

DEVOPS

DOCUMENTAÇÃO PRÁTICA 1

Caio Obatake Watari Gonçalves Cadini - 800383
Lucca Couto Barberato - 800257

SÃO CARLOS - SP
07/2024

1. Visão Geral de Aplicação

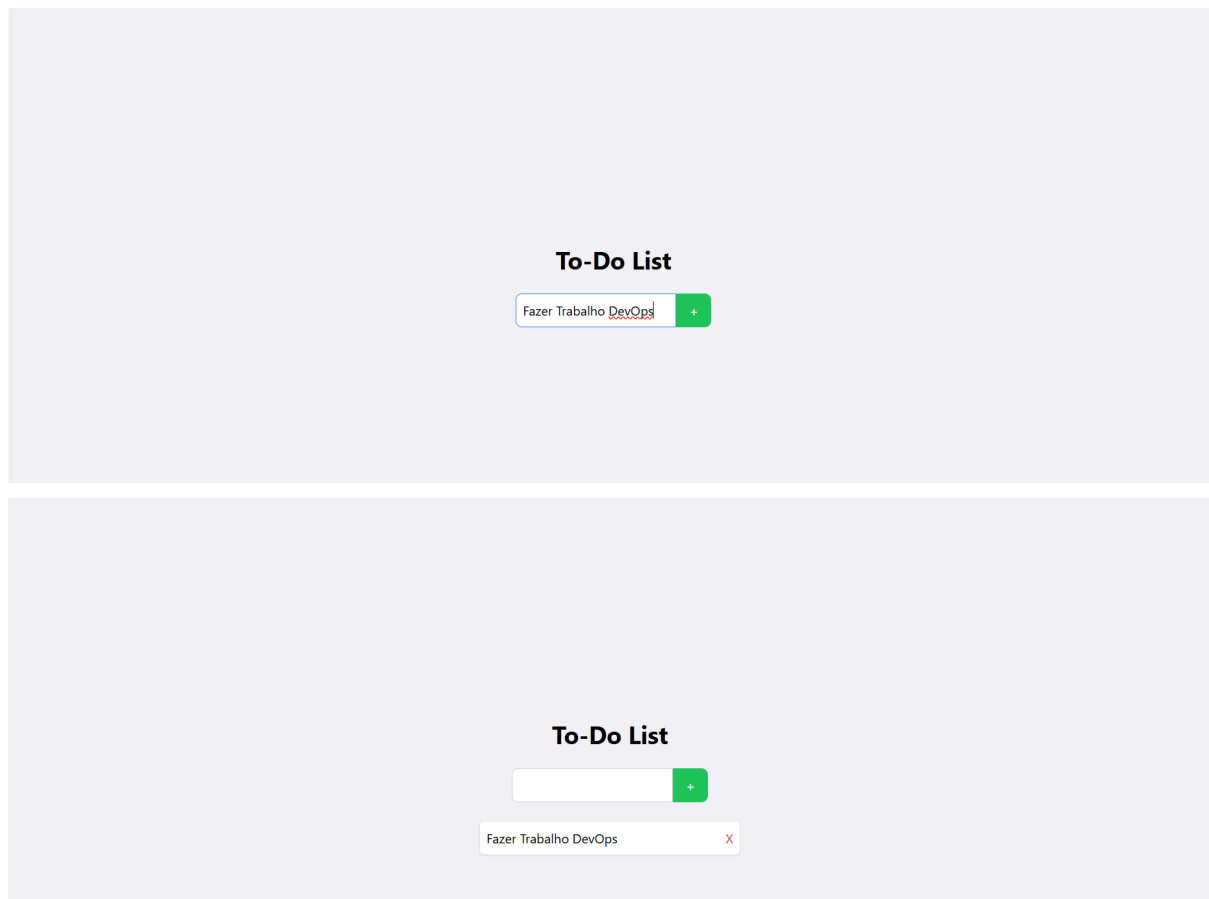
A aplicação desenvolvida se trata de uma ToDo List, sua ideia geral sendo possibilitar que o usuário consiga administrar atividades que precisa fazer durante sua rotina.

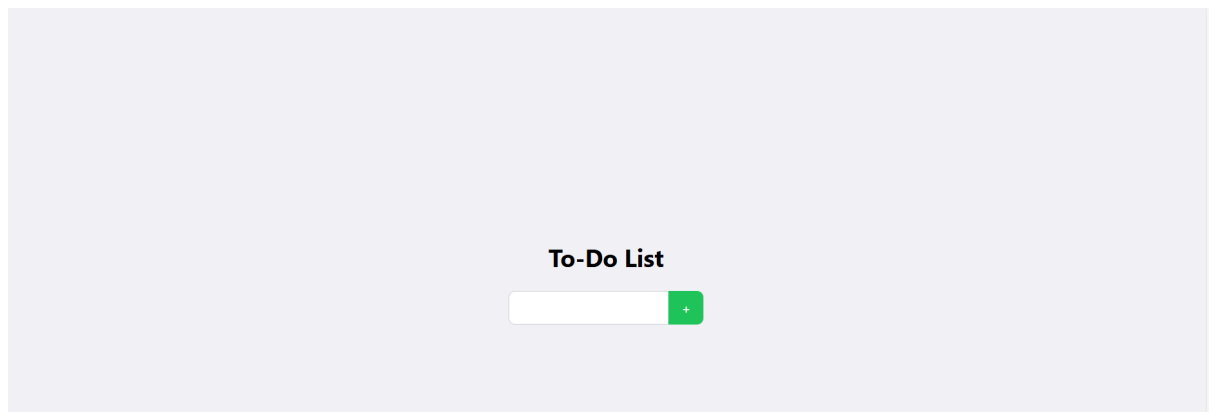
O frontend é representado por um formulário para inserir o nome da tarefa a ser realizada e uma listagem das atividades a serem feitas, assim como um botão em cada uma para que se possa removê-las ao serem realizadas.

O backend por sua vez é composto por 3 endpoints essenciais:

- GET /get/
 - Responsável por obter todas as tarefas presentes no Banco de Dados
- POST /add/
 - Responsável por adicionar uma nova tarefa ao BD
- DELETE /del/{task_id}
 - Responsável por remover do BD a tarefa que contém o ID passado

A aplicação funcional pode ser vista na figura abaixo:





2. Contêineres e Tecnologias utilizadas

O projeto foi dividido em 3 contêineres, um contendo a imagem do frontend, um contendo a imagem do backend e um contendo a imagem do banco de dados.

O frontend foi desenvolvido utilizando Node.js para montar uma aplicação React com Tailwind CSS. Para o processo de build do contêiner do frontend, foi necessário usar a imagem de um servidor responsável por servir conteúdo estático, sendo o NGINX o escolhido.

O backend por sua vez foi desenvolvido em python utilizando o framework FastAPI. Para o processo de build de seu respectivo contêiner, foi necessário realizar o processo de instalação das dependências definidas no arquivo "requirements.txt", além de realizar o processo de conexão do backend com o banco de dados pela utilização da variável de ambiente DATABASE_URL.

O banco de dados foi por sua vez feito utilizando sqlite.

3. Manual de instalação para realizar docker compose

Para conseguir rodar o projeto utilizando os contêineres, basta entrar no diretório do projeto e realizar o comando:

```
docker compose up
```

Após isto, para conseguir acessar e interagir com a aplicação, pelo navegador acessar a porta do 80 do localhost pelo navegador usando o endereço: localhost:80. A partir disto será possível acessar e interagir com a aplicação pelo frontend.

To-Do List