

Desenvolvimento de API - Farmácia Já

Checkpoint 01

Definição do escopo:

A ideia é que o usuário possa pesquisar o nome de um medicamento e descobrir quais farmácias o possuem, além de ter a opção de filtrar os resultados por bairro para facilitar encontrar o local mais próximo.

- Criar uma API para consulta de farmácias e medicamentos.
- Permitir pesquisa por nome do medicamento e filtro por bairro.
- Cadastrar, editar e remover farmácias e medicamentos.
- Estabelecer um relacionamento **muitos-para-muitos** entre farmácias e medicamentos.
- Retornar as informações em formato **JSON**.
- Armazenar os dados em um **banco PostgreSQL**.
- Criar uma **documentação dos endpoints** da API.

Definição dos endpoints:

- **GET /farmacias**
→ Retorna a lista de todas as farmácias cadastradas.
- **GET /farmacias/:id**
→ Retorna os detalhes de uma farmácia específica pelo ID.
- **POST /farmacias**
→ Cadastra uma nova farmácia no sistema.
- **PUT /farmacias/:id**
→ Atualiza as informações de uma farmácia existente.
- **DELETE /farmacias/:id**
→ Remove uma farmácia do sistema.
- **GET /medicamentos**
→ Retorna a lista de medicamentos cadastrados.
- **GET /medicamentos/:id**
→ Retorna informações detalhadas sobre um medicamento específico.
- **POST /medicamentos**
→ Cadastra um novo medicamento no banco de dados.
- **GET /farmacias/medicamentos?nome=dipirona**
→ Pesquisa por farmácias que possuem ou entregam o medicamento pesquisado.

GET /farmacias/medicamentos?nome=dipirone&bairro=Centro

→ Filtra farmácias de um bairro específico que possuem o medicamento procurado

-

OBS: Muitos endpoints poderão contar com a implementação de filtros para auxiliar o usuário a encontrar o medicamento procurado.

Levantamento de requisitos:

Funcionais:

- **RF01:** Cadastrar farmácias com informações como nome, endereço, bairro, telefone e se realiza entregas.
- **RF02:** Cadastrar medicamentos disponíveis no programa Farmácia Popular.
- **RF03:** Associar medicamentos às farmácias que os possuem.
- **RF04:** Permitir a busca de farmácias que tenham determinado medicamento.
- **RF05:** Permitir filtragem por bairro para facilitar a busca.
- **RF06:** Atualizar e remover farmácias e medicamentos quando necessário.

Não funcionais:

- **RNF01:** O sistema deve ser desenvolvido em **Node.js com TypeScript**.
- **RNF02:** O banco de dados deve ser **PostgreSQL**.
- **RNF03:** A API deve seguir o padrão **RESTful**, retornando os dados em **JSON**.
- **RNF04:** O código deve ser organizado em **estrutura MVC** (controllers, services, routes, models).
- **RNF05:** Deve haver **tratamento de erros** e mensagens claras de retorno.
- **RNF06:** Informações sensíveis devem ficar em um arquivo **.env**.
- **RNF07:** A API deve possuir **documentação dos endpoints** (Postman).