Assignment 2 **Chathushi Jayarathna**

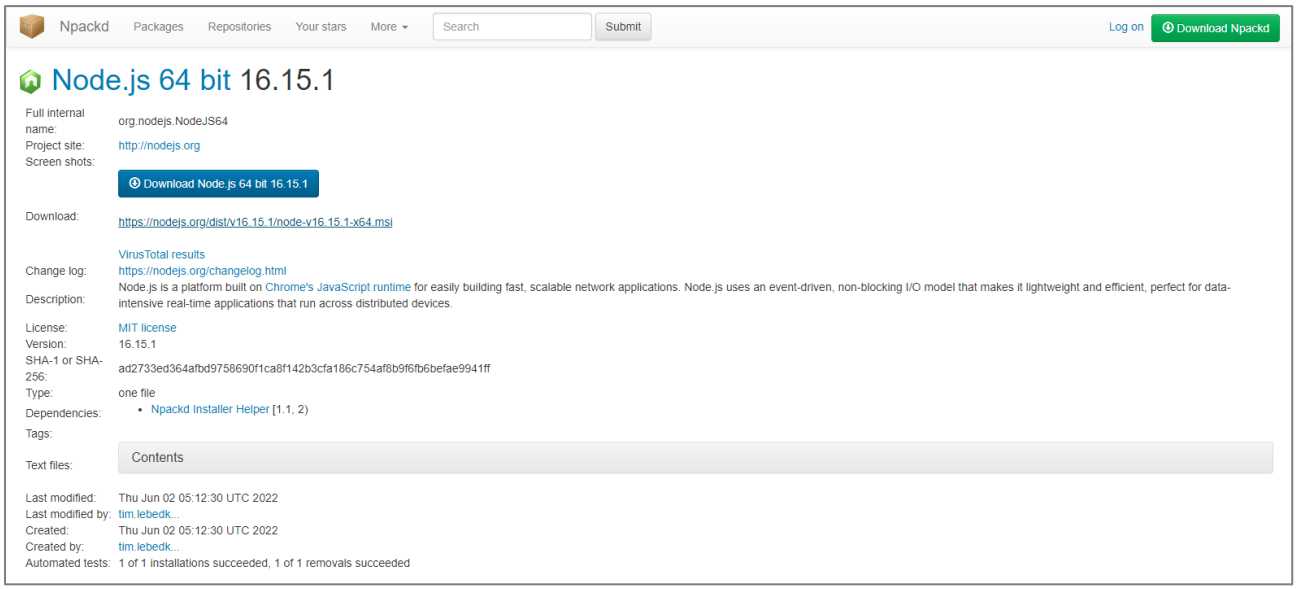
Module No:	1	IU No:	2	Exercise No.	2

Lab Assessment Statement	**Assignment 2 - Build React JS Frontend Application** **Referring to Assignment-1's Project Scenario,** 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. These are the steps provided to build an API. 1. Install Node JS. 2. Create React Application using **npx creact-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. **Provide the source code of developed application.** **Provide screen capture of final result pages for API testing.**
Technical Environment	- REST API, React JS
Guidelines	-
Duration	120 mins

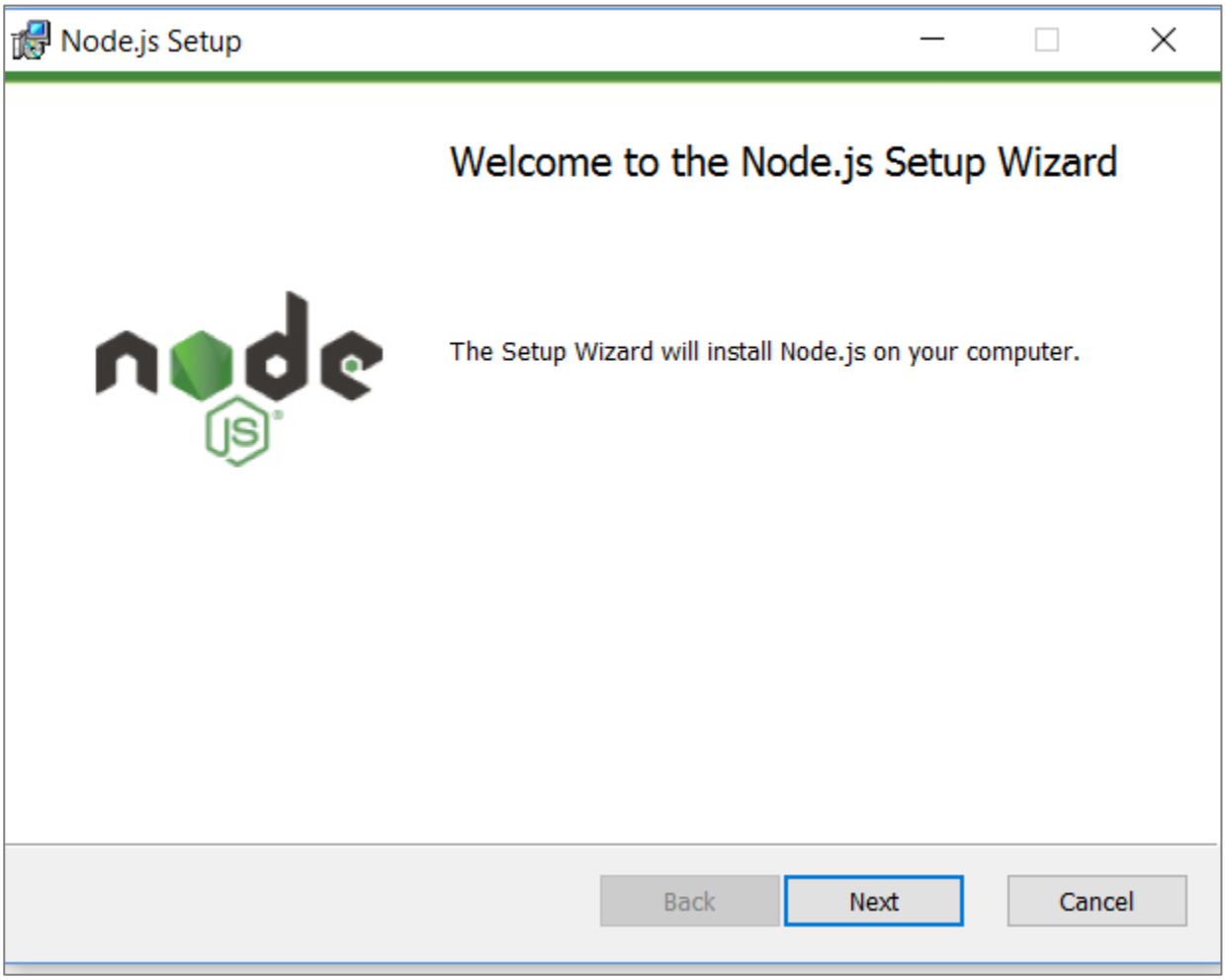
Assignment 2 **Chathushi Jayarathna**

1. Install Node JS.

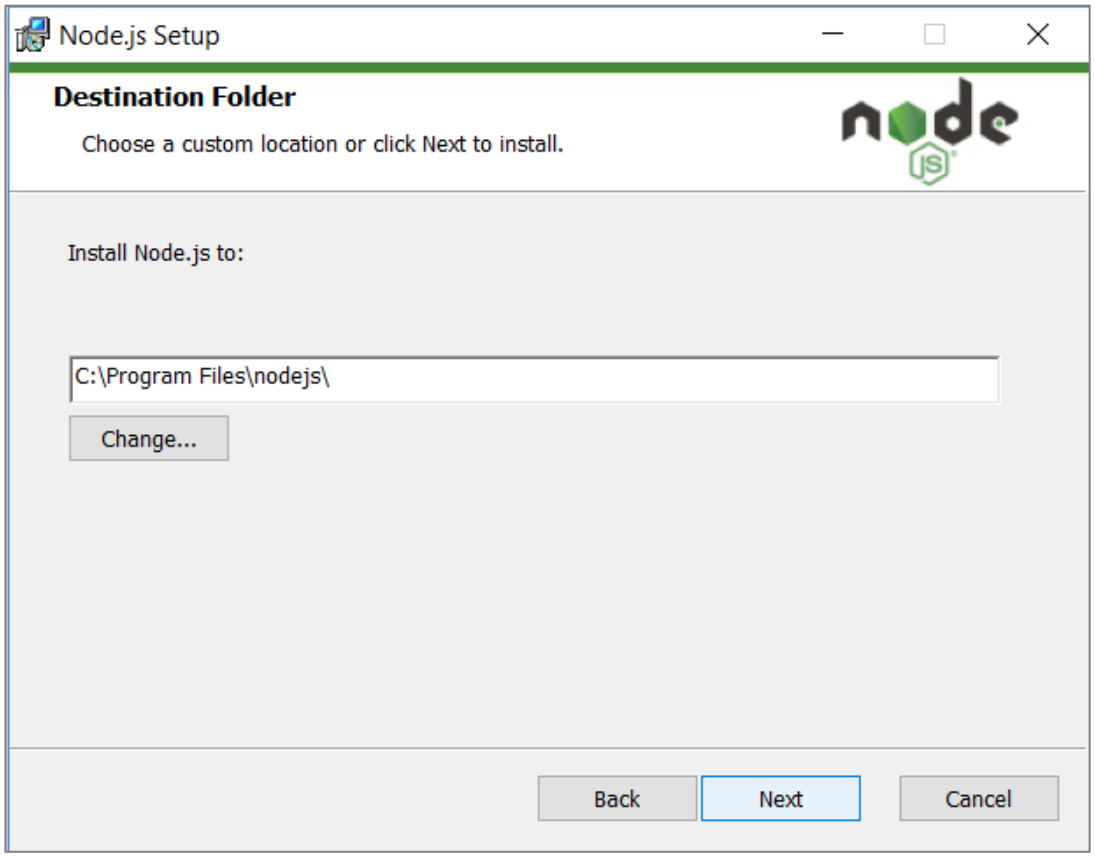
Step 1: Download Node JS

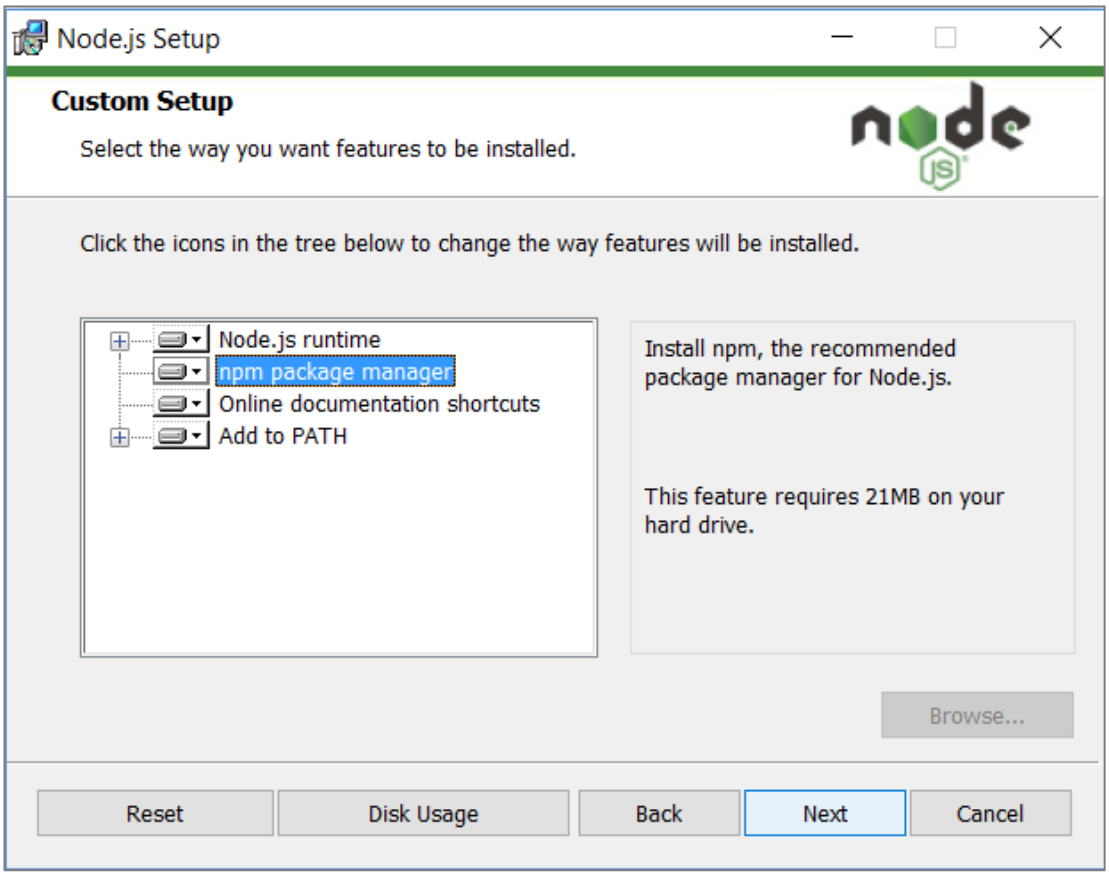


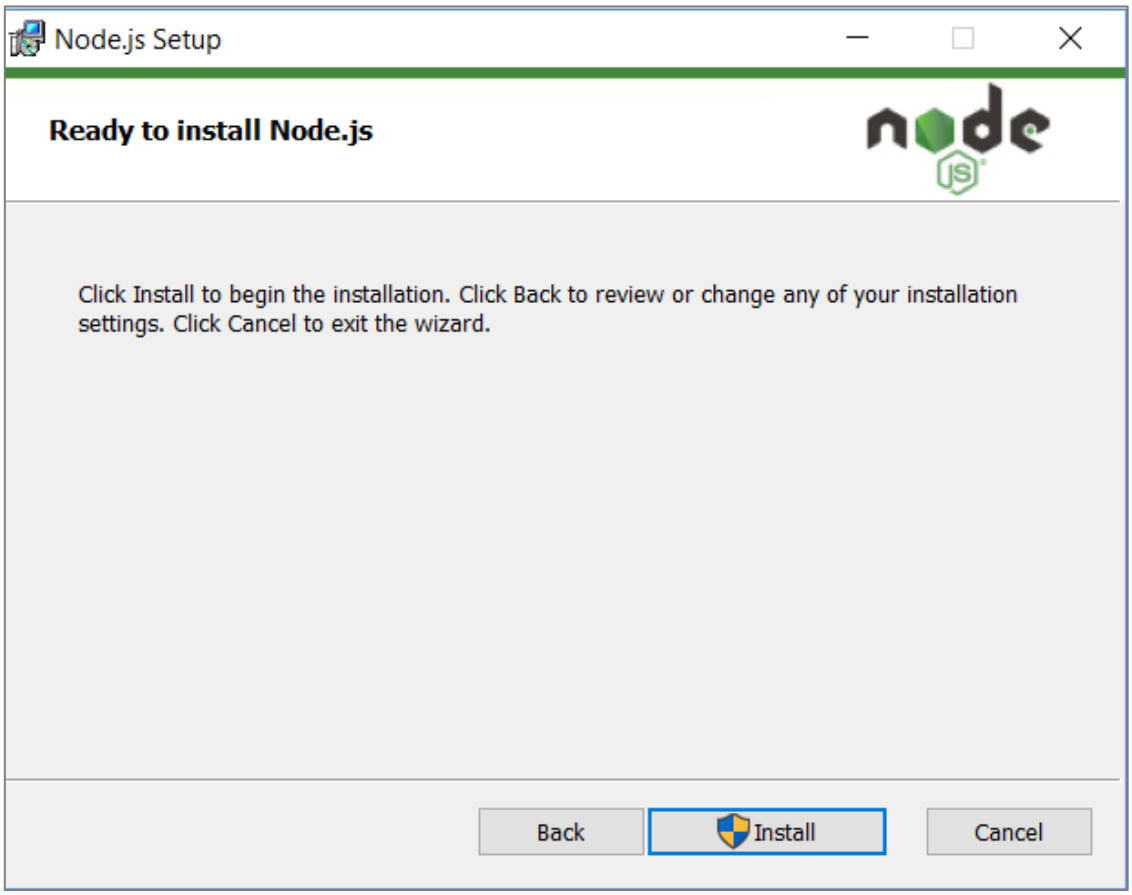
Step 2: Install NodeJS and npm



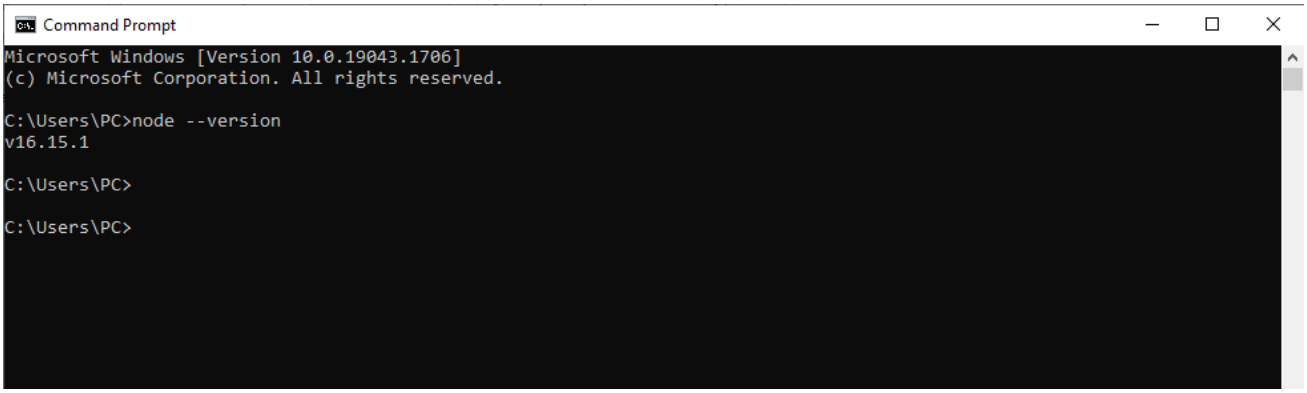
Step 3: Choose the desired path where to install NodeJS







Step 4: Check NodeJS and npm version



2. Create React Application using **npx creact-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-r
eact> 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 = "";
  };
```

Registeration.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

```
// GET List Car
export function getListCar(callback, errorCallback) {
  axios
    .get("http://localhost:8080/api/cars")
    .then((res) => {
      if (callback != null) {
        callback(res.data);
      }
    })
    .catch((err) => {
      if (errorCallback != null) {
        errorCallback(err.message);
      }
    });
}
```

Car-Service.JS – Get Car Details

```
// GET Car Detail
export function getCarDetail(idCar, callback, errorCallback) {
  axios
    .get("http://localhost:8080/api/cars/" + idCar)
    .then((res) => {
      if (callback != null) {
        callback(res.data);
      }
    })
    .catch((err) => {
      if (errorCallback != null) {
        errorCallback(err.message);
      }
    });
}
```

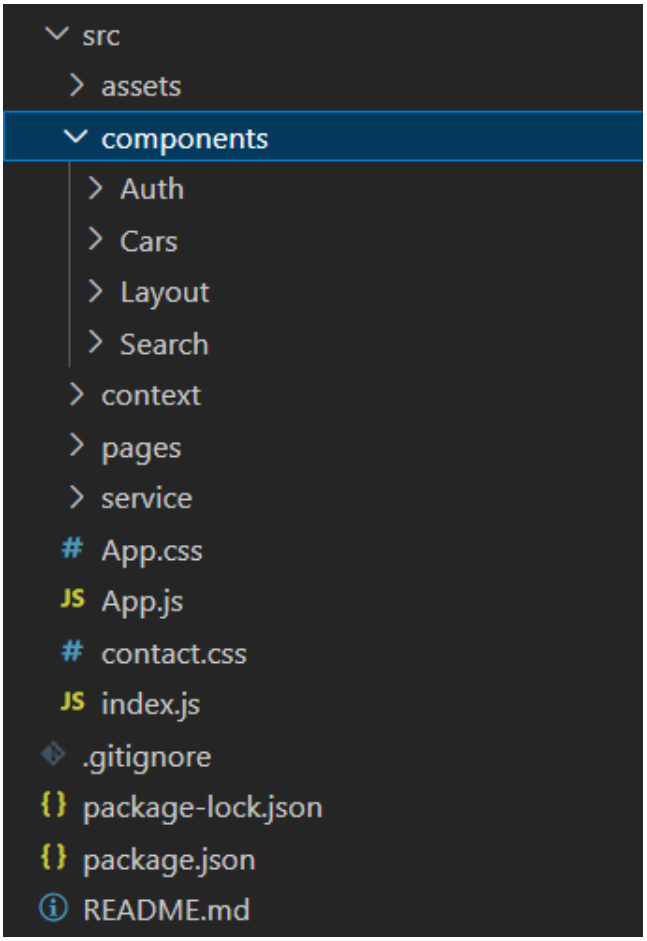
Car-Service.JS – Search Car Details

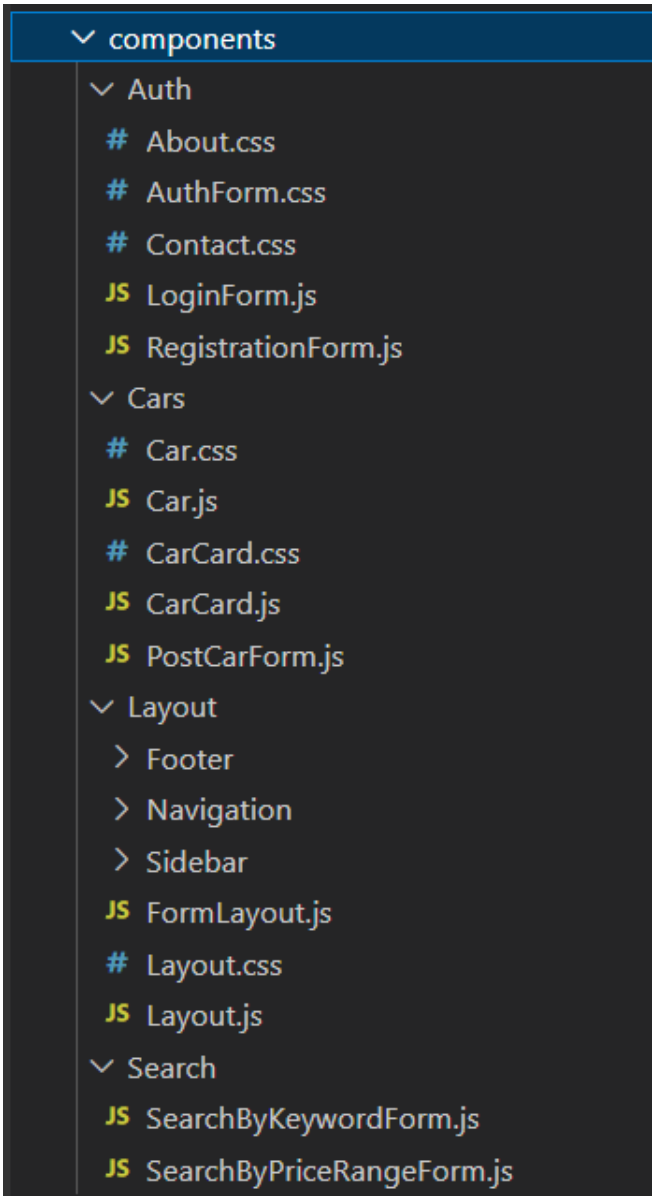
```
35  // GET Search By Keyword
36  export function getSearchByKeyword(keyword, callback, errorCallback) {
37    axios
38      .get(`http://localhost:8080/api/cars?keyword=${keyword}`)
39      .then((res) => {
40        if (callback != null) {
41          callback(res.data);
42        }
43      })
44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

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")}, M
          {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

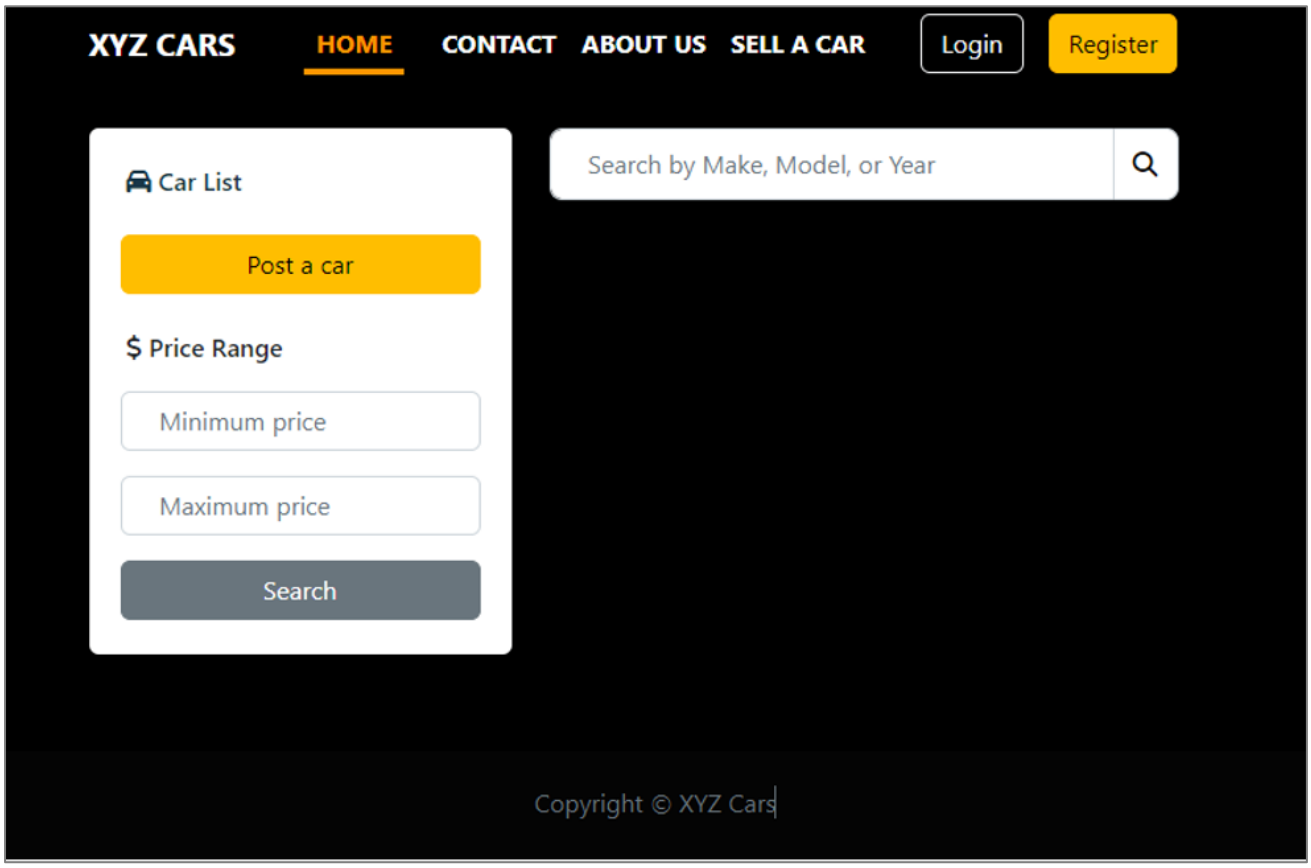
```
1  import { useEffect, useState } from "react";
2  import { useParams } from "react-router-dom";
3  import { getCarDetail } from "../service/car-service";
4
5  import Car from "../components/Cars/Car";
6  import Layout from "../components/Layout/Layout";
7
8  const CarDetail = () => {
9    const [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;
```

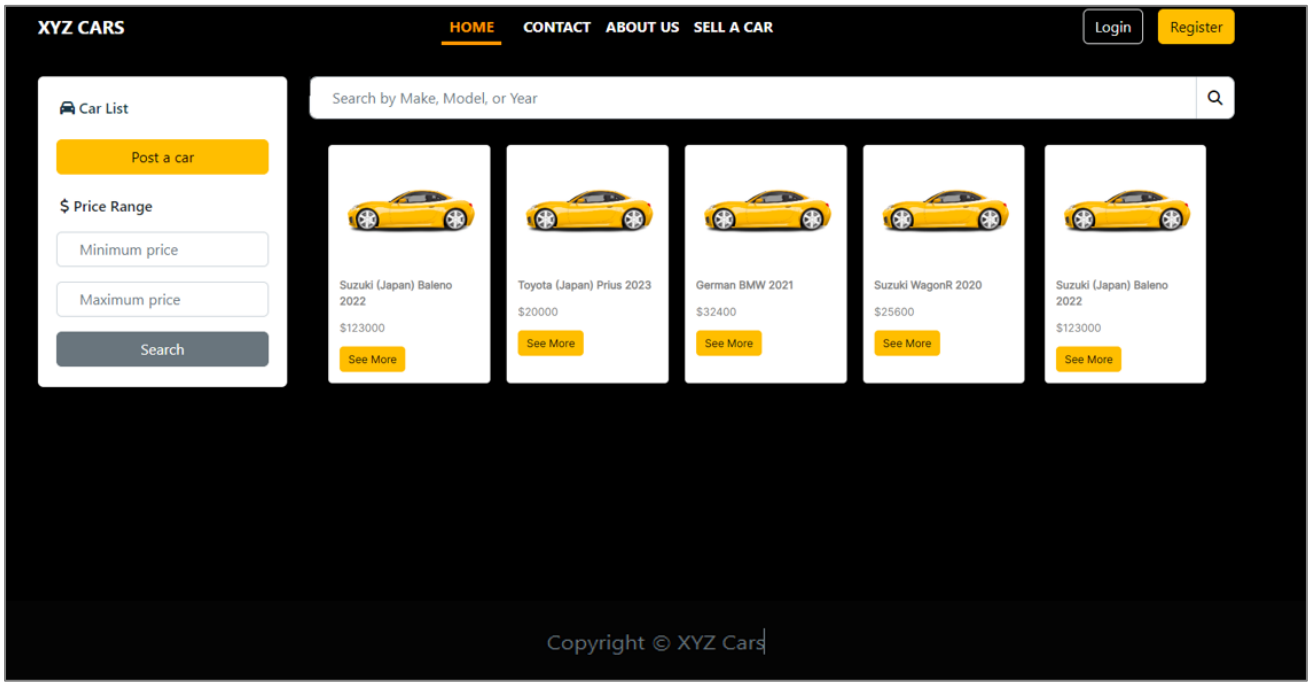
6. Implement React Router to configure routing links.

```jsx
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

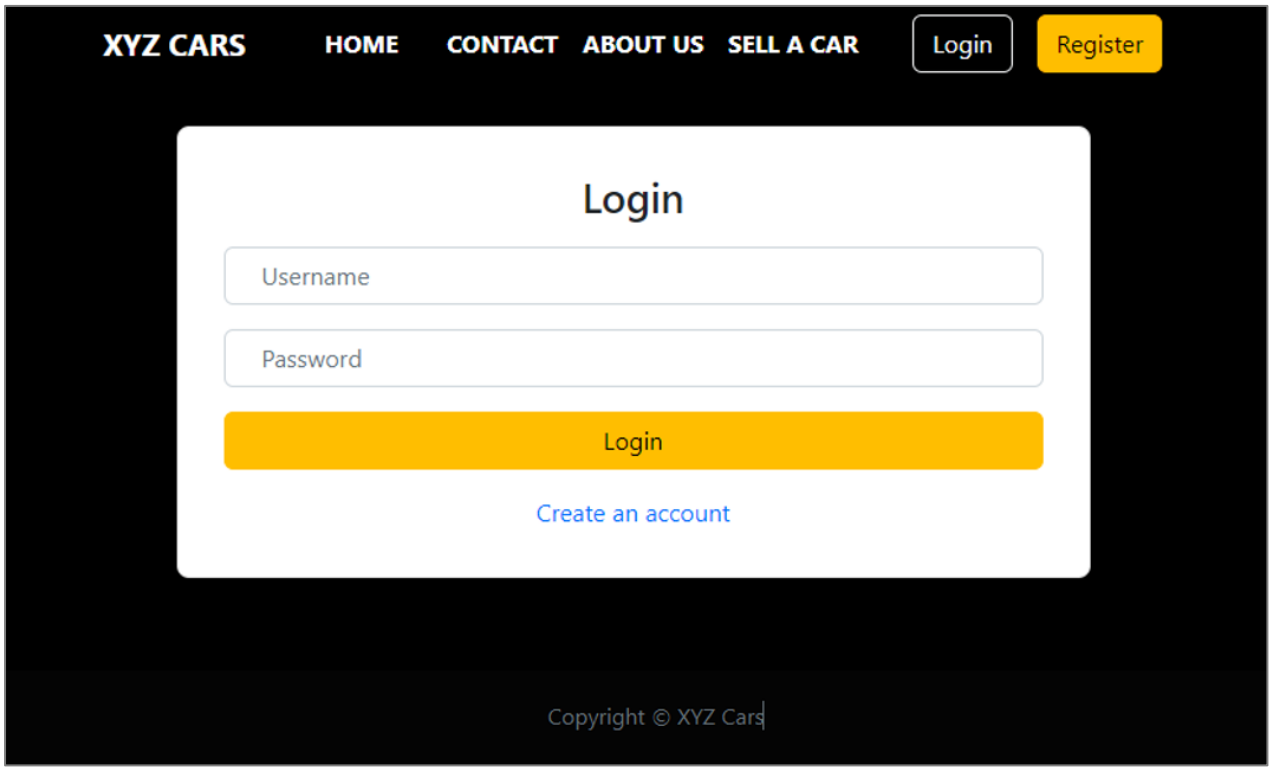
Initial Page



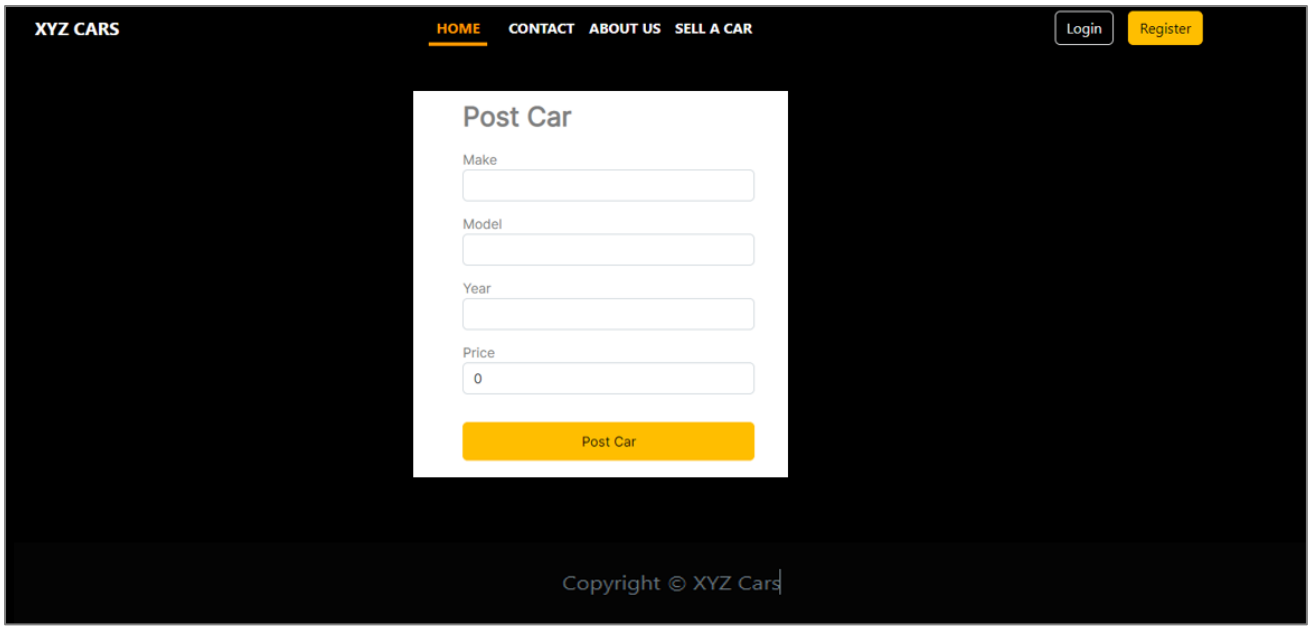


Assignment 2 **Chathushi Jayarathna**

Login

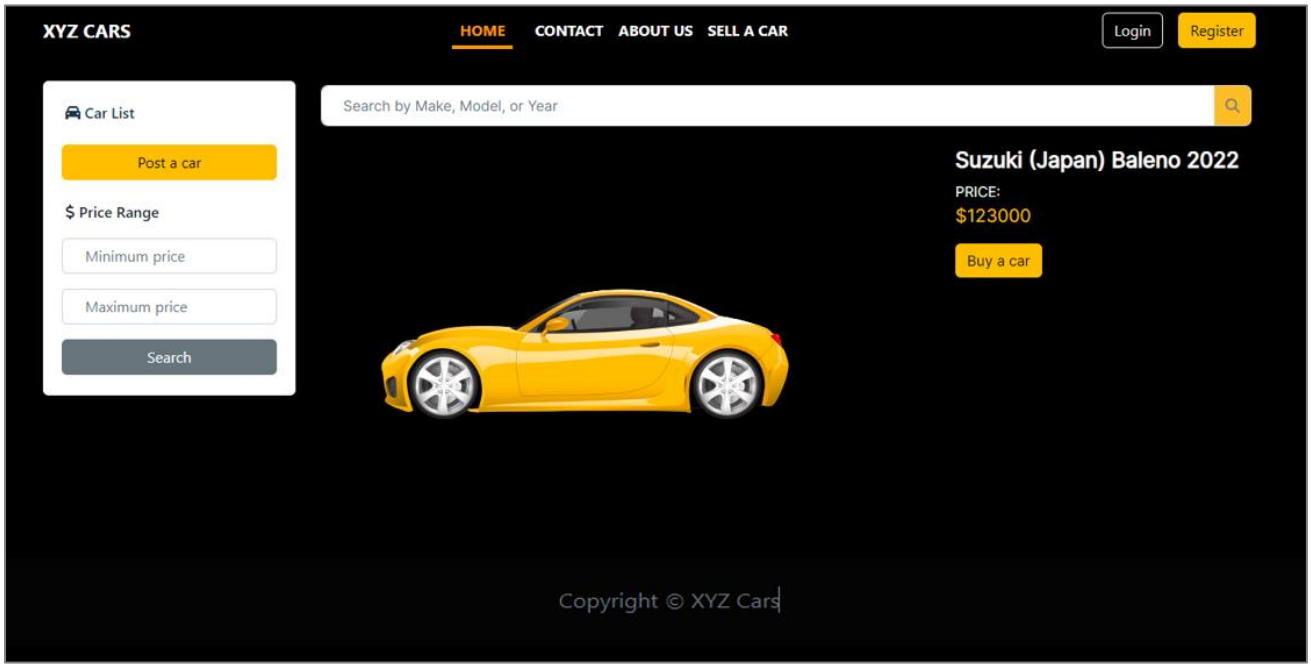


Post cars

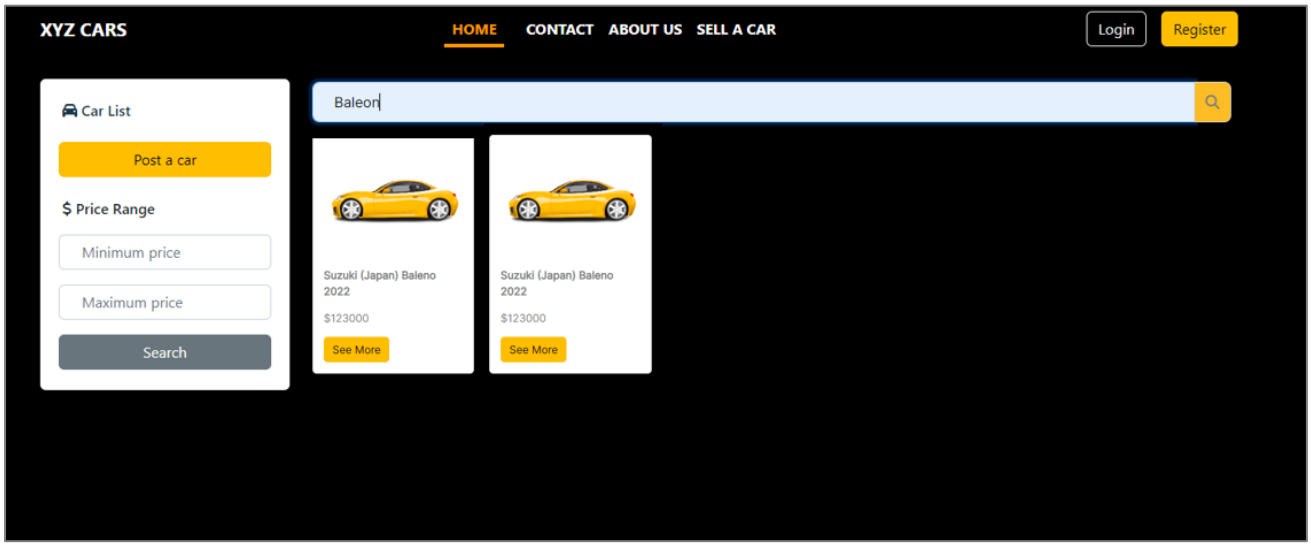


Assignment 2 **Chathushi Jayarathna**

View Car



Search Car



Assignment 2.zip