## **COOKBOOK - Front End Development with React.js**

### **Project Documentation**

#### **Introduction**

**Project Title:** CookBook - Your Virtual Kitchen Assistant

**Team Members:**

1. Saravanan .M
2. Praveen Kumar .G
3. Tharun. A.S
4. Janani. R

#### **Project Overview**

**Purpose:** The CookBook application is designed to help users discover, organize, and manage recipes effortlessly. It caters to both novice and professional chefs, providing a modern and intuitive user experience for culinary exploration.

**Goals:**

* Simplify the recipe discovery and management process.
* Provide a visually appealing and user-friendly interface.
* Offer real-time recipe data integration using APIs.
* Enable users to search and save their favorite recipes.

**Features:**

* Recipe Database Integration
* Category-Based Search
* Recipe Details Page
* Favorites Feature
* Nutritional Information
* Dark Mode
* Community Sharing
* User Authentication
* Multi-Language Support

### **Architecture**

#### **Component Structure**

The CookBook application follows a modular and reusable component structure to enhance maintainability and scalability. Major components include:

* **Navbar:** Navigation across the application.
* **Hero Section:** Welcoming users with a brief intro and featured recipes.
* **Search Bar:** Allows users to search recipes by keyword.
* **Category Cards:** Display available recipe categories.
* **Recipe Details:** Showcases instructions, ingredients, and video tutorials.
* **Footer:** Contains social media links and copyright info.

#### **State Management**

* **Local State:** Managed using React's useState and useEffect hooks.
* **Global State:** Not used as the application is relatively simple and does not require complex state management.
* **API Calls:** Managed using Axios within useEffect for data fetching.

#### **Routing**

The application uses **React Router Dom** for client-side navigation. Key routes include:

* / - Home Page
* /category/:category - Category-wise recipes
* /recipe/:id - Recipe details page

### **Setup Instructions**

#### **Prerequisites**

* **Node.js:** Runtime environment for executing JavaScript on the server side.
* **npm:** Package manager to install dependencies.
* **React:** JavaScript library for building user interfaces.

#### **Installation**

Clone the repository:  
 git clone https://github.com/your-repo/cookbook-app.git

1. Navigate to the project directory:  
    cd cookbook-app
2. Install dependencies:  
    npm install
3. Start the development server:  
    npm start
4. Access the application at: <http://localhost:3000/>

**Folder Structure**

The project is organized as follows:

/cookbook-app

│-- /src

│ │-- /components (Reusable components)

│ │-- /pages (Main pages like Home, Category, Recipe)

│ │-- /styles (CSS files)

│ │-- App.js (Main application file)

│ │-- index.js (Entry point of the app)

│-- /public (Static assets)

│-- package.json (Project configuration)

### **Component Documentation**

#### **Key Components**

1. **Navbar:** Provides navigation between home, categories, and recipe pages.
2. **Hero Section:** Introduces the application with featured content.
3. **Recipe Card:** Displays individual recipes with a thumbnail and title.
4. **Search Bar:** Allows users to filter recipes by name.
5. **Footer:** Displays links to social media and other resources.

#### **Reusable Components**

* **Card Component:** Used to display both recipes and categories.
* **Button Component:** Styled buttons for interactions.
* **Input Component:** Used for search functionality.

### 

### **User Interface**

The UI follows a clean and minimalistic design using **Tailwind CSS** for consistent styling. The layout is responsive and optimized for various devices.

#### **Styling**

* **CSS Framework:** Tailwind CSS
* **Theming:** Supports dark mode for better accessibility.

### **Testing**

#### **Testing Strategy**

* Unit testing using **Jest** for core functionalities.
* Integration testing for API calls.
* End-to-end testing using **React Testing Library**.

#### **Code Coverage**

Tools like **Jest** are used to ensure code quality and coverage.

### **Screenshots or Demo**

* **Homepage:** Shows featured recipes and categories.
* **Search Page:** Filters recipes based on the keyword.
* **Recipe Details Page:** Displays complete cooking instructions.
* **Footer Page:** Includes social media links and app information.

### **Known Issues**

* Some recipes may not have tutorial videos available.
* API rate limits may cause occasional data fetching issues.

### **Future Enhancements**

* Add user authentication for personalized recipe management.
* Implement a shopping list feature to add ingredients directly.
* Enhance the search functionality with advanced filters.
* Include more regional cuisines and dietary preferences.

**THANK YOU.**