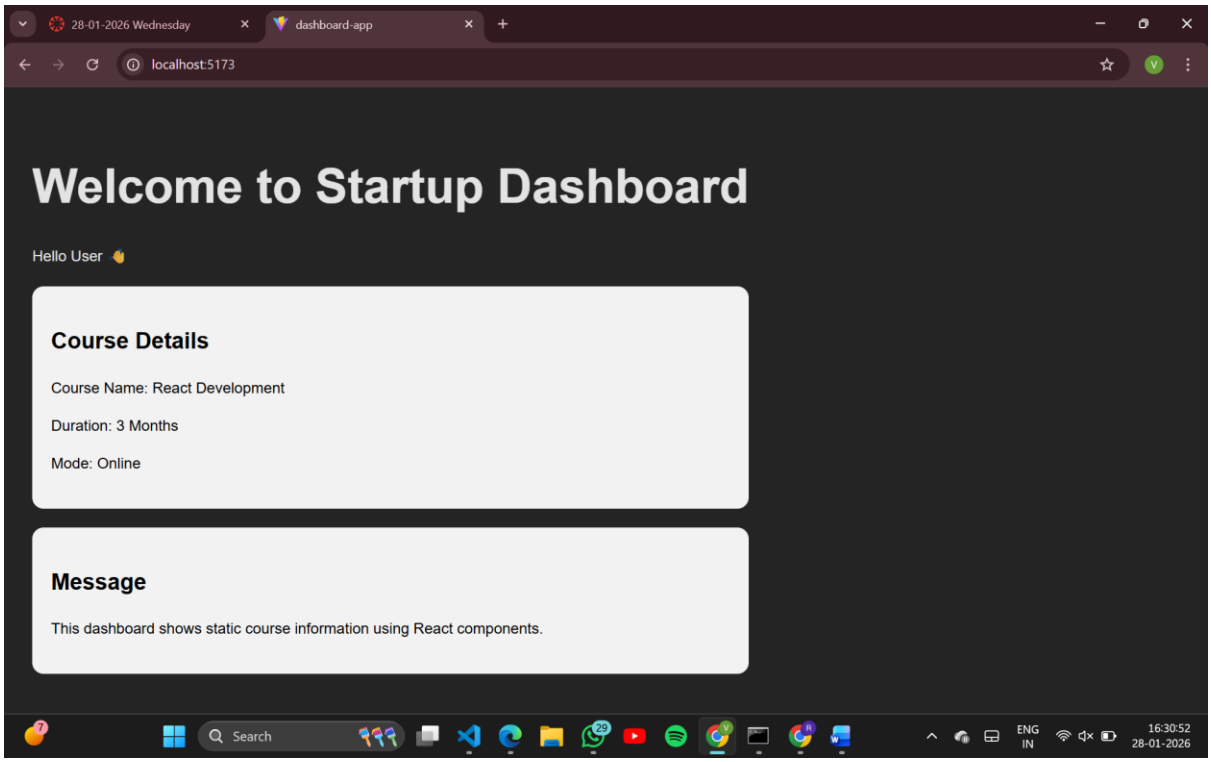
DevOps Full Stack
Assignment- 8/24(week 4) Wednesday

Name: Ravula Vyshnavi
Ht.no: 2303A51216
Batch: 04



1. How would you set up a new React project using tools like Create React App or Vite?

Ans: A React project can be created using tools such as Create React App or Vite. These tools provide a ready-made structure with necessary configuration files. After installing Node.js, the developer runs commands in the terminal to generate the project folder. Required libraries are automatically installed, and a development server is started. This server allows the application to run in a web browser and updates the output instantly when changes are made.

2. What is the role of package.json in a React project?

Ans: The package.json file is an important file that manages the project information and dependencies. It contains the project name, version, and list of libraries required for the application. It also stores scripts that help in running and building the project. By using this file, developers can easily install, update, and manage packages needed for the React application**.**

3. How do you create a functional component in React?

Ans: Functional components are JavaScript functions that return JSX elements to create the user interface. They are simple, lightweight, and reusable. Each functional component represents a part of the UI such as a header, message box, or card. Modern React development mostly uses functional components because they are easy to understand and support powerful features like React hooks.

4. How are components rendered inside the main App component?

Ans: In React, all components are combined inside a main component called App. Other components are imported into the App file and placed inside its JSX structure. The App component acts as the root of the application and controls what appears on the screen. This approach helps in organizing the UI clearly and makes the code easy to manage.

5.What are the benefits of breaking the UI into small reusable components?

Ans: Reusable components help in building clean and efficient applications. They reduce repeated code by allowing the same component to be used in multiple places. Components make maintenance easier because changes can be made in one place and reflected everywhere. They also improve teamwork by dividing the application into smaller logical parts. This approach supports scalability and faster development.