Project Documentation

Cook Book

# Introduction

## Project Title: Cook Book

* + **Team ID :** NM2025TMID47733
  + **Team Leader: M.Sahana & [sahananila2316@gmail.com](mailto:sahananila2316@gmail.com)**

## Team Members:

* + - S.Roopadharshini & [rupadharshini8207@gmail.com](mailto:rupadharshini8207@gmail.com)
    - T.Shalini & [babyshalu684@gmail.com](mailto:babyshalu684@gmail.com)
    - B.Sharmila & [sarmilab932007@gmail.com](mailto:sarmilab932007@gmail.com)
    - S.Sneha & [senthilp6363@gmail.com](mailto:senthilp6363@gmail.com)

1. Project Overview

**Purpose:** A cookbook's purpose is to provide recipes and instructions for preparing meals. It helps people learn how to cook, try new dishes, and improve their skills in the kitchen. It's like a guide that makes cooking easier and more fun!

## Features :

* + - Recipes
    - Ingredients List
    - Measurements
    - Preparation & Cooking Time
    - Serving Size

# Architecture

* + **Frontend:** This is what the user sees and interacts with — the visual part of the cookbook.
  + **Backend:** This is the behind-the-scenes logic that handles requests, processes data, and communicates with the database.
  + **Database:** This is where all the data is stored — recipes, users, ingredients, categories, comments, etc.

1. Setup Instructions

## Prerequisites:

\_ Node.js and npm:

-React.js :

* + - **HTML, CSS, and JavaScript**:
    - **Version Control**
    - **Development Environment**

. Installation Steps:

**Install Dependencies:**

• Navigate into the cloned repository directory and install libraries:

cd fitness-app-react

npm install

Start the Development Server:

• To start the development server, execute the following command:

npm start

**Access the App:**

• Open your web browser and navigate to [http://localhost:3000](http://localhost:3000/).

• You should see the application's homepage, indicating that the installation and setup were successful.

# Folder Structure

cookbook-app/

├── frontend/ # React/Vue/Angular frontend

│ ├── public/ # Static files (index.html, images, icons)

│ ├── src/

│ │ ├── assets/ # Images, fonts, icons

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

│ │ ├── pages/ # Pages (Home, RecipeDetails, Login, etc.)

│ │ ├── services/ # API calls (e.g., axios config)

│ │ ├── context/ # Global state management (e.g., AuthContext)

│ │ ├── App.js # Main app component

│ │ └── index.js # Entry point

│ └── package.json # Frontend dependencies and scripts

├── backend/ # Backend server (Node.js, Express, Django, etc.)

│ ├── controllers/ # Logic for handling requests

│ ├── routes/ # API endpoints (e.g., /recipes, /users)

│ ├── models/ # Database models (e.g., Recipe.js, User.js)

│ ├── middleware/ # Auth, error handling, logging, etc.

│ ├── config/ # DB config, environment setup

│ ├── uploads/ # Image uploads (optional, or use cloud)

│ ├── server.js # Entry point

│ └── package.json # Backend dependencies

├── database/ # DB setup, seeders, migrations

│ ├── schema.sql # For SQL-based DBs

│ ├── seed.js # Dummy data to populate database

│ └── connection.js # DB connection logic

├── .env # Environment variables (DB\_URL, API keys)

├── .gitignore # Ignore node\_modules, env files, etc.

├── README.md # Project overview

└── package.json # If managing both frontend/backend with one tool

# Running the Application

## Frontend:

**├── frontend/ # React app**

**Backend:**

├── backend/ # Node.js + Express app

* + **Access:** Visit [http://localhost:3000](http://localhost:3000/)

# API Documentation

## User:

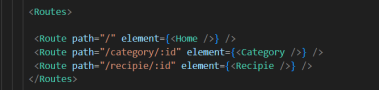
-[http://localhost:3000/api/users](http://localhost:5000/api/users) (or) <https://my-cookbook-api.com/api>

* + - [Sahananila2316@gmail.com](mailto:Sahananila2316@gmail.com)

## Projects:

? Setup the Routing paths

Setup the clear routing paths to access various files in the application.



? Develop the Navbar and Hero components

? Code the popular categories components and fetch the categories from ***themealsdb*** Api.

? Also, add the trending dishes in the home page.

? Now, develop the category page to display various dishes under the category.

? Finally, code the recipe page, where the ingredients, instructions and a demo video  will be integrated to make cooking much easier.

* /api/projects/create
* /api/projects/:id • **Applications:** /api/apply

# Authentication

Authentication allows users to **securely access** and **interact** with the app. It ensures that only registered users can perform certain actions like:

Viewing or editing their profile

Creating or saving recipes

Adding recipes to favorites

Leaving reviews or comments

# User Interface

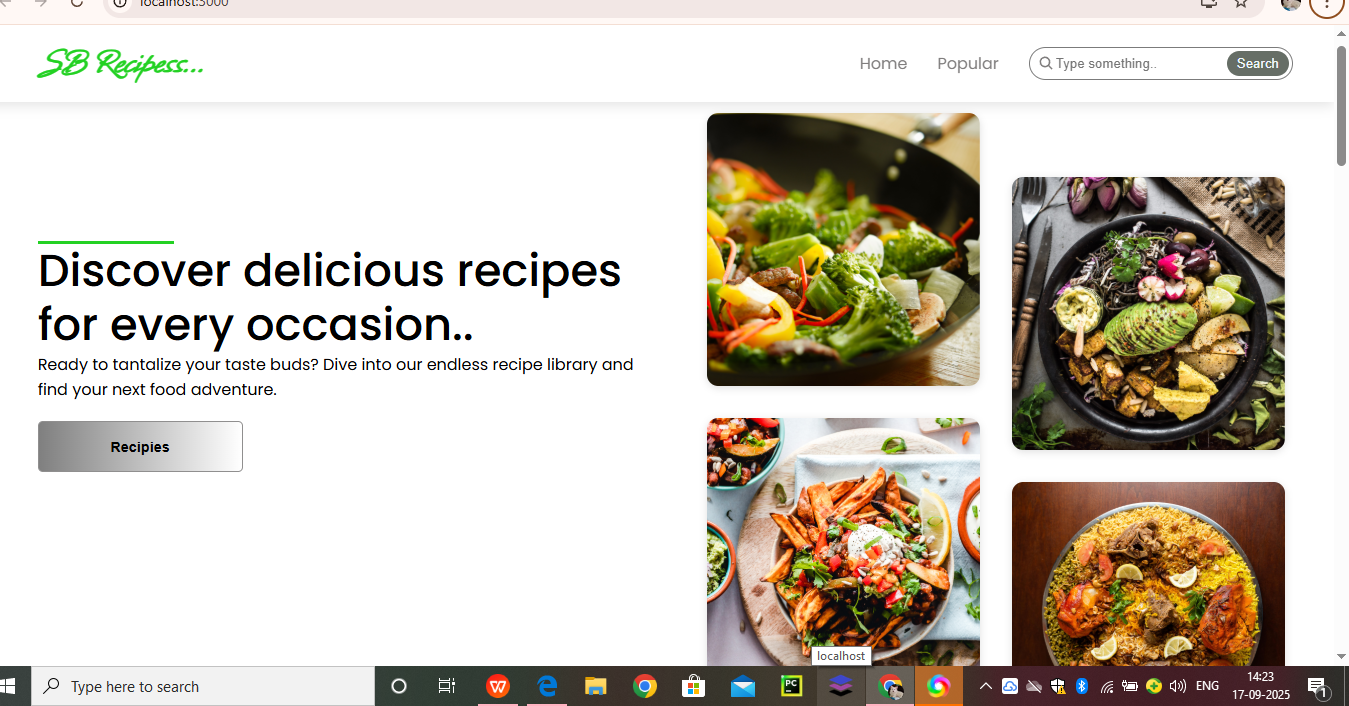
The **User Interface** is the visual part of the cookbook app that users **see and interact with** — usually built using frontend technologies like **HTML**, **CSS**, **JavaScript**, and libraries/frameworks like **React**, **Vue**, or **Angular**.

# Testing

Testing is the process of checking whether your app behaves as expected. It helps catch bugs, improve reliability, and ensure a smooth user experience.

* + 🔐 Authentication
  + 🔐 Recipes
  + ⭐ Favorites

1. **Screenshots or Demo**



1. **Future Enhancements :**
2. ****User Features****
3. ****Recipe Management****
4. ****Community Features****