

❖INTRODUCTION.

Team id : **NM2025TMID47791**

Team members:

S.Sanjai Rajan : sanjairajan1694@gmail.com

K.G.Saranraj : saranrajkg00606@gmail.com

S.Sanjay : gokulgokul1446@gmail.com

R.Sabarivasan : rsabarivasan2007@gmail.com

Purpose:

- ⊖ CookBook is a virtual kitchen assistant that transforms the way users discover, organize, and create recipes, catering to both amateur cooks and professional chefs.

Features:

- ⊖ Intuitive user interface for seamless navigation – Dynamic recipe search and discovery – Personalized recipe organization and creation tools – Community-driven sharing and collaboration – Aesthetic design for immersive culinary exploration – Inclusive


platform for all skill levels – Vision-driven
innovation blending tradition and technology

❖ ARCHITECTURE

Front End:

-  React.js with Bootstrap and Material UI for responsive design and modern UI components.

Back End:

-  : Node.js and Express.js managing server logic, RESTful API endpoints, and real-time interactions.

Data Base

-  MongoDB stores user profiles, recipes, reviews, and chat messages in a flexible document structure.

❖ SETUP INSTRUCTIONS

Prerequisites

Here are the key prerequisites for developing a frontend application using React.js:

Node.js and npm:

Node.js is a powerful JavaScript runtime environment that allows you to run JavaScript code on the local environment. It provides a scalable and efficient platform for building network applications.

Install Node.js and npm on your development machine, as they are required to run JavaScript on the server-side.

- ④ Download: <https://nodejs.org/en/download/>
- ④ Installation instructions: <https://nodejs.org/en/download/package-manager/>

React.js:

- React.js is a popular JavaScript library for building user interfaces. It enables developers to create interactive and reusable UI components, making it easier to build dynamic and responsive web applications.
- Install React.js, a JavaScript library for building user interfaces.
- ④ Create a new React app:
 - `npx create-react-app my-react-app`
 - Replace my-react-app with your preferred project name.
- ④ Navigate to the project directory:
 - `cd my-react-app`
- ④ Running the React App:
 - ④ With the React app created, you can now start the development server and see your React application in action.
- ④ Start the development server:
 - `npm start`
 - This command launches the development server, and you can access your React app at <http://localhost:3000> in your web browser.
- ④ **HTML, CSS, and JavaScript:** Basic knowledge of HTML for creating the structure of your app, CSS for styling, and JavaScript for client-side interactivity is essential.

🔗 **Version Control:** Use Git for version control, enabling collaboration and tracking changes throughout the development process. Platforms like GitHub or Bitbucket can host your repository.

- Git: Download and installation instructions can be found at: <https://git-scm.com/downloads>

🔗 **Development Environment:** Choose a code editor or Integrated Development Environment (IDE) that suits your preferences, such as Visual Studio Code, Sublime Text, or WebStorm.

- Visual Studio Code: Download from <https://code.visualstudio.com/download>
- Sublime Text: Download from <https://www.sublimetext.com/download>
- WebStorm: Download from <https://www.jetbrains.com/webstorm/download>

🔗 To get the Application project from drive:

🔗 Follow below 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>.

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

 You have successfully installed and set up the application on your local machine. You can now proceed with further customization, development, and testing as needed

❖ INSTALLATION STEPS

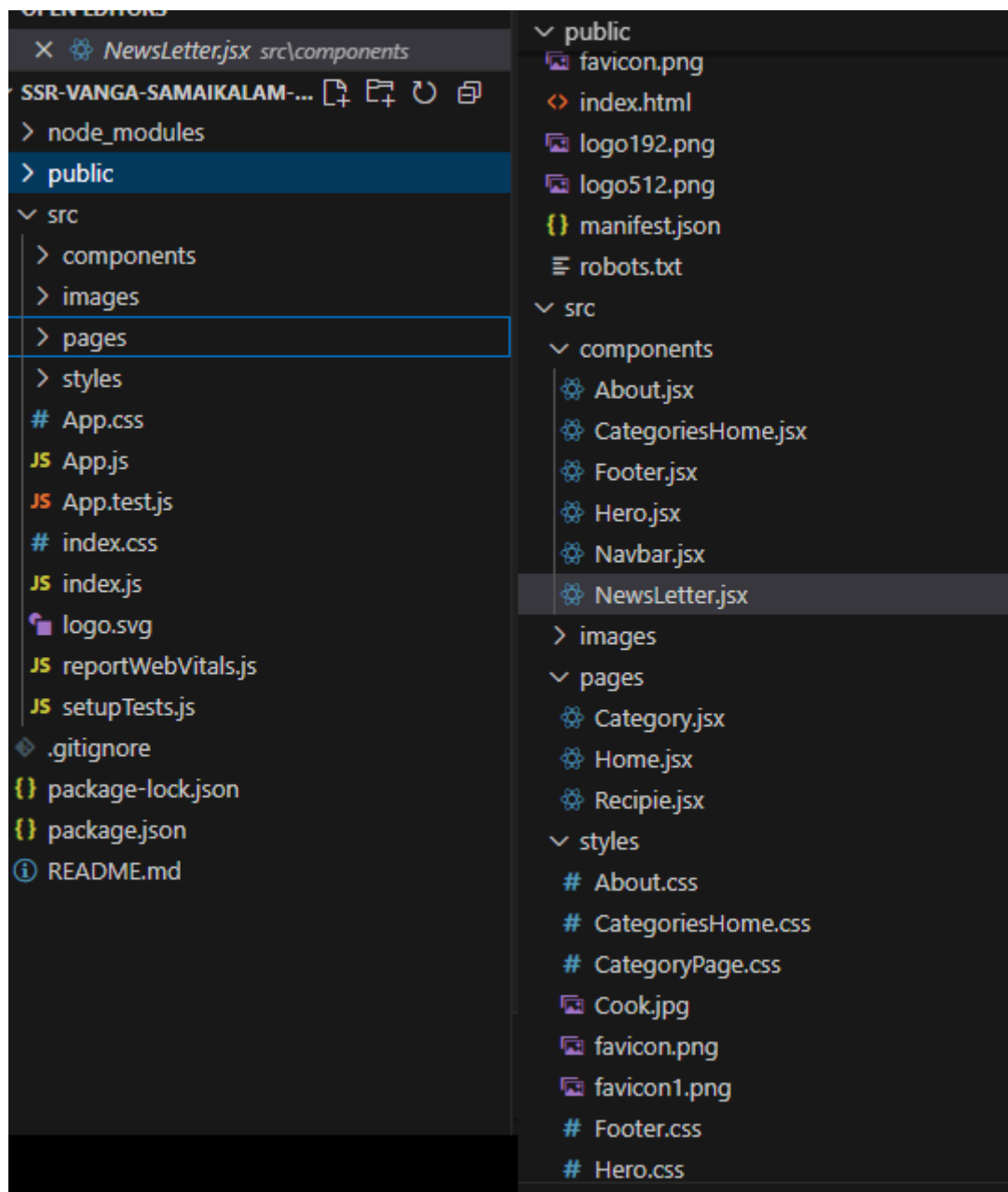
\

 : # Clone the repository git clone

 # Install client dependencies cd client npm install

 # Install server dependencies cd ../server npm install

❖ FOLDER STUTURE



❖ RUNNING THE APPLICATION

🔗 Frontend:

🔗 cd client

🔗 npm start

🔗 Backend:

④ `cd server npm`

④ `start`

④ Access: Visit <http://localhost:3000>

❖ API DOCUMENTATION

④ User:

④ `/api/user/register`

④ `/api/user/login`

④ Projects:

④ `/api/projects/create`

④ `/api/projects/:id` Applications: `/api/apply`

④ Chats:

④ `/api/chat/send`

④ `/api/chat/:userId`

❖ AUTHENTICATION

④ JWT-based authentication for secure login

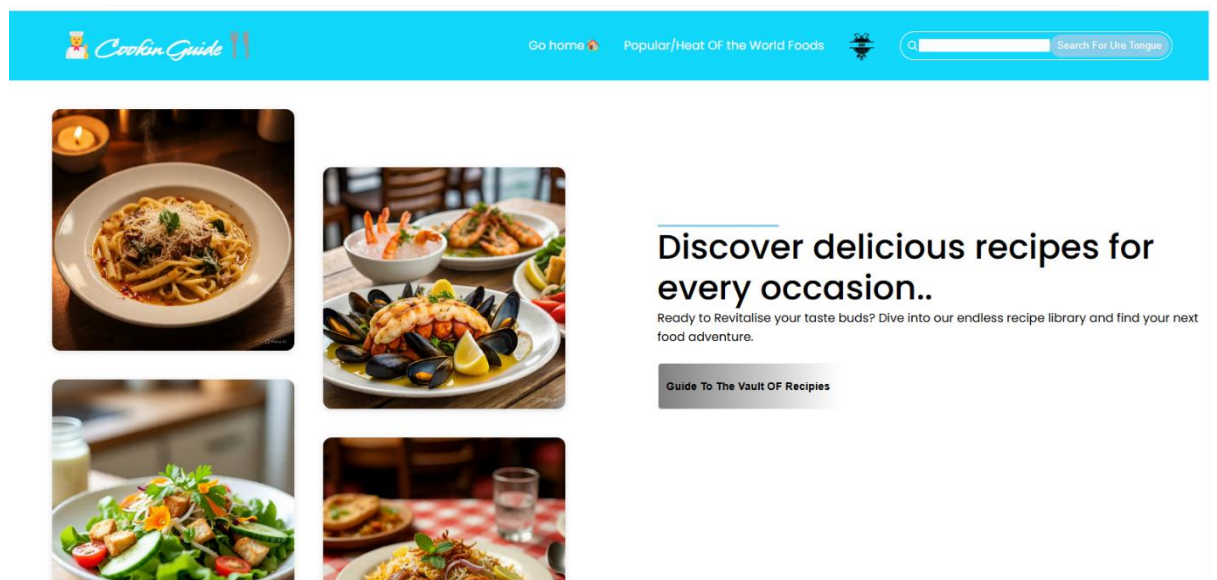
④ Middleware protects private routes

❖ USER INTERFACE

④ Landing Page

④ Freelancer Dashboard

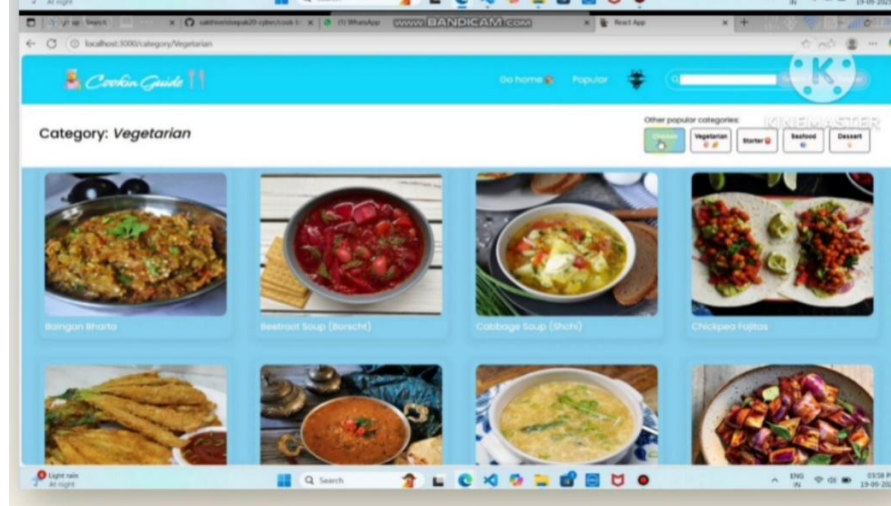
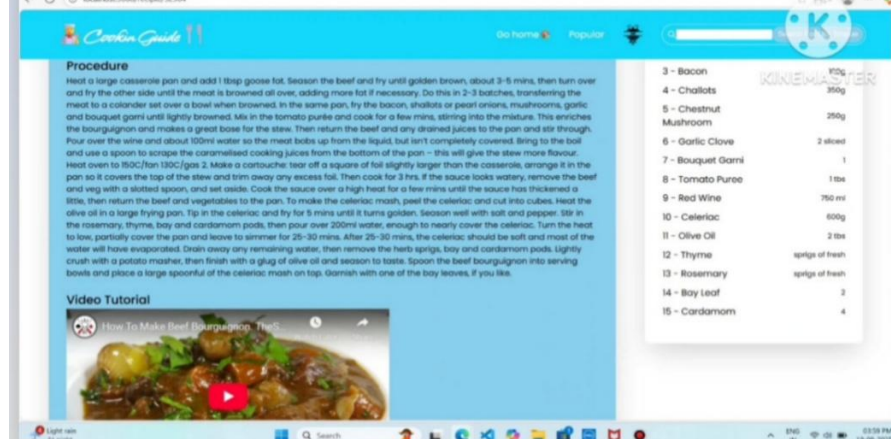
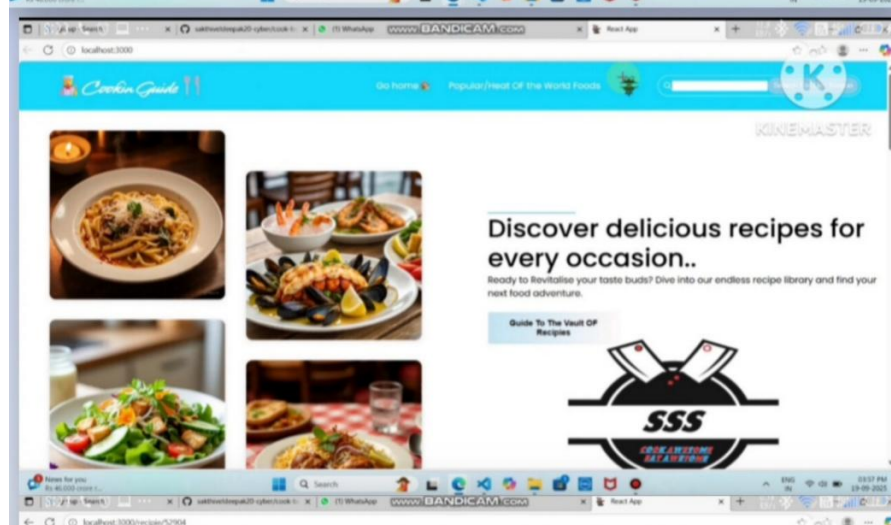
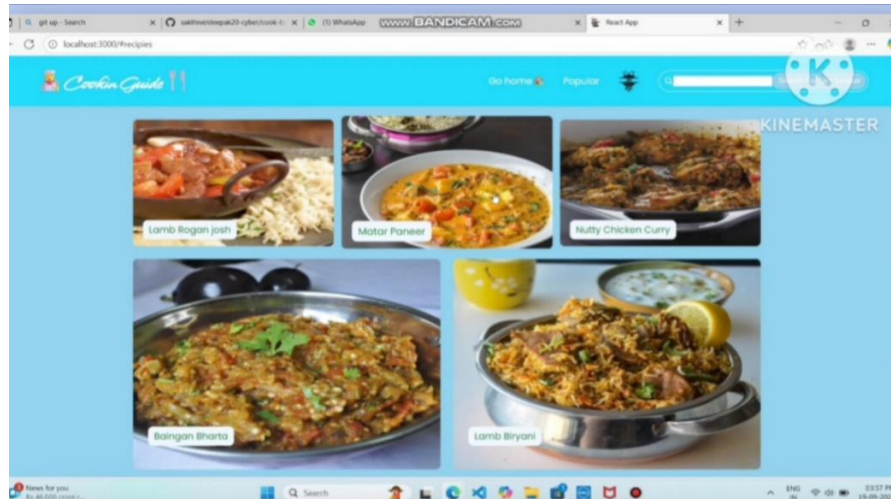
- 🔗 Admin Panel
- 🔗 Project Details Page



❖ TESTING

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

❖ SCREENSHOTS



❖ DEMO VIDEO LINK

https://drive.google.com/file/d/1dkEiqK-YflyZPVmuynky54_bEaPJtF8W/view?usp=drive_link

❖ KNOW ISSUES

⊖ no

❖ FUTURE ENHANCEMENTS

⊖ possible