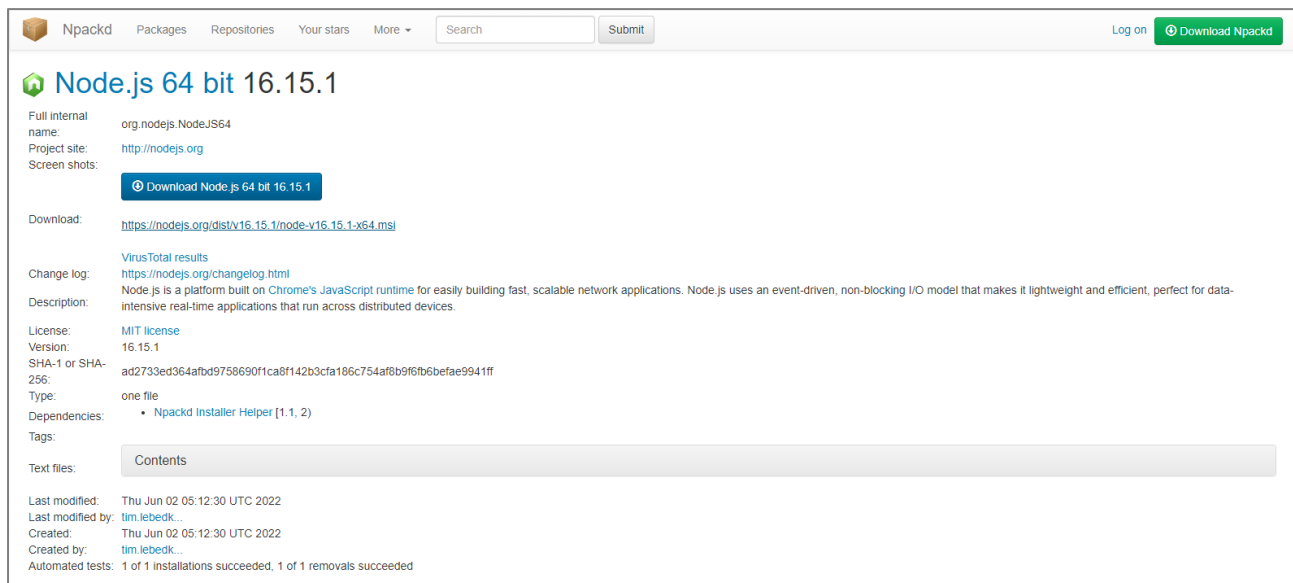


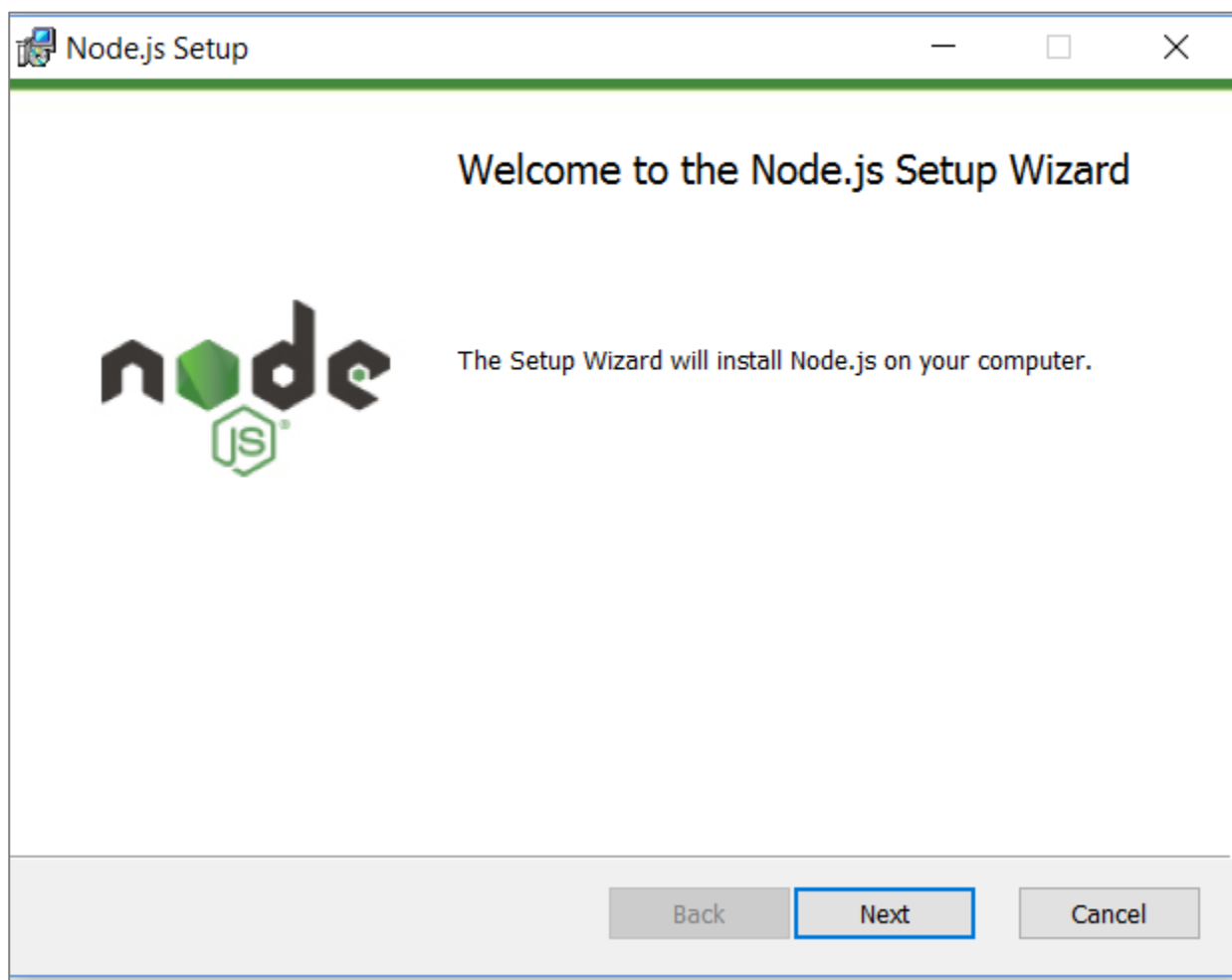
Module No:	1	IU No:	2	Exercise No.	2
------------	---	--------	---	--------------	---

Lab Assessment Statement	<p><u>Assignment 2 - Build React JS Frontend Application</u></p> <p>Referring to Assignment-1's Project Scenario,</p> <p>The scope of this assignment is to develop a frontend for 'XYZ Cars Pte Ltd' as a React developer to develop a Used Car Sales portal.</p> <p>These are the steps provided to build an API.</p> <ol style="list-style-type: none">1. Install Node JS.2. Create React Application using npx create-react-app [project name] command.3. Run Application using npm start command.4. Create Service to fetch the data from API.5. Create methods to make HTTP REST call via Axios.6. Create the required components to display the data coming from services.7. Implement React Router to configure routing links. <p>Provide the source code of developed application.</p> <p>Provide screen capture of final result pages for API testing.</p>
Technical Environment	- REST API, React JS
Guidelines	-
Duration	120 mins

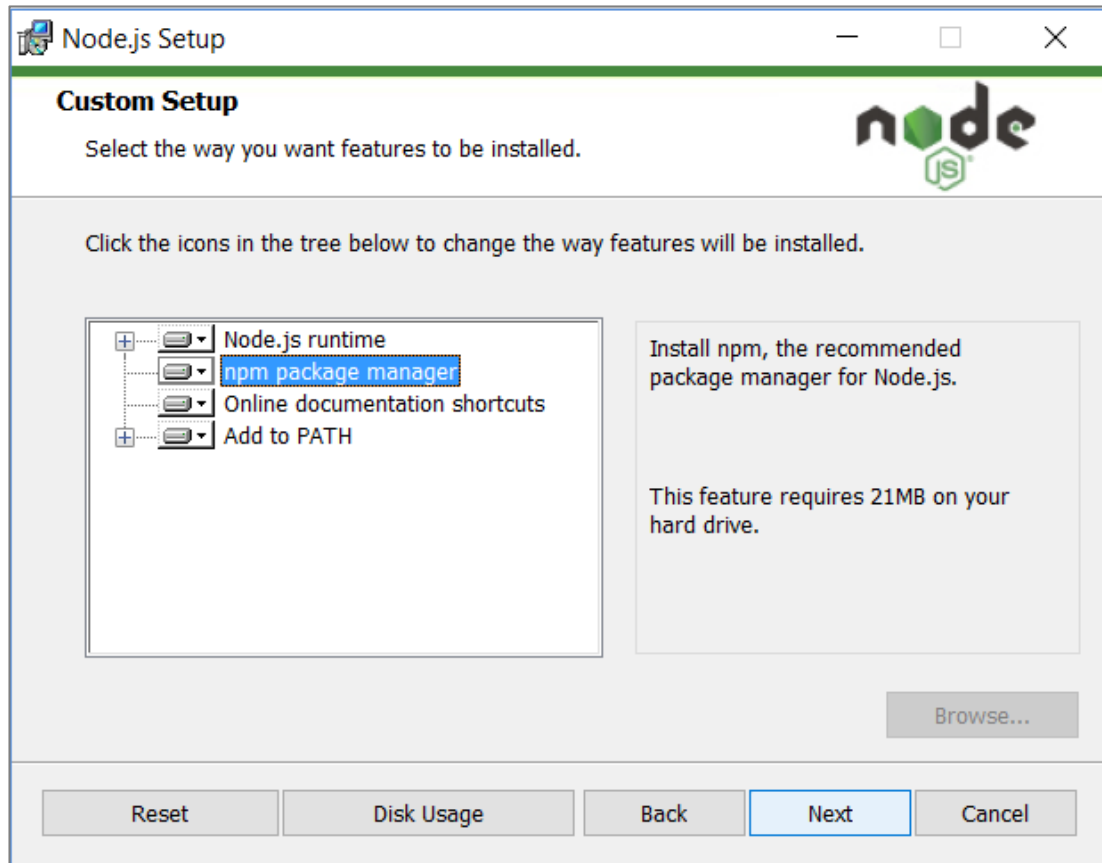
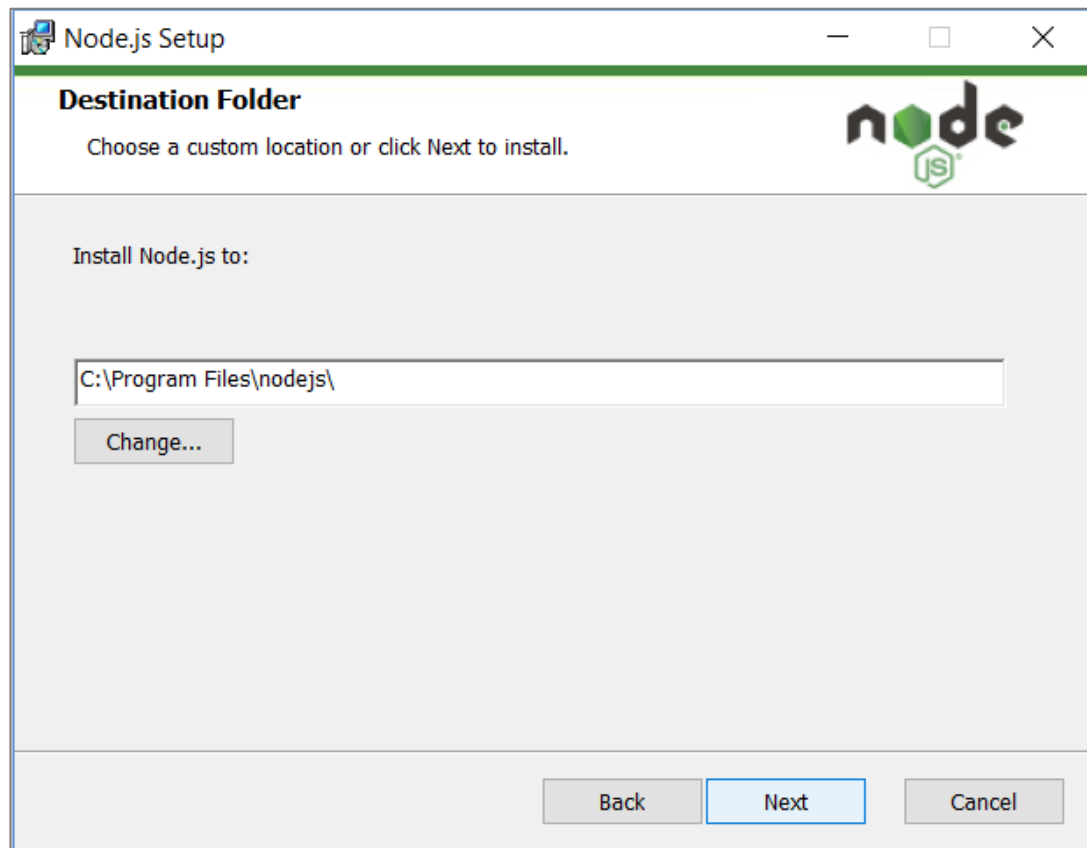
1. Install Node JS.

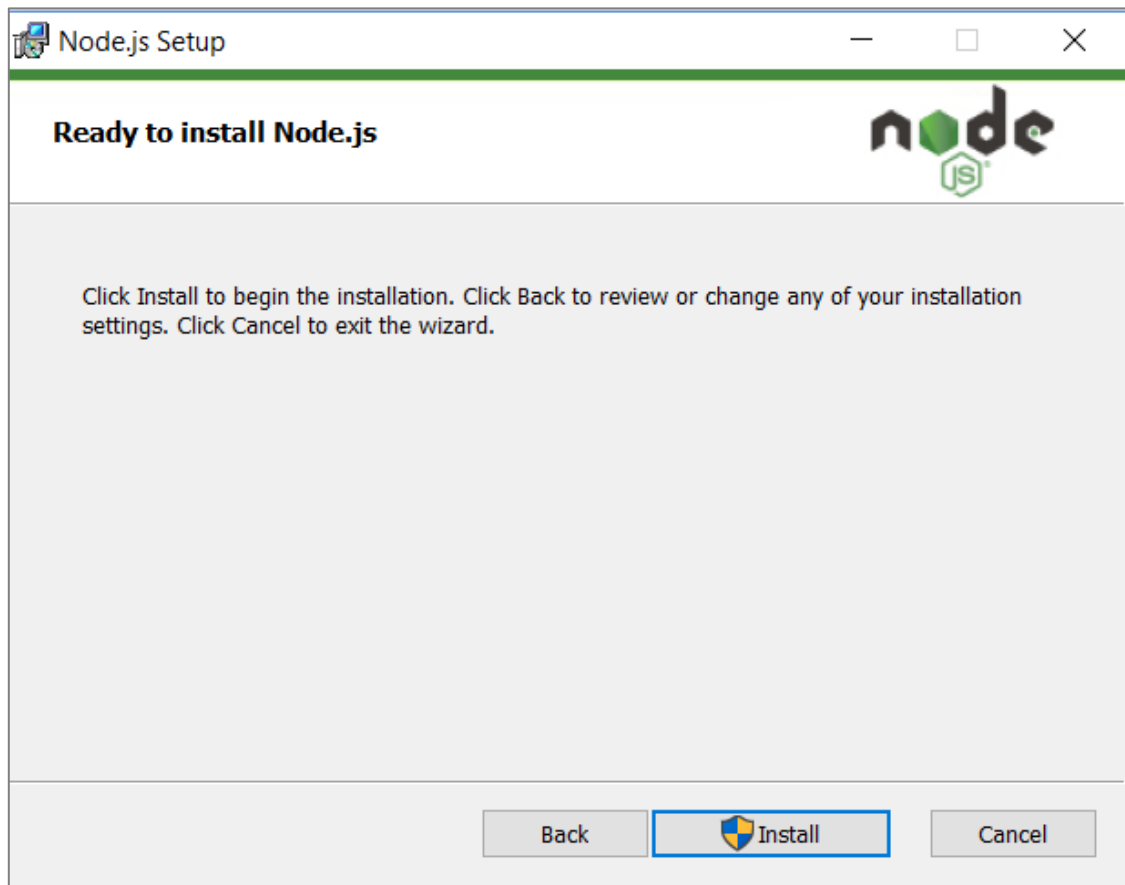
Step 1: Download Node JS

The screenshot shows the Npackd interface for downloading Node.js 64 bit 16.15.1. The page includes a search bar, a 'Download Npackd' button, and a 'Log on' link. The main content area displays the package name 'Node.js 64 bit 16.15.1' and a 'Download Node.js 64 bit 16.15.1' button. Below this, the download URL is provided: <https://nodejs.org/dist/v16.15.1/node-v16.15.1-x64.msi>. The page also lists the full internal name, project site, screen shots, change log, description, license, version, SHA-1 or SHA-256, type, dependencies, tags, and text files. The last modified date is Thu Jun 02 05:12:30 UTC 2022, and the created date is also Thu Jun 02 05:12:30 UTC 2022. The automated tests section shows 1 of 1 installations succeeded and 1 of 1 removals succeeded.

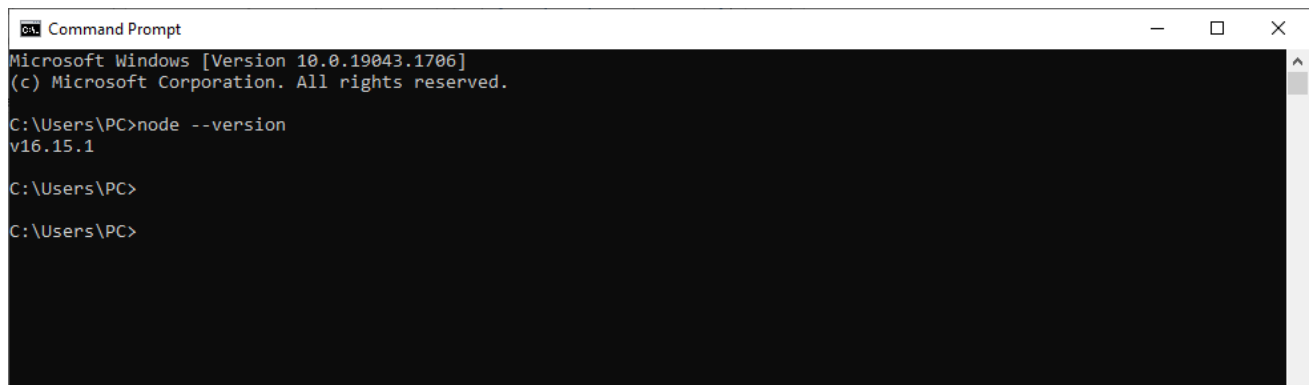
Step 2: Install NodeJS and npm

Step 3: Choose the desired path where to install NodeJS





Step 4: Check NodeJS and npm version

A screenshot of a Windows Command Prompt window. The title bar reads "Command Prompt". The text inside shows the Windows version and copyright information, followed by the command `node --version` being executed, which returns `v16.15.1`. The prompt is currently at `C:\Users\PC>`.

```
Microsoft Windows [Version 10.0.19043.1706]
(c) Microsoft Corporation. All rights reserved.

C:\Users\PC>node --version
v16.15.1

C:\Users\PC>

C:\Users\PC>
```

2. Create React Application using **npx create-react-app [project name]** command.

```
Microsoft Windows [Version 10.0.19043.1706]
(c) Microsoft Corporation. All rights reserved.

C:\Users\PC>npx create-react-app my-app
npm WARN config global `--global`, `--local` are deprecated. Use `--location=global` instead.

Creating a new React app in C:\Users\PC\my-app.

npm WARN config global `--global`, `--local` are deprecated. Use `--location=global` instead.
Installing packages. This might take a couple of minutes.
Installing react, react-dom, and react-scripts with cra-template...
```

```
Success! Created my-app at C:\Users\PC\my-app
Inside that directory, you can run several commands:

  npm start
    Starts the development server.

  npm run build
    Bundles the app into static files for production.

  npm test
    Starts the test runner.

  npm run eject
    Removes this tool and copies build dependencies, configuration files
    and scripts into the app directory. If you do this, you can't go back!

We suggest that you begin by typing:

  cd my-app
  npm start

Happy hacking!
C:\Users\PC>
```

3. Run Application using **npm start** command.

```
PS C:\Users\PC\Documents\Assignment 2\xyz-cars-react> npm start
npm WARN config global `--global`, `--local` are deprecated. Use `--location=global` instead.

> xyz-cars-frontend@0.1.0 start
> react-scripts start
```

4. Create Service to fetch the data from API and Create methods to make HTTP REST call via Axio

Login.JS

```
const LoginForm = () => {  
  const authCtx = useContext(AuthContext);  
  const inputUsernameRef = useRef();  
  const inputPasswordRef = useRef();  
  
  const navigate = useNavigate();  
  
  const [loginStatus, setLoginStatus] = useState("");  
  
  const onSubmitHandler = (e) => {  
    e.preventDefault();  
    const inputUsername = inputUsernameRef.current.value;  
    const inputPassword = inputPasswordRef.current.value;  
  
    console.log(inputUsername);  
    console.log(inputPassword);  
  
    axios  
      .post("http://localhost:8080/api/auth/login", {  
        username: inputUsername,  
        password: inputPassword,  
      })  
      .then((res) => {  
        console.log(res);  
        setLoginStatus("SUCCESS");  
        authCtx.login(res.data.accessToken);  
        navigate("/");  
      })  
      .catch((err) => {  
        console.log(err.message);  
        setLoginStatus("FAILED");  
      });  
  
    inputUsernameRef.current.value = "";  
    inputPasswordRef.current.value = "";  
  }  
};
```

Registration.JS

```
const RegistrationForm = () => {  
  const inputUsernameRef = useRef();  
  const inputPasswordRef = useRef();  
  
  const [registerStatus, setRegisterStatus] = useState("");  
  
  const onSubmitHandler = (e) => {  
    e.preventDefault();  
    const inputUsername = inputUsernameRef.current.value;  
    const inputPassword = inputPasswordRef.current.value;  
  
    console.log(inputUsername);  
    console.log(inputPassword);  
  
    axios  
      .post("http://localhost:8080/api/auth/register", {  
        username: inputUsername,  
        password: inputPassword,  
      })  
      .then((res) => {  
        setRegisterStatus("SUCCESS");  
        console.log(res);  
      })  
      .catch((err) => {  
        setRegisterStatus("FAILED");  
        console.log(err.message);  
      });  
  
    inputUsernameRef.current.value = "";  
    inputPasswordRef.current.value = "";  
  };  
};
```

Car-Service.JS – Get Car List

```
3 // GET List Car
4 export function getListCar(callback, errorCallback) {
5   axios
6     .get("http://localhost:8080/api/cars")
7     .then((res) => {
8       if (callback !== null) {
9         callback(res.data);
10      }
11    })
12     .catch((err) => {
13       if (errorCallback !== null) {
14         errorCallback(err.message);
15       }
16     });
17 }
```

Car-Service.JS – Get Car Details

```
19 // GET Car Detail
20 export function getCarDetail(idCar, callback, errorCallback) {
21   axios
22     .get("http://localhost:8080/api/cars/" + idCar)
23     .then((res) => {
24       if (callback !== null) {
25         callback(res.data);
26       }
27     })
28     .catch((err) => {
29       if (errorCallback !== null) {
30         errorCallback(err.message);
31       }
32     });
33 }
```


Car-Service.JS – Search Car Details

```
35 // GET Search By Keyword
36 export function getSearchByKeyword(keyword, callback, errorCallback) {
37     axios
38         .get(`http://localhost:8080/api/cars?keyword=${keyword}`)
39         .then((res) => {
40             if (callback != null) {
41                 callback(res.data);
42             }
43         })
44         .catch((err) => {
45             if (errorCallback != null) {
46                 errorCallback(err.message);
47             }
48         });
49 }
```

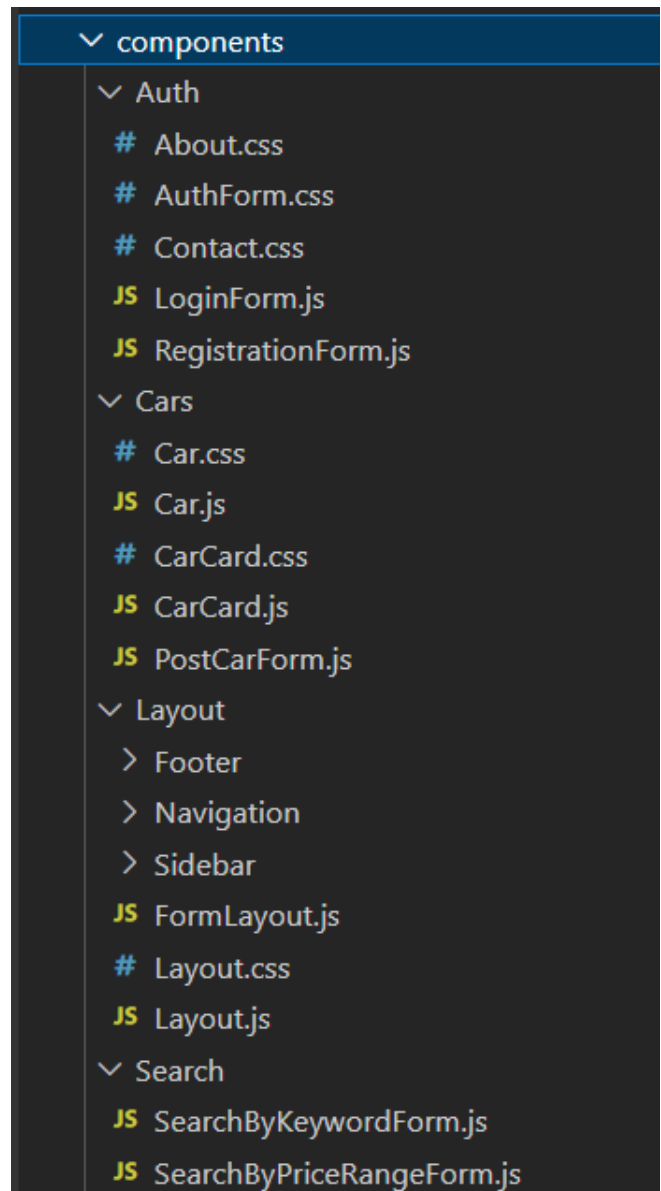
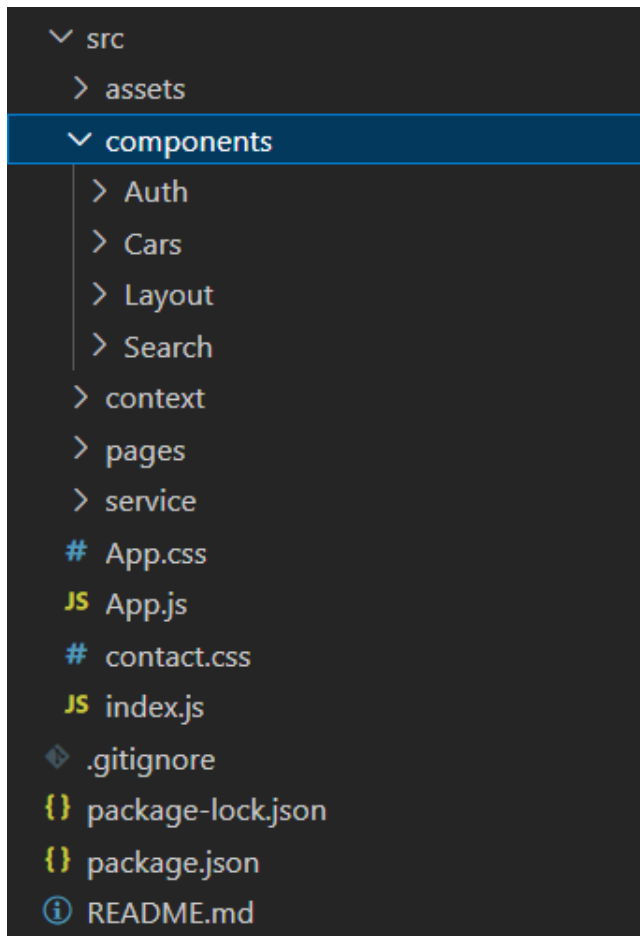
Car-Service.JS – Search car according to price Range

```
51 // GET Search By Price Range
52 export function getSearchByPriceRange(min, max, callback, errorCallback) {
53     axios
54         .get(`http://localhost:8080/api/cars?min=${min}&max=${max}`)
55         .then((res) => {
56             if (callback != null) {
57                 callback(res.data);
58             }
59         })
60         .catch((err) => {
61             if (errorCallback != null) {
62                 errorCallback(err.message);
63             }
64         });
65 }
```

Assignment 2

Chathushi Jayarathna

5. Create the required components to display the data coming from services.



List Car / Search Car (home.js)

```

const Home = () => {
  const [searchParams] = useSearchParams();
  const [listCar, setListCar] = useState([]);

  // Get List Car
  useEffect(() => {
    if (
      searchParams.get("keyword") === null &&
      searchParams.get("min") === null &&
      searchParams.get("max") === null
    ) {
      getListCar(
        (data) => {
          setListCar(data);
        },
        (error) => {
          console.log(error);
        }
      );
    }

    return () => {};
  }, [searchParams]);

  // Search By Keyword (Make, Model, Year)
  useEffect(() => {
    if (searchParams.get("keyword") !== null) {
      const keyword = searchParams.get("keyword");

      getSearchByKeyword(
        keyword,
        (data) => {
          setListCar(data);
        },
        (error) => {
          console.log(error);
        }
      );
    }

    return () => {};
  }, [searchParams]);

```

```

// Search By Price Range
useEffect(() => {
  if (searchParams.get("min") !== null) {
    const min = searchParams.get("min");
    const max = searchParams.get("max");

    getSearchByPriceRange(
      min,
      max,
      (data) => {
        setListCar(data);
      },
      (error) => {
        console.log(error);
      }
    );
  }

  return () => {};
}, [searchParams]);

return (
  <Layout>
    <div className="mt-3 row">
      {searchParams.get("keyword") !== null && (
        <p className="text-secondary ms-1">
          Search "{searchParams.get("keyword")}"
        </p>
      )}
      {searchParams.get("min") !== null && (
        <p className="text-secondary ms-1">
          Min price: ${searchParams.get("min")},
          {searchParams.get("max")}
        </p>
      )}
      {listCar.map((car) => (
        <CarCard
          key={car.idCar}
          idCar={car.idCar}
          make={car.make}
          model={car.model}
          year={car.year}
          price={car.price}
        />
      ))}
    </div>
  </Layout>

```

Car Details

```
1  import { useEffect, useState } from "react";
2  import { useParams } from "react-router-dom";
3  import { getCarDetail } from "../service/car-service";
4
5  import Car from "../components/Cars/Car";
6  import Layout from "../components/Layout/Layout";
7
8  const CarDetail = () => {
9    const [car, setCar] = useState({
10      make: "",
11      model: "",
12      year: "",
13      price: "",
14    });
15
16    const params = useParams();
17
18    >  useEffect(() => { ...
36    }, [params.carId]);
37
38    return (
39      <Layout>
40        <Car {...car} />
41      </Layout>
42    );
43  };
44
45  export default CarDetail;
```

6. Implement React Router to configure routing links.

```
import Home from "../pages/Home";
import { Navigate, Route, Routes } from "react-router-dom";
import CarDetail from "../pages/CarDetail";
import Login from "../pages/Login";
import Register from "../pages/Register";
import PostCar from "../pages/PostCar";
import Contact from "../pages/Contact";

import "../App.css";
import "bootstrap/dist/css/bootstrap.min.css";
import { useContext } from "react";
import AuthContext from "../context/auth-context";

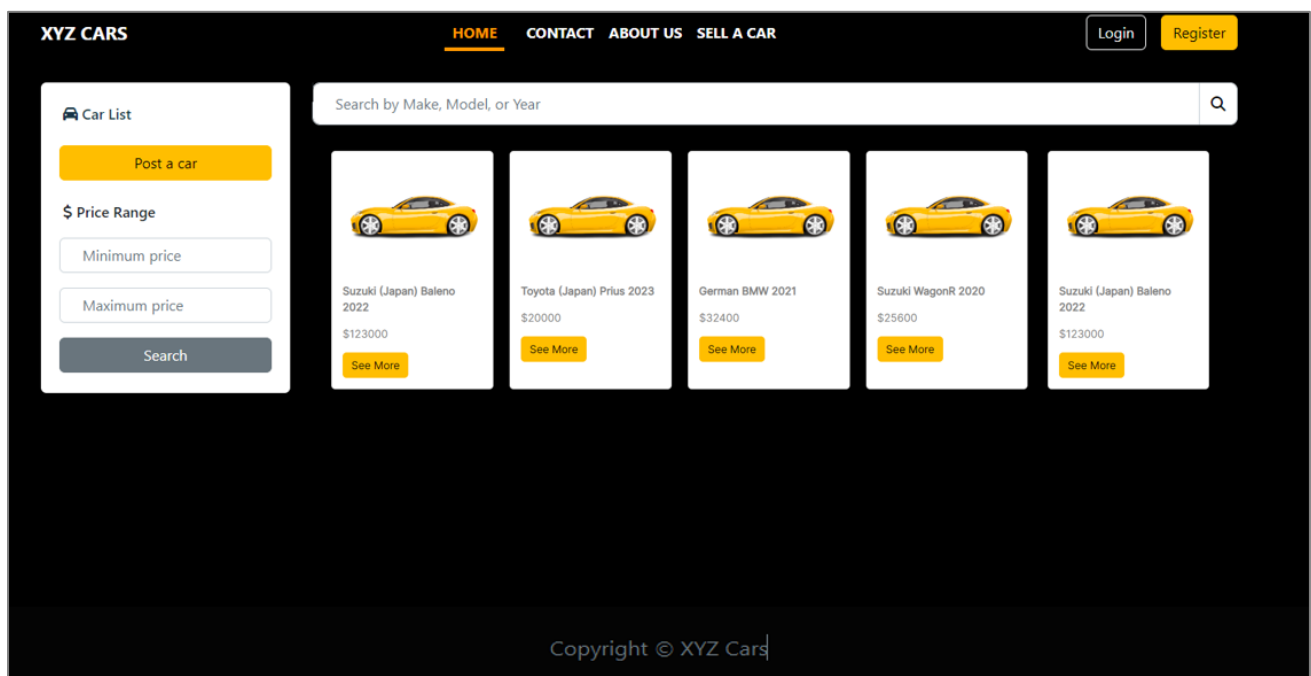
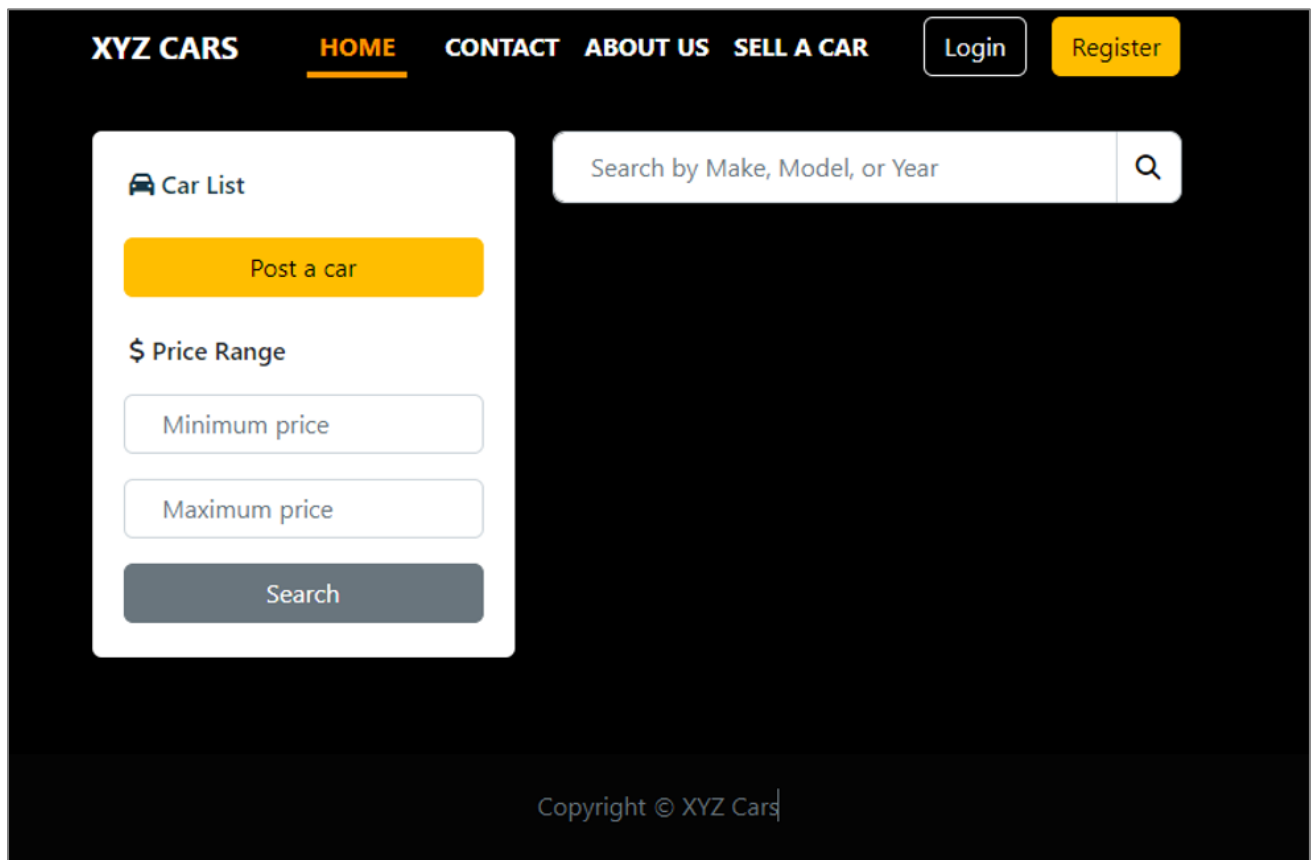
function App() {
  const authCtx = useContext(AuthContext);

  return (
    <Routes>
      <Route path="/" element={<Navigate to="/cars" replace />} />
      <Route path="/cars" element={<Home />} />
      <Route path="/contact" element={<Contact />} />
      <Route path="/cars/:make/:model/:year/:carId" element={<CarDetail />} />
      {!authCtx.isLoggedIn && (
        <>
          <Route path="/login" element={<Login />} />
          <Route path="/register" element={<Register />} />
        </>
      )}
      <Route
        path="/post-car"
        element={authCtx.isLoggedIn ? <PostCar /> : <Navigate to="/login" />}
      />
    </Routes>
  );
}

export default App;
```

7. Screen Captures of Pages

Initial Page



Login

XYZ CARS **HOME** **CONTACT** **ABOUT US** **SELL A CAR**

Login

Register

Login

Login

[Create an account](#)

Copyright © XYZ Cars

Post cars

XYZ CARS **HOME** **CONTACT** **ABOUT US** **SELL A CAR**

Login

Register

Post Car

Make

Model

Year

Price

Post Car

Copyright © XYZ Cars

View Car

XYZ CARS

[HOME](#)
[CONTACT](#)
[ABOUT US](#)
[SELL A CAR](#)

Login
Register

Car List

Post a car

\$ Price Range

Minimum price

Maximum price

Search

Search by Make, Model, or Year

Suzuki (Japan) Baleno 2022

PRICE: \$123000

Buy a car

Copyright © XYZ Cars

Search Car

XYZ CARS

[HOME](#)
[CONTACT](#)
[ABOUT US](#)
[SELL A CAR](#)

Login
Register

Car List

Post a car

\$ Price Range

Minimum price

Maximum price

Search

Baleon

Suzuki (Japan) Baleno 2022
\$123000
See More

Suzuki (Japan) Baleno 2022
\$123000
See More



Assignment 2.zip