COMP 3059 – Capstone Project I

Software Requirements Analysis and Design Assignment

‹MANOS Food Decider›

1.0 Introduction

1.1 Purpose

This document describes the software requirements analysis and design for the MANOS Food Decider application. It outlines the capabilities and features of the system for the intended users and stakeholders.

1.1 1.2 Scope

The MANOS Food Decider application is being developed to provide users with a personalized food recommendation system that considers their dietary restrictions and preferences. The key features in scope include:

- User profile creation and management
- Dietary restriction input and management
- Meal history tracking
- Ingredient-based recipe suggestions
- Shopping list generation
- Recipe and restaurant recommendations based on user preferences

The application will not integrate with wearable devices or fitness trackers, nor will it provide direct ordering of groceries or meal delivery. Advanced nutritional analysis and calorie counting features are also out of scope for this initial release.

2.0 System Overview

The MANOS Food Decider is a mobile-based application that uses powerful learning algorithms to provide personalized meal and recipe recommendations to users based on their dietary restrictions and preferences. The system will integrate with a database of recipes and local restaurant information to generate relevant suggestions for the user.

2.1 Project Perspective

The MANOS Food Decider is a new self-contained system that aims to address the needs of health-conscious consumers who struggle with meal planning and decision-making due to dietary restrictions.

2.2 System Context

The strategic focus is on improving user health outcomes through personalized meal suggestions, efficient grocery planning, and access to suitable restaurant options. By combining these features with a user-friendly interface and robust data security, the system positions itself as a comprehensive solution for individuals seeking to maintain specific dietary lifestyles while reducing the daily stress of meal planning.

2.3 General Constraints

The key constraints for the MANOS Food Decider project include:

- The application is dependent on an active internet connection to fetch data from the central database. Mobile app will be constrained by the availability and performance of the internet connection, as slow or intermittent connectivity can impact the user experience.
- The database that serves as the central data source for the application must have sufficient capacity and performance to handle the increasing volume of user data, recipe information, and restaurant details. As the user base grows, the database may be forced to queue incoming requests, leading to increased latency in data retrieval and potentially impacting the responsiveness of the application.

2.4 Assumptions and Dependencies

The assumptions and dependencies for the Food Decider project include:

- Users will access the application on mobile devices with sufficient hardware resources and performance capabilities
- Users will actively participate in providing accurate and up-to-date information about their dietary preferences and restrictions
- Integration capabilities with external APIs for additional features and services

3.0 Functional Requirements

This section describes specific features of the software project.

3.1 Functional Requirements

3.1.1 User Profile

- The User Profile will hold the user’s data and preferences.
- The user will provide their name, dietary restrictions, and an ingredient list.
- The User Profile will keep a record of the recipes that the user decides to make.
- The User Profile will provide recipe and restaurant recommendations based on dietary restrictions and meal history.

3.1.2 Meal Questionnaire

- The Meal Questionnaire will aid users in narrowing down their recipe or restaurant recommendations based on users' current meal desires.
- The user will complete a simple questionnaire consisting of food category-based questions.
- The Meal Questionnaire will process the users' answers into specific guidelines to find recipes.
- The Meal Questionnaire will provide a list of meal specifications to the Recipe and Restaurant Recommendations functionality.

3.1.3 Ingredient List

- The Ingredient List takes the users’ current household items and returns a list of recipes that can be created.
- The user lists the ingredients they have or wish to use in a recipe.

- The Ingredient List verifies the ingredient name and creates a cohesive list of recipe specifications.
- The cohesive list of recipe specifications is shared with the Recipe and Restaurant Recommender.

3.1.4 Recipe and Restaurant Recommender

- The Recommender takes the users' current meal specifications or ingredient list, with their dietary needs to find and report recipes or restaurants that meet their needs.
- The Meal Questionnaire, Ingredient List, and User Profile provide the users' specific desires and restrictions to the Recipe and Restaurant Recommender.
- The Recommender will process the provided information into a list to search for recipes or restaurants that fulfill the guidelines.
- The Recommender provides users with recipes or restaurants that fit guidelines.

3.2 Use Cases

3.2.1 Use Case #1

- Input ingredients to get recipe recommendations
- As a user, I want to be able to input the ingredients I have at home into the app to get recipe recommendations that help me reduce food waste.

3.2.2 Use Case #2

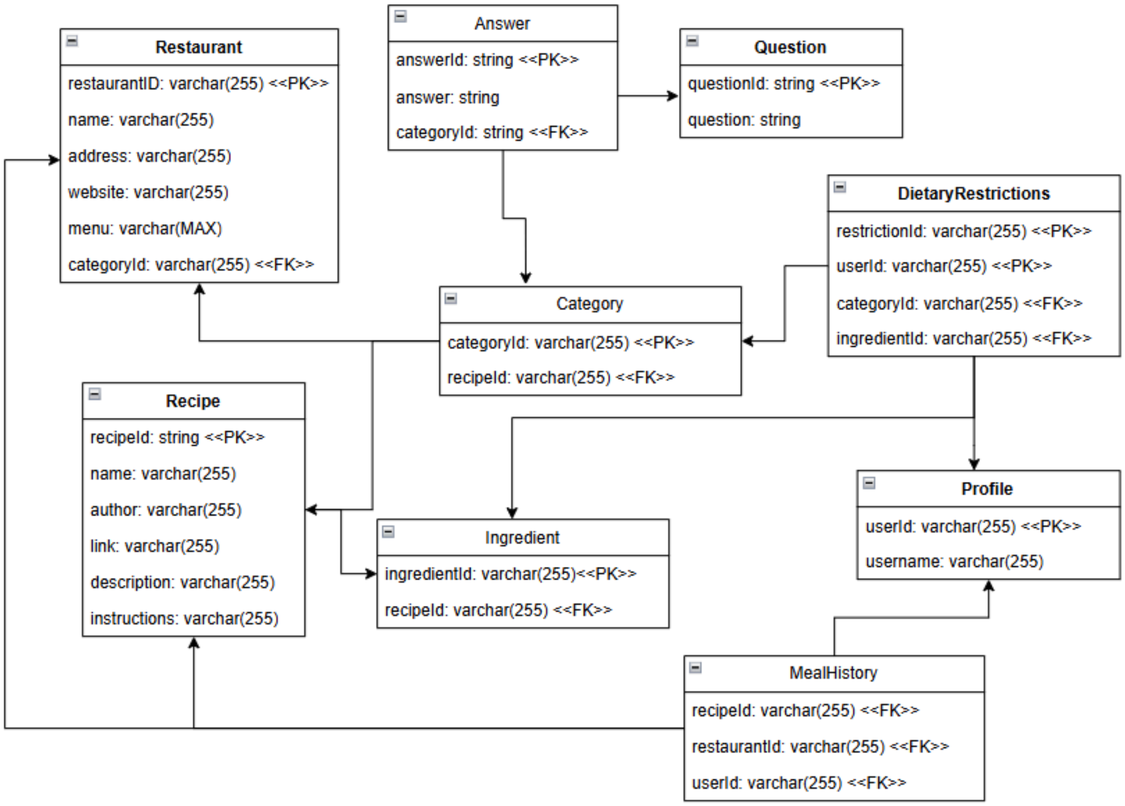
- Explore nut-free, pescatarian recipes
- As a user, I want to explore a variety of nut-free, pescetarian recipes so I can enjoy diverse meals without compromising my health.

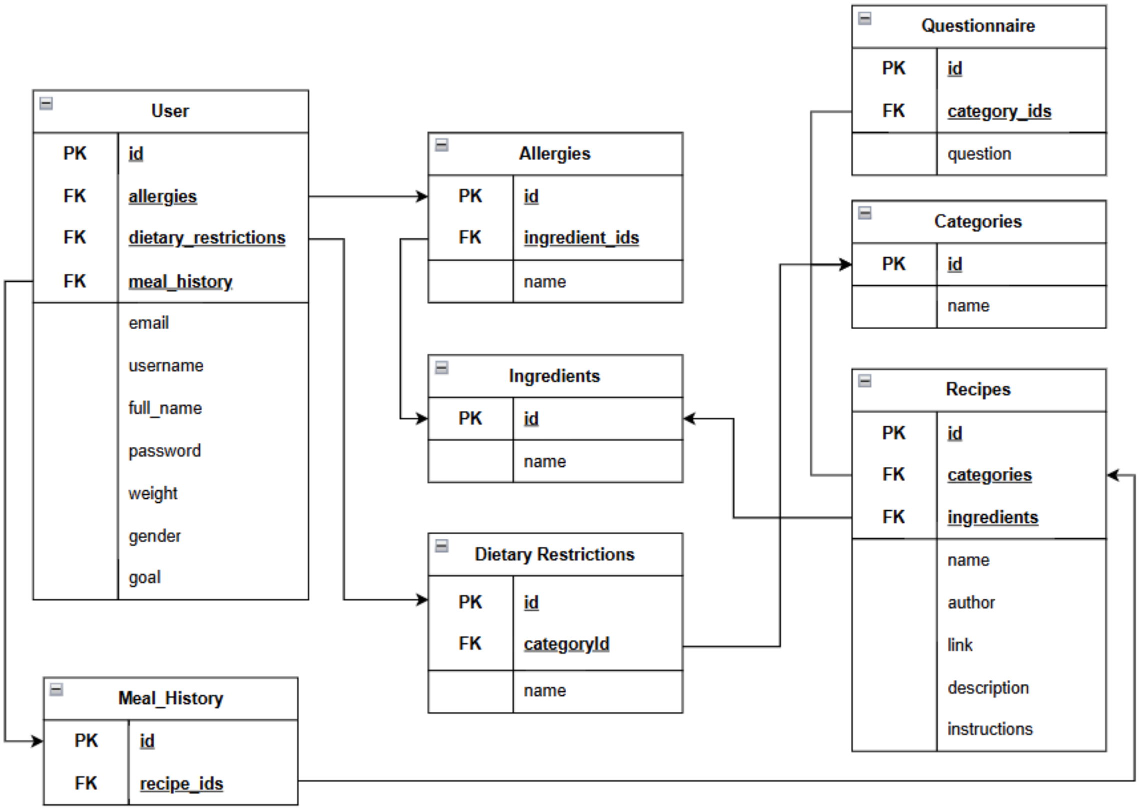
3.2.3 Use Case #3

- Find vegan restaurants with protein-rich options
- As a user, I want to find vegan restaurants that offer a variety of protein-rich options so I can enjoy dining out without compromising my dietary goals.

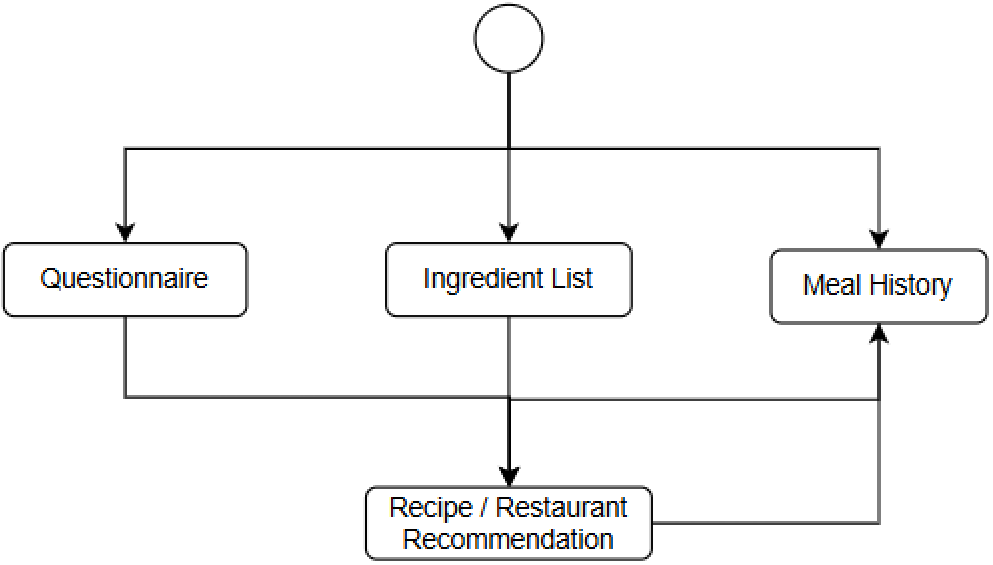
3.3 Data Modelling and Analysis

- Normalized Data Model Diagram

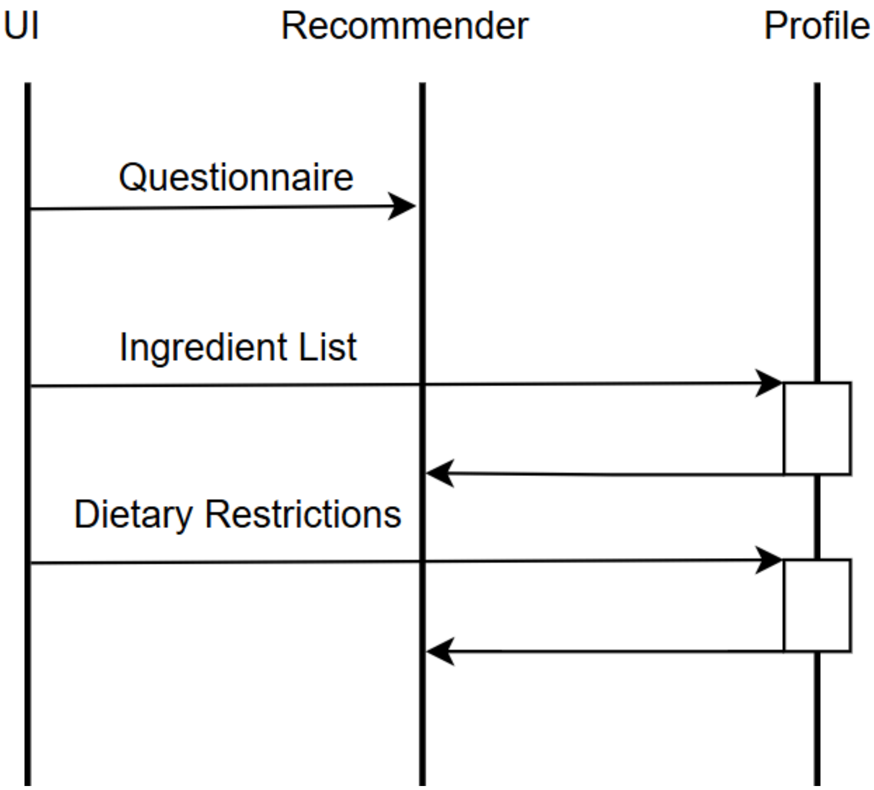

Restaurant
restaurantID: varchar(255) <<PK>>
name: varchar(255)
address: varchar(255)
website: varchar(255)
menu: varchar(MAX)
categoryId: varchar(255) <<FK>>
Answer
answerId: string <<PK>>
answer: string
categoryId: string <<FK>>
Question
questionId: string <<PK>>
question: string
DietaryRestrictions
restrictionId: varchar(255) <<PK>>
userId: varchar(255) <<PK>>
categoryId: varchar(255) <<FK>>
ingredientId: varchar(255) <<FK>>
Category
categoryId: varchar(255) <<PK>>
recipeId: varchar(255) <<FK>>
Recipe
recipeId: string <<PK>>
name: varchar(255)
author: varchar(255)
link: varchar(255)
description: varchar(255)
instructions: varchar(255)
Ingredient
ingredientId: varchar(255)<<PK>>
recipeId: varchar(255) <<FK>>
Profile
userId: varchar(255) <<PK>>
username: varchar(255)
MealHistory
recipeId: varchar(255) <<FK>>
restaurantId: varchar(255) <<FK>>
userId: varchar(255) <<FK>>



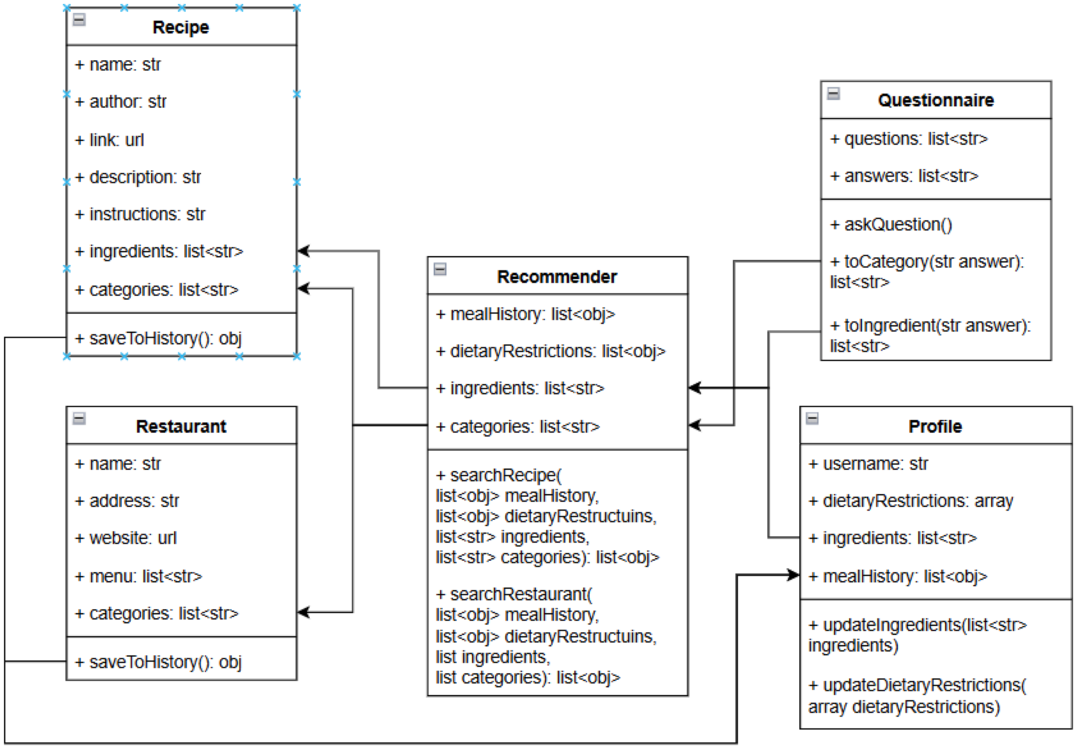
- Activity Diagrams



- Sequence Diagrams

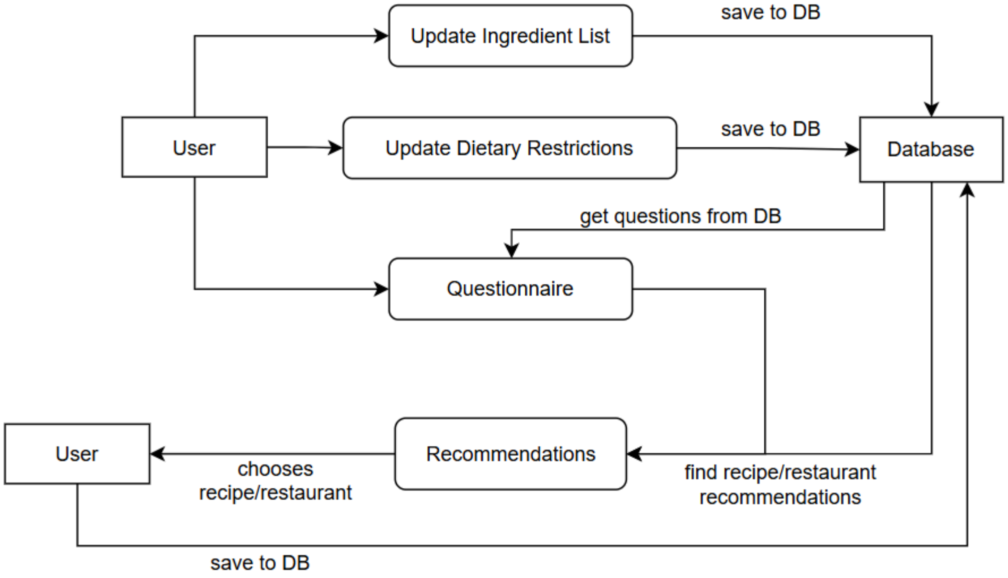


- UML Class Diagram



3.4 Process Modelling

- Data Flow Diagram



4.0 Non-Functional Requirements

4.1 Performance

- The system shall handle up to 10,000 concurrent users without noticeable degradation in performance.
- Database query response times shall not exceed 500 milliseconds under normal load and should not degrade by more than 15% during peak hours.
- The mobile application must load the home screen in under 2 seconds on devices meeting minimum hardware specifications.
- Recommendations should be generated within 3 seconds of user input for optimal user experience.

4.2 Reliability

- The system shall ensure that 99.9% of user interactions are processed without failure or data corruption.
- Error rates in meal and recipe recommendations should remain below 0.1% of all recommendation requests.
- Data backups shall be conducted daily to secure user profiles, preferences, and recommendation history, ensuring data can be restored within an hour of any loss or corruption event.

4.3 Availability

- The application shall maintain an availability rate of 99.5%, accounting for both planned and unplanned maintenance.
- Planned system downtime should not exceed 1 hour per month, with individual sessions lasting no longer than 5 minutes.
- In case of connectivity issues, the app should display cached data (where applicable) to improve user experience without requiring a constant internet connection.

4.4 Security

- All user data must be encrypted both in transit (using TLS/SSL) and at rest, adhering to industry-standard encryption protocols.
- Two-factor authentication (2FA) shall be available to users for enhanced security during the login process.
- User profiles and sensitive data, including dietary preferences, shall be stored securely and be accessible only to authorized users.
- The application shall implement secure password storage using hashing algorithms and adhere to data protection regulations such as GDPR and CCPA.

4.5 Maintainability

- The codebase should follow modular design principles to allow isolated updates or changes without impacting the entire system.
- Documentation for the code, data models, and system architecture should be maintained and updated with each major release.
- The application should log all errors and usage analytics, enabling developers to monitor issues and system performance effectively.
- The system should support deployment automation to streamline updates and reduce manual intervention, allowing patches or minor updates to be deployed with minimal downtime.

4.6 Portability

- The application shall support the latest two major versions of iOS and Android to ensure a wide user base. (iOS 15+ and Android 12+)
- The system should be deployable in cloud environments compatible with multiple platforms, such as AWS, Azure, and Google Cloud.
- The app should maintain consistent performance and functionality across devices of varying screen sizes, including smartphones and tablets.
- Data formats and protocols should be designed to support easy migration to other platforms if needed.

5.0 Logical Database Requirements

The MANOS Food Decider application will utilize a database to store and manage a variety of data, including user profiles, dietary preferences, restaurant information, recipe details, and meal plans.

5.1 Data Formats:

The database will store structured data in a relational format, with tables to represent the core entities of the system. Key data entities will include:

- User Profiles: Storing user information such as name, email, password, and unique user ID.
- Dietary Preferences: Storing details about each user's dietary restrictions, allergies, and preferred food types.
- Restaurants: Storing information about local restaurants, including name, cuisine type, menu items, and nutritional data.
- Recipes: Storing details about various recipes, such as ingredients, preparation instructions, nutritional information, and tags (e.g., cuisine, dietary restrictions).

- Meal Plans: Storing personalized meal plans for users, including recommended recipes, scheduled meals, and shopping lists.

The database will support data types appropriate for storing text, numerical, and date/time information to accommodate the diverse data requirements of the application.

5.2 Storage Capabilities:

The database will have sufficient storage capacity to handle the expected growth in user data, restaurant and recipe information, and personalized meal plans. The database will also provide efficient indexing and querying capabilities to support the application's need for fast retrieval of data for recommendation engines, meal planning, and user-specific features.

5.3 Data Retention:

- User profile and preference data will be stored indefinitely, as long as the user maintains an active account with the application.
- Meal plan and history data will be retained for a period (e.g., 2 years) to support the user's meal tracking and recommendation features.

5.4 Data Integrity & Security:

- The database will enforce data integrity constraints, such as unique user IDs, referential integrity between related entities (e.g., users and their preferences, recipes and their ingredients), and data type validations.
- The database will implement robust security measures, including access controls, data encryption, and password hashing, to protect sensitive user data.
- The application will obtain user consent for the collection and storage of their data and provide mechanisms for users to review, update, and delete their personal information as needed.

6.0 Other Requirements

6.1 Compliance and Regulatory Requirements

- The application must comply with data protection regulations, including GDPR (General Data Protection Regulation) for users within the EU and CCPA (California Consumer Privacy Act) for users within California.
- Accessibility standards (such as WCAG 2.1) should be adhered to, ensuring the app is usable for people with disabilities.
- The application should comply with health and dietary industry guidelines to ensure accuracy in food labeling, dietary recommendations, and allergen disclosures.

6.2 Documentation Requirements

- Detailed user manuals and help documentation must be available within the app to guide users through features like creating profiles, inputting dietary preferences, and navigating meal recommendations.
- Developer documentation should be maintained, covering the system architecture, API integrations, and data models to ensure future developers can easily understand and work on the project.
- Training materials should be provided for customer support representatives to help troubleshoot common issues and assist users effectively.

6.3 User Interface and Experience Requirements

- The application should have a consistent, intuitive design that makes navigation easy for users with different tech experience levels.
- Accessibility features, such as scalable fonts, voice-over support, and alternative text for images, should be incorporated to support users with visual or motor impairments.
- Notifications and alerts should be minimal and non-intrusive, providing reminders for meal planning or shopping list updates without overwhelming the user.

6.4 Ethical and Social Requirements

- The app should avoid promoting unhealthy eating habits by recommending a balanced range of meal options, promoting nutrition and wellness.
- Any collected data on user dietary preferences should be used solely for the purpose of improving user experience and recommendations and should not be shared or sold to third parties.

6.5 Operational and Support Requirements

- A dedicated support channel (such as in-app chat or email support) should be available for users experiencing issues or with questions about app functionality.
- Scheduled maintenance windows should be communicated in advance to avoid unexpected disruptions.
- A feedback mechanism should be incorporated within the app to allow users to provide suggestions, report bugs, or request new features.

7.0 Approval

The signatures below indicate their approval of the contents of this document.

Project Role	Name	Signature	Date

Team Leader	Oleg Chystieiev	*Oleg Chystieiev*	06-11-2024
Product Owner	Anna Shibanova	*Anna Shibanova*	06-11-2024
Project Manager	Nicole Milmin	*Nicole Milmin*	06-11-2024
Lead Developer	Mo Harry Bandukda	*Mo Harry Bandukda*	06-11-2024
QA Specialist	Shirin Ali	*Shirin Ali*	06-11-2024
Reviewer	Prof Laily Ajellu		
Reviewer	Prof Anjana Shah		