  - express-validator
  - node-dev
  - sequelize
  - sqlite3
- No front foi utilizado framework next com TypeScript
- Bibliotecas usadas no front
  - Recharts

## 3. Acesso e Execução do código

### 3.1. Instalação

- 3.1.1. `git clone https://github.com/BrenoNatal/DesafioRadix.git`
- 3.1.2. `cd DesafioRadix`
- 3.1.3. `cd back`
- 3.1.4. `npm install`
- 3.1.5. `cd ..`
- 3.1.6. `cd front`
- 3.1.7. `npm install`

### 3.2. Rodar o back

- 3.2.1. `cd back`
- 3.2.2. `cp .env.example .env`
- 3.2.3. `npm run migrate`
- 3.2.4. `npm run dev`

### 3.3. Rodar o front

- 3.3.1. `cd front`
- 3.3.2. `npm run dev`

## 4. Rotas da API

post: `http://localhost:3333/event`

Cria um evento com base nos parâmetros passados no body da requisição:

`equipmentId:string`  
`value:number`  
`timestamp:date`

post: `http://localhost:3333/eventCSV`

Cria os eventos com base no arquivo csv enviado na requisição

get: `http://localhost:3333/event/:days`

Retorna o valor médio registrado de cada equipamento com base no parâmetro days passado na url.

post: `http://localhost:3333/equipment`

Cria um equipamento com base no parâmetro passado no body da requisição.

equipmentId:string

post: <http://localhost:3333/equipment>

Cria os equipamento com base no arquivo csv enviado na requisição.