**Cookbook Virtual Kitchen Assistant**

1. **Introduction:**

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

**• Team Leader: Asmaa begum B E**

**• Team Members:**

* + - * **Karthik S**
      * **Koushik A**
      * **Arshiya S I**

1. **Project Overview:**

* **Purpose:  
   The Virtual Kitchen Assistant is a React-based web application designed to help users manage their cooking activities more efficiently. It allows users to browse recipes, organize ingredients, create shopping lists, and get cooking instructions in a simple, interactive interface.**

1. **Features:**

**Recipe browsing and search**

* 1. **Ingredient and inventory management**
  2. **Shopping list creation**
  3. **Step-by-step cooking assistant**
  4. **Interactive, responsive user interface**

1. **Architecture:**

**The app is built entirely with React.js, keeping the code modular and easy to maintain.**

* **Frontend: React (Create React App boilerplate)**
* **State Management: Context API + React Hooks**
* **Routing: React Router**
* **Styling: Material UI components and custom CSS modules**
* **Backend: None yet (data is handled on the client side)**

**This lightweight architecture makes it easy to extend with APIs or backend services in the future.**

1. **Setup Instructions:**

**Getting started is simple:**

**Prerequisites**

* **Node.js (v14 or above)**
* **npm or yarn installed**

**Steps**

**# Clone the repository**

**git clone https://github.com/asmacrazy920-netizen/your-virtual-kitchen-assistant.git**

**# Move into the project folder**

**cd your-virtual-kitchen-assistant**

**# Install dependencies**

**npm install**

**# Start the development server**

**npm start**

**Now open** [**http://localhost:3000**](http://localhost:3000) **in your browser and you’re ready to explore.**

1. **Folder Structure:**

**Here’s a quick look at how the project is organized:**

**your-virtual-kitchen-assistant/**

**├── public/ # Static files (HTML, icons, etc.)**

**├── src/**

**│ ├── components/ # Reusable UI components (Navbar, RecipeCard, etc.)**

**│ ├── pages/ # Main pages (Home, Recipes, ShoppingList)**

**│ ├── context/ # Global state management (Context API)**

**│ ├── assets/ # Images, icons, mock data**

**│ ├── App.js # Root app component**

**│ ├── index.js # App entry point**

**├── package.json**

**└── README.md**

**This structure makes it clear where everything belongs—components, pages, state, and assets are all neatly separated.**

1. **Running the Application:**

**To run the app locally:**

**npm start**

**It will open in your browser at** [**http://localhost:3000**](http://localhost:3000)**.**

1. **Components in Action:**

**Some of the main building blocks include:**

* **Navbar – Top navigation across Home, Recipes, and Shopping List.**
* **RecipeCard – A reusable card that shows a recipe’s preview (image, title, and a quick link).**
* **RecipeList – Displays multiple RecipeCards in a grid layout.**
* **ShoppingList – Interactive list where users can add or remove grocery items.**
* **CookingSteps – Walks users through the recipe one step at a time.**

1. **State Management:**

**We’ve kept state management simple but powerful:**

* **Global State: Managed with the Context API (e.g., for shopping list and selected recipes).**
* **Local State: Handled via React Hooks (useState, useEffect) within individual components.**

**This combination gives the app flexibility without overcomplicating things.**

**10.User Interface:**

**The interface focuses on clarity and usability:**

* **A clean landing page that introduces the app.**
* **A recipes page with search and browsing options.**
* **A recipe details page showing ingredients and cooking steps.**
* **A shopping list page to plan your groceries.**
* **We used Material UI for a polished look, alongside custom CSS modules for fine control over styling.**

1. **Testing:**

**Testing ensures reliability as the app grows.**

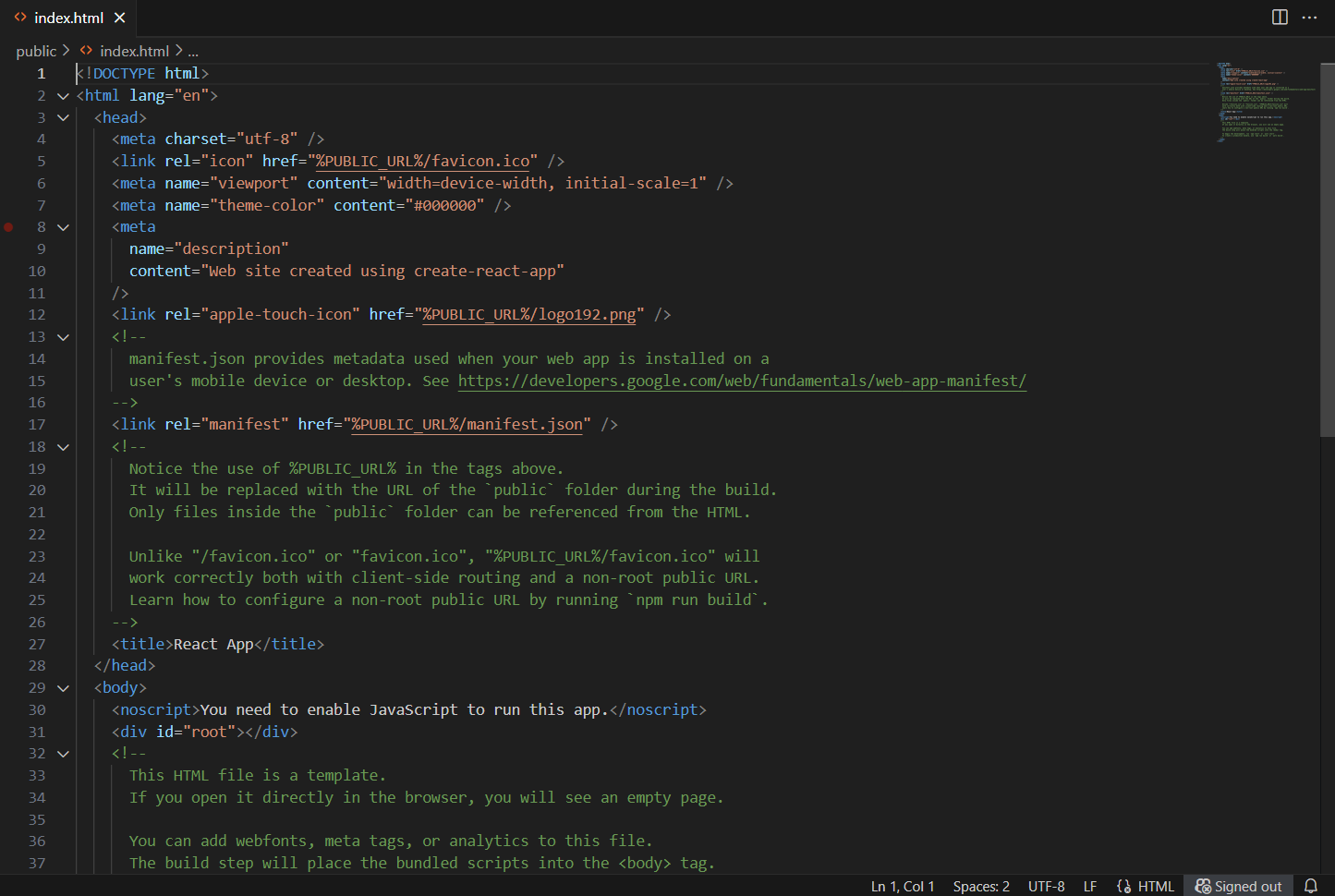
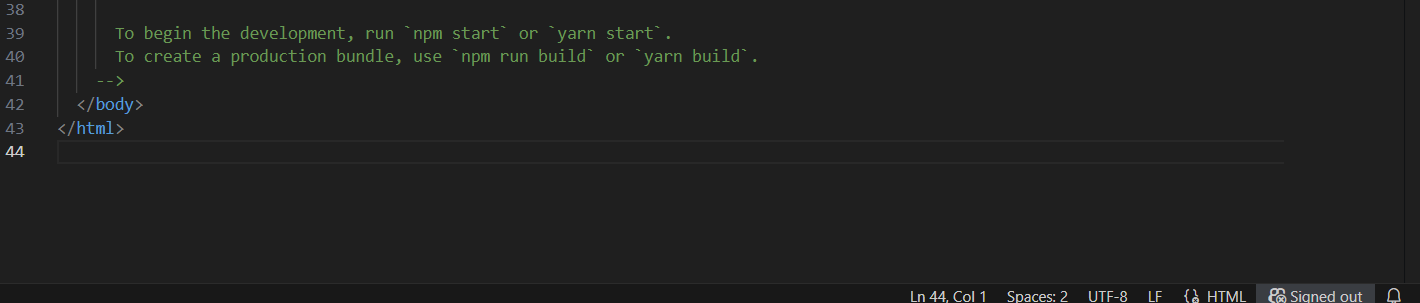
* **Tools: Jest and React Testing Library**
* **Strategy: Component-level testing, snapshot testing, and mock data testing**

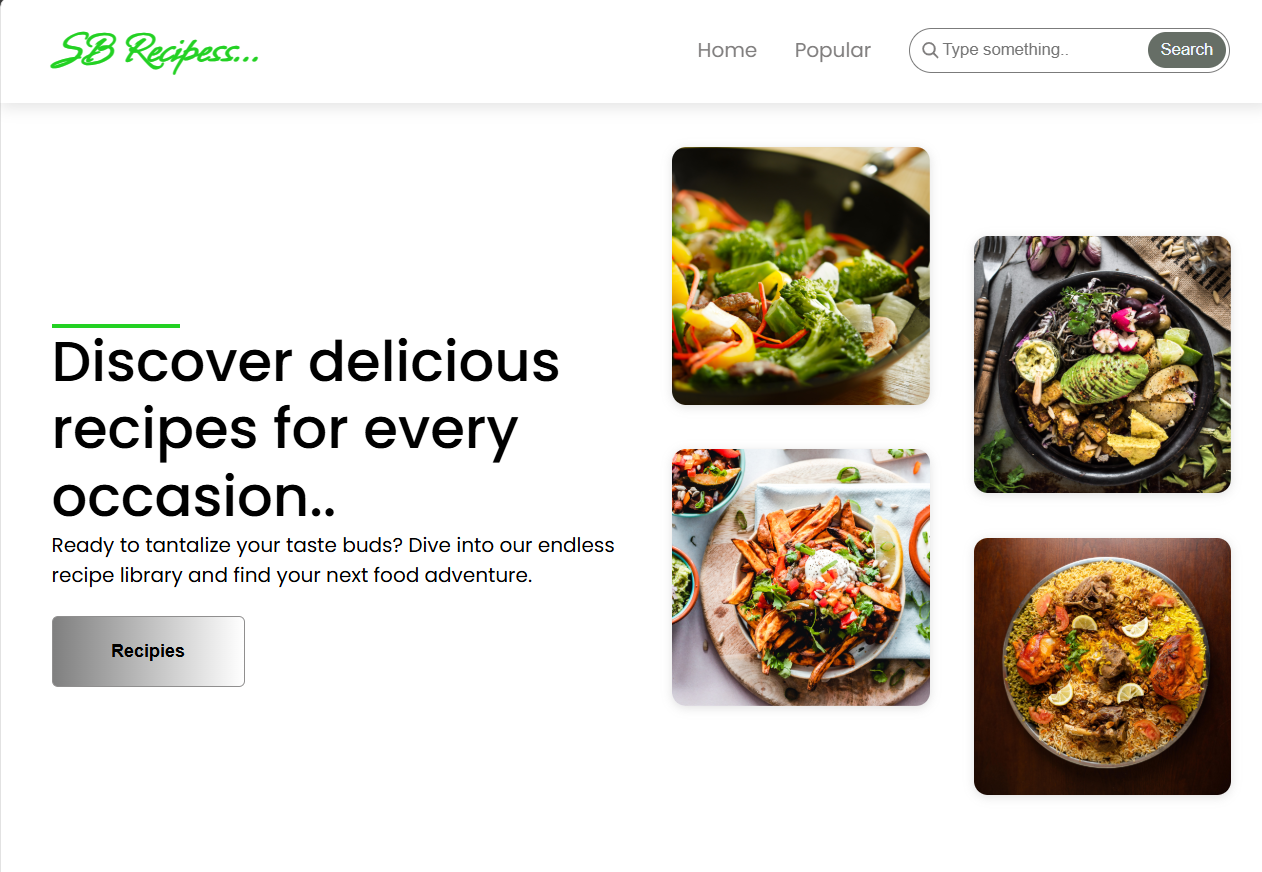
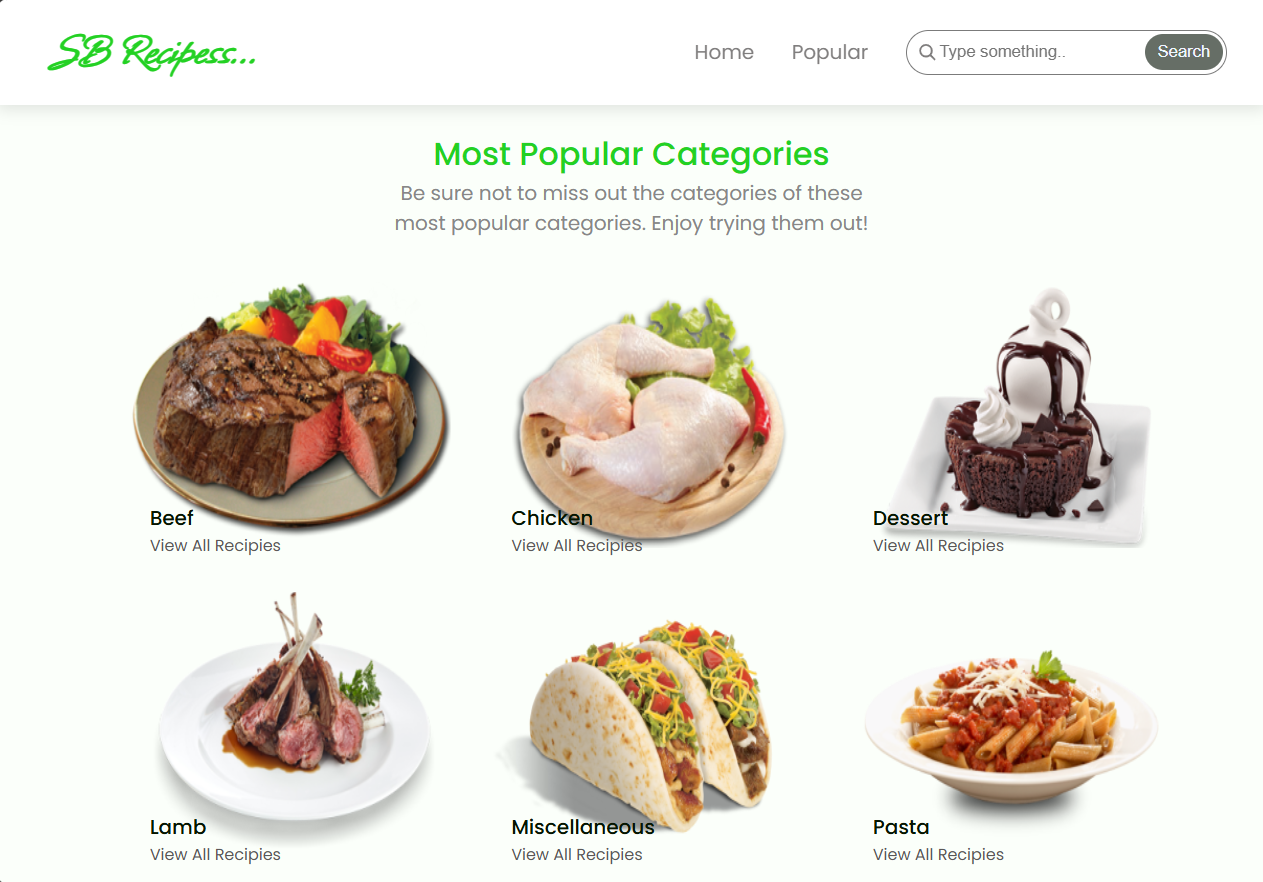
1. **Known Issues:**

* **Currently no backend integration, so data isn’t persisted across sessions.**
* **Voice assistant feature is still in the planning stage.**

1. **Future Enhancements:**

**We’re excited to keep building on this foundation. Potential additions include:**

* **Integration with external recipe APIs (e.g., Spoonacular).**
* **user accounts to save favorite recipes and lists.**
* **Voice-based assistant for hands-free cooking.**
* **Dark mode and custom themes.**
*  **A React Native mobile version.**
* 

* + Output:

