**Project Documentation**

**1**.**Introduction**

* Project title: Insight stream:navigate the news landscape.
* Team ID: NM2025TMID35578.
* Team leader: sindhuja. S &sindhujasivakumar1517@gmail.com
* Team members:
  + Princy prabha. P
  + Rajeshwari. R
  + Ranjitha. R

**2.Project Overview**

* Purpose: A Insight streamcompiles recipes, techniques ,and culinary knowledge, inspiring creativity and preserving cultural traditions. It serves as a resource for cooks to explore new flavors, methods, and cuisines.
* Features:

1.Metadata

2.Datatype

3.Data analyses

**3.Architecture**

* Frontend**:**

1.HTML / CSS: For structuring and styling web-based cookbooks or recipe websites.

2.JavaScript: For interactive features, such as recipe filtering, meal planning, or cooking timers.

3.Read or Angular: For building complex web applications with cooking -related features.

* Backend:

1.Node.js: For building RESTful APIs to serve recipes, ingredients, or cooking techniques.

2.Ruby on Rails: For creating complex web applications with cooking -related features.

3.Python: For data analysis machine learning, or natural language processing in cooking -related applications.

* Database:

1.Mysql : For storing recipes, ingredients, and cooking techniques in a relational database.

2.MongoDB: For storing recipes and related data in a NoSQL database.

**4.Setup Instructions**

* Prerequisites :

- Node. Js

- MongoDB

- Git

- React.js

- Express. Js – Mongoose – Visual studio code

* Installation Steps:

1.Clone the repository: Use visual studio code to clone the cookbook project repository.

2.Install dependencies: Run npm install to install the required dependencies.

3. Configure the project.

4. Run migration.

5. Start the application: Run the command to start the cookbook project (eg.npm start).

**5.Folder structure**

1. Assets: Images, videos, and other media files.

2. Recipes: Recipe data, including ingredients, instructions, and metadata.

3. Components: Reusable UI components (e.g., recipe cards, search bars).

4. Pages: Page templates for recipe details, search results, and other views.

5. Services: API integrations, data fetching, and business logic.

6. Utils: Utility functions for data processing, formatting, and validation.

7. Config: Configuration files for database connections, API keys, etc.

**6.Running the application**

* Frontend**:**

CD code

npm install

* Backend:

npm start

* Access:

1.Open a web browser.

2.Navigate to the frontend URL.

**7.API documentation**

* User API

1.User Registration: Create a new user account.

2.User Login: Authenticate a user and return a token.

3.User Profile: Retrieve a user’s profile information.

4.User Update: Update a user’s profile information.

* Project API

1.Project Creation: Create a new project.

2.Project List: Retrieve a list of projects.

3.Project Details: Retrieve a project’s details.

4.Project Update: Update a project’s details.

* Chat API

1.Message Send: Send a new message.

2.Message List: Retrieve a list of messages.

3.Chat History: Retrieve chat history for a specific project or application.

**8.Authentication**

1. User Registration: Creating a new user account with credentials (username, email, password).

2. User Login: Authenticating a user with their credentials (username/email, password).

3. Token-based Authentication: Issuing a token (e.g., JWT) upon successful login, which is used for subsequent requests.

4. Password Management: Handling password reset, change, and recovery.

**9.User Interface**

1. Recipe Gallery: Showcase of dishes with images and descriptions.

2. Search & Filter: Find recipes by ingredient, cuisine, or dietary need.

3. Recipe Details: Step-by-step instructions, ingredients, and cooking tips.

4. Save & Share: Options to save favorite recipes and share with others.

5. Meal Planning: Tools for planning meals and generating grocery lists.

**10.Testing**

1. Functional Testing: Verify recipe search, filtering, and display.

2. User Interface Testing: Ensure intuitive navigation and usability.

3. Recipe Validation: Check accuracy of ingredients, instructions, and cooking times.

4. Performance Testing: Evaluate loading speed and responsiveness.

5. User Acceptance Testing : Validate user experience and feedback.

**11.Screenshots or Demo**

**12.Known Issues**

* Technical Issues:

- Integration: Difficulty integrating the cookbook feature with existing Naan Mudhalvan platforms or tools.

- Performance: Slow loading times, errors, or bugs affecting user experience.

* Content-Related Issues:

- Recipe Accuracy: Inaccurate ingredient quantities, cooking times, or instructions.

- Recipe Variety: Limited representation of diverse cuisines or dietary preferences.

* User Experience Issues:

- Navigation: Difficulty finding specific recipes or features.

- User Engagement: Low user retention or interaction with the cookbook feature.

**13.Future enhancement**

1. Personalized Recipe Recommendations: Use machine learning to suggest recipes based on user preferences, dietary needs, and cooking history.

2. Augmented Reality (AR) Cooking Guide: Implement AR features to provide interactive, step-by-step cooking instructions.

3 Social Sharing and Community Features: Allow users to share recipes, cooking experiences, and photos, and engage with others in a community forum.

4. Integration with Smart Kitchen Devices: Enable integration with smart kitchen appliances and devices for seamless cooking experiences.

5. Advanced Meal Planning and Grocery List Generation: Enhance meal planning features to include automated grocery list generation, with options for online ordering and delivery.

6. Recipe Customization and Adaptation: Allow users to customize recipes based on dietary restrictions, ingredient availability, and personal preferences.

7. Video Content and Tutorials: Include video tutorials and cooking demonstrations to enhance the learning experience.

8.Gamification and Rewards: Introduce gamification elements, such as cooking challenges, badges, and rewards, to encourage user engagement and retention.