Experiment - 1

Student Name : Jobanjot Singh Grewal
UID : 23BIA50005
Branch : BE CSE (AIML)
Section/Group : 23AIT-KRG/G1
Semester : 6th SEM
Date of Performance : 14/01/26
Subject Name : Full Stack II
Subject Code : 23CSH-382

1. Aim :

To design and implement the foundational frontend architecture of the EcoTrack application using modern React practices, Vite tooling, and ES6+ JavaScript features.

2. Objective :

- To set up a React project using Vite with proper project structure
- To understand component-based architecture in React
- To apply ES6 array methods (map, filter, reduce) for data-driven UI rendering
- To separate concerns using components, pages, and data modules

3. Implementation/Code :

Header.jsx :

```jsx
const Header = ({title}) => {

    return (
        <header style = {
            {
            padding: "1rem",
            backgroundColor: "#4CAF50",
            color: "white",
            textAlign: "center"
            }

        }>
        <h1>{title}</h1>
        </header>
    );

};

export default Header;
```

Dasboard.jsx :

```jsx
import logs from "..data/logs";

const Dashboard = () => {
    const totalCarbon = logs.reduce((total, log) =>
        total + log.carbon, 0);
};
return (
    <div>
        <h1>Dashboard</h1>
        <p>Total Carbon Footprint: {totalCarbon} kg CO2</p>

        <h2>Activity Logs</h2>
        <ul>
            {logs.map(log => (
                <li key={log.id}>
                    {log.activity}: {log.carbon} kg CO2
                </li>
            ))}
        </ul>
    </div>
)

export default Dashboard;
```

App.jsx :

```jsx
import { useState } from 'react'
import Header from './components/Header'
import Dashboard from './pages/dashboard'
import './App.css'

function App() {
  const [currentPage, setCurrentPage] = useState('dashboard')

  return (
    <>
      <Header onNavClick={setCurrentPage} />
      <div className="container">
        {currentPage === 'dashboard' && <Dashboard />}
        {currentPage === 'logs' && <logPage />}
      </div>
    </>
  )
}

export default App
```

logs.jsx :

```jsx
import log from "../data/logs";

const logPage = () => {
    const highCarbonLogs = log.filter(log => log.carbon >= 4);
    const lowCarbonLogs = log.filter(log => log.carbon < 4);
    return (
    <div>
        <h1>High Carbon Logs</h1>
        <ul>
            {highCarbonLogs.map(log =>(
                <li style={{ color: "red" }} key={log.id}>
                    {log.activity}: {log.carbon} kg CO2
                </li>
            ))}
        </ul>
        <h1>Low Carbon Logs</h1>
        <ul>
            {lowCarbonLogs.map(log =>(
                <li style={{ color: "green" }} key={log.id}>
                    {log.activity}: {log.carbon} kg CO2
                </li>
            ))}
        </ul>
    </div>
    )
}

export default logPage;
```

4. Output :

Total Carbon Footprint: 10 Kg

- Car Travel = 4 KG
- Electricity Usage = 6 KG
- Cycling = 0 KG

High Carbon Activity

- Car Travel = 4 Kg
- Electricity Usage = 6 Kg

Low Carbon Activity

- Cycling = 0 Kg

5. Learning Outcomes :

- Analyze Project Structure: Deduce the purpose and architecture of a React application by examining its file and directory organization.
- Component-Based Architecture: Understand the distinction between page-level components (pages/) and reusable UI components (components/).
- React Router (or equivalent): (Assuming App.jsx handles routing) Understand how to implement client-side routing to create a single-page application (SPA) feel.
- Data Handling: Learn how static data can be imported and utilized within React components (as seen with data/logs.js).