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

Experiment 4

Student Name: Rashid Khan   **UID:** 23BCS12551

Branch: BE CSE   **Section/Group:** KRG_3B

Semester: 6th   **Date of Performance:** 06/02/26

Subject Name: Full Stack - II   **Subject Code:** 23CSH-309

Aim: To develop a secure navigation system using Context-based authentication, protected routing, and optimized React hooks to improve performance and UI using Material UI..

Objective:

- To manage **user authentication state globally** using Context
- To implement **routing for page navigation**
- To restrict access to protected routes for unauthenticated users
- To allow page switching **only after successful login**
- To redirect unauthenticated users automatically to the **login page**
- To improve application security and user experience
- To maintain clean, scalable, and reusable code structure

Input/Apparatus Used:

- Programming Language: JavaScript (ES6+)
- Framework / Library: React (Functional Components)
- Build Tool: Vite
- Code Editor: Visual Studio Code
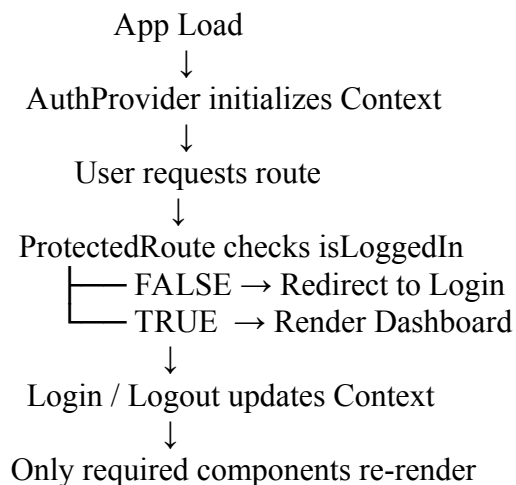- Web Browser: Google Chrome

Tools & Tech Used

- **React (Functional Components)**
- **React Router DOM**
- **Context API**
- **Optimized Hooks: React.memo, useMemo, useCallback**
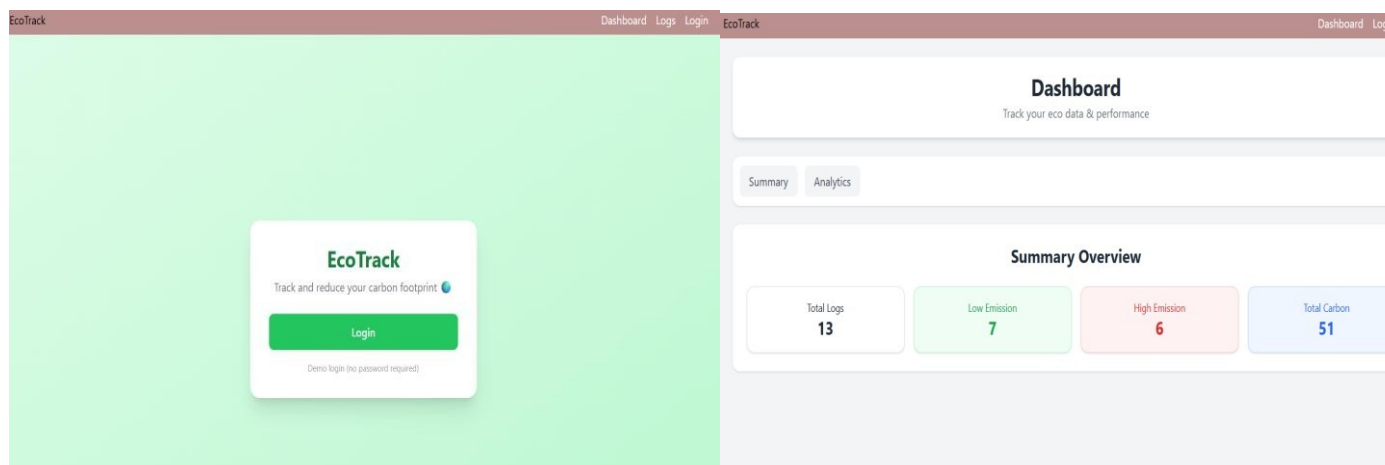- **Material UI (MUI)**

File Structure

```
src/
├── main.jsx
├── App.jsx
├── context/
│   └── AuthContext.jsx
├── components/
│   ├── Login.jsx
│   ├── Dashboard.jsx
│   │   └── ProtectedRoute.jsx
```
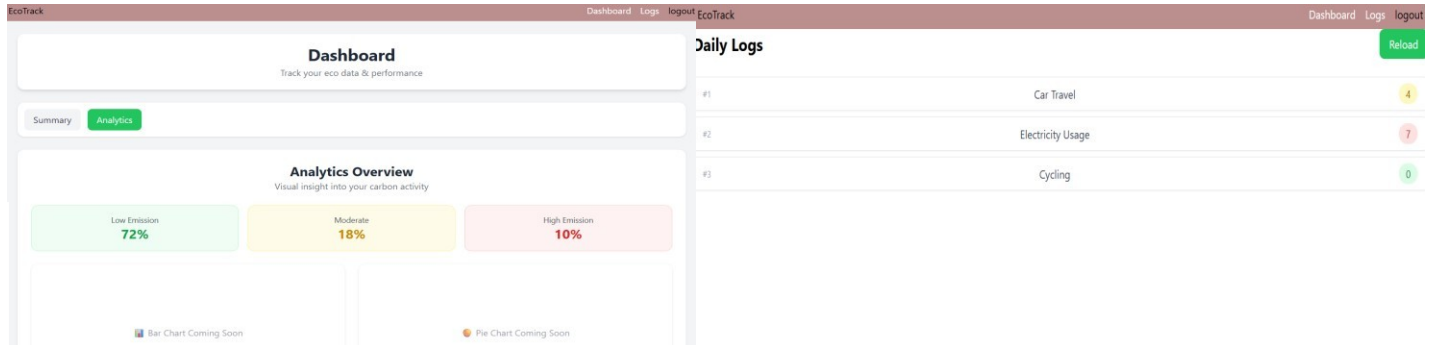
Flow of data

```
                App Load
                   ↓
       AuthProvider initializes Context
                   ↓
            User requests route
                   ↓
       ProtectedRoute checks isLoggedIn
              ├── FALSE → Redirect to Login
              └── TRUE  → Render Dashboard
                   ↓
          Login / Logout updates Context
                   ↓
       Only required components re-render
```

Output:

Learning Outcomes

- Learned performance optimization using React.memo, useMemo, and useCallback
- Implemented secure routing with Context API
- Reduced unnecessary re-renders
- Built modern UI using Material UI
- Improved scalability and maintainability