

Recipedia

Team 36

Timothy To, Yiyi Zhang, Sean Zurcher

## **Project Overview**

- Provide descriptions of the Based Database Application your team will be developing.

Who are the stakeholders? Why is it important?

Our project will be an implementation of a recipe book for casual home use. It will allow users to interact with it in much the same way one would interact with a physical cookbook, mainly by looking up recipes that have been stored in a database. Users will be able to search for specific recipes by directly searching for names or by filtering them based on ingredients, preparation time, dish category (main, side, dessert, etc.), serving size and alphabetization.

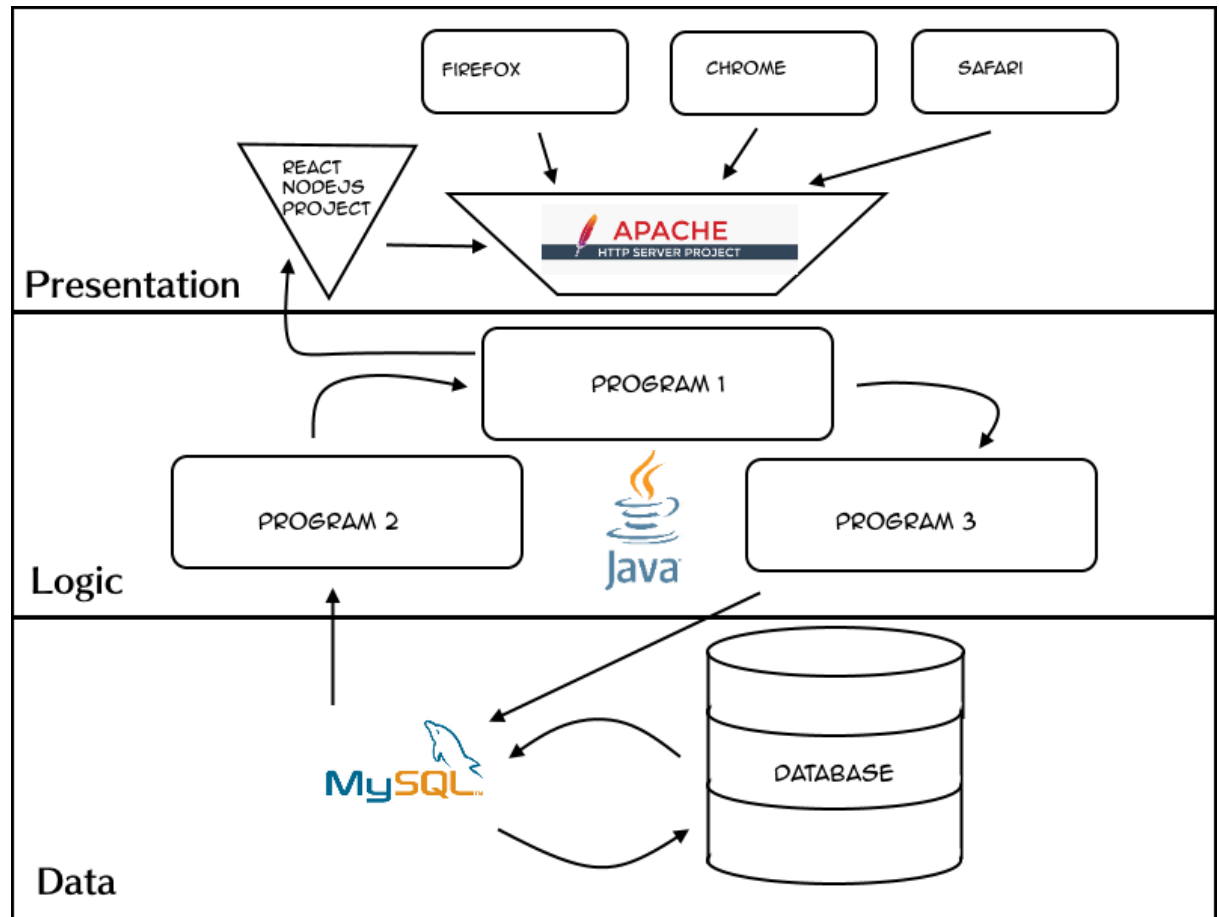
Additionally, users can expand the cookbook by adding their own recipes to the database. Users should also be able to favorite recipes and have those appear on their own separate filter for themselves. Users should be able to rate recipes and on the front page the highest rated recipes should show up. Users should be able to register accounts using an email, username and password to save their own recipes and their favorite recipes.

Stakeholders in our project include anyone who is interested in cooking either in learning cooking with simple to use recipes or simply veteran with too many recipes to keep track of. This project will function as an easy tool for home cooks to keep track of their existing recipes, find new popular recipes, and reference them in the future. Our project will act as an entry point for home cooks to broaden their knowledge and to find new and exciting foods to explore.

## **System Environment**

- A Structure Diagram of the system (graph/diagram based on 3-tier architecture)

- Hardware: It should ideally be able to run on three separate machines and still all connect fine, but for the sake of the project we'll be hosting it all on one computer.
- Software: Any web browser client should be able to access the front end. We'll be using Apache Tomcat for the server that hosts the servlets and MySQL for the database management system.
- RDBMS used: We will be using MySQL for our RDBMS, as stated by the project assignment requirements.
- Programming/Application languages used: For our front end/presentation level, we will use React.js which combines JavaScript, HTML, and CSS to create a nice presentable user interface, and will be hosted by Apache Tomcat. For our backend/logic level, we will use Java and Java servlets. Finally, for the data level, we will use MySQL to create our database.



## Functional Requirements

- Describe users and how users access the system.

Users will be able to search for recipes by different filters. According to users' requirements, system will select the appropriate recipes from database, and list them on the web page. Also, users can create their own account through entering their personal information, such as name and email, to access the recipe book. By logging/logging out of their account, they can access more elaborated functionalities like saving their favorite

recipes, creating/deleting their own specific recipes, and rating the recipes. These will also be recorded in the database.

- Describe each functionality/features, functional processes, and I/O(s).
  - ➔ Register: Users need to register the system by entering their username, password, and email to access the recipe book. The information will be kept in the database. There should be a register button available that leads to a register page. On that register page should be user inputs that allow for the username, password and email to register with.
  - ➔ Login: The users who have registered need to login to access more performance. It will use standard login with username and password.
  - ➔ Edit Profile: Users will be able to edit the profile they saved including their username and password. There should be a button on the user profile page to update the profile page. Then it should have inputs to change the username and password for the user.
  - ➔ Search Recipe: Users will be able to search for a recipe by filtering them based on titles, ingredients, preparation time, dish category (main, side, dessert, etc.), serving size and alphabetization. There should be a default quick search bar input on the homepage somewhere where it works simply based off title. That should be able to be expanded into an advanced search with all the sorts of filters and what not.
  - ➔ Rate Recipe: Users can rate recipe according to their own tastes. The rating would be a user input with a possible score out of five. The rating should remain unseen

until it reaches a sufficient amount of ratings to be an okay average. That rating should be displayed on the recipe page.

- ➔ View Recipe: Users will be able to see the recipes posted by other users and ordered by rate. It should open up the recipe page just by clicking on the desired recipe.
- ➔ Create Recipe: Users will be able to add their own specific recipes. The creation of recipe include entering title, ingredients, preparation time, category and description, steps and other information. There should be a create recipe button somewhere and then have input space to enter all relevant information to the recipe.
- ➔ Delete Recipe: Users can also delete the recipe they created. It should be available to click to delete on the recipe page and have a confirmation button to delete it.
- ➔ Add Favorite: Users can mark the recipes as their favorite. There should be a button on each recipe page that's nice and visible and favorites on button click. That adds the recipe to the user's list of favorites
- ➔ View My (Favorite) Recipe: Users will be able to view all recipes they created/marked as favorite. It should display all the recipes in a list and users should be able to both unfavorite recipes from the list and open up the recipe for viewing.

## **Nonfunctional Issues**

- GUI done in React using HTML and CSS.

- The application will be easy to use, and has a user-friendly interface which is simple and clear. The interface contains clear navigation with buttons and plain view part helps users understand and find the recipe they want. Users will accomplish their goal easily and quickly with few or no user errors.
- Our GUI will be implemented by using a combination of React.js and Java servlets. The servlets, in cooperation with our Java backend, will produce the data that React.js will organize and display to the client.
- The homepage of Recipedia will display highly-rated recipes. The homepage will also have a login prompt, and after a user logs in, the homepage will also display their favorite recipes and their own recipes.
- By entering a query into the search bar, users will see recipes that have been added into the database, ordered by whichever filter restraints they add to the search.
- Upon choosing a recipe, the recipe name, serving size, cooking and preparation time, ingredient list, and instructions will be displayed to the user. The recipe can also be rated and favorited on this page.
- When writing new recipes, users will have fields to enter the recipe name, serving size, cooking and preparation time, ingredient list, and instructions.
- Security
  - The system will be able to automatically log out all customers after a period of inactivity. Also, the system should not leave any cookies on the customer's

computer containing the user's password. The system will be protected from any external danger or attacks.

- Access Control
  - Account access control can be done with cookie based authentication. The system's back-end servers shall only be accessible to authenticated administrators.
  - Users may create and edit their own recipes, and editing rights are given only to the accounts that created them.
  - Users may rate and favorite recipes, and changing these attributes is restricted to each individual user account.