Frontend Development with React.js Project Documentation format.

CookBook: Your Virtual Kitchen Assistant

# 1. Introduction

**Project Title:** Cookbook: Your Virtual Kitchen Assistant

**Team ID :** NM2025TMID30203

**Team Members:**

Team Leader Name: SANTHIYA M & santhiyamurugesanpvt@gmail.com

Team Member Name: ROSHINI R & roshiniroshiniravi84@gmail.com

Team Member Name: SARAL NILA M & keerthimurugesan1991@gmail.com

Team Member Name: SRI DHANA JANANI M & srimathipriya9788@gmail.com

**2. Project Overview Purpose:**

The Cookbook application is designed to simplify meal preparation by offering a virtual kitchen assistant.

Users can browse, save, and manage their favorite recipes with detailed instructions and ingredient lists.

**Features:**

* Extensive Recipe Library with Search & Filter Options
* Step-by-Step Cooking Instructions
* Bookmark Feature for Favorite Recipes
* Light/Dark Mode for Enhanced Visual Comfort
* Responsive Design for Optimal Viewing on All Devices
* Ingredient Checklist for Easy Shopping and Preparation

# 3. Architecture

**Component Structure:**

**App.js** — Root component managing layout and routes

**Header.js** — Navigation bar with search functionality

**RecipeList.js** — Displays categorized recipe cards

**RecipeCard.js** — Individual card displaying recipe details

**RecipeDetails.js** — Detailed view of selected recipe

**Footer.js** — Provides additional resources and links

**State Management:** Context API

**Routing:** React Router

**4. Setup Instructions Prerequisites:**

* Node.js (v18 or higher)
* npm or yarn

**Installation:**

1. Clone the Repository:

`git clone https://github.com/Santhiya1603/cook-book-app.git` `cd cookbook-app`

1. Install Dependencies: `npm install`
2. Create .env file:

`REACT\_APP\_RECIPE\_API\_KEY=your\_api\_key\_here`

1. Run the Application: `npm start`

# 5. Folder Structure

/src

├── /components

│ ├── Header.js

│ ├── RecipeList.js

│ ├── RecipeCard.js

│ ├── RecipeDetails.js

│ └── Footer.js

├── /pages

│ ├── Home.js

│ ├── Categories.js

│ └── Bookmarks.js

├── /assets

│ ├── images

│ ├── icons

│ └── styles

├── /utils

│ ├── api.js

│ ├── helpers.js

│ └── constants.js

├── App.js

├── index.js

# 6. Running the Application

**Frontend Server Command:**`npm start`

**7. Component Documentation Key Components:**

**Header:** Provides navigation and search functionality

**RecipeList:** Displays available recipes categorized by type

**RecipeCard:** Displays individual recipe details **RecipeDetails**: Offers step-by-step instructions

**Reusable Components:**

**Button Component**: Used for actions like 'View Recipe' or 'Bookmark'

**Modal Component:** Displays detailed recipe information

# 8. State Management

**Global State:** Managed via Context API

**Local State:** Managed using React’s `useState()`

# 9. User Interface

* Clean and intuitive layout optimized for smooth navigation
* Mobile-first design for seamless browsing
* Elegant Light/Dark mode toggle for improved accessibility

# 10. Styling

**CSS Frameworks/Libraries:** Tailwind CSS

**Theming:** Integrated light/dark mode

# 11. Testing

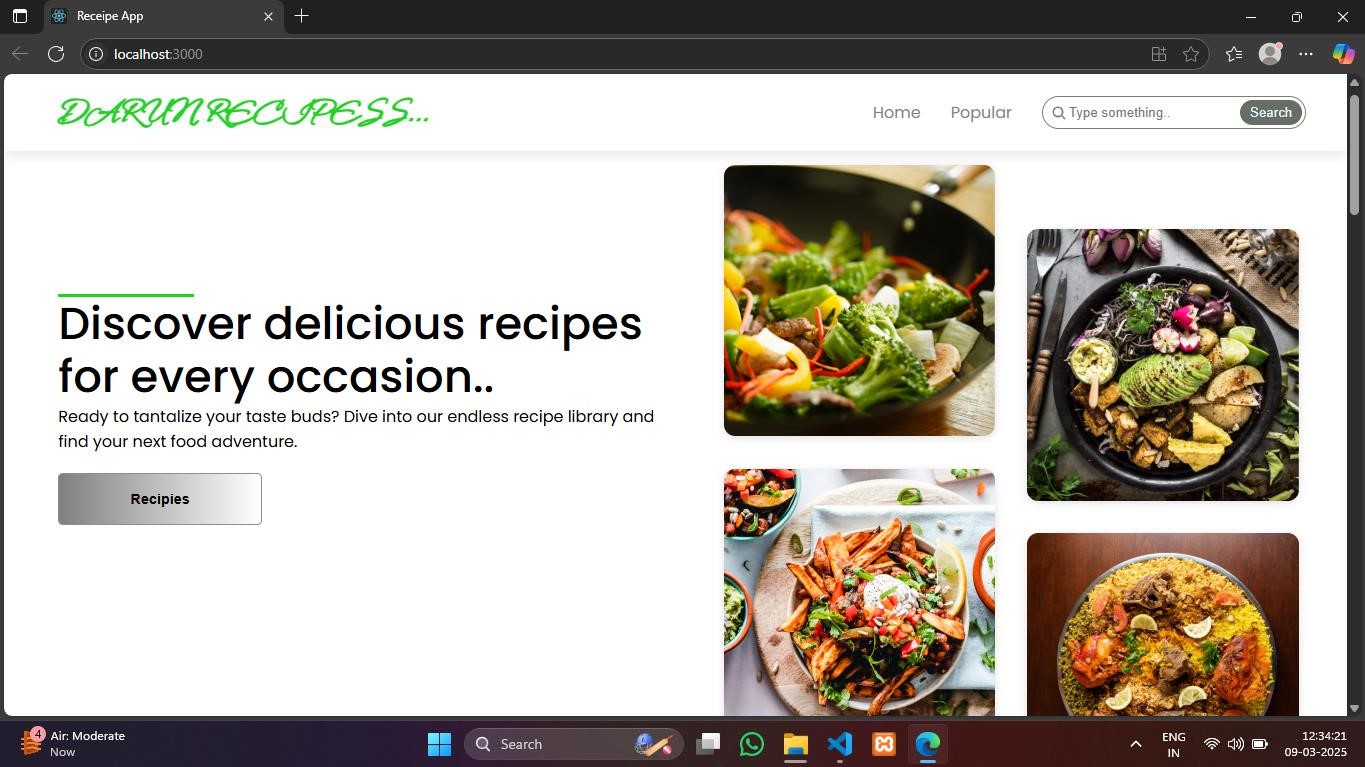
**Testing Strategy:**

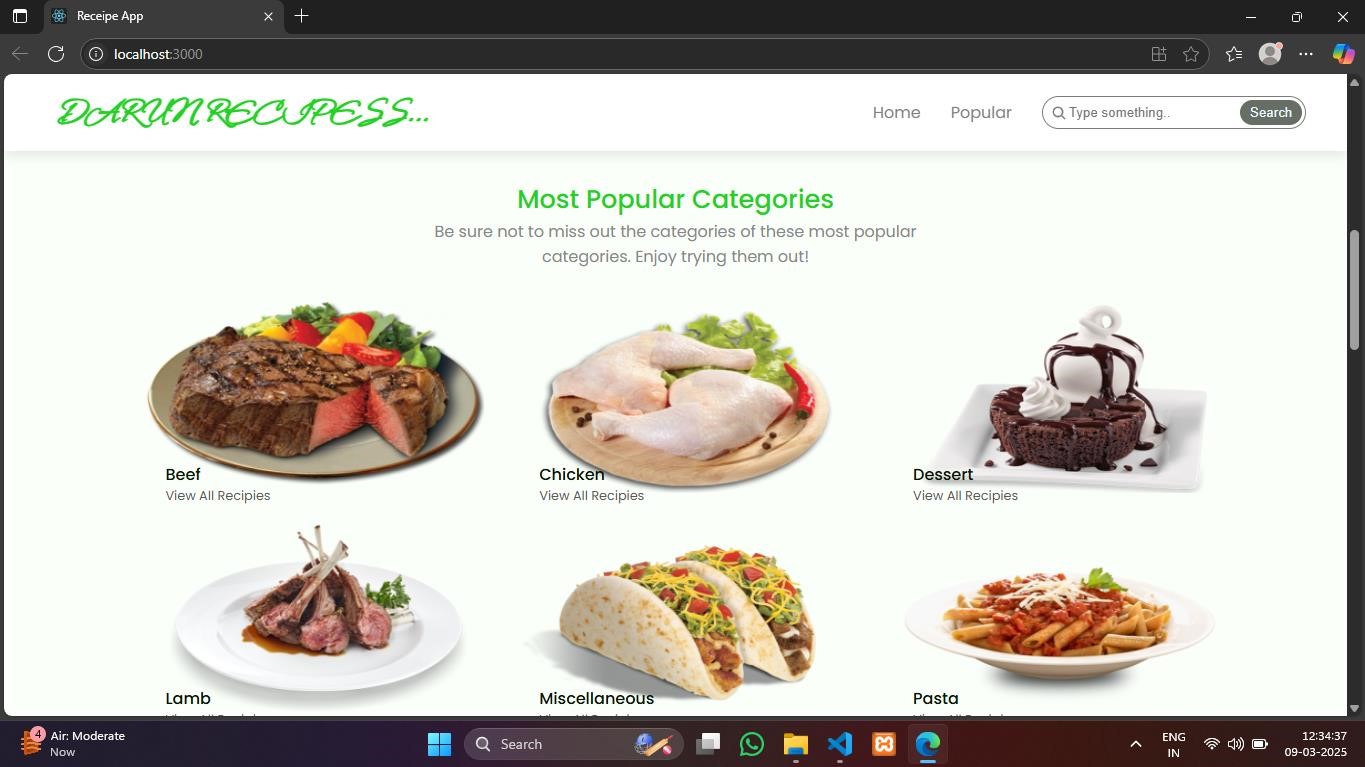
Jest for unit testing

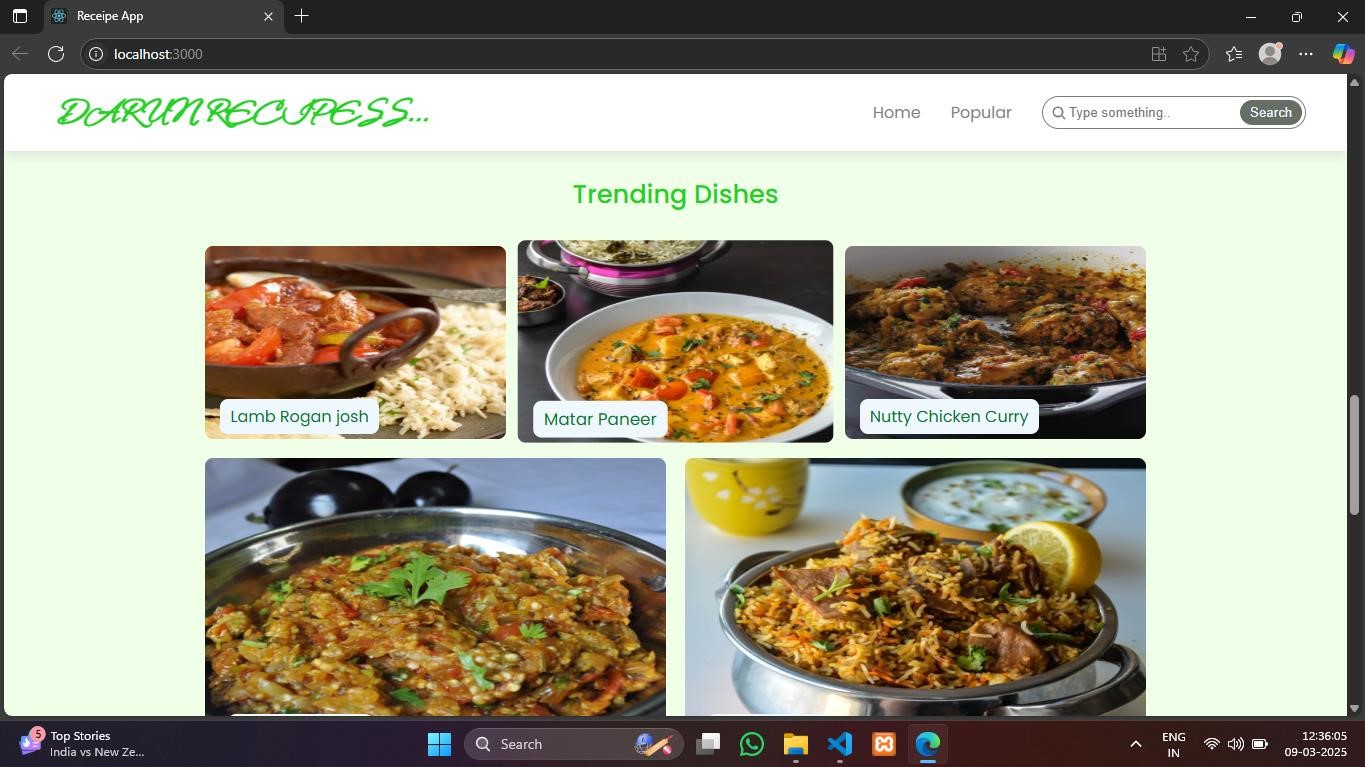
React Testing Library for component testing

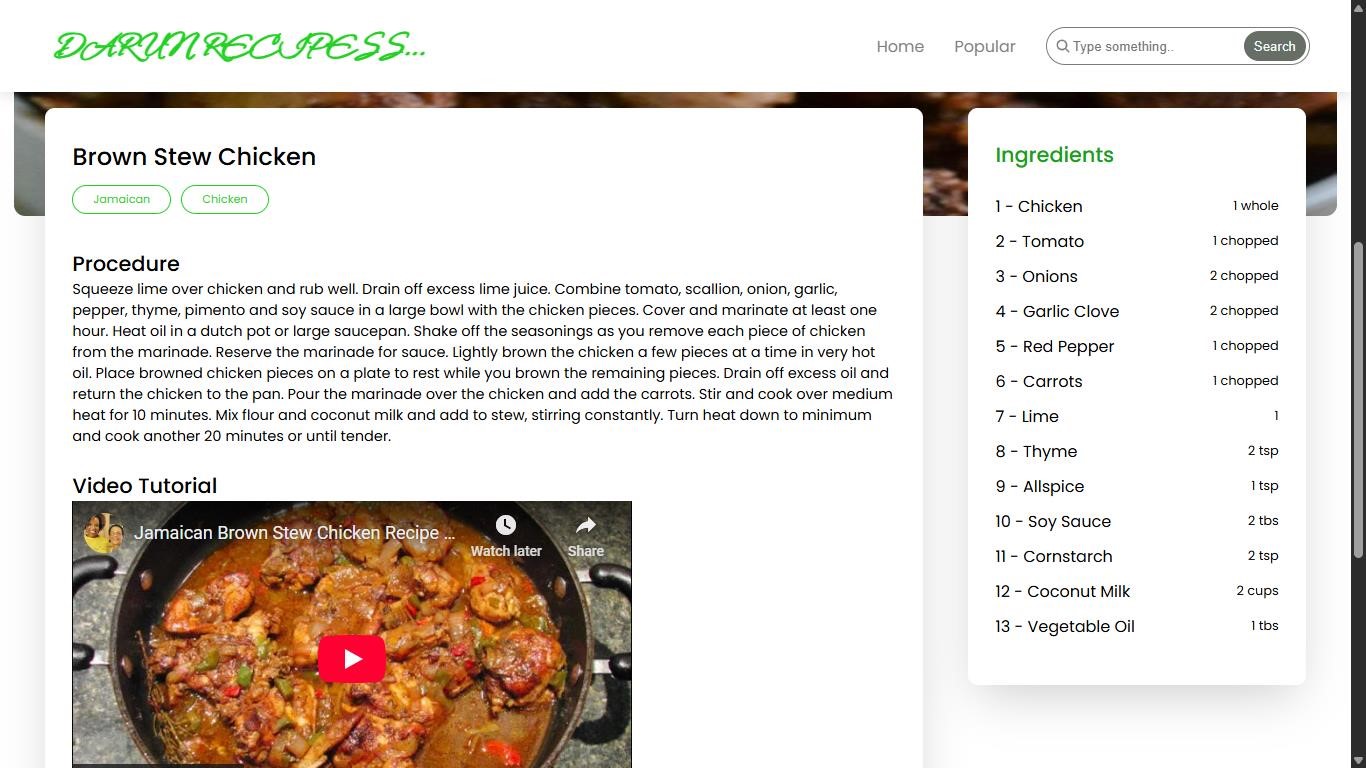
**Code Coverage:** Ensured high test coverage with Jest reports

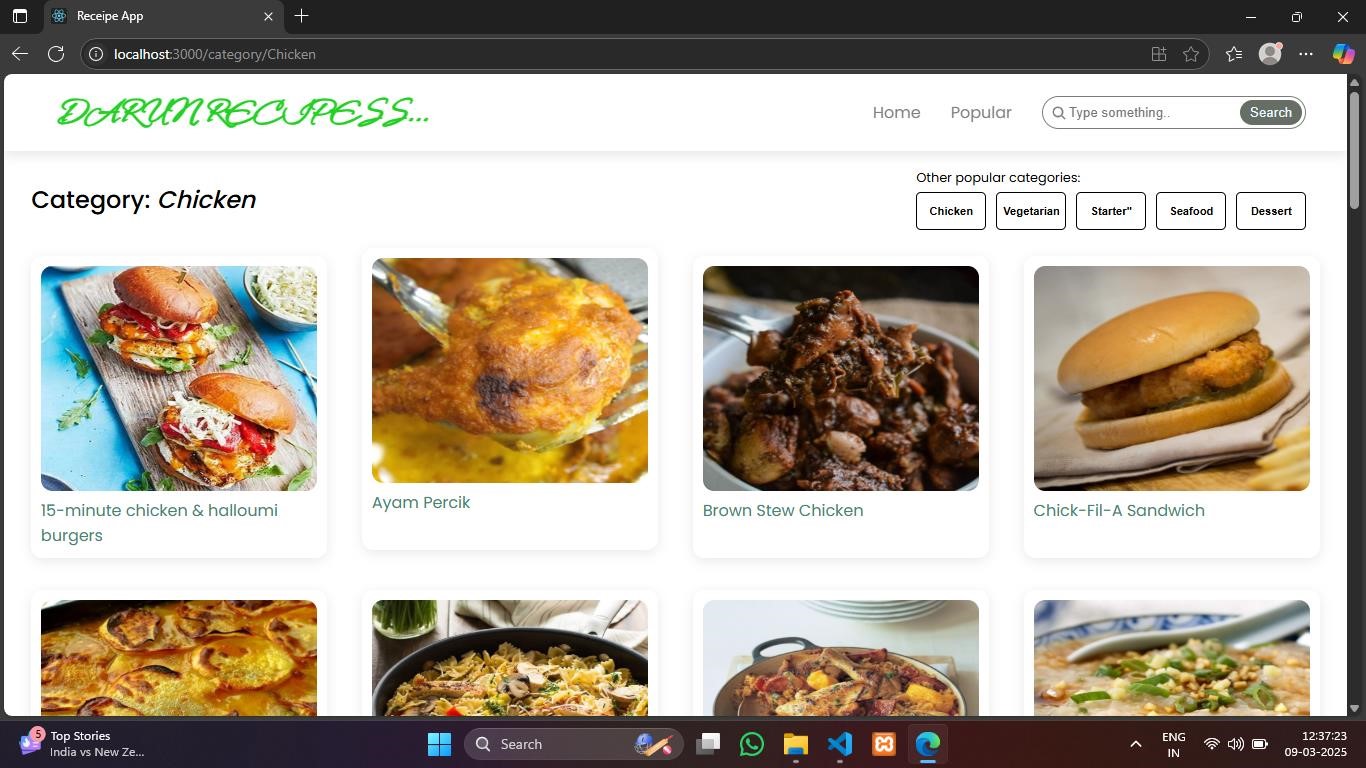
**12. Screenshots or Demo Screenshots:**











**Live Demo Link:**

# 13. Known Issues

* Certain API endpoints may experience delays during peak traffic
* Ingredient checklist feature may require further optimization

# 14. Future Enhancements

* AI-Based Recipe Suggestions
* User Authentication for personalized recipe collections
* Integration with Grocery Shopping Platforms
* Offline Mode for accessing saved recipes