

Project Documentation

COOKBOOK-YOUR VIRTUAL KITCHEN ASSISTANT

1. Introduction

- **Project Title:**cookbook-your kitchen assistant
- **Team ID:**NM2025TMID34284
- **Team Leader:**

Name&mail id: Priyadharshini S (spriyadharisini943@gmail.com)

- **Team Members:**

Name & Mail id:Sathana M (msathana859@gmail.com)

Name & Mail id:Vidya bharathi E (vidyabharathiilavarasi@gmail.com)

Name & Mail id:Nethrashree K (nethrakarthi807@gmail.com)

2. Project Overview

Purpose: The primary purpose of a virtual kitchen assistant, such as the CookBook app, is to **simplify and enhance the cooking experience by managing recipes and providing hands-free guidance during meal preparation**. This is achieved by enabling users to browse, save, customize, and organize recipes, with advanced features like AI-powered recipe scanning and builder, voice-activated instructions, and integration with smart kitchen appliances for a more intuitive and accessible culinary journey.

Features:

A recipe library for browsing and managing recipes.

Step-by-step cooking instructions.

A responsive design and user-friendly interface.

Technical details, including the use of frameworks like React.js

3. Architecture

- **Frontend:** React.js,node.js
- **Backend:** mongodb

- **Database:** mongodb

4. Setup Instructions

- **Prerequisites:**
 - Node.js
 - MongoDB
 - Git
 - React.js
 - Express.js – Mongoose – Visual Studio Code
- **Installation Steps**
- **npm install**
- **npm start**

5. Folder Structure

```
SB-Works/  
|-- client/           # React frontend  
    |-- components/  
        |-- pages/  
|-- server/          # Node.js backend  
    |-- routes/  
        |-- models/  
            |-- controllers/
```

6. Running the Application

- **Frontend:**

Node. Js

Vsc

Code

Backend:

Mongodb

- **Access:** Visit <http://localhost:3000>

7. API Documentation

- **User:**

- /api/user/register
- /api/user/login

- **Projects:**

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

- **Chats:**

- /api/chat/send
- /api/chat/:userId

8. Authentication

- JWT-based authentication for secure login
- Middleware protects private routes

9. User Interface

- github
- whatsapp
- chorme
- google drive

10. Testing

- Manual testing during milestones
- Tools: Postman, Chrome Dev Tools

11. Screenshots or Demo



