



# DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

CHANDIGARH  
UNIVERSITY

Discover. Learn. Empower.

## Experiment 2

**Student Name:** Ashutosh Yadav

**UID:** 23BCS11023

**Branch:** CSE

**Section/Group:** KRG-3-A

**Semester:** 6th

**Date of Performance:** 20/01/2026

**Subject Name:** Full Stack Development – II

**Subject Code:** 23CSH-309

**1. Aim:** To implement Single Page Application (SPA) routing in the EcoTrack application using React Router, secure application routes using protected routing with Context API-based authentication, and manage shared authentication state across components.

### **2. Objective:**

- To configure client-side routing using React Router
- To implement SPA navigation without page reloads
- To protect routes using authentication-based route guards
- To manage shared authentication state using React Context API
- To implement login and logout functionality
- To restrict unauthorized access to protected pages
- To understand redirection logic in protected routing
- To analyze the role of Context API in state management

### **3. Implementation / Code:**

#### **Implementation Description:**

- The EcoTrack application is enhanced by implementing client-side routing using React Router, enabling seamless navigation between different pages without full page reloads.
- An authentication system is implemented using React Context API, which stores and manages the authentication state (isAuthenticated) across the entire application.
- A ProtectedRoute component is created to restrict access to sensitive pages such as Dashboard, Logs, and Data. If the user is not authenticated, they are automatically redirected to the Login page.
- Login functionality updates the authentication state using context, while logout functionality resets the authentication state and redirects the user back to the login page.



# DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

CHANDIGARH  
UNIVERSITY

Discover. Learn. Empower.

- This approach ensures secure navigation, centralized state management, and a smooth SPA user experience.

## Sample Code Snippet:

```
src > App.jsx > App
  4   import { BrowserRouter, Route, Routes } from 'react-router-dom'
  5   import ProtectedRoute from './routes/ProtectedRoute'
  6   import DashboardSummary from './pages/DashboardSummary'
  7   import DashboardAnalytics from './pages/DashboardAnalytics'
  8   import Login from './pages/Login'
  9   import Logs from './pages/Logs'
10
11  function App() {
12
13    return (
14      <main>
15      <>
16        <BrowserRouter>
17        <Header title={"EcoTrack"} />
18
19        <Routes>
20
21          <Route path="/login" element={<Login/>} />
22
23          <Route path="/" element={
24            <ProtectedRoute>
25              <Dashboard/>
26            </ProtectedRoute>
27          }>
28
29
30          <Route index element={<DashboardSummary/>} />
31          <Route path="summary" element={<DashboardSummary/>} />
32          <Route path="analytics" element={<DashboardAnalytics/>} />
33
34        </Route>
35        <Route path="logs" element={<Logs/>} />
36
37      </Routes>
38
39    </BrowserRouter>
40    </>
41    </main>
42
43  )
44
45 }
```



# DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

CHANDIGARH  
UNIVERSITY

Discover. Learn. Empower.

```
src > context > [js] AuthContext.jsx > [o] AuthProvider
1  import {createContext,useState,useContext} from "react";
2
3  const AuthContext = createContext(null);
4
5
6  export const AuthProvider= ({children}) =>[
7
8      const [isAuthenticated,setIsAuthenticated] = useState(false);
9
10
11     return (
12
13         <AuthContext.Provider value={{isAuthenticated,setIsAuthenticated}}>
14             {children}
15         </AuthContext.Provider>
16
17     )
18
19
20 ]
21
22
23
24  export const useAuth = () => useContext(AuthContext);
```

```
src > routes > [jsx] ProtectedRoute.jsx > [o] ProtectedRoute
1  import { Navigate, replace } from "react-router-dom";
2  import { useAuth } from "../context/AuthContext";
3
4  const ProtectedRoute = ({children} ) =>[
5
6      const {isAuthenticated} = useAuth();
7
8      if(!isAuthenticated) {
9          return <Navigate to="/login" replace />;
10     }
11
12
13     return children;
14
15 ]
16
17  export default ProtectedRoute;
```



# DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

CHANDIGARH  
UNIVERSITY

Discover. Learn. Empower.

```
src > components > Header.jsx > Header
  1
  2  import { Link } from "react-router-dom";
  3  import { useAuth } from "../context/AuthContext";
  4  const Header = ({title}) =>{
  5
  6    const {isAuthenticated, setIsAuthenticated} = useAuth()
  7    return [
  8      <header style={{padding:"0.5rem",backgroundColor:"rosybrown",position:"sticky",top:"0",zIndex:1}}>
  9        <h1>{title}</h1>
 10
 11        <nav>
 12
 13          <Link to="/" style={{marginRight:"1rem",color:"white"}}>
 14            Dashboard
 15          </Link>
 16          <Link to="/logs" style={{marginRight:"1rem",color:"white"}}>
 17            Logs
 18          </Link>
 19          {isAuthenticated ? <button onClick={()=>setIsAuthenticated(false)}> logout </button> :
 20            <Link to="/login" style={{marginRight:"1rem",color:"white"}}>
 21              Login
 22            </Link>
 23          }
 24
 25        </nav>
 26
 27      </header>
 28    ]
 29  }
 30
 31
 32
 33
 34  export default Header;
```



# DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

CHANDIGARH  
UNIVERSITY

Discover. Learn. Empower.

## 4. Output:

- The EcoTrack application successfully implements SPA routing
- Navigation occurs without full page reloads
- Unauthorized users are redirected to the login page
- Authenticated users can access Dashboard, Logs, and Data pages
- System logs and environmental data are displayed dynamically

The screenshot shows the login interface of the EcoTrack application. At the top, there's a red header bar with the 'EcoTrack' logo. Below it, the main content area has a dark background with a light red header section containing the word 'Login'. Underneath this, there are two buttons: a light blue 'Login' button and a dark blue 'Login to EcoTrack' button.

The screenshot shows the dashboard interface of the EcoTrack application. At the top, there's a red header bar with the 'EcoTrack' logo. Below it, the main content area has a dark background with a light red header section containing the word 'Dashboard'. Underneath this, there are three navigation links: 'Summary' (in blue), 'Analytics' (in blue), and a dark blue 'Logout' button.



# DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

CHANDIGARH UNIVERSITY  
Discover. Learn. Empower.

## 1. Learning Outcomes

- Implement SPA routing using React Router
- Secure application routes using protected routing
- Manage shared authentication state using Context API
- Implement login and logout functionality
- Understand route redirection logic
- Compare Context API with Redux at an introductory level
- Build scalable and secure React applicatio