**=== Install ===**

npm install jquery popper.js bootstrap@4.6.1 --save

index.js: bootstrap.min.css > index.css > (scss) > App > jquery > popper.js > bootstrap.js

**=== Xây dựng cơ bản cấu trúc HomeTemplate ===**

(src/containers/HomeTemplate/\_components (NavbarHome, FooterHome, ...), HomePage, About Page, ProductsPage(Product),PageNotFound, ...)

**=== Xây dựng bộ định tuyến, tạo đường dẫn đến các page vừa tạo ===**

npm install react-router-dom@5.3.0 –save

- Sử dụng Route của react-router-dom để định tuyến cho mỗi đường dẫn, cấu trúc cơ bản cho mỗi Route:

<*Route* *path*="/" *component*={HomePage} />

- Sử dụng BrowserRouter của react-router-dom để định nghĩa routing, bao bọc nội dung trong App.js (kèm theo cặp thẻ <Switch> của react-router-dom để định tuyến cho đường dẫn mà end-user nhập sai sẽ dẫn về trang Page Not Found – đặt <Route> của Page Not Found ở cuối thẻ <Switch>)

import { BrowserRouter, Route, Switch } from "react-router-dom";

import AboutPage from "./containers/HomeTemplate/AboutPage";

import HomePage from "./containers/HomeTemplate/HomePage";

import PageNotFound from "./containers/HomeTemplate/PageNotFound";

import ProductsPage from "./containers/HomeTemplate/ProductsPage";

function App() {

  return (

    <*BrowserRouter*>

      <*Switch*>

        <*Route* *exact*={true} *path*="/" *component*={HomePage} />

        <*Route* *path*="/about" *component*={AboutPage} />

        <*Route* *path*="/products" *component*={ProductsPage} />

        <*Route* *path*="" *component*={PageNotFound} />

      </*Switch*>

    </*BrowserRouter*>

  );

}

export default App;

- Phát triển web có thể rất nhiều trang => phát sinh nhiều thẻ <Route> => tạo thư mục src/routes để quản lý bộ định tuyến, bao gồm nhiều array quản lý thuộc tính exact, path, component cho từng Template.

- Thay vì duyệt từng mảng array để render ra nhiều thẻ <Route>, ta duyệt mảng và render ra theo từng Template component (truyền props vào Template component trong đó chứa thẻ <Route> và các component chung như Navbar, Footer, ...) để phân luồng cho từng Template

import AboutPage from "../containers/HomeTemplate/AboutPage";

import HomePage from "../containers/HomeTemplate/HomePage";

import ProductsPage from "../containers/HomeTemplate/ProductsPage";

import HomeTemplate from "../containers/HomeTemplate";

// import { Route } from "react-router-dom";

const routesHome = [

  {

    exact: true,

    path: "/",

    component: HomePage

  },

  {

    exact: false,

    path: "/about",

    component: AboutPage

  },

  {

    exact: false,

    path: "products",

    component: ProductsPage

  }

]

// export const renderRoutesHome = () => {

//   return routesHome.map((route, index) => {

//     return <Route

//       key={index}

//       exact={route.exact}

//       path={route.path}

//       component={route.component}

//     />

//   })

// }

// Thay vì duyệt mảng và return về các thẻ <Route/>, ta return về component <HomeTemplate/> để phân luồng với các template sau này, đồng thời truyền props vào

export const renderRoutesHome = () => {

  return routesHome.map((*route*, *index*) => {

    return <*HomeTemplate*

*key*={*index*}

*exact*={*route*.exact}

*path*={*route*.path}

*component*={*route*.component}

    />

  })

}

Trong component HomeTemplate return về:

import React, { Fragment } from 'react';

import { Route } from 'react-router-dom';

import NavbarHome from './\_components/Navbar';

export default function HomeTemplate(*props*) {

  const { exact, path, component } = *props*;

  return (

    <*Fragment*>

      <*NavbarHome* />

      <*Route* *exact*={exact} *path*={path} *component*={component} />

    </*Fragment*>

  )

}

Khi đó App.js được viết lại:

import { BrowserRouter, Route, Switch } from "react-router-dom";

import PageNotFound from "./containers/HomeTemplate/PageNotFound";

import { renderRoutesHome } from "./routes";

function App() {

  return (

    <*BrowserRouter*>

      <*Switch*>

        {renderRoutesHome()}

        <*Route* *path*="" *component*={PageNotFound} />

      </*Switch*>

    </*BrowserRouter*>

  );

}

export default App;

(Lỗi chưa xử lý: VD khi vào path localhost:3000/about/abc => không về trang PageNotFound)

**=== Xây dựng Navbar cho HomeTemplate, các thẻ <a> liên kết với đường dẫn vừa tạo ===**

- Tạo nhanh bằng navbar của bootstrap. Tuy nhiên phải thay toàn bộ thẻ <a> (có thuộc tính href làm refresh lại page => phá vỡ cấu trúc SPA) bằng Link/NavLink của react-router-dom; thay thế href='...' bằng to='...'

(NavLink có props là activeClassName để đặt tên class khi thẻ <li> đang được active (thêm props exact ở thẻ <li> đầu tiên để tránh nhầm lẫn cho các case tiếp theo - tương tự lúc khai báo bộ định tuyến)

import './style.css';

import React from 'react';

import { Link, NavLink } from 'react-router-dom';

export default function NavbarHome() {

  return (

    <nav *className*="navbar navbar-expand-lg navbar-light bg-light">

      <*Link* *className*="navbar-brand" *to*="/">Home Navbar</*Link*>

      <button *className*="navbar-toggler" *type*="button" *data-toggle*="collapse" *data-target*="#navbarSupportedContent01" *aria-controls*="navbarSupportedContent" *aria-expanded*="false" *aria-label*="Toggle navigation">

        <span *className*="navbar-toggler-icon" />

      </button>

      <div *className*="collapse navbar-collapse" *id*="navbarSupportedContent01">

        <ul *className*="navbar-nav mr-auto">

          <li *className*="nav-item">

            <*NavLink* *exact* *activeClassName*="active" *className*="nav-link" *to*="/">Home</*NavLink*>

          </li>

          <li *className*="nav-item">

            <*NavLink* *activeClassName*="active" *className*="nav-link" *to*="/about">About</*NavLink*>

          </li>

          <li *className*="nav-item">

            <*NavLink* *activeClassName*="active" *className*="nav-link" *to*="/products">Products</*NavLink*>

          </li>

        </ul>

      </div>

    </nav>

  )

}

=== Xây dựng components dùng chung cho project ===

(src > components > loader)

npm install styled-components --save để code bằng styled hoặc chèn bằng file css