



# DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

## EXPERIMENT- 1

**Student Name:** SHRESHTA

**Semester:** 6<sup>th</sup>

**UID:** 23BAI70375

**Date of Performance:** 21/01/26

**Branch:** BE CSE AIML

**Subject Name:** FULLSTACK-II

**Section/Group:** 23AIT\_KRG-G1

**Subject Code:** 23CSH-382

### AIM:

To implement Single Page Application (SPA) navigation in the EcoTrack application using React Router, secure application routes using context-based authentication, and extend nested dashboard routing through follow-up enhancements.

### OBJECTIVES:

After completing this experiment and its follow-up tasks, the student will be able to:

1. Configure client-side routing in a React application using React Router
2. Implement SPA navigation without full page reloads
3. Design and apply protected routes using route-guard patterns
4. Manage shared authentication state using React Context API
5. Implement nested routing to build dashboard-style layouts
6. Extend existing nested routes by adding new dashboard sections
7. Implement logout functionality by updating shared context state
8. Analyze route access behavior and explain redirection logic
9. Understand the role of Context API in shared state management and its comparison with Redux at an introductory level



# DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

## IMPLEMENTATION/CODE:

### Logs.js

```
1 | export const logs = [  
2 |   { id: 1, activity: "Car Travel", carbon: 4 },  
3 |   { id: 2, activity: "Electricity Usage", carbon: 6 },  
4 |   { id: 3, activity: "Cycling", carbon: 0 },  
5 | ];
```

### Logs.jsx

```
0 | import { useEffect } from "react";  
1 | import { useNavigate } from "react-router-dom";  
2 | import { useAuth } from "../context/auth";  
3 |  
4 | export default function Logout() {  
5 |   const navigate = useNavigate();  
6 |   const { setIsAuthenticated } = useAuth();  
7 |  
8 |   useEffect(() => {  
9 |     setIsAuthenticated(false);  
10 |    navigate("/login");  
11 |  }, []);  
12 |  
13 |   return <h2>Logging out...</h2>;  
14 | }
```

### Login.jsx

```
0 | import { useNavigate } from "react-router-dom";  
1 | import { useAuth } from "../context/auth";  
2 |  
3 | export default function Login() {  
4 |   const navigate = useNavigate();  
5 |   const { isAuthenticated, setIsAuthenticated } = useAuth();  
6 |  
7 |   function handleLogin() {  
8 |     setIsAuthenticated(true);  
9 |     navigate("/");  
10 |   }  
11 |  
12 |   if (isAuthenticated) {  
13 |     return (  
14 |       <>  
15 |         <h1>You are already logged in</h1>  
16 |         <button onClick={() => navigate("/")}>Go to Dashboard</button>  
17 |       </>  
18 |     );  
19 |   }  
20 |  
21 |   return (  
22 |     <>  
23 |       <h1>Login</h1>  
24 |       <button onClick={handleLogin}>Login Now</button>  
25 |     </>  
26 |   );  
27 | }
```



# DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

## Logout.jsx

```
0 | import { useEffect } from "react";
1 | import { useNavigate } from "react-router-dom";
2 | import { useAuth } from "../context/auth";
3 |
4 | export default function Logout() {
5 |   const navigate = useNavigate();
6 |   const { setIsAuthenticated } = useAuth();
7 |
8 |   useEffect(() => {
9 |     setIsAuthenticated(false);
10 |    navigate("/login");
11 |  }, []);
12 |
13 |   return <h2>Logging out...</h2>;
14 | }
```

## App.jsx

```
0 | import Header from '../components/Header';
1 | import { BrowserRouter, Routes, Route } from 'react-router-dom';
2 | import Logs from './pages/Logs';
3 | import Login from './pages/Login';
4 | import Logout from './pages/Logout';
5 | import Dashboard_layout from './pages/Dashboard_layout';
6 | import DashboardAnalytics from './pages/DashboardAnalytics';
7 | import DashboardSummary from './pages/DashboardSummary';
8 | import AuthProtectedRoutes from './Routes/AuthProtectedRoutes';
9 |
10 | export default function App() {
11 |   return (
12 |     <BrowserRouter>
13 |       <Header />
14 |
15 |       <Routes>
16 |         <Route path="/login" element={<Login />} />
17 |
18 |         <Route
19 |           path="/logout"
20 |           element={
21 |             <AuthProtectedRoutes>
22 |               <Logout />
23 |             </AuthProtectedRoutes>
24 |           }
25 |         />
26 |
27 |         <Route
28 |           path="/"
29 |           element={
30 |             <AuthProtectedRoutes>
31 |               <Dashboard_layout />
32 |             </AuthProtectedRoutes>
33 |           }
34 |         >
35 |           <Route index element={<DashboardSummary />} />
36 |           <Route path="summary" element={<DashboardSummary />} />
37 |           <Route path="analytics" element={<DashboardAnalytics />} />
38 |         </Route>
```



# DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

```
39 |  
40 |     <Route  
41 |       path="/logs"  
42 |       element={  
43 |         <AuthProtectedRoutes>  
44 |           <Logs />  
45 |         </AuthProtectedRoutes>  
46 |       }  
47 |     />  
48 |   </Routes>  
49 | </BrowserRouter>  
50 |   );  
51 | }  
52 |
```

## AuthProtectedRoute.jsx

```
0 | import { Navigate } from "react-router-dom";  
1 | import { useAuth } from "../context/auth";  
2 |  
3 | export default function AuthProtectedRoutes({ children }) {  
4 |   const { isAuthenticated } = useAuth();  
5 |  
6 |   if (!isAuthenticated) {  
7 |     return <Navigate to="/login" replace />;  
8 |   }  
9 |  
10 |   return children;  
11 | }
```

## OUTPUT:

### 1)Login



### Login

Login Now

### 2)DASHBOARD SUMMARY



### DASHBOARD LAYOUT

Dashboard Summary



# DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

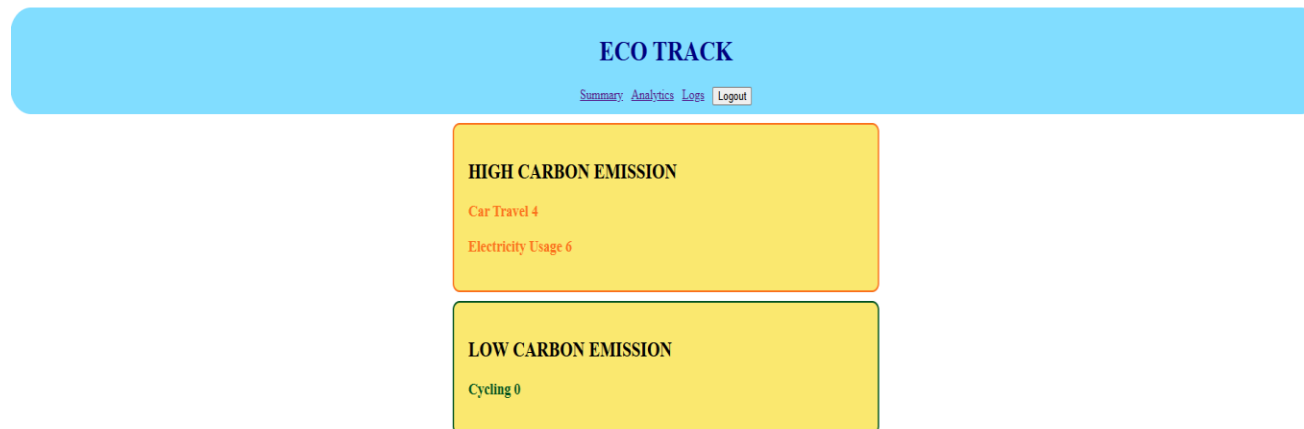
## 3) DASHBOARD ANALYTICS



## DASHBOARD LAYOUT

Dashboard Analytics

## 4) LOGS



## LEARNING OUTCOMES:

1. Understood React authentication flow using login, logout, and protected routes.
2. Implemented Context API to manage global authentication state.
3. Applied route protection using React Router's guarded routes.
4. Handled programmatic navigation with useNavigate() for secure redirects.
5. Debugged common React errors including blank screens and ES module export issues.