PROJECT DOCUMATION

TITLE: COOKBOOK- YOUR VIRTUAL KITCHEN ASSISTANT

1.INTRODUCTION:

> PROJECT TITLE: YOUR VIRTUAL KITCHEN ASSISTANT

> TEAM ID: NM2025TMID37977

> TEAM LEADER: KAMALI.N MAIL ID: ndkamali12@gmail.com

> TEAM MEMBERS:

• KEERTHANA.A MAIL ID:akeerthanadass17@gmail.com

SHEEBA.D MAIL ID:dsheeba025@gmail.com

PRITHINGA DEVI.R MAIL ID:prithingadevi16@gmail.com

2.PROJECT OVERVIEW:

PURPOSE:

- The purpose of "THE COOKBOOK:YOUR VIRTUAL KITCHEN ASSISTANT "project is to provide uses with an interactive platform to explore, organize and prepare recipes easily.
- It helps uses discover new dishes, learn step-y-step cooking instructions, manage ingredients and get personalized suggestions.
- By acting as a virtual kitchen assistant, the project aims to simplify the cooking process, reduce food waste through smart ingredient usage and make cooking Enjoyable for beginners as well as experienced chef's

FEATURES:

- ➤ **Recipe Collection** & **Search** Users can browse and search recipes by cuisine, category, or ingredients.
- > Step-by-Step Cooking Guidance Interactive instructions with text, images, or voice support for easy cooking.
- Virtual Kitchen Assistant Provides cooking tips, ingredient substitutes, and guides users while preparing dishes.
- ➤ **Personalized Recommendations** Suggests recipes based on user preferences, dietary needs, or available ingredients.
- Shopping List Generator Creates a grocery list automatically from selected recipes.
- Favorites & Bookmarking Allows users to save and organize their favorite recipes.

3.ARCHITECTURE:

Component Structure:

- Recipe Component Displays recipes with ingredients and steps.
- Search Component Filters recipes by keywords or ingredients.
- Meal Planner Component Allows weekly meal planning.
- Shopping List Component Generates grocery lists from recipes.
- Assistant Component Provides interactive guidance and suggestions.

State Management:

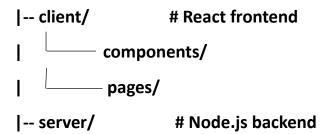
- ♣ Global state managed using Context API/Redux for user data, saved recipes, and preferences.
- Local state for form handling and UI components.

4.SETUP INSTRUCTIONS:

- Prerequisites:
 - Node.js
 - React.js
 - npm
- Installation:
 - 1. Clone the repository.
 - 2. Run npm install to install dependencies.
 - 3. Set up environment variables.
 - 4. Start the development server using npm start.

5.FOLDER STRUCTURE:

Virtual-Kitchen-Assistant/





6.RUNNING THE APPLICATION:

Frontend

- cd client
- npm start

Backend

- cd server
- npm start
- Access: Visit https:// localhost:3000

7.API DOCUMENTATION:

User:

- POST /api/user/register
- POST /api/user/login

Recipes:

- GET /api/recipes/suggest
- GET /api/recipes/:id

Pantry Management:

- POST /api/pantry/add
- GET /api/pantry/list

Cooking Assistance:

- POST /api/cooking/guide
- GET /api/cooking/tips

8.AUTHENTICATION:

- When a user logs in, the server gives a token, this is used to access private pages.
- Middleware checks the token and only allows valid users(pantry , saved recipes)

9.USER INTERFACE:

- Landing Page (app introduction with the recipes)
- Freelancer Dashboard (showing the different category meals.
- Admin Panel for recipe, contents and manage it.

10.TESTING:

- Manual Testing is done at different stages of the recipes
- > Tools used: Postman, Chrome Dev Tools, Jest for unit testing

11. SCREENSHOTS OR DEMO:

✓ Screenshots or a short demo video can be provide a visual representation of the application's functionality.

12.KNOWN ISSUES:

✓ A section detailing known issues or bugs would be helpful.

13.FUTURE ENHANCEMENTS:

- ❖ Al-based meal planning with health goals.
- Smart kitchen appliance integration.
- ❖ Augmented Reality (AR) guided cooking
- Community recipe sharing platform.
- Learning more recipes in easy way.

