Manual do Usuário

Clonar o Repositório

Inicialmente, o repositório do projeto pode ser baixado em sua máquina usando o comando

```
# Caso use HTTPS
git clone https://github.com/izaiasmachado/TrabalhoBD-2022_2-NullBank.git
# Caso use SSH
git clone git@github.com:izaiasmachado/TrabalhoBD-2022_2-NullBank.git
```

Entre na pasta do projeto

```
cd TrabalhoBD-2022_2-NullBank
```

Árvore de diretórios

Utilize essa árvore para identificar a localização dos arquivos do projeto.

```
- LICENSE
--- README.md
- docker-compose.yml
- docs
  - dev
  - entregaveis
  - especificacoes.md
  | — manual.md
 — pdfs
  - especificacoes.pdf
   - manual.pdf
   └─ modelagem.pdf
 - src
   - client
   - server
   L— sql
  - vendor
   └─ roteiro-nullbank-v2.pdf
```

Dependências

Para rodar o projeto, é necessário instalar todas as dependências do documento de especificações. Faça isso antes de prosseguir para as próximas etapas.

Subir o Banco de Dados

Para subir o banco de dados, execute o comando

```
docker-compose up -d
```

O banco MySQL deve ficar acessível em http://localhost:3306, com usuário root e senha root . Também é possível acessar o banco de dados pelo usuário admin e senha root . Essas informações podem ser alteradas no arquivo docker-compose.yml .

Rodar Scripts SQL

Para rodar os scripts SQL, utilize um cliente MySQL de sua preferência. Os scripts SQL estão na pasta src/sql e devem ser executados na seguinte ordem:

- 1. schema.sql
- migrations.sql
- 3. triggers.sql

Onde schema.sql cria o banco de dados e as tabelas, migrations.sql insere os dados de povoamento inicial e triggers.sql cria os gatilhos.

Iniciar o Servidor

Navegue para a pasta do servidor

```
cd src/server
```

Instale as dependências do servidor

```
npm install
```

Crie um arquivo .env copiando o arquivo .env.example

```
cp .env.example .env # Linux e MacOS
copy .env.example .env # Windows
```

Certifique-se de que o arquivo .env foi criado corretamente e também que as informações de conexão com o banco de dados estão corretas nesse arquivo.

Com isso, o servidor pode ser iniciado com o comando

```
npm start
```

A API deve ficar acessível em $\mbox{http://localhost:5001}$. Caso queira mudar a porta do servidor, basta alterar a variável \mbox{port} no arquivo $\mbox{.env}$.

Iniciar o Cliente

Abra o arquivo index.html disponível raíz do projeto. Você pode utilizar qualquer browser para isso.

Isso só pode ser realizado após subir a API. No arquivo script.js , a variável global link deve conter a URL da API. Caso a API esteja rodando na porta 5001, a variável deve ser http://localhost:5001 .

A primeira tela é a tela de credenciais do banco de dados. O usuário é 'admin' e a senha é 'root'. Essas informações podem ser alteradas no arquivo src/client/login.js .

Após isso, você será redirecionado para uma tela com a opção de login como DBA ou Funcionário. Se você escolher DBA, o usuário é 'Admin' e a senha é 'Root. No caso do funcionário, o usuário é 'qwerty' e a senha é '12345'. Esses são os níveis de acesso disponíveis no sistema.

Encerrar Execuções

Para encerrar a execução do servidor e também do cliente, basta pressionar Ctrl+C no terminal onde o servidor está sendo executado.

Para encerrar a execução do banco de dados, execute o comando

docker-compose down