***Project Documentation***

*Cook Book - Your Virtual Kitchen Assistant*

**1. *Introduction***

• ***Project Title:* Cook Book - Your Virtual Kitchen Assistant**

• ***Team ID* - NM2025TMID36981**

• ***Team Leader*: Charumathi P (charumathi.p7781@gmail.com)**

• ***Team Members:***

**– Bharathy R (bharathybhanu2112@gmail.com)**

**–Deepika G (madeshmadesh45180@gmail.com)**

–**Dhanusree K (dhanusreek2911@gmail.com)**

**2. *Project Overview***

*The primary goal of CookBook is to provide a user-friendly platform that* caters to

individuals passionate about cooking, baking, and exploring new culinary horizons.

***Our objectives include:***

***• User-Friendly Experience:*** Create an interface that is easy to navigate, ensuring

users can effortlessly discover, save, and share their favourite recipes.

***• Comprehensive Recipe Management:*** Offer robust features for organizing and

managing recipes, including advanced search options.

***• Technology Stack:*** Leverage modern web development technologies, including React.js, to ensure an efficient, and enjoyable user experience.

**3*. Architecture***

• ***Frontend*:**

➔ **Framework -** React.js is used to build the user interface for the digital cookbook. ➔ **Styling -** Css files are used to design and style the recipe pages and layout. ➔ **Component Structure -** The src folder contains reusable components such as

RecipeCard, CategoryList, and SearchBar.

➔ **Assets** - Static files like recipe images, icons, and logos are stored in the src directory. **4. *Setup Instructions***

• ***Prerequisites:***

**–** Node.js and npm

**–** React.js

**–** Git

**–** HTML, CSS, and JavaScript

**–** Visual Studio Code

● **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.

● Download: https://nodejs.org/en/download/

● 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.

● Create a new React app:

npx create-react-app my-react-app

Replace my-react-app with your preferred project name.

● Navigate to the project directory:

cd my-react-app

● Running the React App:

With the React app created, you can now start the development server and see your React application in action.

● 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.

● **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:***

***• Installation of required tools:***

To build CookBook, we'll need a developer's toolkit. We'll use React.js for the interactive interface, React Router Dom for seamless navigation, and Axios to fetch news data. For visual design, we'll choose either Bootstrap or Tailwind CSS for pre-built styles and icons.

Open the project folder to install necessary tools, In this project, we use:

o React Js

o React Router Dom

o React Icons

o Bootstrap/tailwind css

o Axios

***•*** For further reference, use the following resources

***o https://react.dev/learn/installation***

***o***

***https://react-bootstrap-v4.netlify.app/getting-started/introducti***

***on/ o https://axios-http.com/docs/intro***

***o https://reactrouter.com/en/main/start/tutorial***

**5. *Project Structure***

|  |
| --- |

|  |
| --- |

In this project, we’ve split the files into 3 major folders, Components, Pages and Styles. In the pages folder, we store the files that acts as pages at different url’s in the application. The components folder stores all the files, that returns the small components in the application. All the styling css files will be stored in the styles folder.

**6. *Running the Application***

• ***Frontend:***

cd code

npm install

npm start

cd server npm

start

• ***Access*:** Visit http://localhost:3000

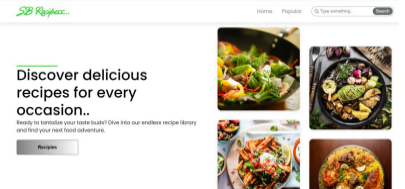
**7. *Testing***

• Manual testing done against the user interface.

**8. *Screenshots:***

**● Hero components**

The hero component of the application provides a brief description about our application and a button to view more recipes.



● **Popular categories**

This component contains all the popular categories of recipes..

|  |
| --- |

? **Trending Dishes**

This component contains some of the trending dishes in this application.

|  |
| --- |

? **News Letter**

The news letter component provides an email input to subscribe for the recipe newsletters.

|  |
| --- |

**? Category dishes page**

The category page contains the list of dishes under a certain category.

|  |
| --- |

**? Recipe page**

The images provided below shows the recipe page, that includes images, recipe instructions, ingredients and even a tutorial video.

|  |
| --- |

|  |
| --- |

|  |
| --- |