Project Documentation – CookBook

**1.** **Introduction**

• Project Title: Cookbook – Your Virtual Kitchen Assistan

• Team ID: NM2025TMID36639

• Team Leader: BenJoel.J & benjoel22072006@gmail.com

• Team Members:

– Akash.U & akashudayakumar2006@gmail.com

– Atshaya.R & atshayarajamani.18@gmail.com

-Ashwini.V & v.ashwiniachu@gmail.com

**2. Project Overview**

• Purpose: Cookbook is a smart virtual kitchen assistant that helps users discover, organize, and prepare recipes efficiently. It connects culinary enthusiasts with step-by-step guidance, personalized recommendations, and interactive cooking support.

• Features:

– Recipe search and filtering by ingredients, cuisine, and dietary needs

– Step-by-step guided cooking mode with voice/video assistance

– Personalized meal planning and grocery list generation

– Integration with preparation videos for visual learning

– Feedback and rating system for recipes

– Admin dashboard for recipe management and user support

**3. Architecture**

• Frontend: React.js with Bootstrap and Material UI

• Backend: Node.js and Express.js for server logic and API endpoints

• Database: MongoDB to store recipes, user preferences, shopping lists, and cooking history

**4. Setup Instructions**

• Prerequisites:

– Node.js

– MongoDB

– Git

– React.js

– Express.js

– Mongoose

– Visual Studio Code

• Installation Steps:

Clone the repository: git clone <repo\_url>

Install client dependencies: cd client && npm install

Install server dependencies: cd ../server && npm install

**5. Folder Structure**

Cookbook/

|-- client/ # React frontend

| |\_\_ components/

| |\_\_ pages/

|-- server/ # Node.js backend

| |\_\_ routes/

| |\_\_ models/

| |\_\_ controllers/

**6. Running the Application**

• Frontend:

cd client

npm start

• Backend:

cd server

npm start

• Access: Visit http://localhost:3000

**7. API Documentation**

• User:

– /api/user/register

– /api/user/login

• Recipes:

– /api/recipes/create

– /api/recipes/:id

– /api/recipes/search

• Meal Plans:

– /api/mealplan/create

– /api/mealplan/:id

• Grocery List:

– /api/grocery/add

– /api/grocery/:id

**8. Authentication**

• JWT-based authentication for secure login

• Middleware to protect private routes

**9. User Interface**

• Landing Page with recipe discovery

• User Dashboard with saved recipes and meal plans

• Guided Cooking Page with video/voice support

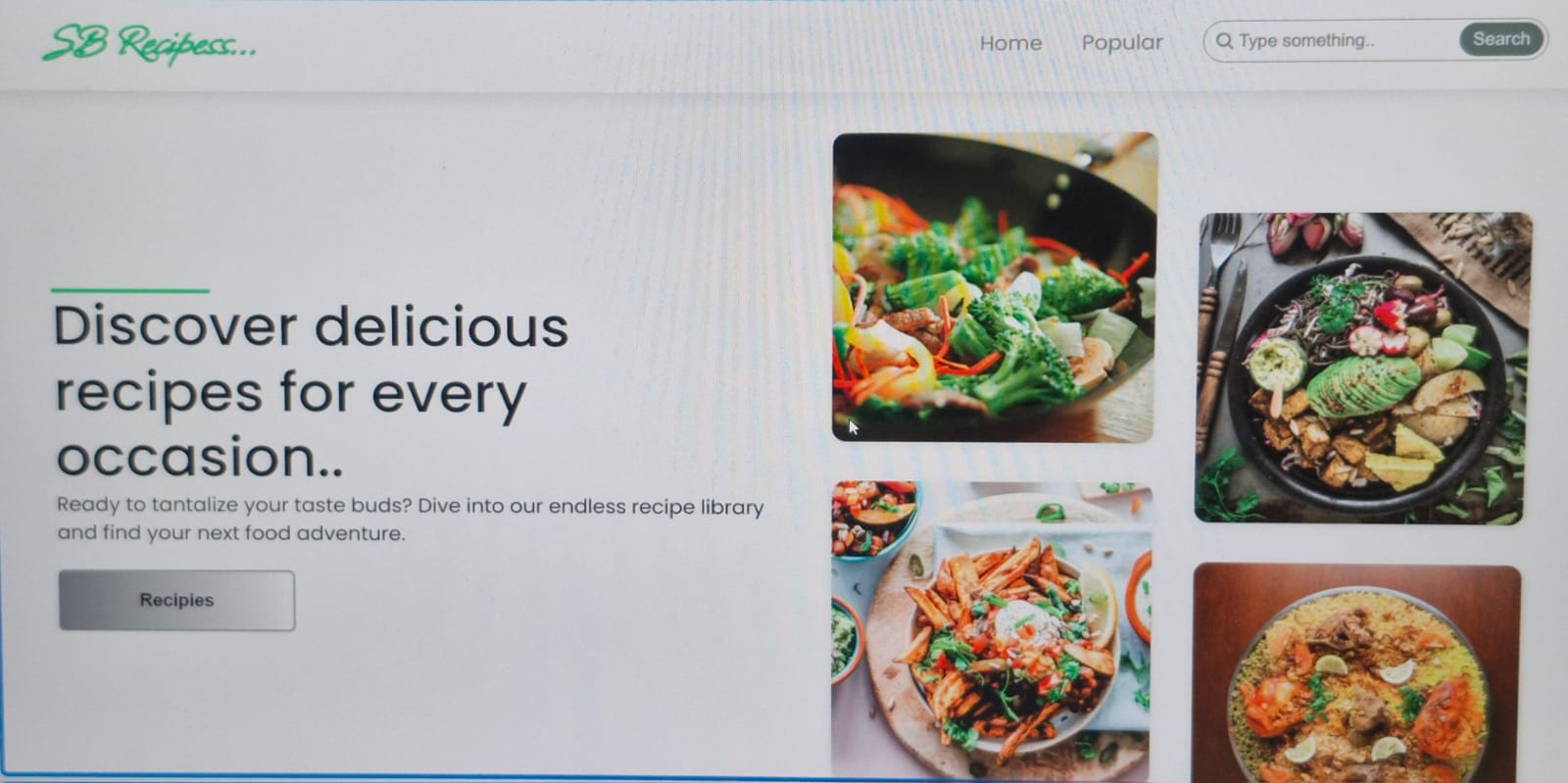
• Admin Panel for recipe and content management

**10. Testing**

• Manual testing during development milestones

• Tools: Postman, Chrome Dev Tools

**11. Screenshots or Demo**

****

**12. Known Issues**

• Limited recipe dataset in initial phase

• Performance optimization required for large media files

**13. Future Enhancements**

• AI-powered recipe recommendations based on user preferences

• Multi-language support for global users

• Integration with smart kitchen devices (IoT)

• Community features like sharing recipes and cooking tips