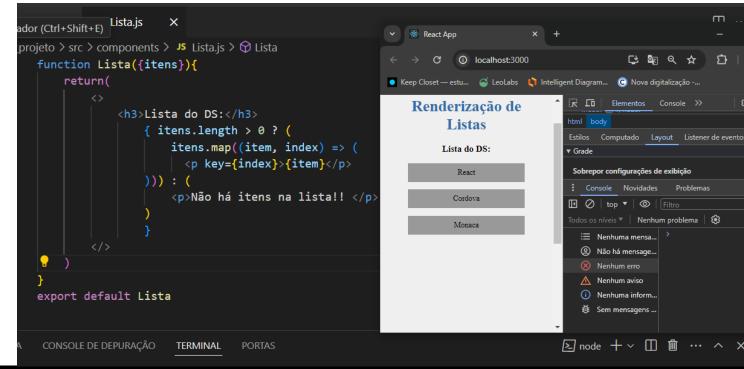


GOVERNO DO ESTADO DE SÃO PAULO

PROGRAMAÇÃO WEB III APLICAÇÃO DE PÁGINA ÚNICA (SPA)



Prosseguindo em relação a aula passada, onde executamos a renderização de listas e login, agora passamos para o State Lift:





State Lift

Basicamente é um técnica utilizada para compartilhar o state, porque é normal vários componentes dependerem do mesmo estado, então somos de certa maneira obrigados a eleva os níveis do mesmo componente a um componente pai, então seguindo as regras e orientação a objeto centralizamos o state no pai e por definição quem usa é quem define (setState).



Primeiramente vamos modificar a estrutura do site no App.js e limpar as diretivas:

```
JS App.js
                JS Lista.js
                                 JS Form.js
meu projeto > src > JS App.js > 分 App
      import './App.css';
      import Lista from './components/Lista'
       function App() {
         const meusItens = ['React', 'Cordova', 'Monaca']
         return (
           <div className="App">
             <h1> Renderização de Listas </h1>
 10
             <Lista itens = {meusItens}/>
 11
 12
 13
       export default App;
```

```
JS SeuNome.js
JS App.js
                                   JS Lista.js
meu projeto > src > JS App.js > ♥ App
       import './App.css';
       function App() {
          return (
           <div className="App">
              <h1> Aula State Lift </h1>
            </div>
 10
 11
       export default App;
 12
```



Vamos criar um componente da mesma forma que criamos no começo os outros, porem ao invés de inserir a constante dentro desta função vamos direcionar ela diretamente no "Pai" App.js:



Prosseguindo inserimos a const na função importamos o useState, além do SeuNome e setamos o

SeuNome={SetNome} :

```
EXPLORADOR ... JS App.js 1 X JS SeuNome.js
PROJET... [♣ 🛱 ひ 🗗
                    meu_projeto > src > JS App.js > 🕅 App
                              import './App.css'

✓ meu_projeto

                              import {useState} from 'react'
                              import SeuNome from './components/SeuNome'

∨ components

  -- iviyi varricija
                              function App() {
  JS Pessoa.js
                                const [nome, SetNome] =useState()
  JS PwebIII.js
                                 return (
  # PwebIII.module.css
                                  <div className="App">
  JS SeuNome.js
                                    <h1> Aula State Lift </h1>
  # App.css
                                    <SeuNome SetNome={SetNome}/>
                        10
 JS App.js 1
                        11
                                  </div>
 JS App.test.js
                        12
                        13
  # index.css
                        14
                              export default App;
 JS index.js
```



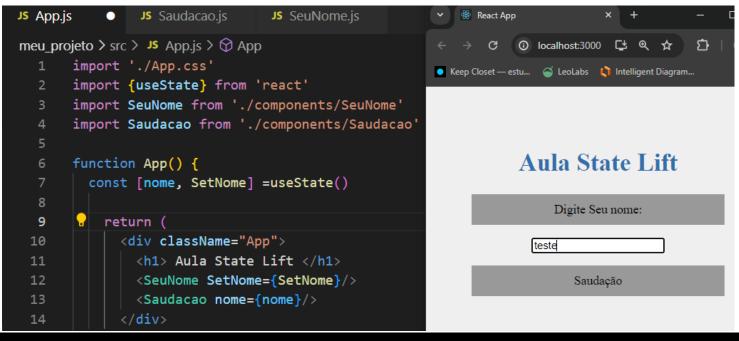
Voltamos para o SeuNome.js, implementamos o SetNome como propriedades, e criamos um onChange que vai capturar na tela a informação do SetNome e salvar no cache do navegador, lembrando que devemos colocar a variável {nome} dentro da função:

```
JS App.js 1
                JS SeuNome.js X
meu_projeto > src > components > JS SeuNome.js > ♦ SeuNome
       function SeuNome({SetNome}){
  3
           return(
                <div>
                    Digite Seu nome: 
                    <input type="text"</pre>
                    placeholder="Qual seria seu nome?"
                    onChange={(e) => SetNome(e.target.value)}/>
               </div>
 10
 11
 12
       export default SeuNome
```

```
JS App.js
                JS SeuNome.js
meu projeto > src > JS App.js > 🕥 App
       import './App.css'
       import {useState} from 'react'
       import SeuNome from './components/SeuNome'
       function App() {
         const [nome, SetNome] =useState()
          return (
           <div className="App">
             <h1> Aula State Lift </h1>
            <SeuNome SetNome={SetNome}/>
            {nome}
 11
       export default App;
```

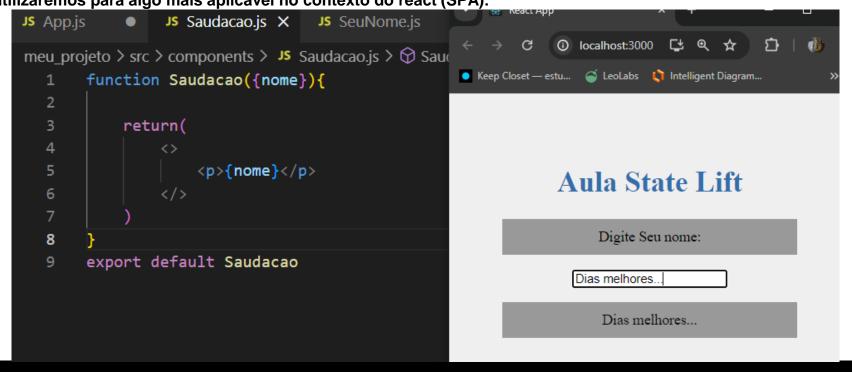


Prosseguindo já com esta parte desenvolvida entre o componente State filho e no App.js, vamos criar agora mais um componente para fazer toda aquela transição entre o get e o set mas de forma separadas, como vocês podem ver Saudação vai buscar o próprio no nome e mostrar na tela em baixo, mas para isto temos que trocar a propriedade no componente:



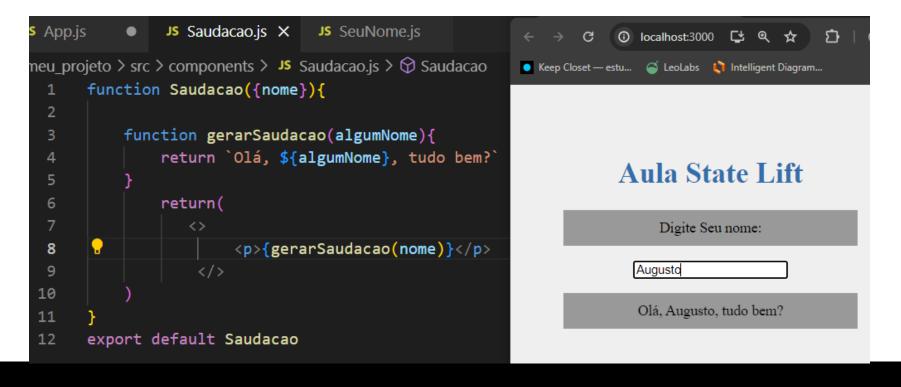


Prosseguindo em outro exemplo substituímos a saudação pelo nome e na função colocamos o nome como propriedade, como podem perceber agora a informação aparece diretamente embaixo de onde digitamos, no próximo slide utilizaremos para algo mais aplicável no contexto do react (SPA):



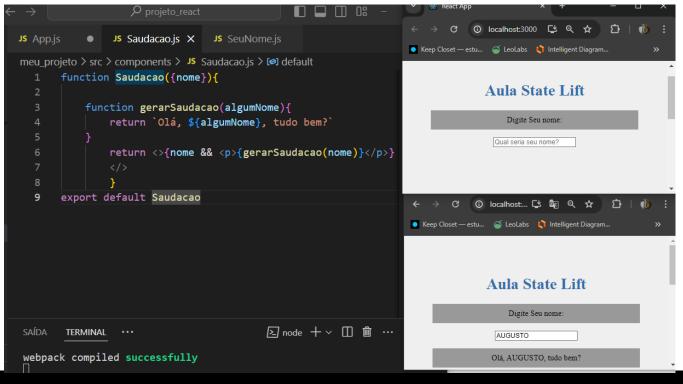


Dentro ainda do componente Saudacao.js vamos fazer outras modificações básicas, criando mais um função podemos agora concatenar uma frases por exemplo diretamente com o nome digitado:





Dentro ainda do componente Saudacao.js, podemos criar uma condição caso queiram, sendo assim só aparecera os dados, quando for digitado algo na parte de baixo:





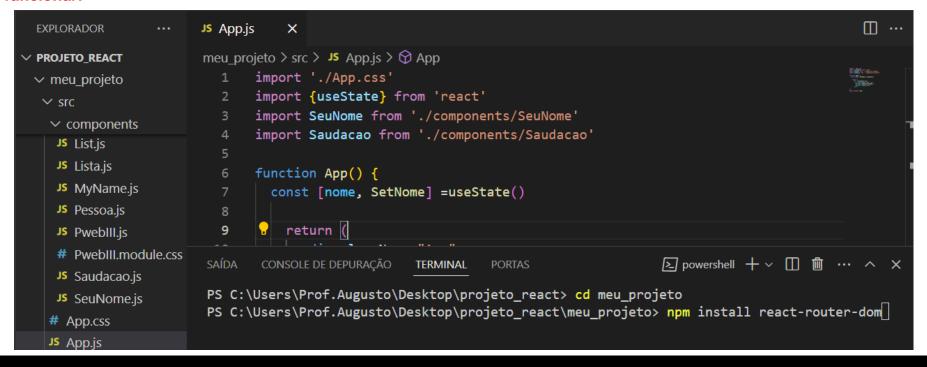
React Router

O React Router é um pacote para mudanças de URLs das aplicações para podermos acessar outras Views, sem o Page do Reload, trocando apenas uma parte do layout da aplicação, ou seja, o que muda de view para view.

Precisamos instalar este pacote no projeto, e também realizar algumas mudanças em como o App vai ser estruturado.



Prosseguindo, dentro do terminal entramos no projeto e executamos a instalação com npm install reactrouter-dom, caso não funciona esta versão mais adiante coloquei todas as possíveis causas e modificações para funcionar:





Damos uma limpada do App.js para montar um novo projeto e a estrutura do layout da página, importamos o Router, Switch, Route e link destes react DOM:

```
EXPLORADOR
                        JS App.js
                                    ×
                       meu_projeto > src > JS App.js > 🗘 App
PROJET... [♣ 🛱 🖔 🗗
                               import {BrowserRouter as Router, Switch, Route, link} from 'react-router-dom'

∨ meu_projeto

✓ src

                                function App() {
  components
                                    return (
   JS PwebIII.js
                                     <Router>
   # PwebIII.module.css
  JS Saudacao.js
  JS SeuNome.js
                                           <link to ="/">Home</link>
  # App.css
 JS App.js
                                           <link to ="/empresa">Empresa</link>
 JS App.test.js
  # index.css
 JS index.js
                                           <link to ="/contato">Contato</link>
 ¹ logo.svg
 JS reportWebVitals.js
 Js setupTests.js
                                     </Router>
 .gitignore
{} package-lock.json
                                export default App;
{} package.json
(I) DEADME ...
```



Na sequência criamos mais uma pasta, para as rotas dos links do App.js, são três, Contato, Empresa e Home dentro de um pasta chamada pages bem simples:

```
EXPLORADOR
                         JS App.js
                                          JS Home.js X
                                                           JS Contato.js
                                                                             JS Empresa.js
                         meu_projeto > src > pages > JS Home.js > ♥ Home

✓ PROJETO_REACT

                                 function Home(){

∨ meu_projeto

                                     return(
   > node_modules
                                          <div>
   > public
                                              <h1>Home</h1>

✓ src

                                              Conteúdo da Página
    components
                                          </div>
    pages
    JS Contato.js
                            8
                                 export default Home
    JS Empresa.js
    JS Home.js
     App.css
    JS App.is
```



Continuando - Depois que colocar os Imports vamos na parte de baixo e inserimos as rotas lembrando que com o BrowserRouter podemos acessar como Switch o menu das páginas sempre setando o router:

```
EXPLORADOR
                                                                             JS Empresa.js

∨ PROJETO REACT

                          meu projeto > src > JS App.js > 😭 App
                                  import {BrowserRouter as Router, Switch, Route, Link} from 'react-router-dom'
 meu projeto
                                 import Home from './pages/Home'
   > node_modules
                                 import Empresa from './pages/Empresa'
   > public
                                 import Contato from './pages/Contato'

✓ src

    > components
                                 function App()

∨ pages

     JS Contato.js
                                     return (
                                       <Router:
     JS Empresa.js
     JS Home.is
    # App.css
                                            <Link to="/">Home</Link>
    JS App.js
   JS App.test.is
                                            <Link to="/empresa">Empresa</Link>
    # index.css
   JS index.js
   logo.svq
                                            <Link to="/contato">Contato</Link>
   JS reportWebVitals.js
   JS setupTests.js
  aitianore
                                        <Switch>
  () package-lock.json
                                          <Route exact path="/">
  () package.json
                                            <Home />
                                          </Route>
  ① README.md
                                          <Route path="/empresa">
                                            <Empresa />
                                          </Route>
                                          <Route path="contato/">
                                            <Contato />
                                          </Route
                                        </Switch>
                                       </Router>
                                 export default App;
```



Parte do Switch do código neste exemplo, você está buscando e roteando no próprio Switch das páginas e renderizando todas em um único bloco dentro da página principal PAI e executando o

npm start:

```
<Switch>
        <Route exact path="/">
          <Home />
        </Route>
        <Route path="/empresa">
          <Empresa />
        </Route>
        <Route path="contato/">
          <Contato />
        </Route>
      </Switch>
     </Router>
export default App;
```

• Home • Empres: • Contato	<u>a</u>	
Home	₽.	
Conteúdo da	ı página	



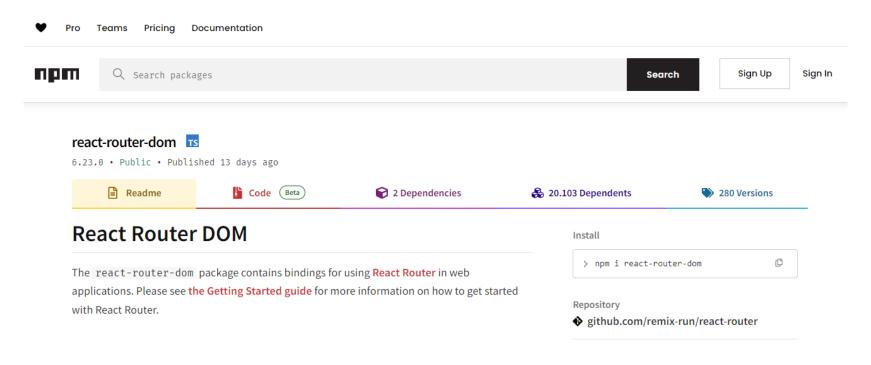
Caso aconteça este erro no nível de segurança SSL3 do server significa que pode estar desatualizado ou com falhas de segurança os pacotes do react, então devemos atualizar o node o react e outras aplicações de desenvolvimento, caso precisem instalar outras versões manualmente do node ou npm existe o nvm para isto https://github.com/coreybutler/nvm-windows/releases comandos nvm install vX.X.X, outra alternativa é utilizar o comando set NODE_OPTIONS=--openssl-legacy-provider antes do npm start que vai deixar no modo legado assim também deverá funcionar, em última instancia, caso ainda não funcione utilizem a versão mais antiga do router npm install react-router-dom@5.2.0 e isto serve para todos os módulos quando precisarem trocar lembrem das versões e

compatibilidades em qualquer projeto react:

```
meu_projeto > src > JS App.js > 🗘 App
                    import {BrowserRouter as Router, Switch, Route, Link} from 'react-router-dom'
                    import Home from './components/pages/Home';
                    import Empresa from './components/pages/Empresa';
                   import Contato from './components/pages/Contato';
                    function App() {
                      😯 return (
                                                    <Link to="/">Home</Link>
             at C:\Users\Prof.Augusto\Desktop\projeto_react\meu_projeto\node_modules\loader-runner\lib\LoaderRunner.js:233:18
             at \ context. callback \ (C:\Users\Prof.Augusto\Desktop\projeto\_react\meu\_projeto\node\_modules\loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loader-runner\lib\Loade
rRunner.is:111:13)
             at C:\Users\Prof.Augusto\Desktop\projeto react\meu projeto\node modules\babel-loader\lib\index.js:51:103 {
        opensslErrorStack: [ 'error:03000086:digital envelope routines::initialization error' ].
       library: 'digital envelope routines',
       reason: 'unsupported',
       code: 'ERR OSSL EVP UNSUPPORTED'
 Node.js v18.17.0
PS C:\Users\Prof.Augusto\Desktop\projeto react\meu projeto>
```

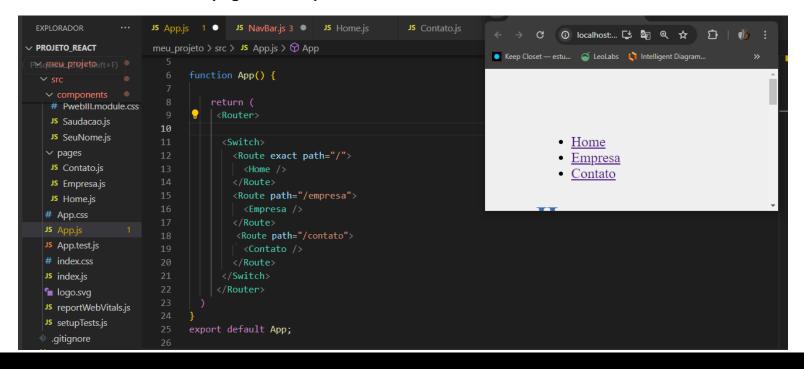


Mais informações das dependências ou se vocês tiverem alguma desatualizada basta irem no site npm lá existem várias informações para usarem nos projetos exemplo https://www.npmjs.com/package/react-router-dom :





Próximo passo, criamos o navbar e o footer dentro do escopo do layout componentes, simplificando toda a tarefa, no laravel é bem mais visível e no react também para o reaproveitamento dos componentes, depois podemos usar esta navbar e footer em todas as páginas sem precisar reescrever novamente:





Passo 2, criamos o navbar jogamos tudo no navbar e importamos o link do Router Dom:

```
JS NavBar.js
                                                             JS Home.js
                                                                                JS Contato.js
 EXPLORADOR
                                                                                                                 ① localhost:... 🚅 💁 🍳 🟠 🗀
                          meu projeto > src > components > JS NavBar.js > ...

∨ PROJETO REACT

    Keep Closet — estu...
    LeoLabs
    Intelligent Diagram...

                                  import {Link} from 'react-router-dom'

✓ meu projeto

                                  function NavBar(){

✓ src

    Home

                                      return(
   components

    Empresa

    Contato

    JS Condicional.is
    JS Evento.js
                                             <Link to="/">Home</Link>
                                                                                                           Home
    JS Form.js
    JS HelloWorld.js
                                             <Link to="/empresa">Empresa</Link>
    JS Item.js
                                                                                                            Conteúdo da Página
    JS List.js
    JS Lista.js
                                             <Link to="/contato">Contato</Link>
    JS MyName.js
    JS NavBar.js
    JS Pessoa.js
    JS PwebIII.js
                                  export default NavBar
     # PwebIII.module.css
```

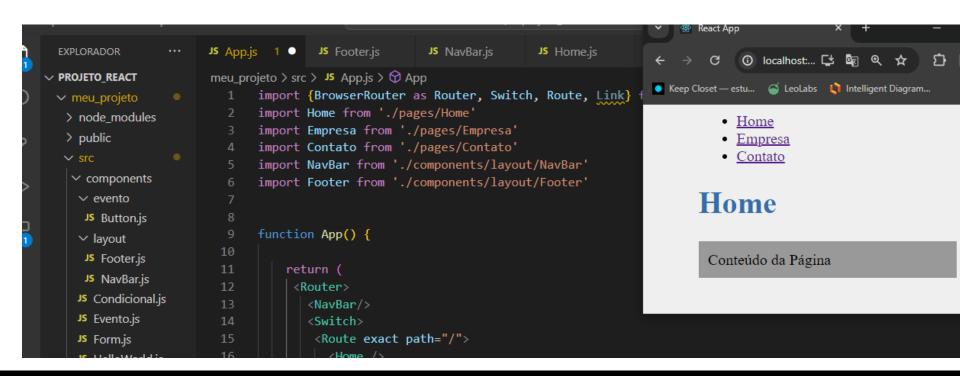


Passo 3, voltamos para o App.js e colocamos o link e o import do NavBar:

```
JS NavBar.js • JS Home.js
                                                JS Contato.js
                                                                 JS Empresa.js
meu_projeto > src > JS App.js > ...
       import {BrowserRouter as Router, Switch, Route, Link} from 'react-router-dom'
      import Home from './pages/Home'
       import Empresa from './pages/Empresa'
       import Contato from './pages/Contato'
       import NavBar from './components/NavBar'
       function App() {
           return (
            <Router>
              <NavBar/>
             <Switch>
               <Route exact path="/">
                 <Home />
               </Route>
               <Route path="/empresa">
                 <Empresa />
               </Route>
                <Route path="/contato">
                 <Contato />
```

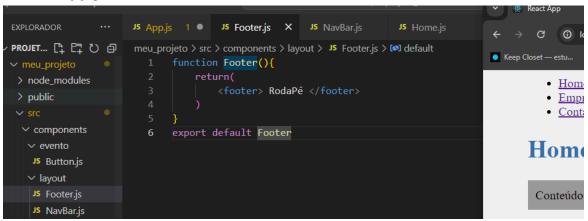


Passo 4, voltamos para o App.js e colocamos o link e o import do NavBar e já vamos montando também um footer no componente e para padronizar criamos uma pasta chamada layout:



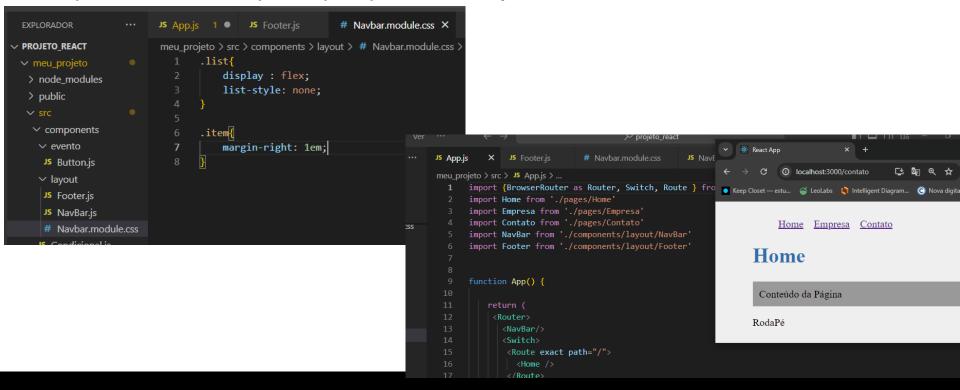


Passo 5, Footer no mesmo esquema e no App.js colocado lá no final do Switch:



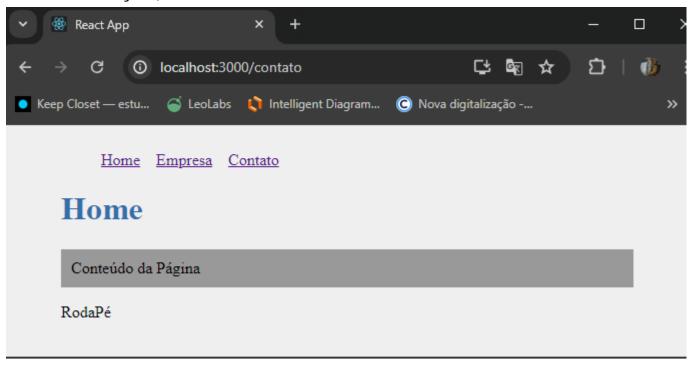


Passo 6, Utilizar um CSS no mesmo esquema dentro do componentes e ai vocês podem montar de sua maneira para testar, lembrando que tem que importar no navbar.js:





TAREFA PARA PRÓXIMA ENTREGA, VOCÊS VÃO RENDERIZAR O LAYOUT DE PÁGINA ÚNICA, FICA A SEU CRITÉRIO A PERSONALIZAÇÃO, DESDE QUE UTILIZEM O PADRÃO REACT SPA QUE FOI PASSADO:





GOVERNO DO ESTADO DE SÃO PAULO