

Metodologia Frontend

- Para desenvolver nosso frontend, primeiro foram levantados os requisitos necessários para que o projeto fosse bem sucedido e atendesse as expectativas. Portanto, viu se a necessidade de construir uma página WEB, que poderia ser acessada de qualquer dispositivo com internet no mundo, porém voltado para dispositivos Desktop e Notebooks, por proporcionar uma melhor visualização da página web.
- Para desenvolver a página WEB, optamos por utilizar o framework ReactJS, desenvolvido pelo Facebook, juntamente com a estrutura de desenvolvimento NextJS, desenvolvida pela Vercel. A escolha do ReactJS juntamente com o NextJS, foi feita por conta da facilidade do uso e de sua atualidade no desenvolvimento de aplicações WEB.
- Para criar o projeto ReactJS com a estrutura NextJS, foi necessário a instalação do ambiente de execução NodeJS, na versão 12.16.1, que pode ser feito o download através do link: <https://nodejs.org/en/download/>, que traz consigo o gerenciador de pacotes do NodeJS, NPM, na versão 6.13.4. Além disso, também foi instalado o gerenciador de pacotes Yarn, que foi desenvolvido pelo facebook, na versão 1.22.5. Utilizamos o Yarn pela familiaridade, porém cumpre a mesma função do Npm. Para instalar o Yarn através da linha de comando, é necessário ter o npm instalado, e executar o seguinte comando: `npm install --global yarn`. Mais informações sobre o Yarn estão disponíveis no link: <https://classic.yarnpkg.com/en/docs/install/#windows-stable>
- O projeto foi criado utilizando a estrutura do NextJS, através do comando: `yarn create next-app --typescript`, executado no CMD do Windows. Todas as informações sobre como criar um projeto React, utilizando a estrutura do NextJS, podem ser acessadas através do link: <https://nextjs.org/docs/api-reference/create-next-app>
- Como linguagem de programação principal utilizamos o TypeScript, que nada mais é do que o próprio JavaScript, porém trazendo a tipagem estática a este.
- Tendo em mente que, através do nosso cenário físico, poderíamos captar a data e hora que um identificador foi lido por um sensor, começamos a avaliar como dispor

essas informações de maneira relevante, e que ajudasse a montar estratégias para o negócio, com base nos dados de localização.

- Com isso em mente, foram levantadas duas principais maneiras de visualização dos dados, que são, primeiramente, um mapa de calor do local, que possibilita uma visão das áreas mais visitadas da planta baixa do local, e secundamente, um aglomerado de gráficos, que exibem os números coletados, de maneira agrupada, onde pode ser feita uma comparação entre diferentes setores de um local, em um determinado período.
- Para desenvolver o mapa de calor utilizamos a biblioteca HeatmapJS, que pode ser acessada através do link: <https://www.patrick-wied.at/static/heatmapjs/> que contém toda a documentação da biblioteca. Além disso, esta biblioteca também está disponível no gerenciador de pacotes do NodeJS, npm, no link: <https://www.npmjs.com/package/heatmapjs>. A versão utilizada nesse projeto foi a 2.0.2
- Para desenvolver os gráficos foi utilizada a biblioteca ChartsJS, que pode ser acessada através do link: <https://www.chartjs.org/>, a versão utilizada foi adquirida através do gerenciador de pacotes NPM no link: <https://www.npmjs.com/package/react-chartjs-2>, e possui o nome de react-chartjs-2, que é uma versão da biblioteca ChartsJS própria para ReactJS. A versão utilizada dessa biblioteca foi a 3.0.4
- Com tudo isso estruturado, precisamos que os dados cheguem até a nossa aplicação WEB. Essas dados vem da nossa API, através de requisições HTTP. Para realizar essas requisições do nosso frontend para a API, utilizamos a biblioteca AXIOS, adquirida através do gerenciador de pacotes NPM, no link: <https://www.npmjs.com/package/axios>. Foi utilizada a versão 0.21.1