**Lesson 02 Demo 03**

**Creating a React Router**

**Objective:** To create a React router with protected features to ensure that only authenticated users can access specific components.

**Tools required:** Node.js and React.js

**Prerequisites:** None

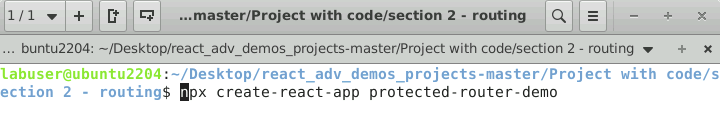
Steps to be followed:

1. Create and set up the React project
2. Update index.js file
3. Create the required files for the src folder
4. Add the bootstrap features
5. Test the application

**Step 1: Create and set up the React project**

1. Open a terminal window to run the following command to create a React application:

**npx create-react-app protected-router-demo**

****

1. Open the created React application folder (**protected-router-demo**) in VS Code by clicking on **File** in the top left corner and selecting **Open Folder**

**A screenshot of a computer

Description automatically generated**

1. Click on **Open** button

A screenshot of a computer

Description automatically generated

The folder structure appears as follows:

**A screenshot of a black screen

Description automatically generated**

1. Inside the project, open the **TERMINAL** and run the following command to install the required dependencies:

**npm install**

**A screenshot of a computer program

Description automatically generated**

1. Run the following command to install routing dependencies:

**npm install react-router-dom**

**A screen shot of a computer

Description automatically generated**

**Note**: These commands help to install all the required dependencies mentioned in the **package.json** file on the local machine as a **node\_module** folder.

1. View the **react-router-dom** dependencies in the **package.json** file

**A screenshot of a computer program

Description automatically generated**

**Step 2: Update index.js file**

1. Inside the **src** folder, open the **index.js** file, and add the **BrowserRouter** tag which helps to enable routing features to the whole application.

**import React from 'react';**

**import ReactDOM from 'react-dom/client';**

**import './index.css';**

**import App from './App';**

**import reportWebVitals from './reportWebVitals';**

**import {BrowserRouter} from 'react-router-dom'**

**const root = ReactDOM.createRoot(document.getElementById('root'));**

**root.render(**

**<React.StrictMode>**

**<BrowserRouter>**

**<App />**

**</BrowserRouter>**

**</React.StrictMode>**

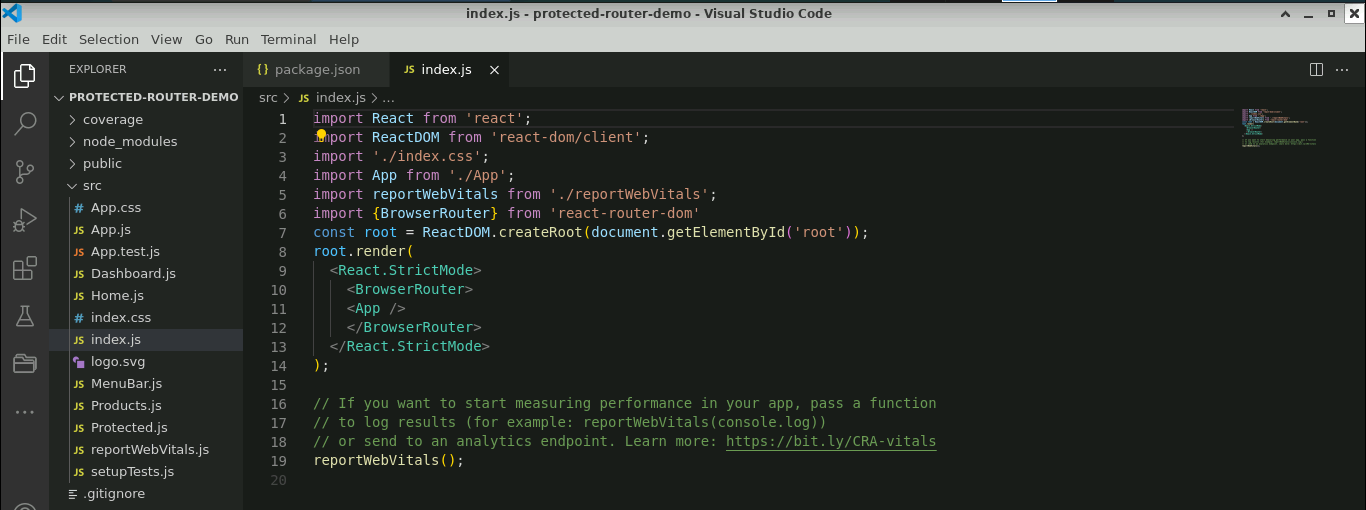
**);**

// If you want to start measuring performance in your app, pass a function

// to log results (for example: reportWebVitals(console.log))

// or send to an analytics endpoint. Learn more: https://bit.ly/CRA-vitals

**reportWebVitals();**

****

**Step 3:** **Create the required files for the src folder**

1. Right-click on the **src** folder and select **New File**

**A screenshot of a computer

Description automatically generated**

* 1. Create a file named **Home.js**

**A computer screen with a black and white screen

Description automatically generated**

* 1. Inside the **Home.js** file, add the following code as shown below:

**import React from 'react'**

**function Home() {**

**return (**

**<div className="card mt-5 text-center">**

**<div className="card-body">Home</div>**

**</div>**

**)**

**}**

**export default Home**

**A screenshot of a computer

Description automatically generated**

* 1. Inside the **src** folder, create a **Products.js** file and add the following code as shown below:

**import React from 'react'**

**function Products() {**

**return (**

**<div className="card mt-5 text-center">**

**<div className="card-body">Products</div>**

**</div>**

**)**

**}**

**export default Products**

**A screenshot of a computer

Description automatically generated**

* 1. Create a **Dashboard.js** file and add the following code as shown:

**import React from 'react'**

**function Dashboard() {**

**return (**

**<div className="card mt-5 text-center">**

**<div className="card-body">Dashboard</div>**

**</div>**

**)**

**}**

**export default Dashboard**

**A computer screen shot of a black screen

Description automatically generated**

* 1. Create a **Producted.js** file and add the following code as shown below:

**import React from 'react'**

**import { Navigate } from 'react-router-dom'**

**function Protected({ isSignedIn, children }) {**

**if (!isSignedIn) {**

**return <Navigate to="/" replace />**

**}**

**return children**

**}**

**export default Protected**

**A screenshot of a computer

Description automatically generated**

* 1. Create the **MenuBar.js** file and add the following code as shown below. This file provides a routing link for all components (**Home.js**, **Product.js**, **Dashboard.js** file)

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

**const MenuBar = () => {**

**return (**

**<nav className="d-flex justify-content-center">**

**<div className="p-2">**

**<Link to="/">Home</Link>**

**</div>**

**<div className="p-2">**

**<Link to="/products">Products</Link>**

**</div>**

**<div className="p-2">**

**<Link to="/dashboard">Dashboard</Link>**

**</div>**

**</nav>**

**)**

**}**

**export default MenuBar**

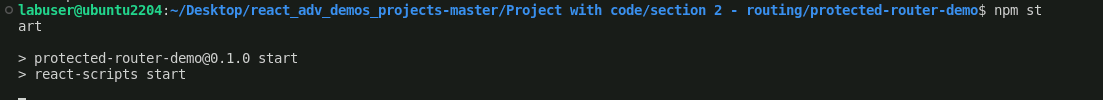
**A screenshot of a computer program

Description automatically generated**

**Step 4: Add the Bootstrap features**

* 1. Open the terminal and run the below command to execute the application:

**npm start**

****

* 1. The following output will appear then, right-click on the screen and select **Inspect**

**A screenshot of a computer

Description automatically generated**

* 1. Navigate to **sources**, under the **Page** tabselect **(index)**

**A screenshot of a computer

Description automatically generated**

The following screenshot shows the **(index)** file

**A screenshot of a computer

Description automatically generated**

* 1. Inside the **src** folder, create an **index.html** file, then copy and paste the **(index)** file content under the **index.html** file

**A computer screen shot of a computer code

Description automatically generated**

**Step 5: Test the application**

* 1. Open the **App.js** file which manages the user's authentication status (**isSignedIn**) with the help of the **state** hook

**import './App.css'**

**import { useState } from 'react'**

**import {Routes, Route } from 'react-router-dom'**

**import MenuBar from './MenuBar'**

**import Home from './Home'**

**import Dashboard from './Dashboard'**

**import Products from './Products'**

**import Protected from './Protected'**

**export default function App() {**

**const [isSignedIn, setIsSignedIn] = useState(null)**

**const signin = () => {**

**setIsSignedIn(true)**

**}**

**const signout = () => {**

**setIsSignedIn(false)**

**}**

**return (**

**<div className="container mt-2">**

**<h2 className="mb-5 text-center">React Protected Routes Example</h2>**

**<MenuBar />**

**<Routes>**

**<Route path="/" element={<Home />} />**

**<Route**

**path="/dashboard"**

**element={**

**<Protected isSignedIn={isSignedIn}>**

**<Dashboard />**

**</Protected>**

**}**

**/>**

**<Route**

**path="/products"**

**element={**

**<Protected isSignedIn={isSignedIn}>**

**<Products />**

**</Protected>**

**}**

**/>**

**</Routes>**

**{isSignedIn ? (**

**<div className="d-grid mt-2">**

**<button className="btn-danger" onClick={signout}>**

**Sign out**

**</button>**

**</div>**

**) : (**

**<div className="d-grid mt-2">**

**<button className="btn-dark" onClick={signin}>**

**Sign in**

**</button>**

**</div>**

**)}**

**</div>**

**)**

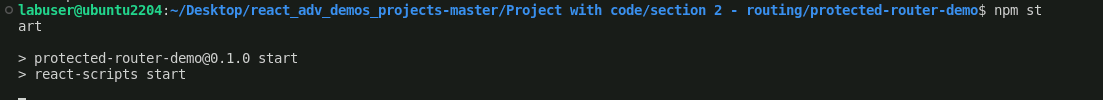
**}**

**A screen shot of a computer

Description automatically generated**

* 1. Open the terminal and run the below command to execute the application:

**npm start**

****

The output appears as shown below:

A screenshot of a computer

Description automatically generated

Note: The project runs on the default browser with the URL [**http://localhost:3000**](http://localhost:3000)

* 1. Click on **Products** and **Dashboard**. Navigation is restricted.

**A screenshot of a computer

Description automatically generated**

* 1. Click on the **Sign In** button to activate the **isSignIn** boolean value. Now user can proceed to navigate to either the **Products** or **Dashboard** components.

**A screenshot of a computer

Description automatically generated**

The output appears as shown below:

**A screenshot of a computer error

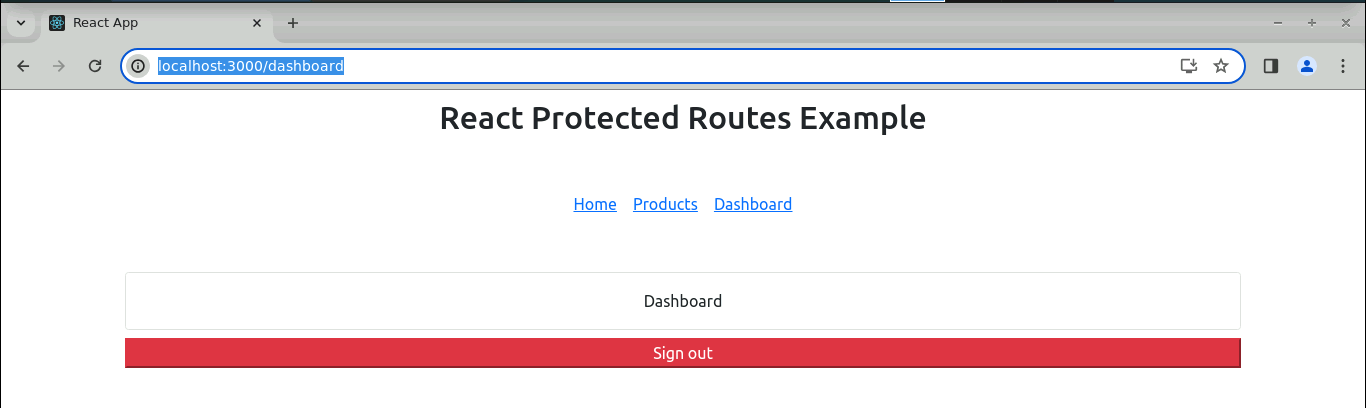
Description automatically generated**

* 1. Click on **Products**, it navigates to the **Products** component with **localhost:3000/products**

A screenshot of a computer

Description automatically generated

* 1. Click on **Dashboard**, it navigates to the **Dashboard** component with **localhost:3000/dashboard**



With this, you have successfully created a React router with protected features to ensure that only authenticated users can access specific components.