



Experiment 2

Student Name: Aaditya Anand

Branch: BE CSE

Semester: 6th

Subject Name: Full Stack Development-II

UID: 23BCS10446

Section/Group: 23BCSKRG_3A

Date of Performance: 19/01/26

Subject Code: 23CSH-309

1. **Aim:** To implement Single Page Application (SPA) navigation in the Eco track application using React Router, secure application routes using context-based authentication, and extend nested dashboard routing through follow-up enhancements.
2. **Objective:**
 - To enable smooth Single Page Application (SPA) navigation in Eco Track using React Router.
 - To protect application routes using context-based authentication.
 - To create nested dashboard routes for better and scalable navigation.
 - To manage user login and session data using React Context API.
 - To build a flexible routing system that supports future dashboard features.
3. **Implementation/Code:**

dashboard.jsx:

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

const Dashboard = () => {  const totalcarbon = logs.reduce((total,
log) => total + log.carbon, 0);

  return (
    <div className="dashboard">
      <header className="dashboard-header">
        <h2 className="dashboard-title">Dashboard</h2>
        <p className="dashboard-summary">Total Carbon Footprint:
<strong>{totalcarbon} kg CO2</strong></p>
      </header>

      <section className="logs">
        <ul className="log-list">
```



```
{logs.map((log) => (  
  <li className="log-item" key={log.id}>  
    <span className="activity">{log.activity}</span>  
    <span className="carbon">{log.carbon} kg CO2</span>  
  </li>  
  )  
))  
</ul>  
</section>  
</div>  
);  
};
```

```
export default Dashboard;
```

login.jsx:

```
import {useAuth} from '../context/AuthContext';  
import {useNavigate} from 'react-router-dom';  
const LoginPage=()=>useAuth(); const  
{setIsAuthenticated}=useAuth(); const  
navigate=useNavigate(); const  
handleLogin=()=>{  
  // Perform authentication logic here (e.g., API call)  
  // On successful authentication:  
  setIsAuthenticated(true);  
  navigate('/'); // Redirect to a protected route after login  
  
  return(  
    <  
      <div>  
        <h2>Login Page</h2>  
        <button onClick={handleLogin}>Login to ECOTRACK</button>  
      </div>  
    </>  
  );  
};  
  
export default handleLogin;
```



ProtectedRoute.jsx:

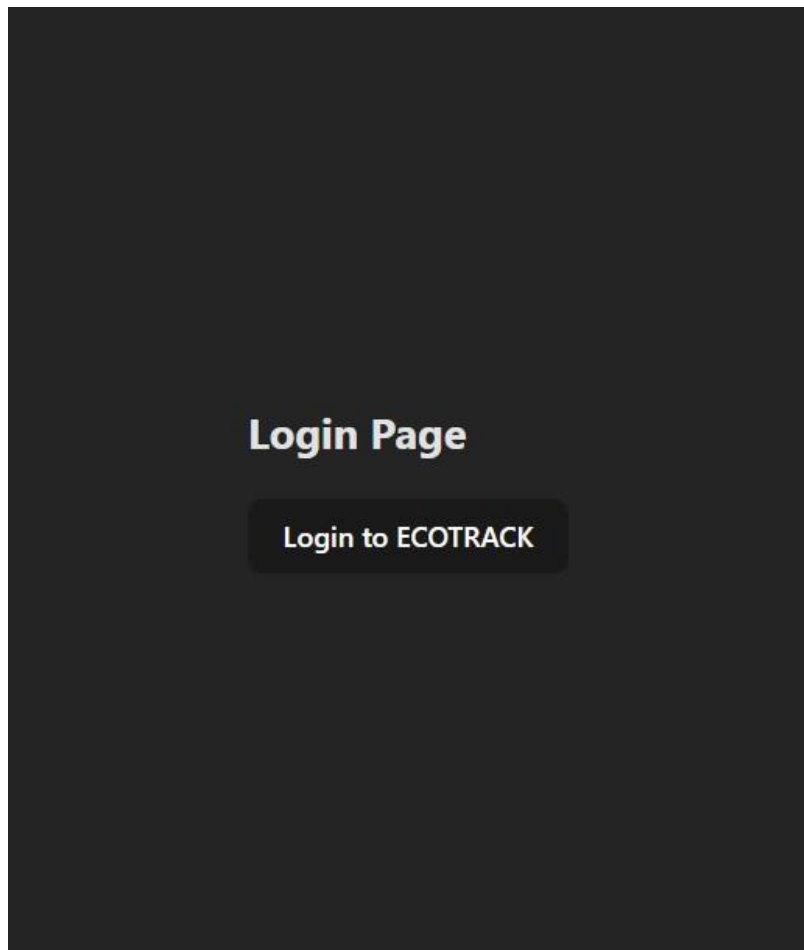
```
import { Navigate } from "react-router-dom";
import { useAuth } from
"../context/AuthContext";

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

  return isAuthenticated ? children : <Navigate to="/login" />;
};

export default ProtectedRoute;
```

4. Output:





DEPARTMENT OF

COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

5. Learning Outcome:

- To understand how to create Single Page Applications using React Router for smooth navigation.
- To learn how to protect routes using Context API–based authentication.
- To practice using nested routing for organized dashboard layouts.
- To develop skills in managing global user data and sessions in React.
- To learn how to build scalable and maintainable routing systems for real-world projects.