FindFood:

A recipe review web-application

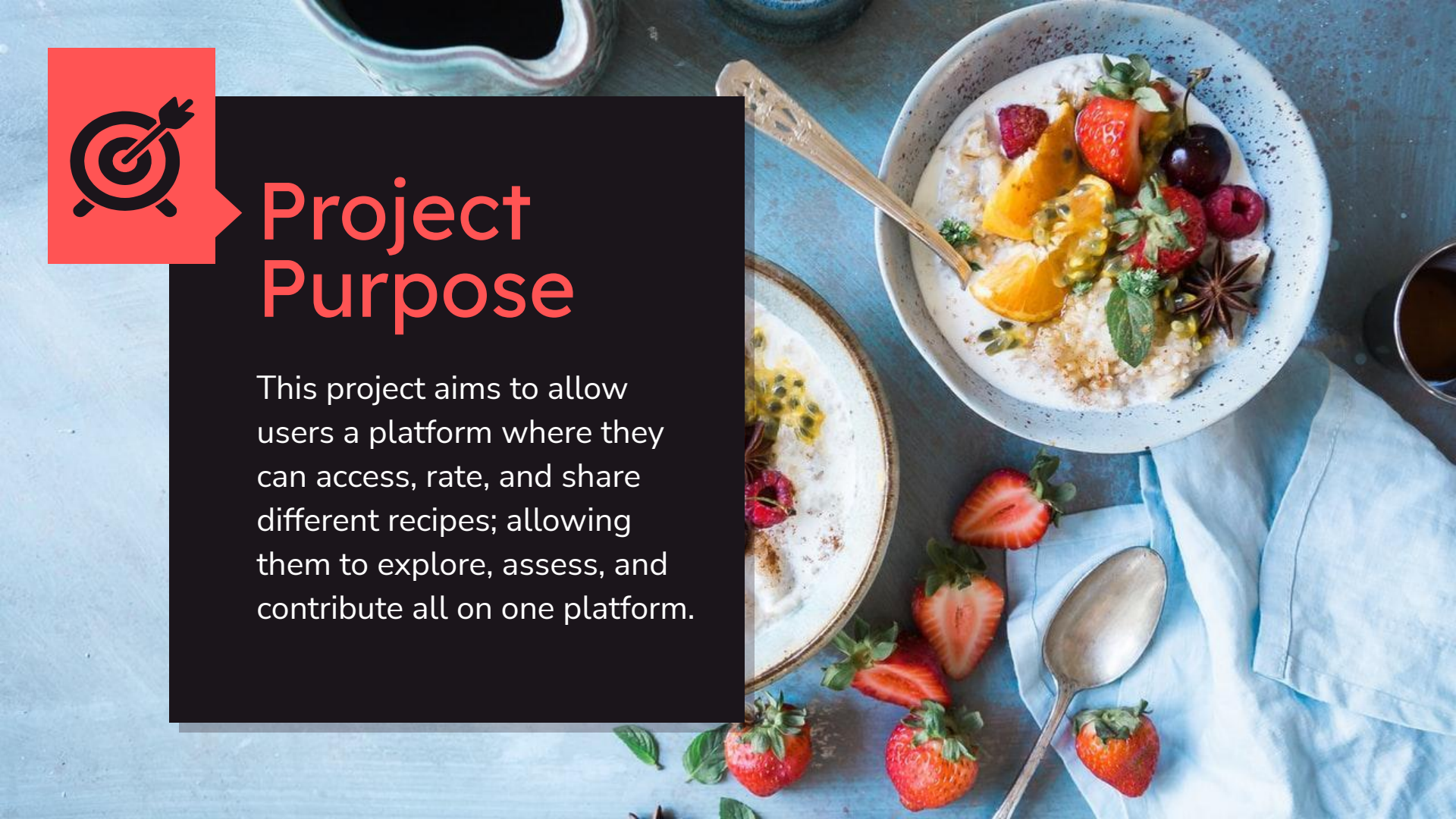
Mentor: Ms Ami Pandat

Presented by:

Pragati Sharma: 202012047

Mahi Patel: 201801039

Vanshika Kochhar: 201801104

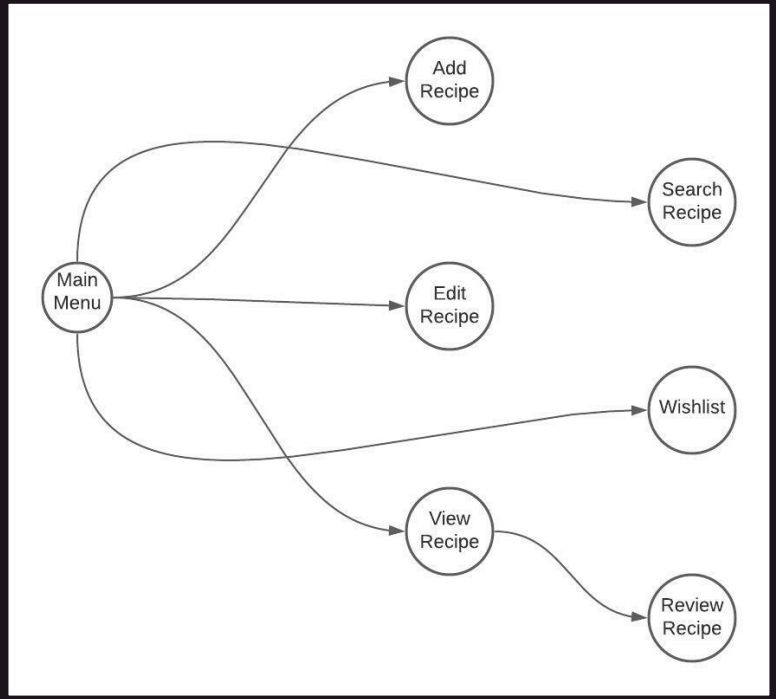


Project Purpose

This project aims to allow users a platform where they can access, rate, and share different recipes; allowing them to explore, assess, and contribute all on one platform.

Project Scope

- Allow access to all recipes for all users
- Allow users to rate all available recipes
- Allow users to add recipes



Users and Stakeholders

These are the website users who wish to contribute to community by adding the recipes and the project development team who maintain the database.

Stakeholders

Users

Any and all individuals looking for recipes for particular dishes or simply trying to find a dish to try and want to share their recipes on the platform.

Business Rules

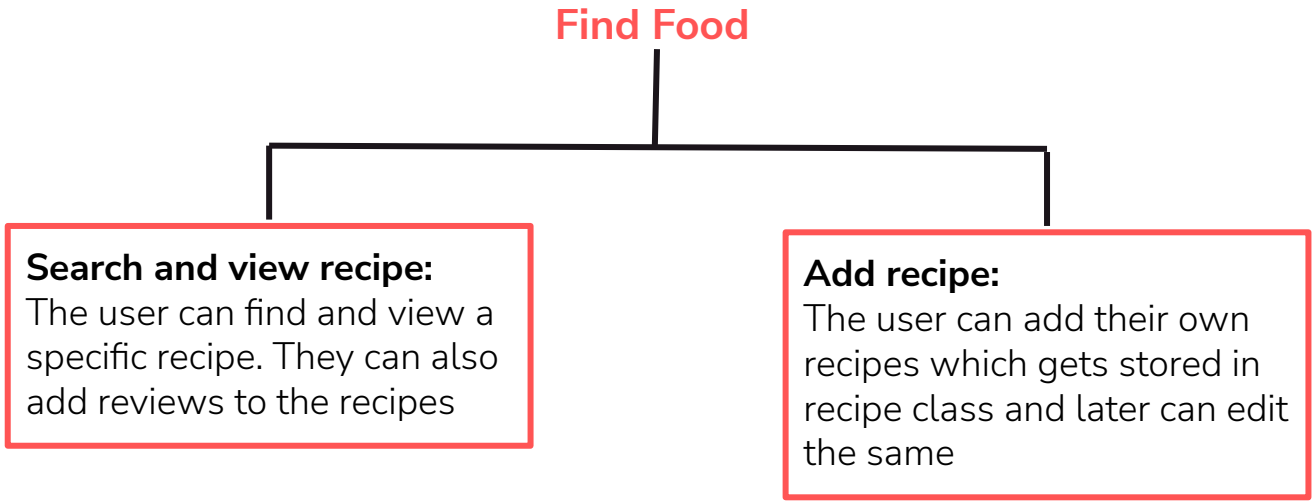
Only users will be allowed to rate recipes.

All recipes added to the website must follow a particular pattern.

All recipes must be accessible by all users irrespective of the rating of the recipe.

System Information

- The recipe review web application is titled as **“Find Food”**



Functional & non-functional requirements

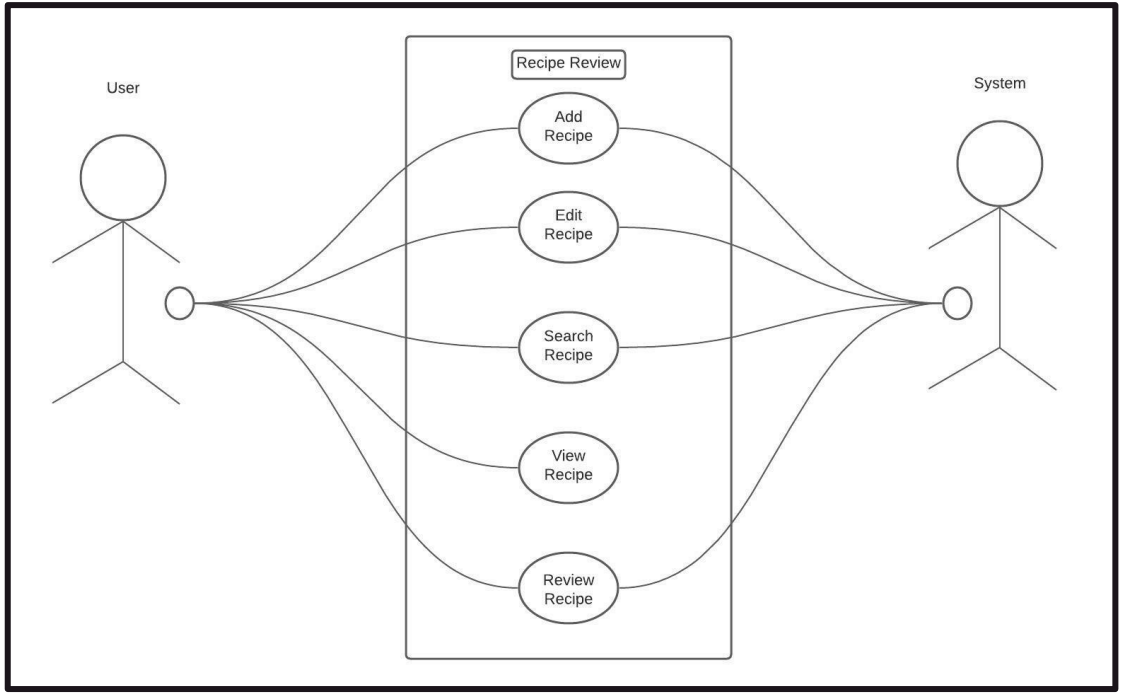
Functional Requirements

- **Login:** Users will be able to login into the system
- **Search Recipe**: Users will be able to search for any particular recipes on the website.
- **Review a Recipe**: Users should be able to review and rate any recipe
- **Add Recipe**: Users will be able to share any new original recipes of their own with all other users of the platform.
- **Edit Recipe**: Users will be able to make simple edits to any recipes that they have shared.
- **View Recipe**: Users will be able to view all shared recipes by all users of the website.
- **Wishlist:** The users should shortlist recipes.

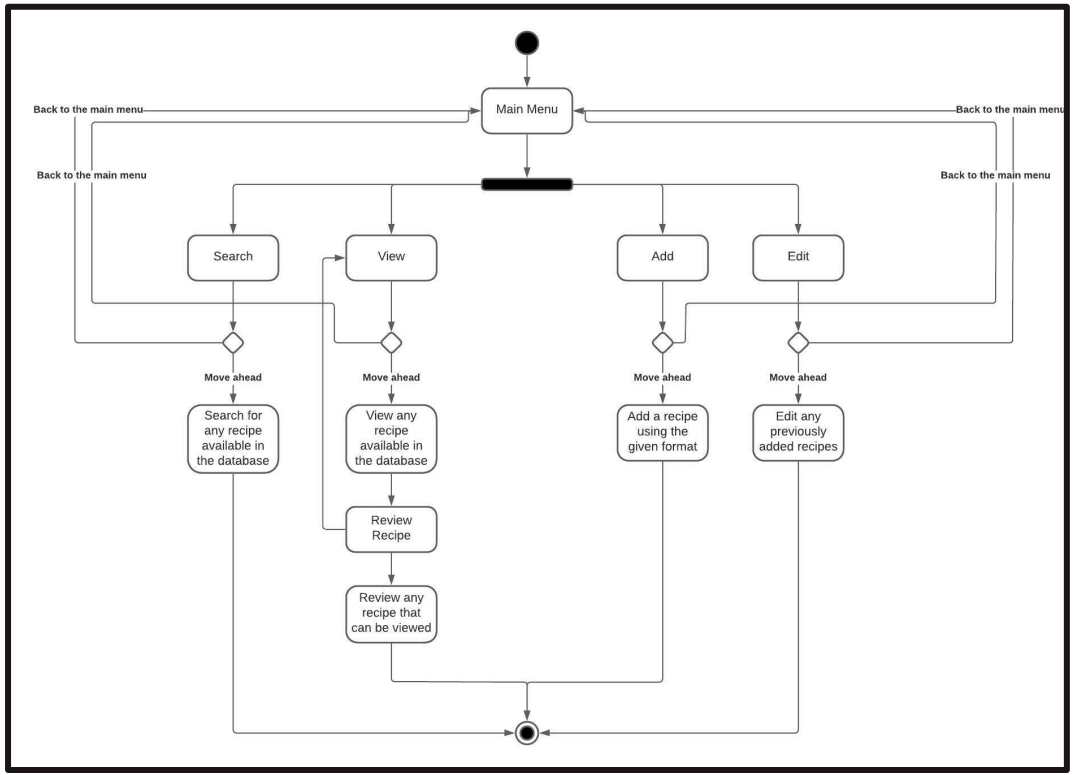
Non-Functional Requirements

- **Scalable:** Scalable database should be used to store huge recipe set.
- **Consistency:** Additions and edits to any information on the website should be consistent with the data stored in the server.
- **Performance:** The response time for any query to the server should be as short as possible to enhance the user experience.
- **Ease of Use:** The interface developed should be user-friendly so as to accommodate all and any users of the platform.

Use case Diagram

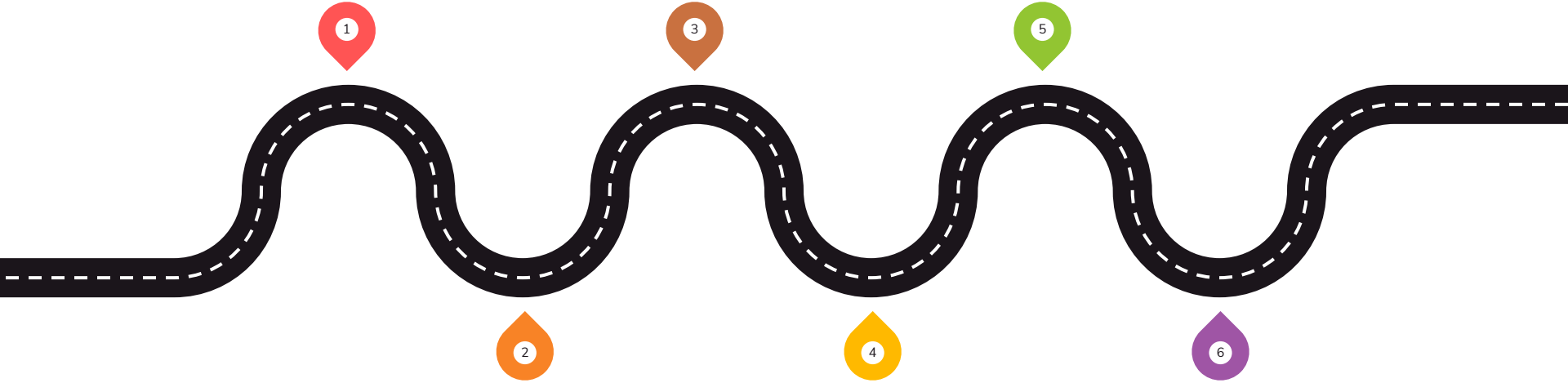


Activity Diagram

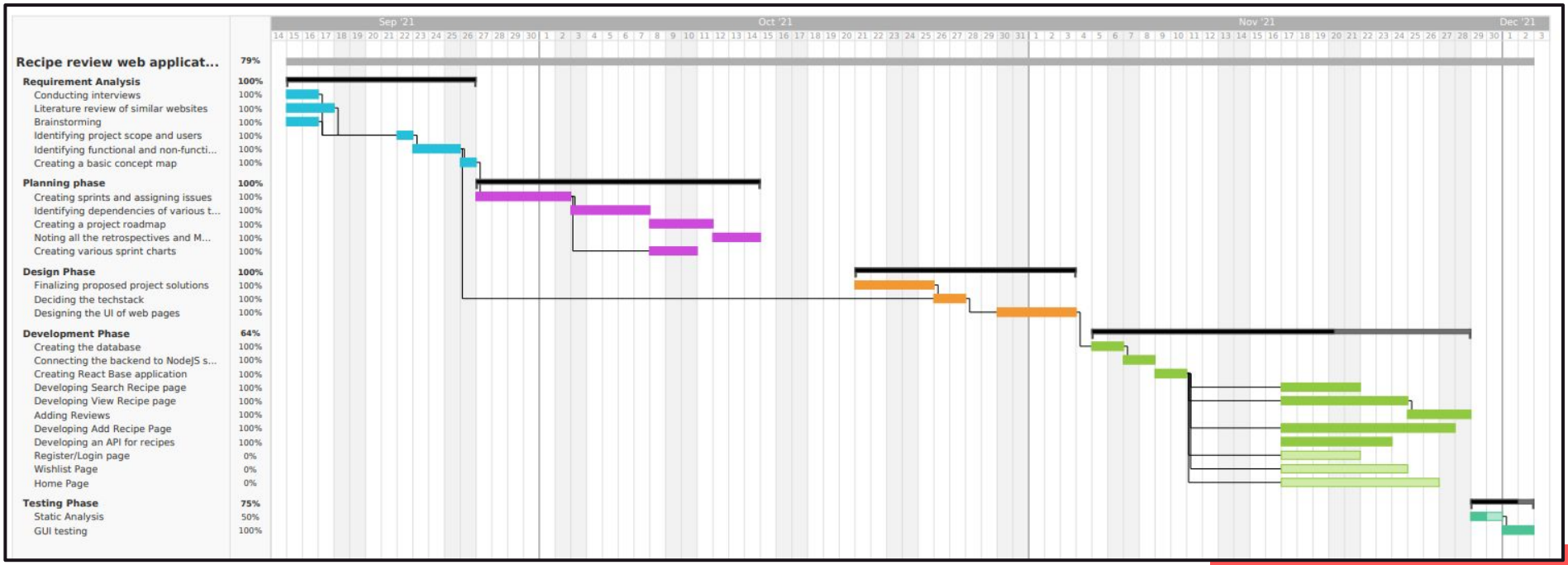


Tasks performed

1. **Requirement Analysis:** Interviews, Literature review, Brainstorming
2. **Planning:** Created sprints and issues, Roadmaps, Identified dependencies
3. **Designing:** Finding solutions of the problem using technologies and creating UI
4. **Backend development:** Using MongoDB as database with NodeJS server to store recipes
5. **Frontend development:** Developing different web pages using ReactJS
6. **GUI testing:** Checking GUI responsiveness



Project Plan

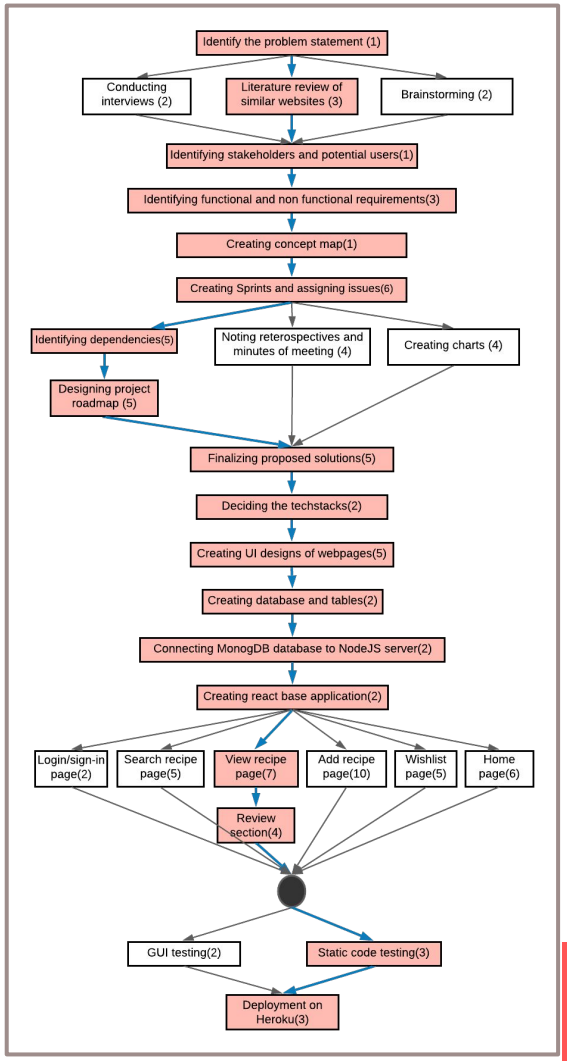


Critical Path

The activities in the in the diagram has their duration written in the bracket daywise.

The critical path is calculated by finding the longest route with maximum duration depict the optimal plan and duration of the project.

For our project, the total duration calculated from critical path is: **57 days (2 months approx.)**



Assumptions

- The recipes entered by the users are authentic.
- The users don’t fake the data while editing the information.
- The ratings given by the users are from 0-5 stars and the average review is also represented as a whole number between 0-5.
- The users prefer to add the recipes so as to expand the search API

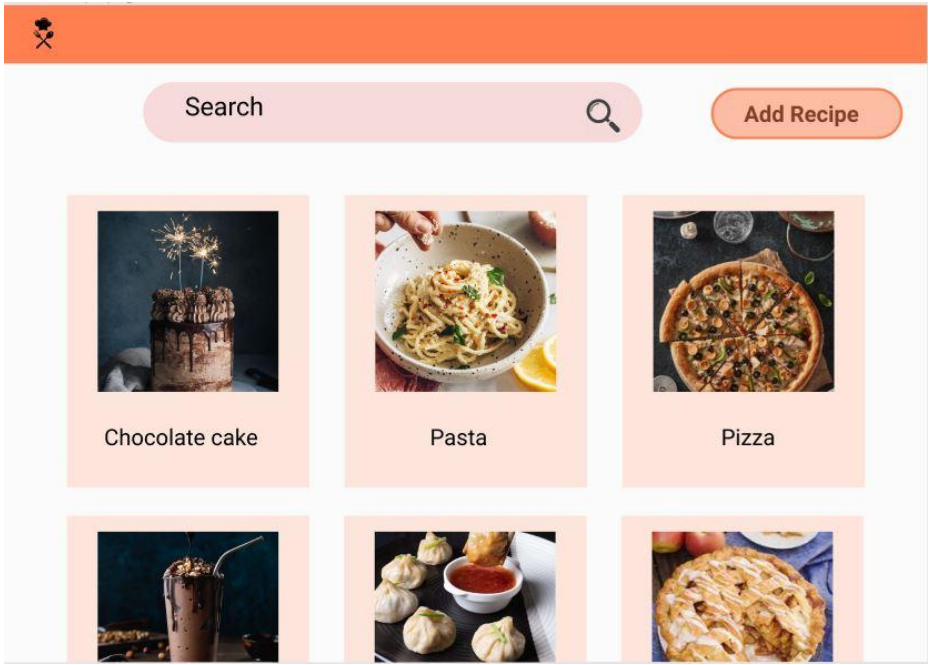
Design and Implementation Constraints

- Using self-built API with limited recipe data instead of pre-built API
- Due to limited data, unable to apply filters for search results.
- We were unable to convert the UI designs on Figma to JS code using available plugins and else were paid.
- Due to time constraints, we left some of the issues unresolved.

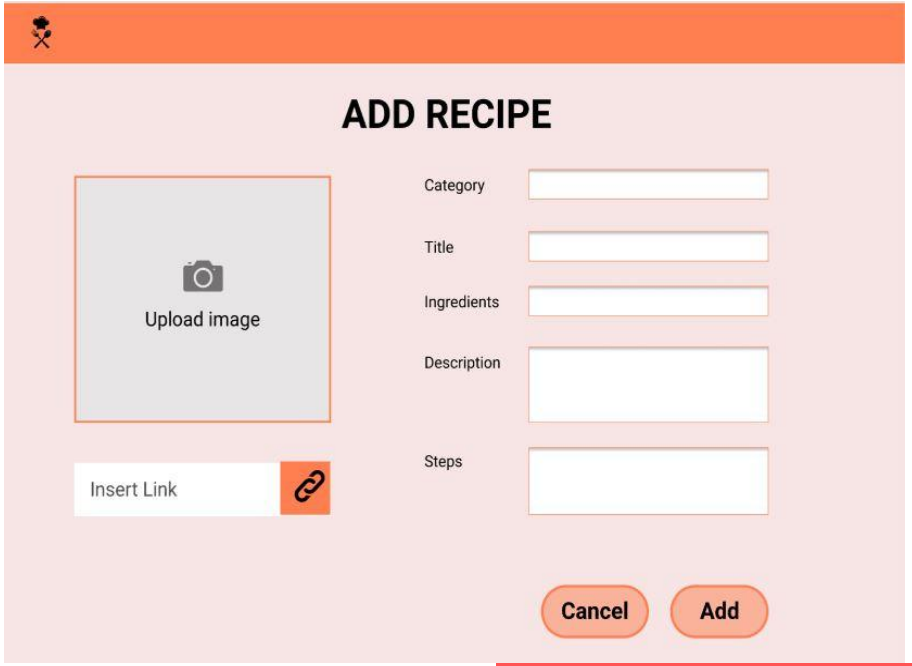
Risks and its mitigation

RISKS	MITIGATION STRATEGIES
The website is highly user-dependent i.e. the database expands only when the user adds recipes.	We can create an API consisting of a huge number of verified recipes with the description. This will help the users to get the desired recipe.
Users might not trust numerical ratings to judge a recipe.	We can add a comment section wherein the user can add their respective reviews for each recipe and using NLP and ML techniques analyze the overall score of the recipes.

UI/UX Designs



Home Page



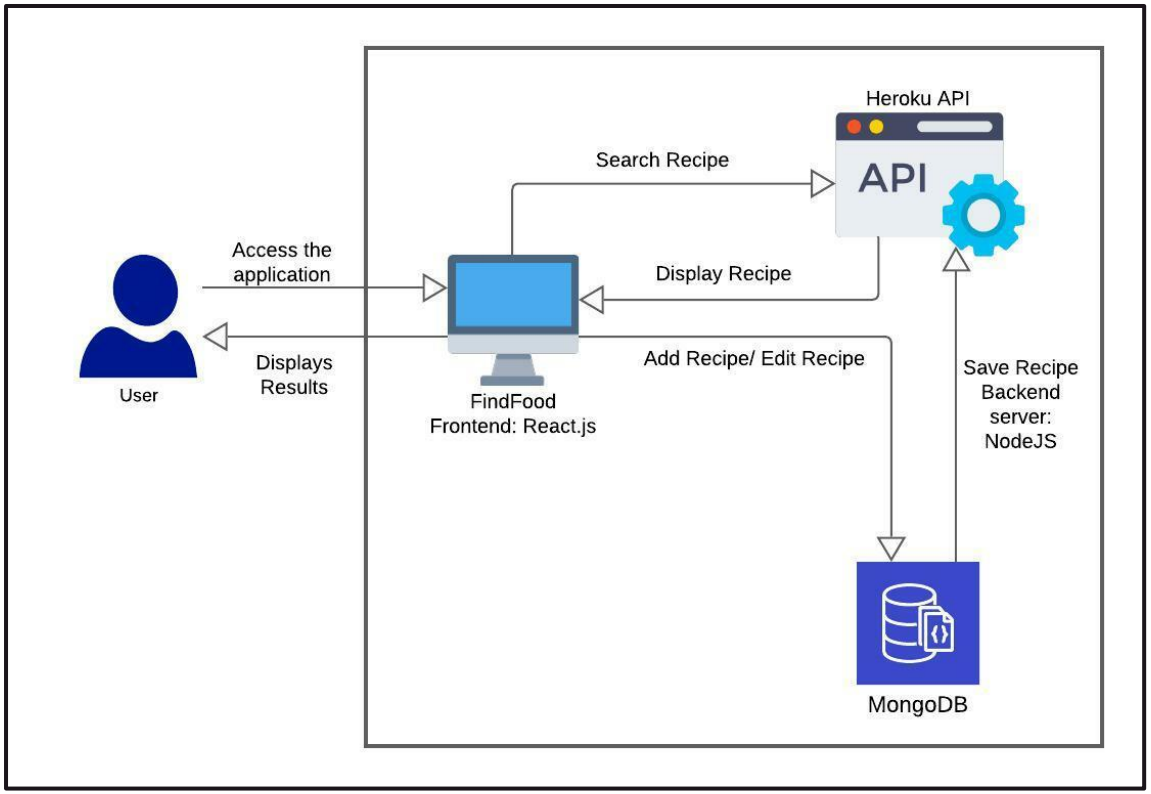
Add Recipe Page

UI/UX Designs

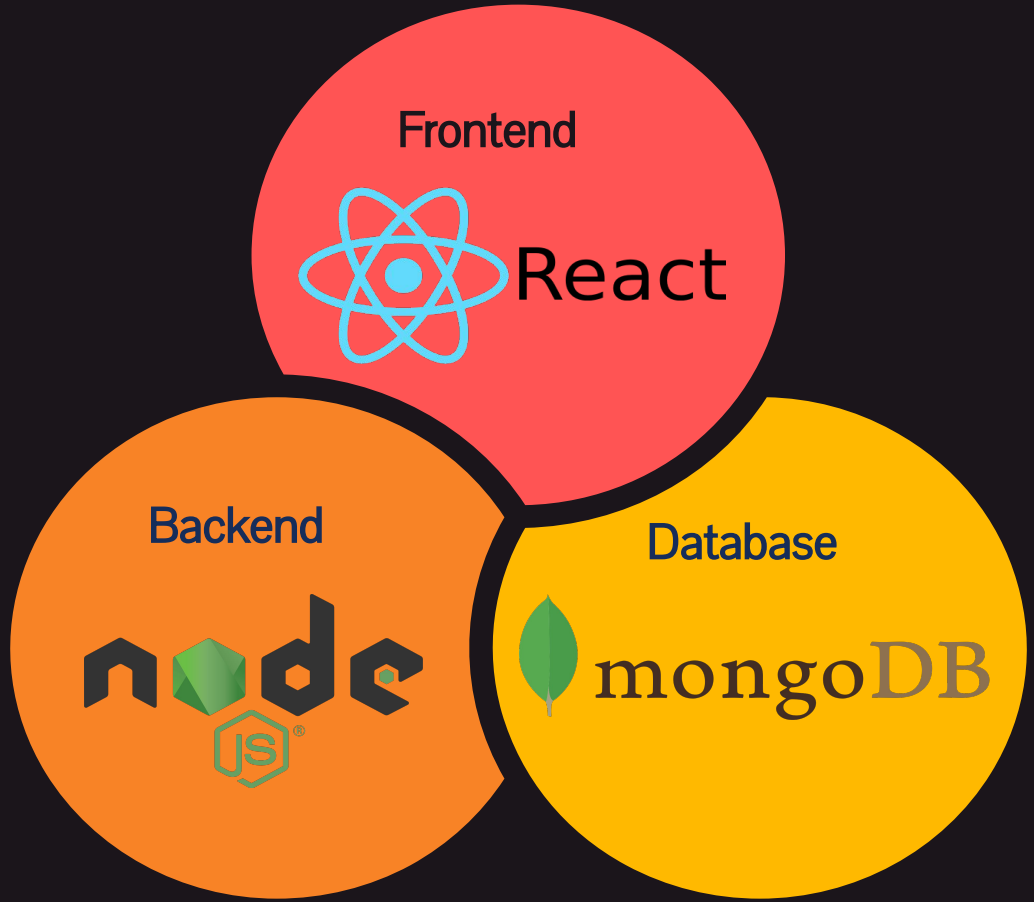
View recipe Page

View recipe Page

System Architecture Model



Techstack used



Future enhancements

- The recipe review web application can be enhanced by allowing the users to enter textual reviews to get a better judgment about the recipe.
- More filters can be added to search like filtering based on recipe category, ingredients, veg or non-veg, cuisine type, etc.
- The API can be expanded more by adding authentic recipes so that a wider range of searches is possible.

Unresolved Issues

- We decided to implement a wishlist page wherein the user can add his/her favorite recipes.
- For this, we needed users to register themselves. So, we decided to do it using Google Login API but we were unable to do so.
- Due to limited API, we were unable to sort the recipes based on reviews and also apply filters to the search results.

Work distribution

NAME	CONTRIBUTION IN PROJECT IMPLEMENTATION	CONTRIBUTION IN PROJECT DOCUMENTATION
Pragati Sharma (202012047)	• Collected and analyzed the requirements of the project. • Designed the UI of web pages • Created the project roadmap • Developed dynamic search page. • Developed Add and edit recipe page. • Added Data to the API	• Managed the workflow of the entire project. • Added UX/UI designs. • Created System architecture diagram • Added design and implementation constraints. • Highlighted unresolved issues.

Work distribution

NAME	CONTRIBUTION IN PROJECT IMPLEMENTATION	CONTRIBUTION IN PROJECT DOCUMENTATION
Mahi Patel (201801039)	• Collected and analyzed the requirements of the project. • Created and assigned dependencies to the project tasks • Created various charts to understand project progress. • Created the database to store recipe data. • Integrated MongoDB to NodeJS server • Created an API in Heroku	• Added system information • Added the assumptions made • Identified potential risks and their mitigation strategies. • Created Project plan along with the description of various tasks.

Work distribution

NAME	CONTRIBUTION IN PROJECT IMPLEMENTATION	CONTRIBUTION IN PROJECT DOCUMENTATION
Vanshika Kochhar (201801104)	• Collected and analyzed the requirements of the project. • Noted retrospective of sprints and minutes of meetings • Created concept map • Created ReactJS application. • Developed View recipe page along with the timeline. • Developed the mechanism to assign reviews to the recipe. • Added Data to the API	• Added project background • Created concept map • Created use case and activity diagram • Noted the task dependencies. • Added the future enhancements

Important Links

- Github : FindFood
- Project Report
- Requirement Analysis Document
- UI/UX Designs
- Concept Map
- Use Case Diagram
- Activity Diagram
- System Architecture Model
- Project Plan

THANK YOU!