

# CPSC 304 Project Cover Page

Milestone #: 1

Date: 10.01.2024

Group Number: 107

Name	Student Number	CS Alias (Userid)	Preferred E-mail Address
Keisuke Yamamoto	39088984	g1d4h	ykei2356@gmail.com
Yuto Kikuta	32572265	v4k9h	jordan2002222@gmail.com
Seokhee Hong	91166660	t9z3e	tjrgml1207@naver.com

By typing our names and student numbers in the above table, we certify that the work in the attached assignment was performed solely by those whose names and student IDs are included above. (In the case of Project Milestone 0, the main purpose of this page is for you to let us know your e-mail address, and then let us assign you to a TA for your project supervisor.)

In addition, we indicate that we are fully aware of the rules and consequences of plagiarism, as set forth by the Department of Computer Science and the University of British Columbia

# **1. The Domain of our group's application**

The domain of the application is recommendation and management for culinary and recipe through AI. The focus is on helping users discover, create, and manage recipes based on their dietary preferences, available ingredients, and nutritional needs. The application aims to cater to home cooks, food enthusiasts, and individuals with specific dietary restrictions, such as allergies, veganism, or keto. The goal is to assist users in planning meals, trying new cuisines, and adhering to health-conscious or ingredient-based constraints.

## **2. Aspects of the domain modeled by the database**

### **1. Ingredient-based Recipe Recommendation**

The system is designed to provide ingredient-based recipe recommendations. Users can input ingredients they have on hand, and the application will search for recipes that include these ingredients, helping reduce food waste and making meal planning easier.

### **2. Ingredient management**

The database stores information about various ingredients and their potential allergens. This aspect addresses the need for quickly obtaining the information about the ingredients.

### **3. Recipe storage and categorization**

The database structures recipes with attributes like title, instructions, servings, cooking time, and nutritional details. It categorizes them by cuisine, such as Italian or Japanese, and type, like dessert or fast food, making it easier for users to explore and filter recipes based on preferences.

### **4. Allergen and Dietary Restrictions Management**

The database models common allergens and dietary restrictions to help users with specific dietary needs. This functionality is critical for making the application accessible to users with food sensitivities or ethical food choices.

### **5. Nutritional Information**

Each recipe is associated with a detailed breakdown of its nutritional content. This addresses the need for users to evaluate recipes based on health metrics and dietary goals.

### Real-Life

Suppose a user has a gluten allergy and only has ingredients like rice, chicken, and spinach in their kitchen. The application can search recipes using an AI model to suggest gluten-free recipes using these ingredients and save the recipes to our database. Thanks to the database, users can review their recipes they used before anytime with the useful information. This real-life situation highlights the application's ability to accommodate various constraints, reduce food waste, and make healthy meal choices easier.

## 3. Database specifications

The database will provide functionality to store, retrieve, and manage a wide variety of recipe-related data. Users will be able to look histories for recipes that have been searched using AI and saved to the database, filter recipes by cuisine, category, or dietary restriction, and receive suggestions that meet their health and allergen requirements. Additionally, the database will allow for storing detailed nutritional information, making it easier for users to select recipes that align with their dietary goals. Users can also explore recipes by categories or cuisines.

## 4. Application Platforms

The project will use **PostgreSQL** as the primary database to store and manage the application's data, chosen for its strong support for complex queries, scalability, and integration with the tech stack. The expected technology stack includes **Next.js** and **React.js** for building a dynamic and responsive user interface, along with **TypeScript** to ensure type safety and reduce runtime errors during development. This stack will enable the creation of a full-featured, efficient, and maintainable application.

