

Stony Brook University

AI-Powered Recipes Recommendation Application

Hyomin Kim

hyomin.kim@stonybrook.edu

Nahyun Kim

nahyun.kim.4@stonybrook.edu

TABLE OF CONTENTS

I. Introduction.....	3
A. Product Description.....	3
B. Scope.....	3
C. Users.....	4
D. User Feedback.....	5
E. Existing Alternatives.....	5
F. Definitions.....	6
G. References.....	7
II. Requirements.....	7
A. Functional Requirements.....	7
B. Use Cases.....	8
C. User Interfaces.....	11
D. Non-functional Requirements.....	11
III. Contributions.....	13
A. Hyomin Kim.....	13
B. Nahyun Kim.....	13
C. Both Members.....	13

I. Introduction

A. Product Description

Smart AI Recipes is an AI-powered web application designed to help users find and follow recipes based on the recipe's name or the ingredients they already have at home. The application aims to reduce food waste, save time, and promote healthier eating habits, particularly for individuals who struggle with meal planning.

Many people, especially those living alone, often find it difficult to cook meals due to a lack of ingredients or recipe ideas. As a result, they frequently rely on expensive and unhealthy delivery food. Our solution addresses this issue by providing an AI-powered recipe suggestion system. Users can input the ingredients they have in their fridge, and our system will analyze available recipes to suggest the best matches.

In addition to recommendations, our platform provides step-by-step cooking instructions, required ingredients, cooking methods, and nutritional information, making it easy for anyone to follow along and prepare meals efficiently.

B. Scope

The AI-Powered Recipes Recommendation Application is designed to assist individuals who cook at home, particularly those living alone and encountering difficulties in meal planning. Many individuals struggle with determining what to cook due to a limited selection of available ingredients or a lack of recipe ideas. Our platform aims to mitigate these challenges by providing AI-powered recipe recommendations, thereby enhancing convenience and accessibility in home cooking.

To achieve this objective, we are developing a web-based application utilizing React as the frontend framework. This system employs machine learning techniques to generate personalized meal recommendations based on users' available ingredients, offering an efficient and intelligent cooking solution. Through this approach, our platform helps users make optimal use of their pantry while reducing food waste and promoting healthier eating habits.

The core of our recommendation system is powered by TensorFlow, a widely recognized and robust artificial intelligence framework. By leveraging TensorFlow, we can train and implement an advanced recipe recommendation algorithm that delivers accurate and contextually relevant meal suggestions. This ensures that users receive meal options that align with their dietary preferences and ingredient availability, thereby optimizing their cooking experience.

In addition to recommendation accuracy, the application prioritizes security and user authentication. To safeguard user data and provide a seamless login experience, we have integrated Google OAuth authentication. This ensures that users can securely access their accounts while maintaining data privacy and protection against unauthorized access.

Overall, this project aims to integrate artificial intelligence into meal planning in a manner that enhances user convenience, promotes healthier eating choices, and minimizes food wastage. By combining an intelligent recommendation system with a user-friendly interface, the AI-Powered Recipes Recommendation Application seeks to transform the way individuals approach home cooking, making it more efficient, enjoyable, and accessible.

C. Users

The AI-Powered Recipes Recommendation Application is designed to cater to a wide range of users who encounter difficulties in meal planning and recipe selection. Home cooks who prepare meals regularly often struggle to come up with new ideas based on the ingredients they have on hand. Individuals living alone, particularly single-person households, may find it challenging to plan and prepare meals efficiently, often resulting in food waste or reliance on unhealthy convenience foods. Health-conscious individuals seeking to maintain a balanced diet benefit from the application's ability to provide nutritional insights along with meal recommendations. Additionally, beginner cooks who lack experience in the kitchen can utilize the platform's structured guidance to follow step-by-step recipes with ease. Moreover, busy professionals and students who have limited time for meal preparation can rely on the application for quick and easy-to-follow recipe suggestions tailored to their needs.

To meet the diverse needs of these users, the application provides an intuitive recipe recommendation system that allows users to input available ingredients and receive relevant meal suggestions. Alongside this, it offers detailed nutritional information to support informed eating choices. The user-friendly interface ensures accessibility for individuals of all skill levels, while Google OAuth authentication enhances security and enables a personalized experience. By addressing these specific requirements, the AI-Powered Recipes Recommendation Application streamlines the meal planning process, making cooking more efficient, accessible, and enjoyable for a broad spectrum of users.

D. User Feedback

The solicitation of input from outside individuals will be most applicable in the prototype testing phase of our solution, for which feedback on the UI design and core features of our product will be collected via usability tests with a cohort of our target users. We found that many users struggle to keep track of what's in their fridge and often feel uncertain about what meals to cook.

From the midterm presentation, we also received useful feedback such as making the UI more user-friendly to clearly differentiate it from ChatGPT and adding a rating system so users can evaluate the recipes. Additionally, users highlighted the importance of having a user-friendly UI, which can set our app apart from AI tools like ChatGPT. This feedback helped us realize the importance of standing out through better user interaction and intuitive design. Users emphasized that having a clean and simple interface could significantly improve their experience and make our app more approachable compared to existing AI tools. And with a rating feature, they can share their thoughts on the recipes they've tried, which also helps future users find the most popular or helpful recipes more easily. These insights have not only influenced our current feature set but have also shaped how we prioritize user experience in future updates.

E. Existing Alternatives

While working on our project, we looked at *Simply Recipes* as an existing recipe website to see what it does well and what it could do better. One of its biggest strengths is its huge collection of recipes. Each recipe includes step-by-step instructions and high-quality photos, which make it easy to follow. The site also shares extra content like recommended grocery items and food-related news.

However, we also noticed some problems. There are too many features and categories and they are confusing. It's hard to know where to click or how to find what you're looking for. Also, there's no way to search by ingredients you already have, and makes it less helpful if we're just trying to figure out what to cook with what's in our fridge. The website doesn't suggest recipes based on each taste either, and sometimes it feels like too much information is on one page to focus on.

These things helped us make better choices for our app. We decided to keep our design simple and clean, add an ingredient-based recipe search, and include a personalized

recommendation system that learns what users like to eat. By learning from the strengths and weaknesses of *Simply Recipes*, we were able to build a tool that's easier to use and more helpful for everyday cooking.

F. Definitions

1. AI (Artificial Intelligence): A branch of computer science that enables machines to perform tasks that normally require human intelligence, such as making decisions or recommendations. In this project, AI is used to voice recognition and recommend recipes that match the user's ingredients or preferences.
2. React: A JavaScript library used to build user interfaces, especially for web applications. We used React to create a fast and interactive front-end for our recipe application.
3. TensorFlow: An open-source platform developed by Google for building and training machine learning models. Our system uses TensorFlow to train the model that powers the recipe recommendation engine.
4. Google OAuth: A secure login system that lets users sign into apps using their Google account. It allows users to log in easily without creating a new account, while keeping their data secure.
5. UI (User Interface): The part of the application that users see and interact with. Our app focuses on a clean and simple UI so users can find recipes quickly and easily.
6. UX (User Experience): The overall experience a user has while using a product or service. We prioritized good UX by reducing complexity and focusing on user-friendly design.
7. Recipe Recommendation System: A system that suggests recipes based on user preferences or available ingredients. Our system suggests the best possible recipes based on what users have in their fridges.

G. References

1. Simply Recipes. (n.d.). *Dinner Ideas & Recipes*. Retrieved March 22, 2025, from <https://www.simplyrecipes.com>

II. Requirements

A. Functional Requirements

1. Recipe Search by Name: Users can search for recipes by entering the name of a dish. The application will retrieve and display relevant recipes that match the entered name, allowing users to quickly find the specific dish they wish to prepare.
2. Ingredient-Based Recipe Search: If users do not know what to cook, they can input the ingredients they have into the search bar. The application will filter and suggest recipes that can be made using those ingredients. For example, if a user enters eggs, cheese, and tomatoes, the system may suggest recipes such as an egg and cheese sandwich.
3. Smart Recipe Filtering: The application employs an intelligent filtering system to refine recipe suggestions. Users can filter recipes based on available ingredients, meal type, and cooking preferences. This ensures that the recommendations align with the user's needs and constraints.
4. Categorized Recipes: To enhance usability, recipes are systematically categorized based on meal types. This classification simplifies the selection process, making it easier for users to browse and choose a suitable dish.
5. Step-by-Step Cooking Instructions: Each recipe includes detailed and easy-to-follow instructions to guide users through the cooking process. These instructions specify the main cooking method, such as frying or boiling, and outline each step clearly to help users, including beginners, successfully prepare the dish.
6. Nutritional Information Display: The application provides comprehensive nutritional information for each recipe. This includes data on calories, carbohydrates, protein, fat, and sodium content. Such information is

particularly beneficial for users who are health-conscious or tracking their dietary intake.

7. **User Account Management:** Users can create accounts to save their favorite recipes, allowing them to easily revisit dishes they enjoyed. The account system ensures convenience by storing user preferences and previously accessed recipes.
8. **Secure Authentication via Google OAuth:** To protect user data and enhance security, the application integrates Google OAuth authentication. This allows users to sign in using their Google accounts, ensuring a private and secure login process.
9. **Recipe Difficulty and Cooking Time Estimation:** Each recipe is assigned a difficulty level and an estimated cooking time. This helps users gauge the complexity of the dish and plan their cooking accordingly.
10. **Dynamic Ingredient Adjustment:** The system includes a feature that adjusts ingredient quantities based on the number of servings selected by the user. This ensures accurate portioning and reduces food waste.
11. **Personalized Taste Preferences:** By utilizing the login system, the application can store user taste preferences and provide personalized recipe recommendations. Users can set additional filtering options based on their dietary habits and ingredient preferences.

B. Use Cases

Use Case 1: Viewed Detailed Recipe Instructions without searching recipe's name or ingredients

- **Primary Actor:** Any user and System
- **Priority:** Essential
- **Scenario:**
 - The recipe on the homepage shows the high rated recipe if the user isn't logged-in.
 - The user selects a recipe from the search page or homepage.
 - The system loads the recipe page.
 - The page shows the ingredients, step-by-steps cooking instructions, nutritional information, and rated system.

- Extensions:
 - If a category page fails to load due to some reason, the system displays a retry option.

Use Case 2: Ingredient-Based Recipe Recommendation

- Primary Actor: Logged-in user and System
- Priority: Essential
- Scenario:
 - The user navigates to the search page.
 - The user clicks the button which can change the name based search bar to the Ingredient-based search bar.
 - The user enters ingredients in the input field (e.g. egg, cheese, tomato).
 - The system displays the list of recipes.
 - The user selects a recipe to view full details.
 - The page shows the ingredients, step-by-steps cooking instructions, and nutritional information.
- Extension:
 - If there is no recipe regarding the user's search input, the system displays a message: "There is no search result."

Use Case 3: Filter Recipes by User's Preferences

- Primary Actor: Logged-in user and System
- Priority: Expected
- Goal in Context: To recommend the recipes that match the user's preferences.
- Preconditions: The user is logged in.
- Scenario:
 - The system automatically retrieves the user's saved preferences from their profile.
 - The AI-based recommendation engine uses the user's preferences and available data to generate a list of suitable recipes.
 - The user can see the system's recommendations on the homepage.
 - The user selects one to see full details.

- Extensions:
 - If the user has not set any preferences, the system generates high rate recipes with a message: “Set your preferences for our recommendation!”
 - If no perfect match is found for recommendations, the system shows the closet matches.

Use Case 4: Rate and Review a Recipe

- Primary Actor: Logged-in User and System
- Priority: Expected
- Preconditions: The user has selected a recipe and made it before.
- Scenario:
 - On the recipe details page, the user can see the button for rate.
 - The page displays the Like/DisLike button.
 - The system updates the rating and the review.
 - Other users can see the feedback or rate on the recipe page.
- Extension:
 - If the submission fails, the system displays the message: “Fail to submission. Try again”
 - If a non-logged-in user tries to rate the recipes, the system displays a message: “This is for the logged-in user. Please log-in first.”

Use Case 5: Browse Recipes by Category

- Primary Actor: Any User
- Priority: Nice to have
- Scenario:
 - The user navigates to the “Categories” section from the navigation bar or homepage.
 - The user can see a list of categories such as Main Dish, Side Dish, Soup, Rice, and Desserts.
 - The user selects one category from the list.
 - The system displays a list of recipes that belong to the category.

- The user can see more recipes by scrolling down and clicking the recipe to view the details.
- Extension:
 - If a category page fails to load due to some reason, the system displays a retry option.

C. User Interfaces

1. Home screen: Displays a search bar where users can input ingredients or recipe names. Showcases recommended recipes based on trending dishes and user preferences. Provides quick access to recently viewed and favorited recipes.
2. Search screen: Allows users to enter ingredients they have at home to generate suitable recipe suggestions. Features filters for dietary preferences, cooking time, and difficulty level. Displays search results in a visually structured format with images, titles, and brief descriptions.
3. Recipe detail screen: Presents step-by-step cooking instructions with an intuitive layout. Lists required ingredients along with possible substitutions. Displays nutritional information such as calories, protein, and fat content. Includes interactive elements like a "Start Cooking" mode that guides users through the cooking process.
4. User information screen: Enables users to bookmark recipes for future reference. Provides an organized view of saved recipes categorized by type.

D. Non-functional Requirements

1. Performance and Responsiveness: The application should provide recipe search results within two seconds to ensure a seamless user experience. The system should handle up to 100 concurrent users without performance degradation.
2. Scalability: The architecture should be designed to support future expansion, including an increased number of users and additional features.

Cloud-based infrastructure should be used to ensure dynamic resource allocation and load balancing.

3. **Security:** All user authentication and authorization processes must be handled securely through Google OAuth to protect personal data. User data must be encrypted both in transit and at rest using industry-standard encryption methods. The system should comply with relevant data protection regulations, such as GDPR.
4. **Usability and Accessibility:** The user interface should follow best practices in UI/UX design to ensure ease of use for individuals of all skill levels. The application should be accessible to users with disabilities, following WCAG 2.1 guidelines.
5. **Maintainability and Modularity:** The system should be developed using a modular architecture to facilitate future updates and maintenance. The codebase should follow best practices and be well-documented for easy troubleshooting and enhancements.
6. **Reliability and Availability:** The system should maintain a 99.9% uptime, ensuring availability for users at all times. Automated backups should be performed daily to prevent data loss in case of system failure.
7. **Cross-Platform Compatibility:** The application should be optimized for both desktop and mobile devices, ensuring consistent performance across different screen sizes. It should be compatible with modern web browsers, including Chrome, Firefox, Safari, and Edge.
8. **Data Integrity and Consistency:** Recipe data and user preferences should be consistently updated and synchronized to prevent discrepancies. Changes made by a user should be reflected in their profile and recommendations in real-time.
9. **Localization and Internationalization** The measurement units in recipes should be adjustable based on the user's location (metric/imperial system).
10. **Logging and Error Handling:** The system should log all critical operations and errors for debugging and security auditing purposes. Error messages should be user-friendly and provide clear instructions for resolving issues.

III. Contributions

A. Hyomin Kim

1. Researched secure account management and methods to strengthen login-related security, including OAuth integration
2. Define the Scope section, clearly outlining the project's purpose, target users, technical architecture (React + TensorFlow), and key goals such as reducing food waste, supporting solo cooks, and integrating AI for smarter meal planning
3. Researched non-functional requirements, including performance, scalability, security, accessibility, and cross-platform compatibility, and helped define them in detail

B. Nahyun Kim

1. Shared several outside sources and applications similar to our project, which helped the team shape both the UI design direction and functional requirements based on common patterns and user expectations
2. Searched for and evaluated open APIs related to recipes for possible integration into the system
3. Sketched simple mockups by hand to visualize the app layout and key features

C. Both Members

1. Collaborated on brainstorming the Functional Requirements section
2. Worked together to prepare the Midterm Presentation, including slides and content planning