Experiment 2

Name : Astik Joshi	**UID :** 23BCS10627
Branch: BE-CSE	**Section/Group:** KRG-3B
Semester: 6th	**Date of Performance:** 19-01-2026
Subject Name: Full Stack-II	**Subject Code:** 23CSH-309

1. **Aim:** The aim of this implementation is to develop a secure navigation system using Contextbased authentication along with route protection, ensuring that only logged-in users can access authorized pages of the application.

2. **Objective:-**
   - To centrally manage authentication state across the entire application
   - To enable secure navigation using protected and public routes
   - To ensure only authenticated users can access restricted pages
   - To automatically redirect unauthenticated users to the login page while maintaining a scalable code structure
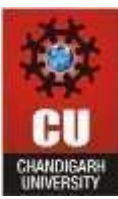
3. **Implementation/Code:**

   - **Header.jsx :-**

```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
```

```jsx
style={{
  backgroundColor: "#ffffff",
  borderBottom: "1px solid #e5e7eb",
  padding: "0.9rem 1.8rem",
  display: "flex",
  alignItems: "center",
  justifyContent: "space-between",
}}
>
{/* App Name */}
<h1
  style={{
    fontSize: "1.4rem",
    fontWeight: "600",
    color: "#065f46",
    margin: 0,
  }}
>
  {title}
</h1>

{/* Links */}
<nav style={{ display: "flex", gap: "1.4rem" }}>
  <Link
    to="/"
    style={{
      textDecoration: "none",
      color: "#374151",
      fontSize: "0.95rem",
    }}
  >
    Dashboard
  </Link>

  <Link
    to="/log"
    style={{
      textDecoration: "none",
      color: "#374151",
      fontSize: "0.95rem",
    }}
  >
    Logs
```
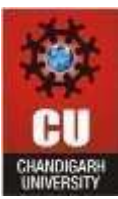
```
    </Link>
  </nav>

  {/* Logout */}
  <button
    onClick={handleLogOut}
    style={{
      backgroundColor: "transparent",
      border: "1px solid #dc2626",
      color: "#dc2626",
      padding: "0.35rem 0.9rem",
      borderRadius: "4px",
      fontSize: "0.85rem",
      cursor: "pointer",
    }}
  >
    Logout
  </button>
</header>
);
};

export default Header;
```
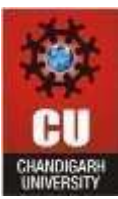
- **AuthContext.jsx :-**

```
1   import { createContext, useContext, useState } from "react";
2
3   export const AuthContext = createContext(null);
4
5   export const AuthProvider = ({ children }) => {
6     const [isAuthenticated, setIsAuthenticated] = useState(false);
7
8     return (
9       <AuthContext.Provider value={{ isAuthenticated, setIsAuthenticated }}>
10        {children}
11      </AuthContext.Provider>
12    );
13  };
14
15  export const useAuth = () => useContext(AuthContext);
```

- **Login.jsx :-**

```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: "#eef2f7",
        display: "flex",
        alignItems: "center",
        justifyContent: "center",
      }}
    >
      <div
        style={{
          width: "380px",
          backgroundColor: "#ffffff",
          border: "1px solid #e5e7eb",
          padding: "2rem",
        }}
      >
        <h2
          style={{
            fontSize: "1.6rem",
            fontWeight: "600",
            color: "#065f46",
            marginBottom: "0.5rem",
          }}
        >
          Login
        </h2>

        <p
          style={{
```

```
        fontSize: "0.9rem",
        color: "#4b5563",
        marginBottom: "1.5rem",
      }}
    >
      Access your EcoTrack dashboard
    </p>

    <button
      onClick={handleLogin}
      style={{
        width: "100%",
        padding: "0.7rem",
        backgroundColor: "#065f46",
        color: "#ffffff",
        border: "none",
        fontSize: "0.95rem",
        cursor: "pointer",
      }}
    >
      Sign In
    </button>

    <p
      style={{
        marginTop: "1.3rem",
        fontSize: "0.8rem",
        color: "#6b7280",
      }}
    >
      Simple authentication demo
    </p>
    </div>
  </div>
 );
};

export default Login;
```
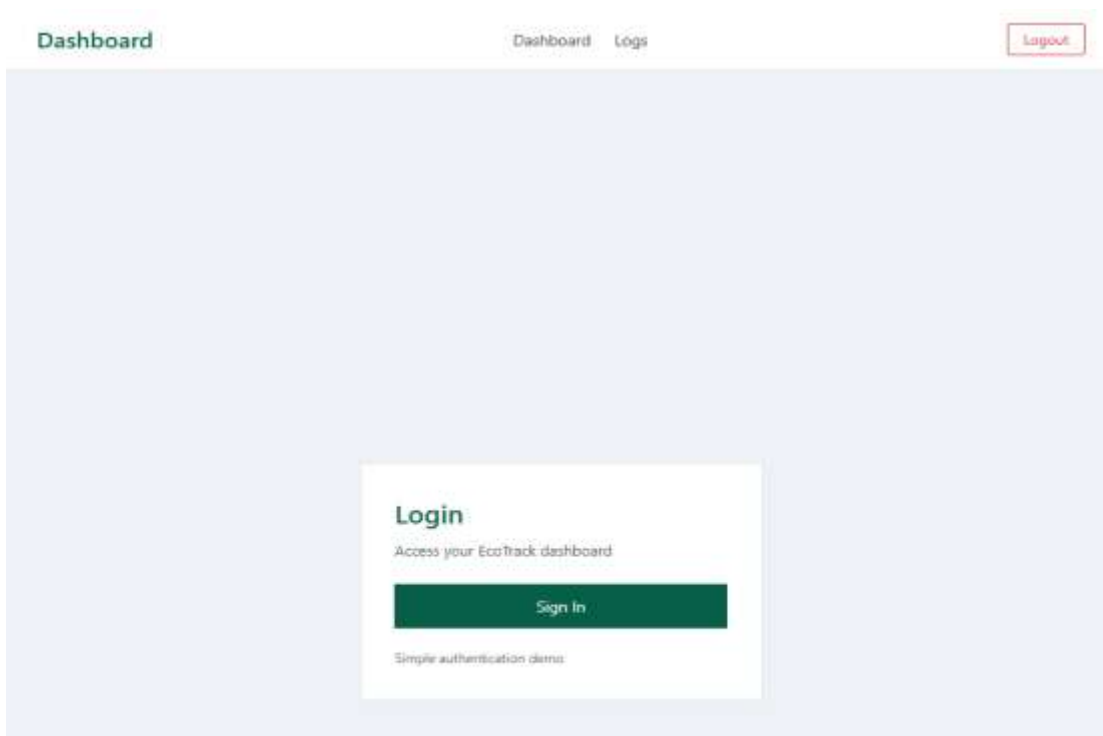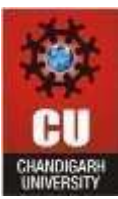
- **App.jsx:-**

```
1   import Dashboard from "./pages/dashboard.jsx";
2   import Log from "./pages/log.jsx";
3   import Header from "./components/Header.jsx";
4   import { Routes, Route, Link, BrowserRouter } from "react-router-dom";
5   import Login from "./pages/login.jsx";
6   import ProtectedRoute from "./Routes/ProtectedRoute.jsx";
7   import DashboardLayout from "./pages/dashboardLayout.jsx";
8
9   function App() {
10    return (
11      <>
12        <BrowserRouter>
13          <Header title="Dashboard" />
14          <main>
15            <Routes>
16              <Route path="/login" element={<Login />} />
17              <Route
18                path="/dashboardLayout"
19                element={
20                  <ProtectedRoute>
21                    <DashboardLayout />
22                  </ProtectedRoute>
23                }
24              />
25              <Route
26                path="/"
27                element={
28                  <ProtectedRoute>
29                    <Dashboard />
30                  </ProtectedRoute>
31                }
32              />
33              <Route
34                path="/log"
35                element={
36                  <ProtectedRoute>
37                    <Log />
38                  </ProtectedRoute>
39                }
40              />
41            </Routes>
42          </main>
43        </BrowserRouter>
44      </>
45    );
46  }
47
48  export default App;
49
```

- ProtectRoute.jsx

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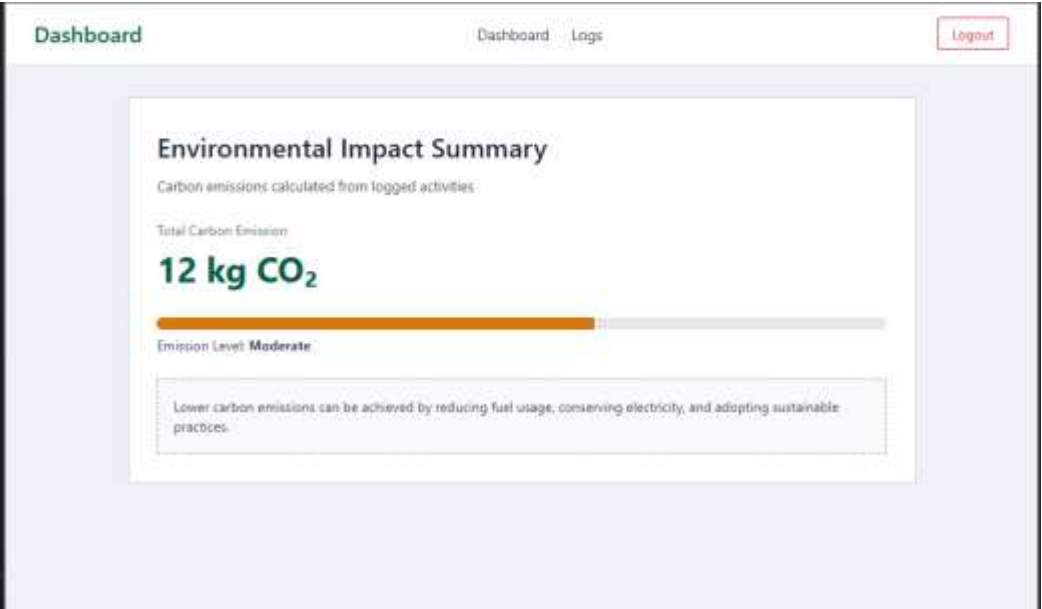
4. Output

Dashboard   Dashboard   Logs   Logout

Login
Access your EcoTrack dashboard

Sign In

Simple authentication demo

DEPARTMENT OF
COMPUTER SCIENCE & ENGINEERING
Discover. Learn. Empower.

Dashboard   Dashboard   Logs   Logout

Environmental Impact Summary

Carbon emissions calculated from logged activities

Total Carbon Emission

12 kg CO_2

Emission Level: **Moderate**

Lower carbon emissions can be achieved by reducing fuel usage, conserving electricity, and adopting sustainable practices.
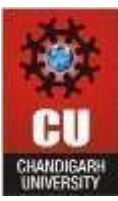

Dashboard   Dashboard   Logs   Logout

Activity Logs

Track your activities and their carbon impact

Activity	Carbon Emission (kg CO_2)
Car Travel	4
Electricity Usage	6
Public Transport	2

Tip: Choose sustainable options like biking, walking, or public transport to lower your emissions.

5. Learning Outcome :-

- Learned global state management using Context
- Implemented protected routing for secure navigation
- Controlled page access based on authentication status
- Enabled automatic redirection for unauthenticated users
- Improved application security and code maintainability
- Enhanced user experience through seamless navigation
- Built a scalable and reusable authentication structure