



Experiment 2

Student Name: Gourav Sharma

Branch: BE CSE

Semester: 6th

Subject Name: Full Stack Development-II

UID: 23BCS10857

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 EcoTrack application using React Router, secure application routes using context-based authentication, and extend nested dashboard routing through follow-up enhancements.

2. Objective:

- To implement seamless Single Page Application (SPA) navigation in EcoTrack using React Router.
- To secure application routes through context-based authentication and protected routing.
- To design nested dashboard routes for modular and scalable UI navigation.
- To manage user session state globally using React Context API.
- To enhance dashboard functionality with future-ready, extensible routing architecture.

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}</strong> kg CO2</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;
```



DEPARTMENT OF

COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

ProtectedRoute.jsx:

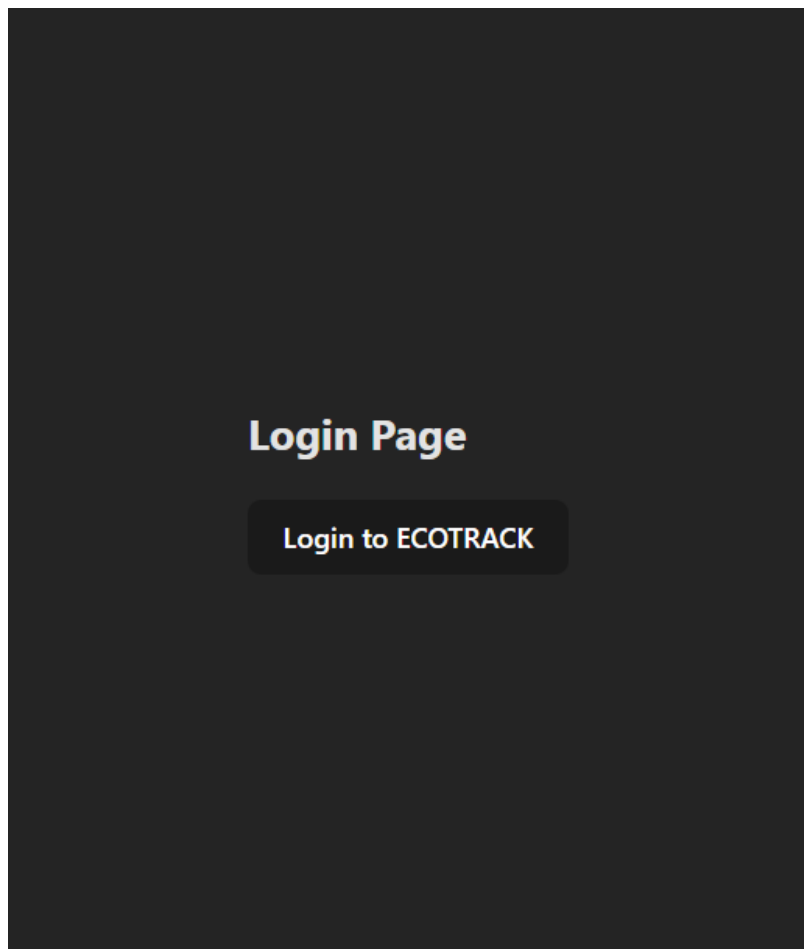
```
import { Navigate } from "react-router-dom";
import { useAuth } from
"../context/AuthContext";

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:

- Understand how to build Single Page Applications using React Router for smooth client-side navigation.
- Learn to implement protected routes using Context API-based authentication.
- Gain hands-on experience with nested routing for structured dashboard layouts.
- Develop skills in managing global user state and session handling in React.
- Learn to design scalable and maintainable routing architectures for real-world applications.