**1. Introduction**

**Project Title**: SB Recipess

**Team Members**:

1. C.Pradhap
2. M.Thangavael
3. S.Kishore Kumar
4. S.Mythreyan

## 2. Project Overview

* **Purpose**:  
  The **SB Recipess** project is a frontend web application designed to provide users with recipes across various categories. It allows users to search, browse, and explore recipes, while also giving them an option to subscribe for exclusive updates..
* **Features**:
  + 🏠 Homepage with navigation bar, search, and category listings.
  + 🔎 Search functionality for recipes.
  + 📧 Newsletter subscription form.
  + 📂 Categorized recipes (Chicken, Vegan, Pasta, Dessert, etc.).
  + 🔐 User authentication (Sign in / Sign up).
  + 📱 Responsive design for mobile and desktop.

**3. Architecture**

* **Component Structure**:
* App.js
* ├── Navbar
* ├── SearchBar
* ├── Newsletter
* ├── Footer
* ├── Pages
* │ ├── Home
* │ ├── Popular
* │ └── CategoryPages
* **State Management**:
  + Local state is managed using **React useState**.
  + For global state (user authentication, search results), **Context API** is used.
* **Routing**:  
  Implemented with **react-router-dom**:
  + / → Home
  + /popular → Popular Recipes
  + /category/:name → Recipe categories
  + /signin → Login page

## 4. Setup Instructions

* **Prerequisites**:
  + Node.js (>=16.x)
  + npm or yarn
* **Installation**:

# Clone repository

git clone https://github.com/your-repo/sb-recipess.git

# Navigate to project folder

cd sb-recipess

# Install dependencies

npm install

# Start development server

npm start

## 5. Folder Structure

sb-recipess/

│── public/

│── src/

│ ├── assets/ # Images, icons, logos

│ ├── components/ # Reusable components (Navbar, Footer, Newsletter)

│ ├── pages/ # Different pages (Home, Popular, Categories)

│ ├── context/ # Context API for global state

│ ├── utils/ # Helper functions, API calls

│ ├── App.js # Main App component

│ └── index.js # Entry point

│── package.json

* **Utilities**:
  + api.js → Helper for fetching recipe data.
  + auth.js → Helper for login/logout.

## 6. Running the Application

* **Start Frontend**:
* npm start

Runs the app in development mode → <http://localhost:3000>.

## 7. Component Documentation

* **Navbar**
  + Purpose: Provides navigation and search.
  + Props: links, onSearch.
* **Newsletter**
  + Purpose: Collects user emails for subscription.
  + Props: onSubscribe(email).
* **Footer**
  + Purpose: Displays categories and site info.
  + Props: None (static).
* **Reusable Components**:
  + Button → Used across app.
  + InputField → Used in forms.

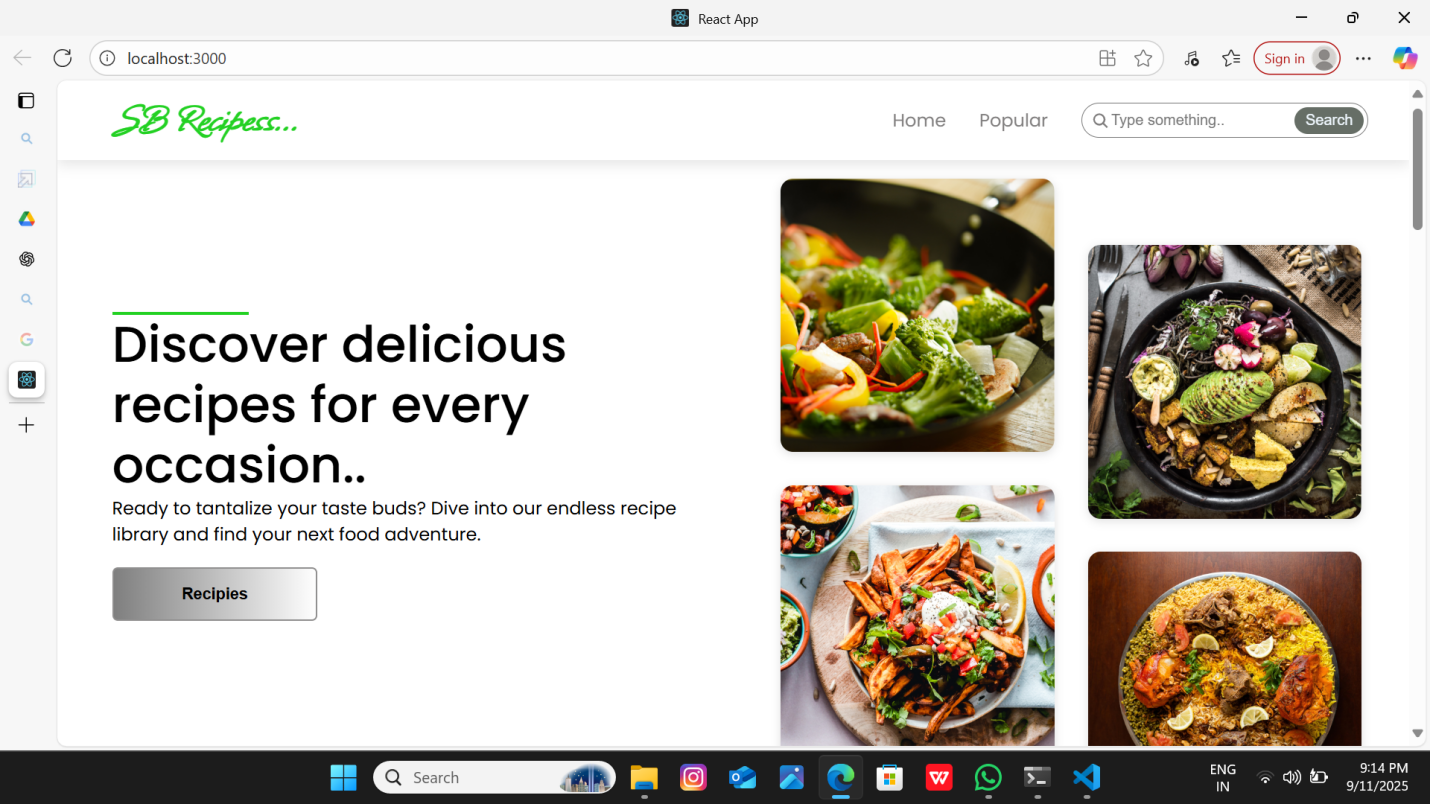
## 8. State Management

* **Global State** (Context API):
  + Authentication (login/logout status).
  + Search results shared across pages.
* **Local State**:
  + Newsletter form email input.
  + Navbar search query.

## 9. User Interface

### Screenshots

✅ **Home Page**



## 10. Styling

* **CSS Frameworks/Libraries**:
  + TailwindCSS (utility-first CSS framework).
* **Theming**:
  + Custom color palette (green theme for food/health).
  + Consistent font styling for headings and text.