**Naan Mudhalvan - Frontend Development and Database Administration**

**VALLIAMMAL COLLEGE FOR WOMEN**

**(College code: 1363)**

**Department of Computer Applications**

**Project title: CookBook: Your Virtual Kitchen Assistant**

**(React Application)**

**NM Team ID: 151575**

**Team leader: Sangari. S**

**Team size: 4**

**Team members:** **1.** **Sangari. S**

**2. Sujitha. V**

**3. Pavithra. B**

**4. Kaviya. K**

**GitHub link (Includes coding and documentation):**

**https://github.com/sangarishanmugam/cookbook**

**Google drive link (Includes project demo video):**

**https://drive.google.com/file/d/1-4e0QV5M99mpjpcGhfc7ic6eV9OM0voI/view?usp=drivesdk**

**Frontend Development with React.js**

**Project Documentation**

1. **Introduction**
   * **Project Title:** CookBook: Your Virtual Kitchen Assistant (React Application)
   * **Team Members**: **1.** **Sangari. S** (Role: Leading the team and assigning tasks)

**2. Sujitha. V** (Role: Coding and assistance)

**3. Pavithra. B** (Role: Documentation and assistance)

**4. Kaviya. K** (Role: Project video demo-making and assistance)

1. **Project Overview**
   * **Purpose**: Briefly describe the purpose and goals of the project.
   * **Features**: Highlight the key features and functionalities of the frontend.
2. **Architecture**
   * **Component Structure**: Outline the structure of major React components and how they interact.
   * **State Management**: Describe the state management approach used (e.g., Context API, Redux).
   * **Routing**: Explain the routing structure if using react-router or another routing library.
3. **Setup Instructions**
   * **Prerequisites**: List software dependencies (e.g., Node.js).
   * **Installation**: Provide a step-by-step guide to clone the repository, install dependencies, and configure environment variables.
4. **Folder Structure**
   * **Client**: Describe the organization of the React application, including folders like components, pages, assets, etc.
   * **Utilities**: Explain any helper functions, utility classes, or custom hooks used in the project.
5. **Running the Application**
   * Provide commands to start the frontend server locally.
     + **Frontend**: npm start in the client directory.
6. **Component Documentation**
   * **Key Components**: Document major components, their purpose, and any props they receive.
   * **Reusable Components**: Detail any reusable components and their configurations.
7. **State Management**
   * **Global State**: Describe global state management and how state flows across the application.
   * **Local State**: Explain the handling of local states within components.
8. **User Interface**
   * Provide screenshots or GIFs showcasing different UI features, such as pages, forms, or interactions.
9. **Styling**

* **CSS Frameworks/Libraries**: Describe any CSS frameworks, libraries, or pre-processors (e.g., Sass, Styled-Components) used.
* **Theming**: Explain if theming or custom design systems are implemented.

1. **Testing**

* **Testing Strategy**: Describe the testing approach for components, including unit, integration, and end-to-end testing (e.g., using Jest, React Testing Library).
* **Code Coverage**: Explain any tools or techniques used for ensuring adequate test coverage.

1. **Screenshots or Demo**

* Provide screenshots or a link to a demo showcasing the application’s features and design.

1. **Known Issues**

* Document any known bugs or issues that users or developers should be aware of.

1. **Future Enhancements**

* Outline potential future features or improvements, such as new components, animations, or enhanced styling.