

COOKBOOK : YOUR VIRTUAL KITCHEN ASSISTANT

1. Induction

- **Project Title: CookBook: Your Virtual Kitchen Assistant**
- **Team ID: NM2025TMID35090**
- **Team Leader: Mythili Rakkammal P**
- **(suryaravir9@gmail.com)**
- **Team Members:**

1.Syed ali fathima M (syedalifathima220207@gmail.com)

2.Meharaj fathima S (mehanafil@gmail.com)

3.Bruntha T (pruntha6374431900@gmail.com)

2.project overview

Purpose

CookBook is a web-based application designed to help users explore, organize, and manage recipes. It allows browsing and searching recipes, creating personalized collections, and offering step-by-step cooking assistance for an enhanced kitchen experience.

Key Features:

- ✓ Browse and search recipes by cuisine, ingredients, or categories
- ✓ Create and manage personalized recipe collections
- ✓ Mark favorite recipes for quick access
- ✓ Step-by-step cooking mode with timers and instructions
- ✓ Modern, responsive interface for web and mobile

3.Architecture

- Frontend: React.js with Bootstrap for UI styling
- Backend: JSON Server (simulated REST API)
- Database: db.json (recipes, ingredients, cuisines, collections)

4.Technologies Used

- React.js
- JSON Server
- Visual Studio Code
- Git

5.Installation & Setup

- Navigate to the project folder : `cd CookBook`
- Install dependencies : `npm install`
- Start the React app : `npm run dev`

- In a separate terminal, start JSON Server : `npx json-server --watch db.json --port 3000`

6.Folder Structure

CookBook/ | | --

code/ db/ # Database

config

| | -- node _ modules/ #

Dependencies

| | -- public/ # Static files

| | -- src/ # Source code

| | -- assets/ # Media (images, recipes)

| | -- Components/ # React components

| | -- App.css

| | -- App.jsx

| | -- index.css

| | -- main.jsx

| | -- package.json

| | -- vite.config.js

| | -- README.md

7.Running the Application

- ❖ Frontend : `npm run dev`
- ❖ Access via: `http://localhost:5173`

8.API Documentation

Recipes

- GET `/api/recipes` → All recipes
- GET `/api/recipes/:id` → Recipe by ID
- POST `/api/recipes/create` → Add recipe
- PUT `/api/recipes/:id/update` → Update recipe
- DELETE `/api/recipes/:id/delete` → Delete recipe

Collections & Favorites

- Similar CRUD endpoints for /api/collections and /api/favorites

9.Authentication

- Currently not implemented (planned as a future enhancement).

10.User Interface

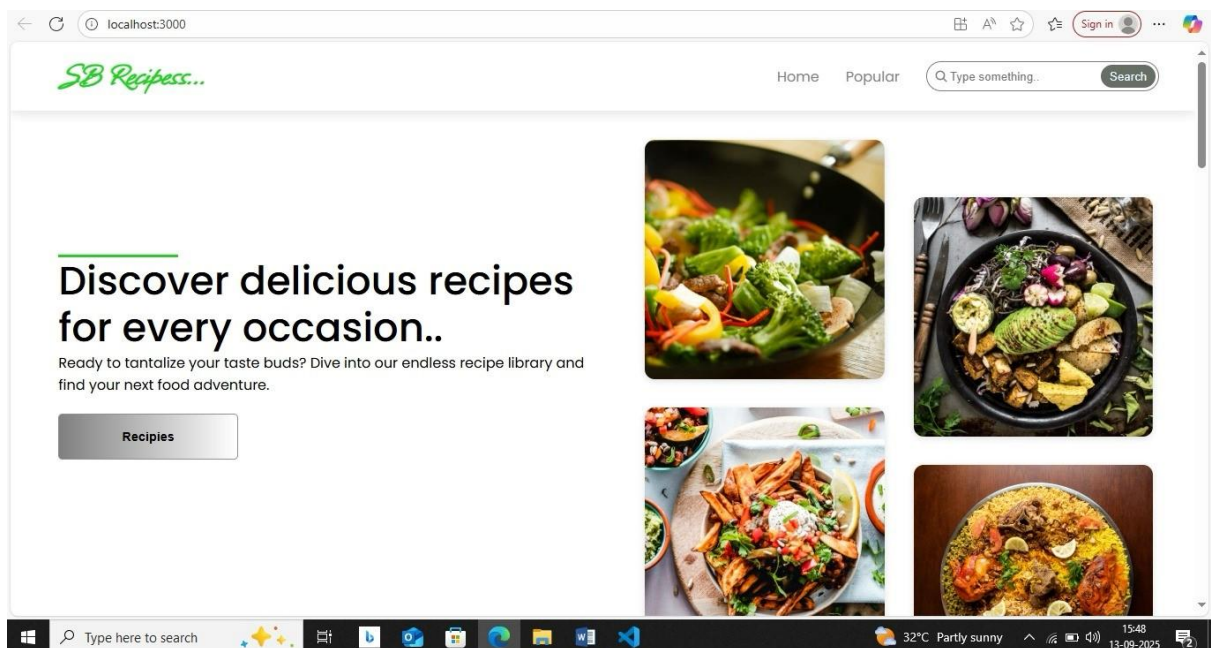
- Landing Page: Explore all recipes
- Sidebar Navigation: Home, Favorites, Collections
- Recipe Cards: Title, cuisine, ingredients, cook time
- Cooking Mode: Step-by-step guide with timers

11. Testing

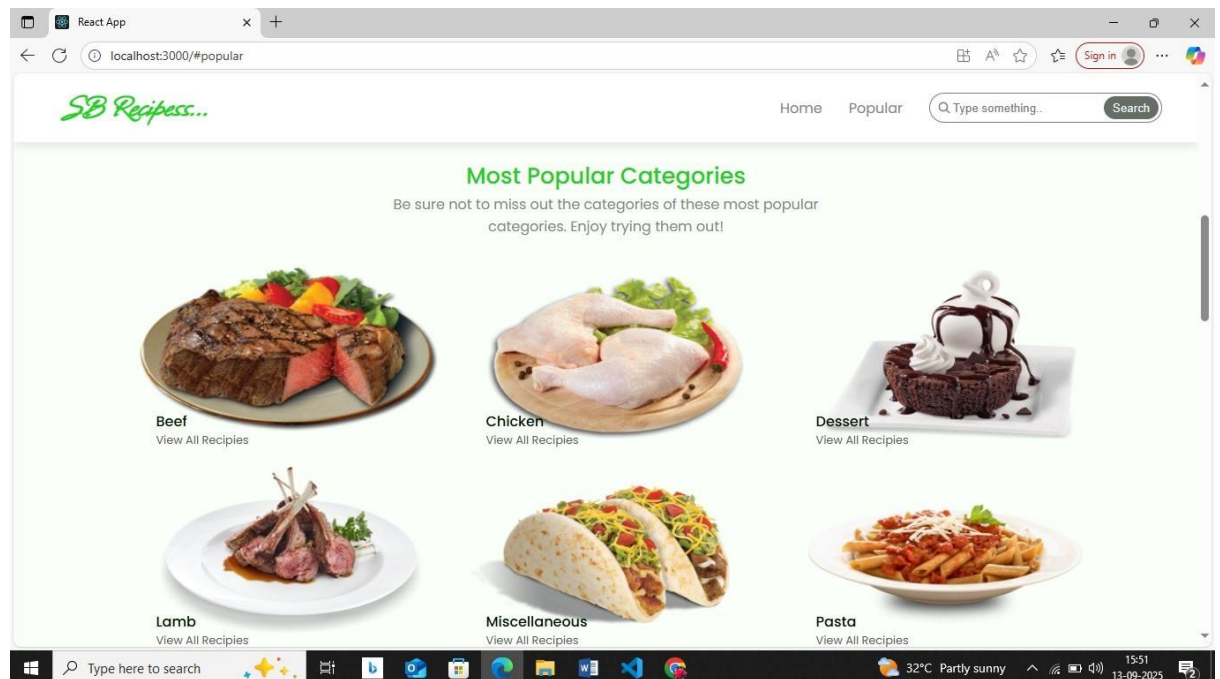
- Method: Manual testing
- Verified: Recipe listing, search, cooking mod | | --Tools Used: Chrome DevTools, JSON Server

12.Screenshots

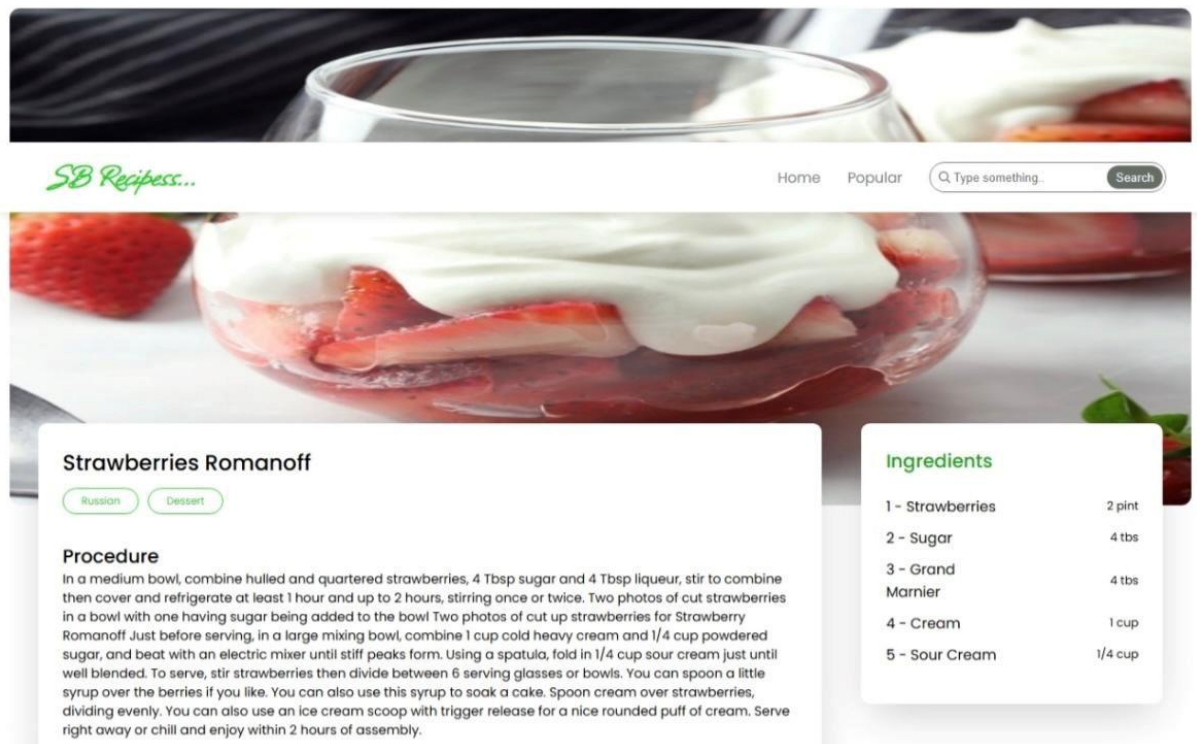
❖ Dashboard view



❖ Landing Page



❖ Recipe Details Page



13. Known Issues

- No login/authentication system
- Limited to static JSON data
- JSON Server must run separately

14. Future Enhancements

- User authentication (JWT)
- Cloud-based recipe storage
- AI-powered meal recommendations
- Grocery list generation
- Mobile app version (React Native)