**Frontend Development with React.js**

**Project Documentation**

1. **Introduction**

* **Project Title**: Cookbook- Your Virtual Kitchen Assistant.
* **Team Members**: **ANEES SAIDHANI A**(Team Lead), **VARSHINI M** (Team Member), **ENIYA M** (Team Member)

1. **Project Overview**

**Purpose**:

* The purpose of the Cookbook Virtual Assistant is to provide users with a smart, personalized cooking companion that simplifies every aspect of meal preparation.
* From discovering recipes based on available ingredients to offering step-by-step cooking guidance, the assistant helps users cook with confidence and ease.
* The Cookbook Virtual Assistant enhances your kitchen experience by making meal planning, preparation, and execution seamless and tailored to your needs.

**Features**

* The Cookbook Virtual Assistant comes packed with a range of features designed to make cooking easier, faster, and more enjoyable.
* It offers personalized recipe recommendations based on your dietary preferences, available ingredients, and cooking skill level.
* With interactive, step-by-step instructions, it guides you through each recipe, including helpful tips and cooking techniques.

1. **Architecture**

**Component Structure**:

* Natural Language Processing (NLP) Module
* Recipe Management System
* User Profile & Personalization
* Shopping List & Pantry Manager
* Meal Planning Module
* Recommendation Engine
* Cloud Backend & Data Management

**State Management**:

* **GLOBAL STATE:** The **global state** of the Cookbook Virtual Assistant refers to the persistent, shared data that remains consistent across all user sessions, devices, and features.
* **LOCAL STATE:** The **local state** of the Cookbook Virtual Assistant refers to the temporary, session-specific data that is only relevant during an active interaction.

**Routing**:

/src

/pages

HomePage.jsx

RecipeList.jsx

RecipeDetail.jsx

ShoppingList.jsx

MealPlanner.jsx

NotFound.jsx

1. **Folder Structure**

/cookbook-assistant

│

├── /public

│ └── index.html

│

├── /src

│ ├── /assets # Images, icons, styles

│ ├── /components # Reusable UI components (e.g., Button, Header, RecipeCard)

│ ├── /pages # Route-level components (e.g., RecipeList, ShoppingList)

│ ├── /features # Feature-specific logic & UI

│ │ ├── /recipes

│ │ │ ├── RecipeList.jsx

│ │ │ ├── RecipeDetail.jsx

│ │ │ └── recipeSlice.js # (if using Redux Toolkit or similar)

│ │ ├── /shoppingList

│ │ │ ├── ShoppingList.jsx

│ │ │ └── shoppingListSlice.js

│ │ ├── /mealPlanner

│ │ │ ├── MealPlanner.jsx

│ │ │ └── mealPlannerSlice.js

│ │ └── /user

│ │ ├── Profile.jsx

│ │ └── userSlice.js

│ │

│ ├── /routes # All app routes using React Router v6

│ │ └── AppRoutes.jsx

│ │

│ ├── /services # API services to interact with the backend

│ │ ├── api.js # Axios or Fetch instance

│ │ ├── recipeService.js

│ │ ├── shoppingListService.js

│ │ └── authService.js

│ │

│ ├── /state # Global state management (Redux, Zustand, Context)

│ │ └── store.js

│ │

│ ├── /hooks # Custom React hooks (e.g., useFetchRecipes)

│

│ ├── /utils # Helper functions (e.g., formatTime, calculateCalories)

│

│ ├── App.jsx # Main App component

│ └── main.jsx # Entry point for React (Vite) or index.js (CRA)

│

├── /backend (optional) # Backend folder if full-stack in one repo

│ └── /api

│ └── recipes.js

│

├── package.json

└── README.md

1. **Running the Application**

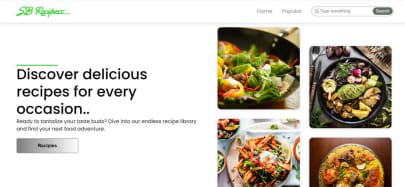
Node.js (v16+)

npm or yarn installed

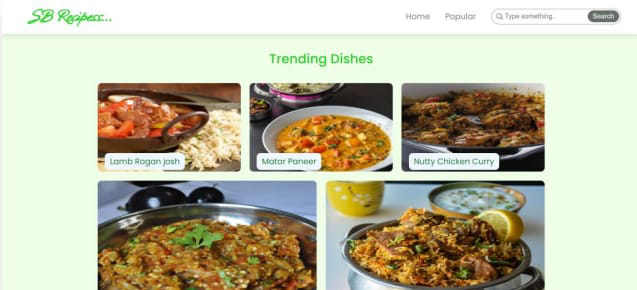
1. **User Interface**

Screenshots from the demo:

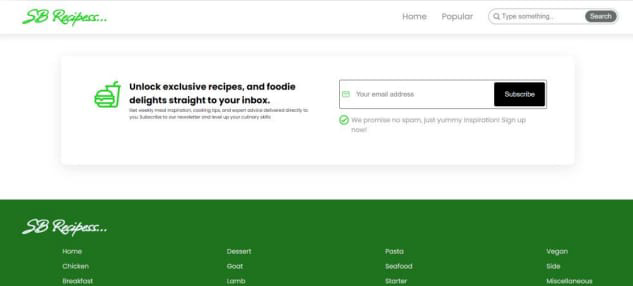
* frame\_0.jpg → Home page.



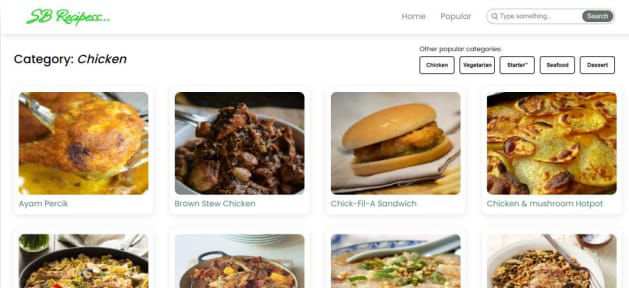
* frame\_150.jpg → Search page.



* frame\_300.jpg → Subscribe page.



* frame\_450.jpg → Category page.



1. **Styling**

* Primary: #6B8E23 (Olive Green)
* Secondary: #FFA726 (Warm Orange)
* Background: #FFF8E1 (Soft Cream)
* Text: #333333 (Dark Gray for readability)

1. **Testing**

**Strategy**

* Jest for logic and component testing
* React Testing Library for testing React components

1. **Screenshots or Demo**

Demo video: rhythmic tunes.mp4

Extracted screenshots: frame\_0.jpg, frame\_150.jpg, frame\_300.jpg, frame\_450.jpg, frame\_600.jpg

1. **Known Issues**

* Inaccurate Ingredient Recognition
* Voice Command Misinterpretation
* Recipe Filter Inconsistency
* Slow Loading Times
* Limited Offline Functionality
* Accessibility Gaps
* State Loss on Refresh
* Third-Party API Limitations

1. **Future Enhancements**

* Offline Mode Support
* Voice Navigation Improvements
* Smart Ingredient Substitution
* AI-Powered Meal Recommendations
* Nutritional Analysis Integration
* Barcode Scanner for Pantry Tracking
* Recipe Step-by-Step with Timers
* Social Recipe Sharing
* Enhanced Accessibility Features
* Meal Plan Export & Sync
* Multilingual Support
* Smart Home Integration