

Desenvolvendo um frontend Web com React JS

My SQL:

- Criar uma tabela de Login, conforme o script Login.sql no Git Aula 04

Prompt de Comando (VS Code):

- Instalar o pacote react-app. Acessar o diretório C:\Projetos\AgendaContatos via prompt de comando
(O React App é um pacote que já traz algumas dependências para desenvolvimento de um frontend em React JS)
- Executar o seguinte comando: yarn create react-app frontend
(frontend será a pasta que irá ser criada)
- Na pasta C:\Projetos\AgendaContatos\frontend\public remover todos os arquivos, exceto o index.html
- Na pasta C:\Projetos\AgendaContatos\frontend\src remover os arquivos: serviceWorker.js, App.test.js, index.css e logo.svg
- Deixar a pasta C:\Projetos\AgendaContatos\frontend\public com o arquivo index.html igual ao arquivo index.html que está no Git na pasta Aula 04/Parte 01
- Deixar a pasta C:\Projetos\AgendaContatos\frontend\src com os arquivos App.css, App.js e index.js igual aos arquivos App.css, App.js e index.js que estão no Git na pasta Aula 04/Parte 01
- Executar o nosso primeiro projeto em React acessando a pasta C:\Projetos\AgendaContatos\frontend e executando o comando: yarn start
(Neste momento, o VS Code vai abrir o browser com a página web em React)
(Experimente alterar o arquivo App.js e salve sua alteração e você verá a alteração direto na página web, sem precisar executar nada)
- Alterar o arquivo C:\Projetos\AgendaContatos\backend\src\routes.js, incluindo uma API de autenticação. Copiar o arquivo routes.js do Git Aula 04/Parte02/backend/src
- Alterar o arquivo C:\Projetos\AgendaContatos\backend\src\server.js, alterando a porta do servidor e incluindo mais algumas declarações. Copiar o arquivo routes.js do Git Aula 04/Parte02/backend/src
- No console vá até a pasta C:\Projetos\AgendaContatos\frontend e instale o seguinte pacote com o comando: yarn add react-router-dom
(Esse pacote permite o React a ter acesso a todos os controles do nosso html)
- No console vá até a pasta C:\Projetos\AgendaContatos\frontend e instale o seguinte pacote com o comando: yarn add axios
(Esse pacote nos permitirá chamar APIs a partir do React)
- Criar a pasta pages dentro de C:\Projetos\AgendaContatos\frontend\src
- Criar a pasta assets dentro de C:\Projetos\AgendaContatos\frontend\src
- Criar a pasta services dentro de C:\Projetos\AgendaContatos\frontend\src

- Dentro da pasta pages criar os arquivos Login.js e Login.css. Copiar os arquivos que estão no Git Aula 04/Parte 02/frontend/src/pages
- Dentro de assets colocar uma figura jpg por exemplo escola.jpg. Copiar o arquivo que está no Git Aula 04/Parte 02/frontend/src/assets
- Dentro de services criar o arquivo api.js. Copiar o arquivo que está no Git Aula 04/Parte 02/frontend/src/services
- Alterar os arquivos C:\Projetos\AgendaContatos\frontend\src\ App.css, App.js, index.js e routes.js, conforme os 4 arquivos que estão no Git Aula 04/Parte 02/frontend/src
- Abra dois terminais dentro do VS Code e execute em um terminal o servidor Node porta 3333 para as APIs (backend) e no outro terminal o servidor Node porta 3000 para as páginas em React (frontend) e vamos testar o nosso login