



## Experiment 2

**Student Name: Raj Kumar Sing**

**Branch: CSE**

**Semester: 6th**

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

**UID: 23BCS11393**

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

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

**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:**

#### **Tools & Technologies Used:**

- AWS Free Tier Account
- Web Browser (Google Chrome / Firefox)
- Amazon EC2 Service
- RDP Client (Microsoft Remote Desktop)
- Internet-enabled Laptop/Desktop

#### **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.

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

## Sample Code Snippet:

### src/Header.jsx

```
import { Link, useNavigate } from "react-router-dom";
import { useAuth } from "../context/AuthContext.jsx";
```

```
const Header = ({ title }) => {
  const { setIsAuthenticated } = useAuth();
  const navigate = useNavigate();
```

```
  const handleLogOut = () => {
    setIsAuthenticated(false);
    navigate("/login");
  };
}
```

```
return (
  <header
    style={{
      backgroundColor: "#0f172a",
      color: "white",
      padding: "0.8rem 1.5rem",
      display: "flex",
      alignItems: "center",
      justifyContent: "space-between",
      boxShadow: "0 2px 8px rgba(0,0,0,0.15)",
    }}
  >
    {/* App Title */}
    <h1
      style={{
        fontSize: "1.4rem",
        fontWeight: "700",
        letterSpacing: "0.5px",
        margin: 0,
      }}
    >
      {title}
    </h1>

    {/* Navigation */}
    <nav style={{ display: "flex", gap: "1.5rem" }} >
      <Link
        to="/"
        style={{
          color: "#cbd5f5",
          textDecoration: "none",
          fontSize: "0.95rem",
        }}
      >
        Dashboard
      </Link>
    </nav>
  </header>
)
```



# DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

</Link>

```
<Link
  to="/log"
  style={{
    color: "#cbd5f5",
    textDecoration: "none",
    fontSize: "0.95rem",
  }}
>
```

```
  Logs
</Link>
```

</nav>

```
{/* Logout */}
<button
  onClick={handleLogOut}
  style={{
    backgroundColor: "#ef4444",
    color: "white",
    border: "none",
    padding: "0.35rem 0.9rem",
    borderRadius: "6px",
    fontSize: "0.85rem",
    fontWeight: "600",
    cursor: "pointer",
  }}
  onMouseOver={(e) =>
    (e.target.style.backgroundColor = "#dc2626")}
  }
  onMouseOut={(e) =>
    (e.target.style.backgroundColor = "#ef4444")}
  }
>
  Logout
</button>
</header>
);
};
```

export default Header;

## src/pages/dashboard.jsx

```
import { logs } from "../data/logs";
```

```
const Dashboard = () => {
  const totalCarbon = logs.reduce((acc, log) => acc + log.carbon, 0);
```

```
  return (
    <div
      style={{
        minHeight: "100vh",
        backgroundColor: "#f0fdf4",
        display: "flex",
```



# DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

```
alignItems: "flex-start",
justifyContent: "center",
padding: "2rem",
}}
>
<div
  style={{
    width: "100%",
    maxWidth: "900px",
  }}
>
  { /* Header Section */ }
  <div
    style={{
      marginBottom: "1.5rem",
    }}
  >
    <h1
      style={{
        fontSize: "2rem",
        fontWeight: "700",
        color: "#065f46",
        marginBottom: "0.4rem",
      }}
    >
      Carbon Dashboard
    </h1>
    <p
      style={{
        color: "#047857",
        fontSize: "0.95rem",
      }}
    >
      Overview of your daily environmental impact 🌍
    </p>
  </div>

  { /* Stats Card */ }
  <div
    style={{
      backgroundColor: "#ffffff",
      borderRadius: "14px",
      padding: "2rem",
      boxShadow: "0 8px 20px rgba(0,0,0,0.08)",
      display: "flex",
      alignItems: "center",
      justifyContent: "space-between",
      flexWrap: "wrap",
    }}
  >
    <div>
      <p
        style={{
          fontSize: "0.9rem",
```



# DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

```
        color: "#6b7280",
        marginBottom: "0.4rem",
    }}
>
    Total Carbon Footprint
</p>

<p
  style={{
    fontSize: "3rem",
    fontWeight: "800",
    color: "#16a34a",
  }}
>
  {totalCarbon}
<span
  style={{
    fontSize: "1.1rem",
    fontWeight: "500",
    color: "#374151",
    marginLeft: "0.5rem",
  }}
>
  kg CO2
</span>
</p>
</div>
```

```
{/* Visual Indicator */}
<div
  style={{
    width: "120px",
    height: "120px",
    borderRadius: "50%",
    background:
      totalCarbon > 15
        ? "#fee2e2"
        : totalCarbon > 8
        ? "#fef3c7"
        : "#dcfce7",
    display: "flex",
    alignItems: "center",
    justifyContent: "center",
    fontSize: "1.2rem",
    fontWeight: "700",
    color:
      totalCarbon > 15
        ? "#dc2626"
        : totalCarbon > 8
        ? "#d97706"
        : "#15803d",
  }}
>
  {totalCarbon > 15
```



# DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

```
? "High"
: totalCarbon > 8
? "Medium"
: "Low"}
</div>
</div>

{/* Footer Note */}
<p
  style={{
    marginTop: "1.2rem",
    fontSize: "0.85rem",
    color: "#6b7280",
  }}
>
  Tip: Reduce travel emissions and switch to renewable energy to lower
  your footprint
</p>
</div>
</div>
);
};
```

export default Dashboard;

## src/context/AuthContext.jsx

```
import { createContext, useContext, useState } from "react";

export const AuthContext = createContext(null);

export const AuthProvider = ({ children }) => {
  const [isAuthenticated, setIsAuthenticated] = useState(false);

  return (
    <AuthContext.Provider value={{ isAuthenticated, setIsAuthenticated }}>
      {children}
    </AuthContext.Provider>
  );
};

export const useAuth = () => useContext(AuthContext);
```

## src/pages/login.jsx

```
import { useAuth } from "../context/AuthContext.jsx";
import { useNavigate } from "react-router-dom";

const Login = () => {
  const { setIsAuthenticated } = useAuth();
  const navigate = useNavigate();

  const handleLogin = () => {
    setIsAuthenticated(true);
    navigate("/");
  };
};
```

```
return (  
<div  
  style={{  
    minHeight: "100vh",  
    backgroundColor: "#f0fdf4",  
    display: "flex",  
    alignItems: "center",  
    justifyContent: "center",  
    padding: "2rem",  
  }}  
>  
<div  
  style={{  
    width: "100%",  
    maxWidth: "420px",  
    backgroundColor: "#ffffff",  
    borderRadius: "14px",  
    padding: "2rem",  
    boxShadow: "0 10px 24px rgba(0,0,0,0.1)",  
  }}  
>  
  { /* Title */}  
  <h2  
    style={{  
      fontSize: "1.9rem",  
      fontWeight: "700",  
      color: "#065f46",  
      textAlign: "center",  
      marginBottom: "0.5rem",  
    }}  
  >  
    Welcome Back  
  </h2>  
  
  <p  
    style={{  
      textAlign: "center",  
      color: "#047857",  
      fontSize: "0.95rem",  
      marginBottom: "2rem",  
    }}  
  >  
    Sign in to access your EcoTrack dashboard  
  </p>  
  
  { /* Login Button */}  
  <button  
    onClick={handleLogin}  
    style={{  
      width: "100%",  
      padding: "0.8rem",  
      borderRadius: "10px",  
      backgroundColor: "#16a34a",  
      color: "white",  
      fontWeight: "600",  
      fontSize: "1.05rem",  
      border: "none",  
      cursor: "pointer",  
      boxShadow: "0 4px 12px rgba(0,0,0,0.15)",  
    }}  
    onMouseOver={(e) =>
```



# DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

```
(e.target.style.backgroundColor = "#15803d")
}
onMouseOut={ (e) =>
  (e.target.style.backgroundColor = "#16a34a")
}
>
  Login
</button>

{/* Footer note */}
<p
  style={{
    marginTop: "1.5rem",
    fontSize: "0.85rem",
    color: "#6b7280",
    textAlign: "center",
  }}
>
  Secure access • Environment-first tracking
</p>
</div>
</div>
);
};

export default Login;
```

## src/Routes/ProtectedRoute.jsx

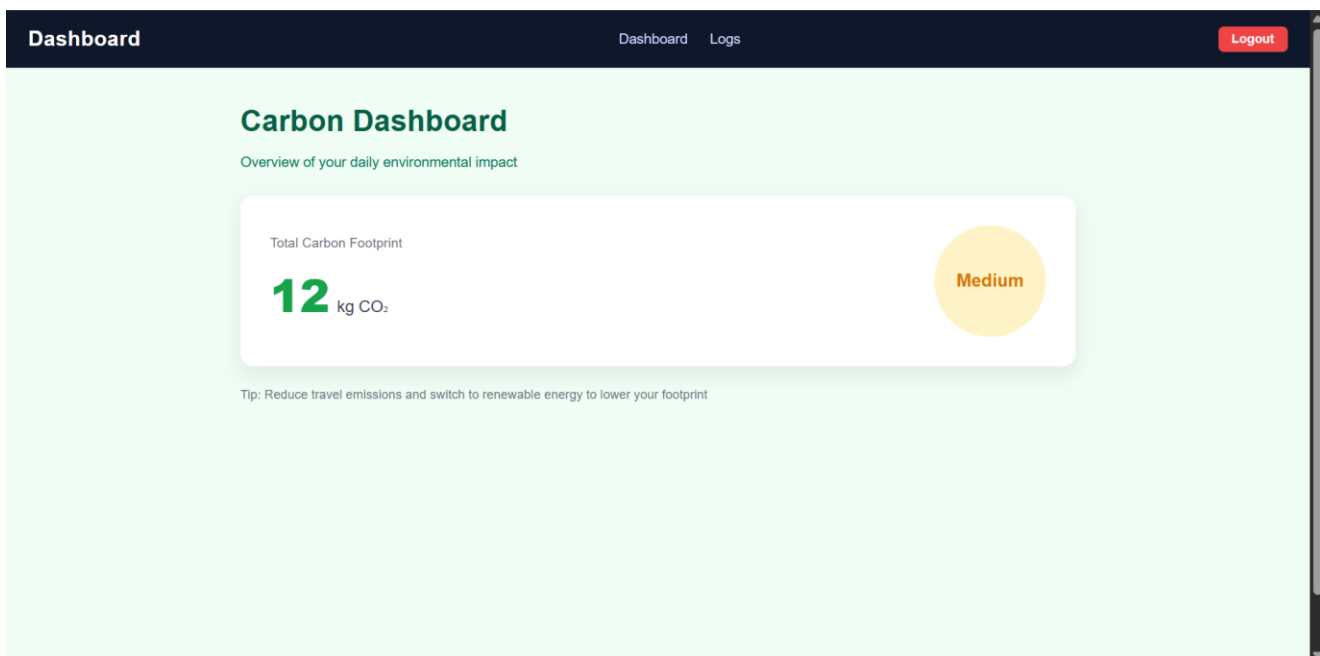
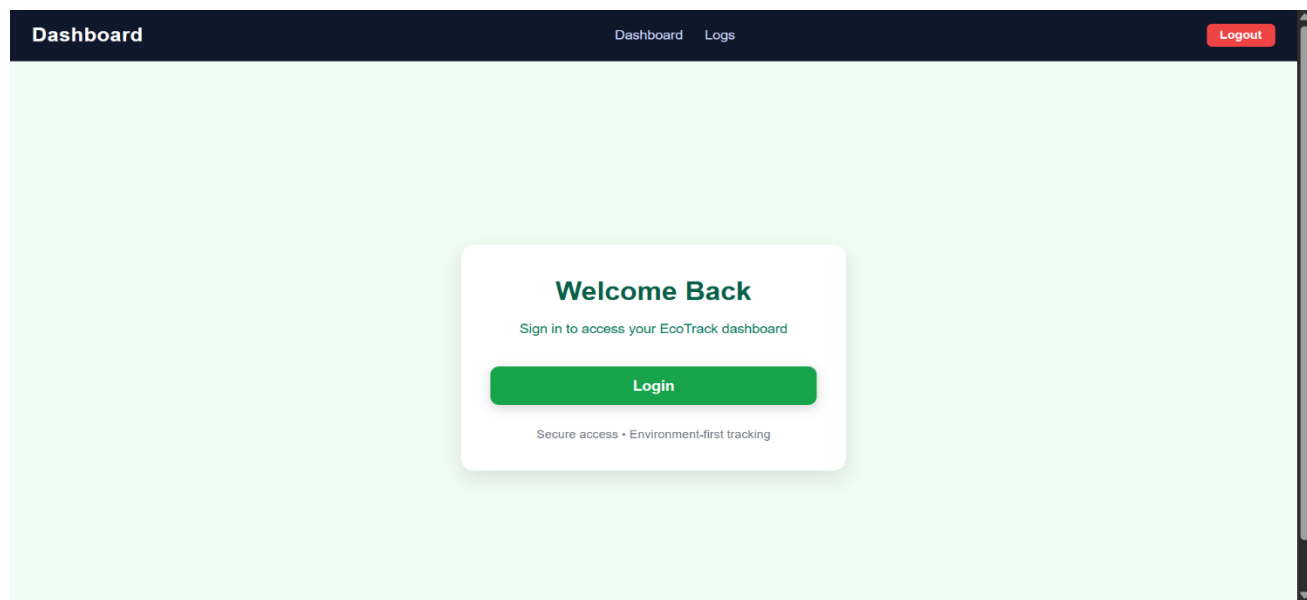
```
import {Navigate} from "react-router-dom";
import {useAuth} from "../context/AuthContext";

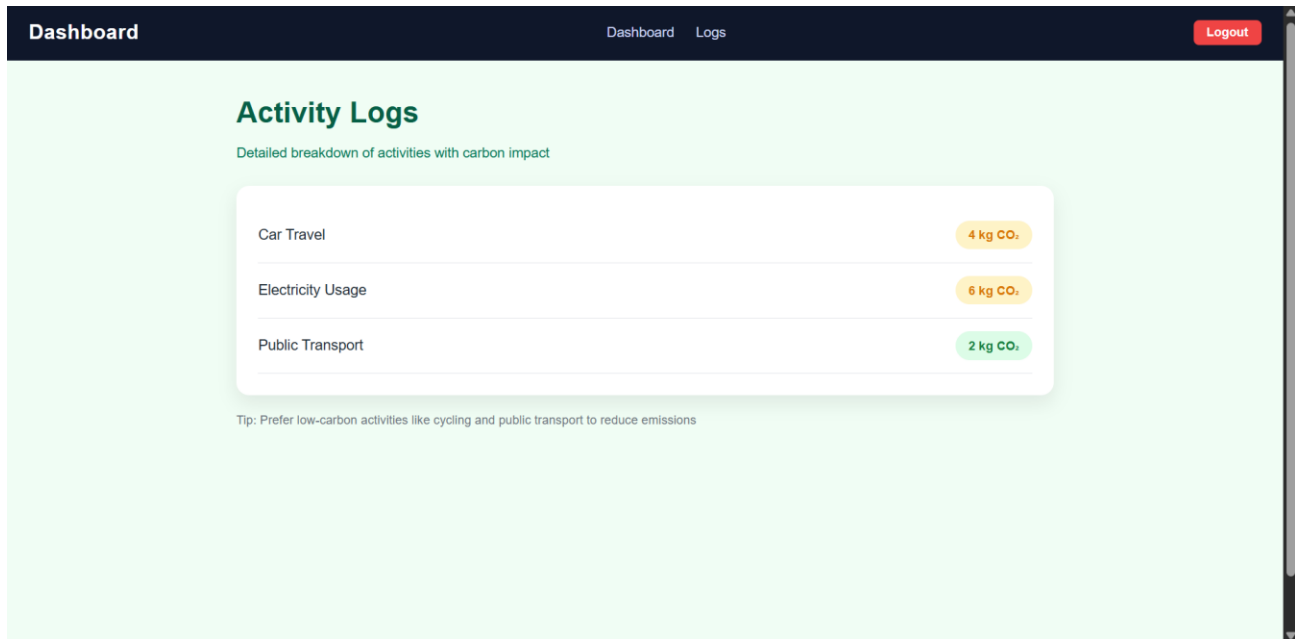
const ProtectedRoute = ({children}) => {
  const {isAuthenticated} = useAuth();

  if(!isAuthenticated){
    return <Navigate to="/login" replace />;
  }
  return children;
}
export default ProtectedRoute;
```

## 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
- Logout functionality securely ends the session
- Proper route protection is verified using ProtectedRoute





## 5. Learning Outcomes (What I Have Learnt)

After completing this experiment, the student is able to:

- 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 applications



# **DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING**

Discover. Learn. Empower.



# **DEPARTMENT OF** **COMPUTER SCIENCE & ENGINEERING**

Discover. Learn. Empower.



# **DEPARTMENT OF** **COMPUTER SCIENCE & ENGINEERING**

Discover. Learn. Empower.