ASSIGNMENT 04

Build an Authentication System with React

Prepared By: ARM. Afrih

Student Id: TED/240063R

Contents

Table of Figure

1. Login Page

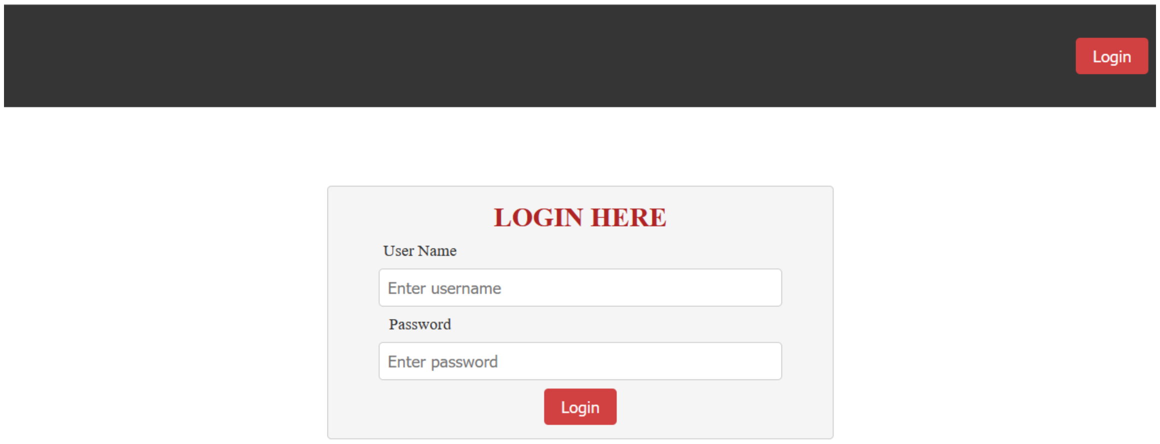
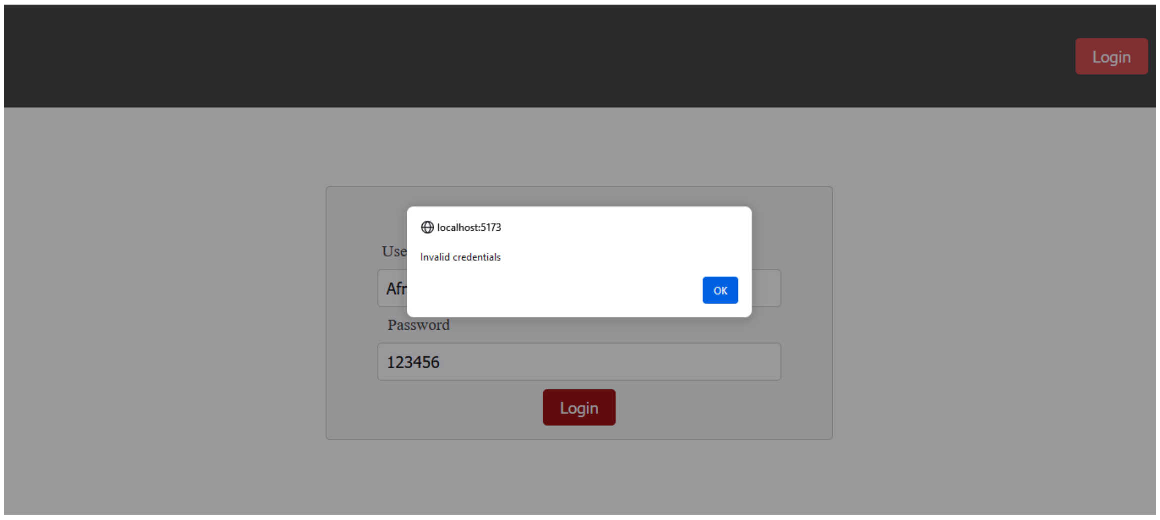


Figure 1Login Page

2. Login Invalid Message

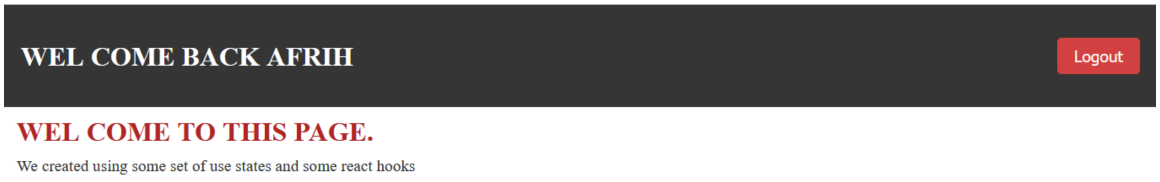


Invalid credentials

Figure 2 Login Invalid Message

.

3. Code - Home


WEL COME BACK AFRIH

Logout

WEL COME TO THIS PAGE.

We created using some set of use states and some react hooks

Figure 3 Home Page

4. Authentication code

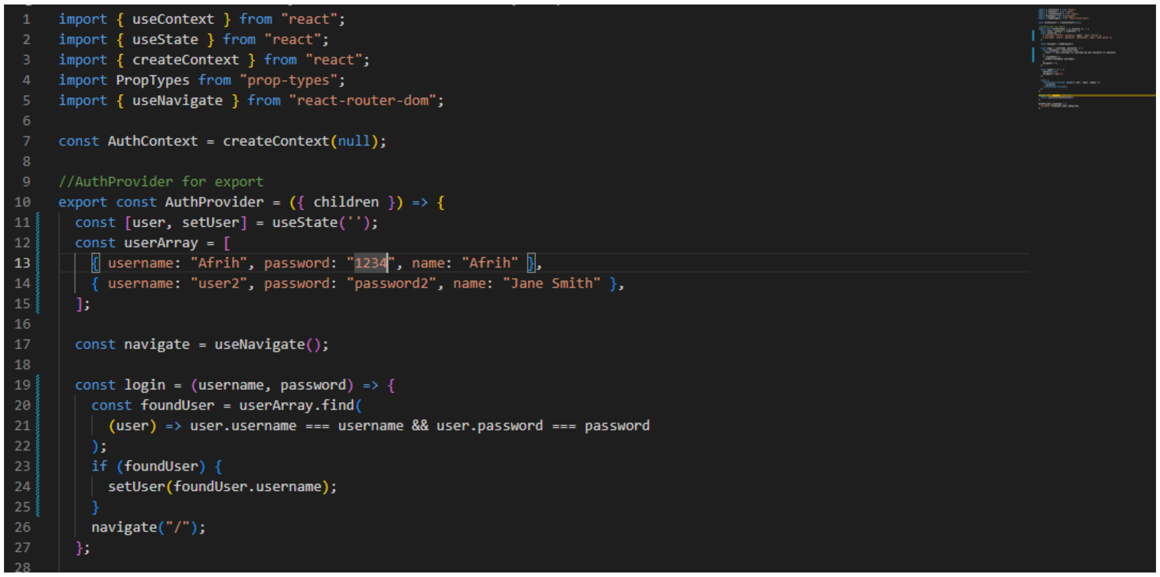

```
import { useContext } from "react";
import { useState } from "react";
import { createContext } from "react";
import PropTypes from "prop-types";
import { useNavigate } from "react-router-dom";

const AuthContext = createContext(null);

//AuthProvider for export
export const AuthProvider = ({ children }) => {
  const [user, setUser] = useState('');
  const userArray = [
    { username: "Afrih", password: "1234", name: "Afrih" },
    { username: "user2", password: "password2", name: "Jane Smith" },
  ];

  const navigate = useNavigate();

  const login = (username, password) => {
    const foundUser = userArray.find(
      (user) => user.username === username && user.password === password
    );
    if (foundUser) {
      setUser(foundUser.username);
    }
    navigate("/");
  };
```

Figure 4Authentication code 1

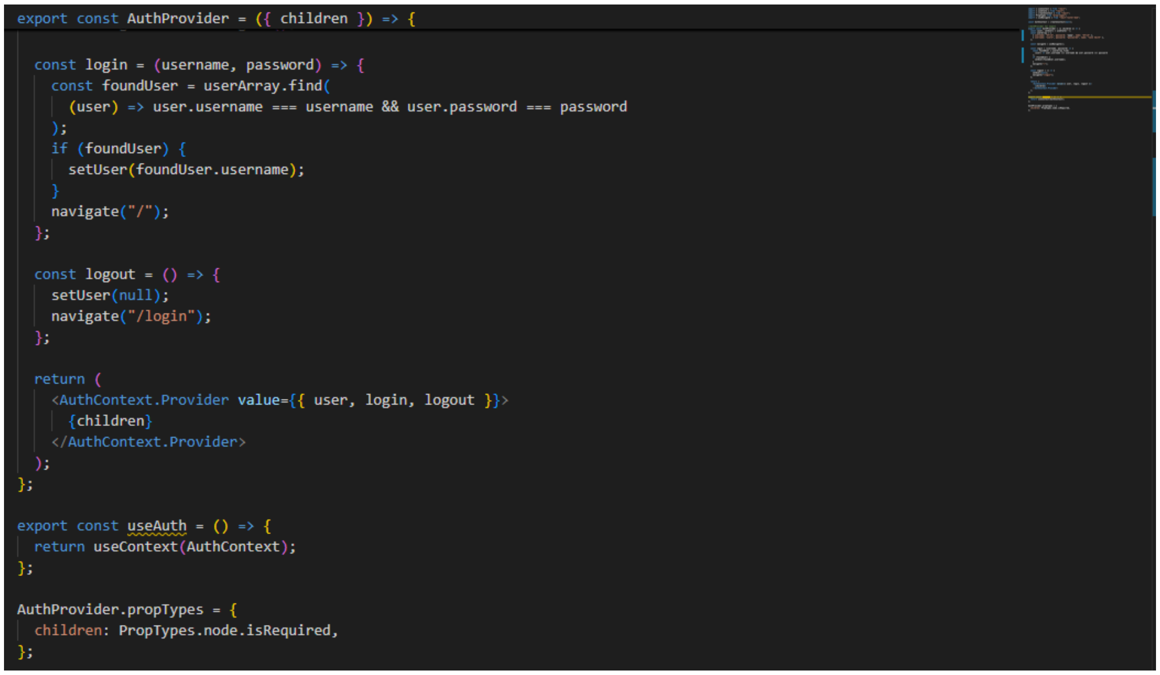

```
export const AuthProvider = ({ children }) => {

  const login = (username, password) => {
    const foundUser = userArray.find(
      (user) => user.username === username && user.password === password
    );
    if (foundUser) {
      setUser(foundUser.username);
    }
    navigate("/");
  };

  const logout = () => {
    setUser(null);
    navigate("/login");
  };

  return (
    <AuthContext.Provider value={{ user, login, logout }}>
      {children}
    </AuthContext.Provider>
  );
};

export const useAuth = () => {
  return useContext(AuthContext);
};

AuthProvider.propTypes = {
  children: PropTypes.node.isRequired,
};
```

Figure 5 Authentication code 2

5. Protection Route code

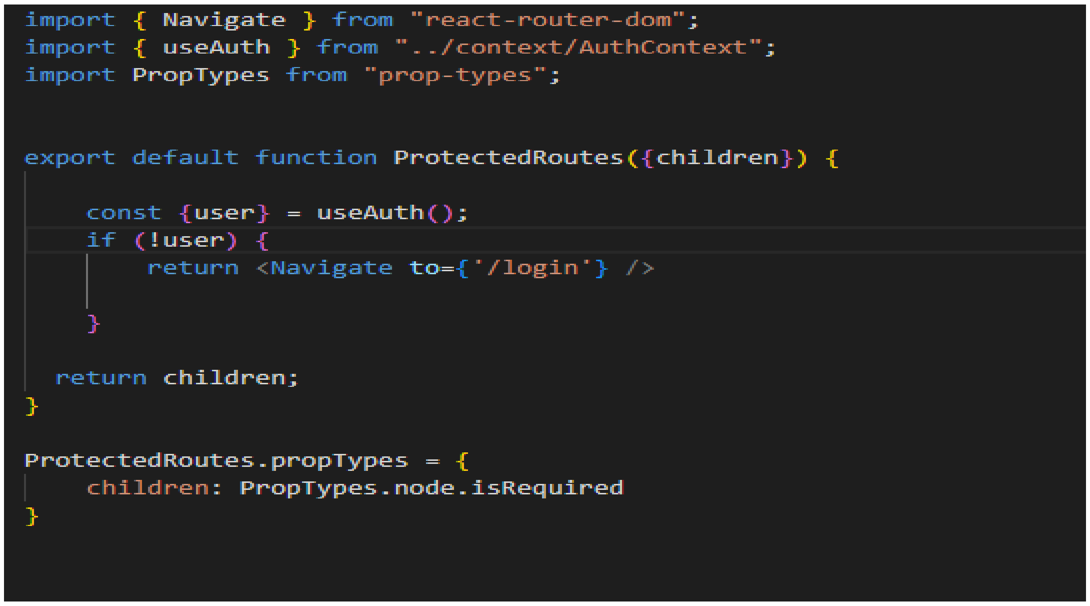
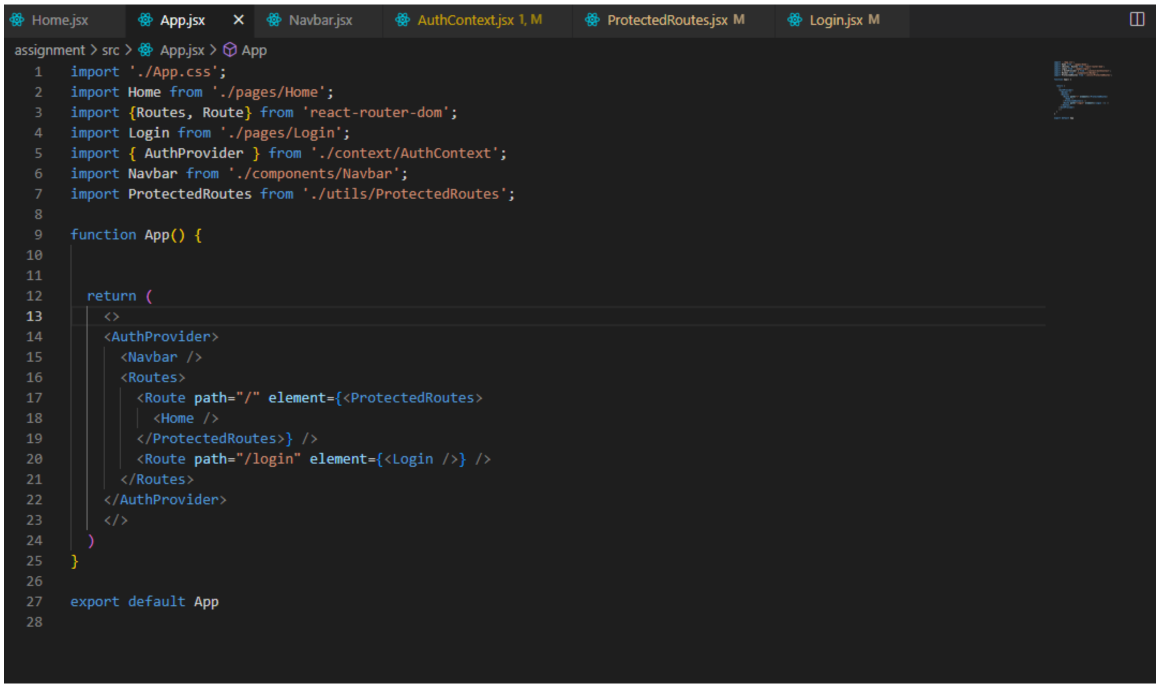

```
import { Navigate } from "react-router-dom";
import { useAuth } from "../context/AuthContext";
import PropTypes from "prop-types";


export default function ProtectedRoutes({children}) {

    const {user} = useAuth();
    if (!user) {
        return <Navigate to={'/login'} />

    }

  return children;
}

ProtectedRoutes.propTypes = {
    children: PropTypes.node.isRequired
}
```

Figure 6 Protection Route

6. App.jsx code



```jsx
import './App.css';
import Home from './pages/Home';
import {Routes, Route} from 'react-router-dom';
import Login from './pages/Login';
import { AuthProvider } from './context/AuthContext';
import Navbar from './components/Navbar';
import ProtectedRoutes from './utils/ProtectedRoutes';

function App() {


  return (
    <>
    <AuthProvider>
      <Navbar />
      <Routes>
        <Route path="/" element={<ProtectedRoutes>
          <Home />
        </ProtectedRoutes>} />
        <Route path="/login" element={<Login />} />
      </Routes>
    </AuthProvider>
    </>
  )
}

export default App

```

Figure 7 App.jsx