

Facecook - A Rede Social das Receitas

1. Objetivo

Nossa aplicação web tem como objetivo a criação de uma API RESTful utilizando a stack MERN, com a implementação de um CRUD para adicionar, editar, visualizar e remover receitas do nosso site. Assim, com a nossa aplicação, achar e compartilhar as suas receitas favoritas será muito mais fácil.

2. Tecnologias utilizadas

MongoDB

Express

React

NodeJS

Docker

3. Como utilizar

Para acessar o nosso site, basta acessar a URL <https://facecook-frontend.blacksky-4c211341.australiaeast.azurecontainerapps.io/>, que te levará para a nossa página inicial, onde você verá todas as receitas que foram criadas no nosso site. Clicando no botão “Adicionar”, você poderá adicionar a sua receita ao nosso site e compartilhá-la com todos. Abaixo de cada receita, você poderá editar ou excluir uma receita previamente criada clicando em seus respectivos botões.

4. Como funciona a API

Você pode utilizar a nossa API pela URL <https://facecook-backend.blacksky-4c211341.australiaeast.azurecontainerapps.io/>

Endpoints da API:

Pegar todas as receitas:

- URL: /
- Método: GET
- Descrição: Pega todas as receitas do banco de dados.
- Corpo da resposta:

```
{  
  "receitasLista": [Array de objetos de receitas]  
}
```

Criar uma nova receita:

- URL: /adicionar
- Método: POST
- Descrição: Adiciona uma nova receita ao banco de dados.
- Corpo da requisição:
 - imagem (Arquivo): Imagem da receita (multipart/form-data)
 - nome (String): Nome da receita
 - ingredientes (String): Ingredientes da receita (separados por uma vírgula)
 - modoPreparo (String): Modo de preparo da receita
 - categorias (String): Categoria da receita (padrão: salgado)
 - tempoPreparo (String): Tempo de preparo da receita
 - rendimento (String): Rendimento da receita
- Corpo da resposta:

```
{  
  "message": "Receita adicionada com sucesso"  
}
```

Pegar detalhes da receita para fins de edição:

- URL: /editar/:id
- Método: GET
- Descrição: Recupera detalhes de uma receita com base no seu ID para fins de edição.
- Parâmetros da requisição:
 - id (String): ID da receita
- Corpo da resposta:

```
{  
  
  "receitaSelecionada": { Objeto receita }  
  
}
```

Editar a receita:

- URL: /editar/:id
- Método: PUT
- Descrição: Atualiza uma receita existente com base no seu ID.
- Parâmetros da requisição:
 - id (String): ID da receita
- Corpo da requisição:
 - Mesmo do endpoint " Criar uma nova receita"
- Corpo da resposta:

```
{  
  
  "message": "Receita atualizada com sucesso"  
  
}
```

Deletar a receita:

- URL: /:id
- Método: DELETE
- Descrição: Exclui uma receita com base no seu ID.
- Parâmetros da requisição:
 - id (String): ID da receita
- Corpo da resposta:

{ Objeto receita }

Notas:

- Todas as respostas estão no formato JSON.
- Os erros são tratados com códigos de status e mensagens de erro adequadas.

5. Passo a passo sobre como criar o banco de dados, tabelas e campos

- Crie uma conta no MongoDB Atlas
- Crie um novo projeto
- Crie uma nova implantação
- Em “Database”, vá em “Browse Collections”, onde você criará um banco e após isso uma coleção
- Habilite o ip de sua conexão na aba “Network Access” para poder acessar o banco (você pode habilitar o ip 0.0.0.0/0 para dar acesso a todos os ips)
- Após isto, em “Overview”, clique em “Connect”, e então em “MongoDB for VSCode”
- Copie a String de conexão, substitua a área <password> pela senha de sua conexão (você pode achá-la em “Database Access” e clicando em “edit” na conta deseja)
- Cole a String de conexão no seu código dentro de `mongoose.connect()`
- Modele a sua coleção usando mongoose do jeito que desejar