**SOFTWARE**

**REQUIREMENTS SPECIFICATION**

**For**

**Recipe Recommendation System**

**Prepared by: -**

**Team Better\_Call\_Chef**

## Introduction

### Purpose

Outlining the complete requirements for the Recipe Recommendation System is the main goal of this publication. The document clarifies the client's functional and non-functional needs. Establishing a user-friendly framework for managing recipe details and user preferences is the main goal of this project. The objective is to provide a computer-based recipe suggestion system that is effective and produces a range of reports to improve user experience.

### 1.2 Scope of Development Project

Our development project's scope includes a multimodal strategy to transform people's approach to their fitness and nutrition goals. Our platform focuses on integrating user profiles and preferences to provide a highly customised experience that supports a wide range of fitness objectives, including maintenance, muscle gain, and weight loss. The incorporation of an all-encompassing survey guarantees that we obtain profound understandings of users' distinct lifestyles, enabling us to precisely and pertinently customise meal plans. This project aims to offer an immersive user experience enhanced with nutritional insights, going beyond a simple recipe compilation.

Frequent updates, community involvement tools, and partnerships with professionals in the fitness, nutrition, and culinary arts highlight our dedication to giving users a comprehensive, indisputable value experience. Our development project is all-encompassing in nature, with the goal of revolutionising the field of health-conscious living by providing a smooth, AI-driven platform that combines culinary creativity with individual wellness objectives.

### 1.3 Definitions, Acronyms and Abbreviations

JAVA -> platform independence

SQL-> Structured query Language

ER -> Entity Relationship

UML -> Unified Modelling Language

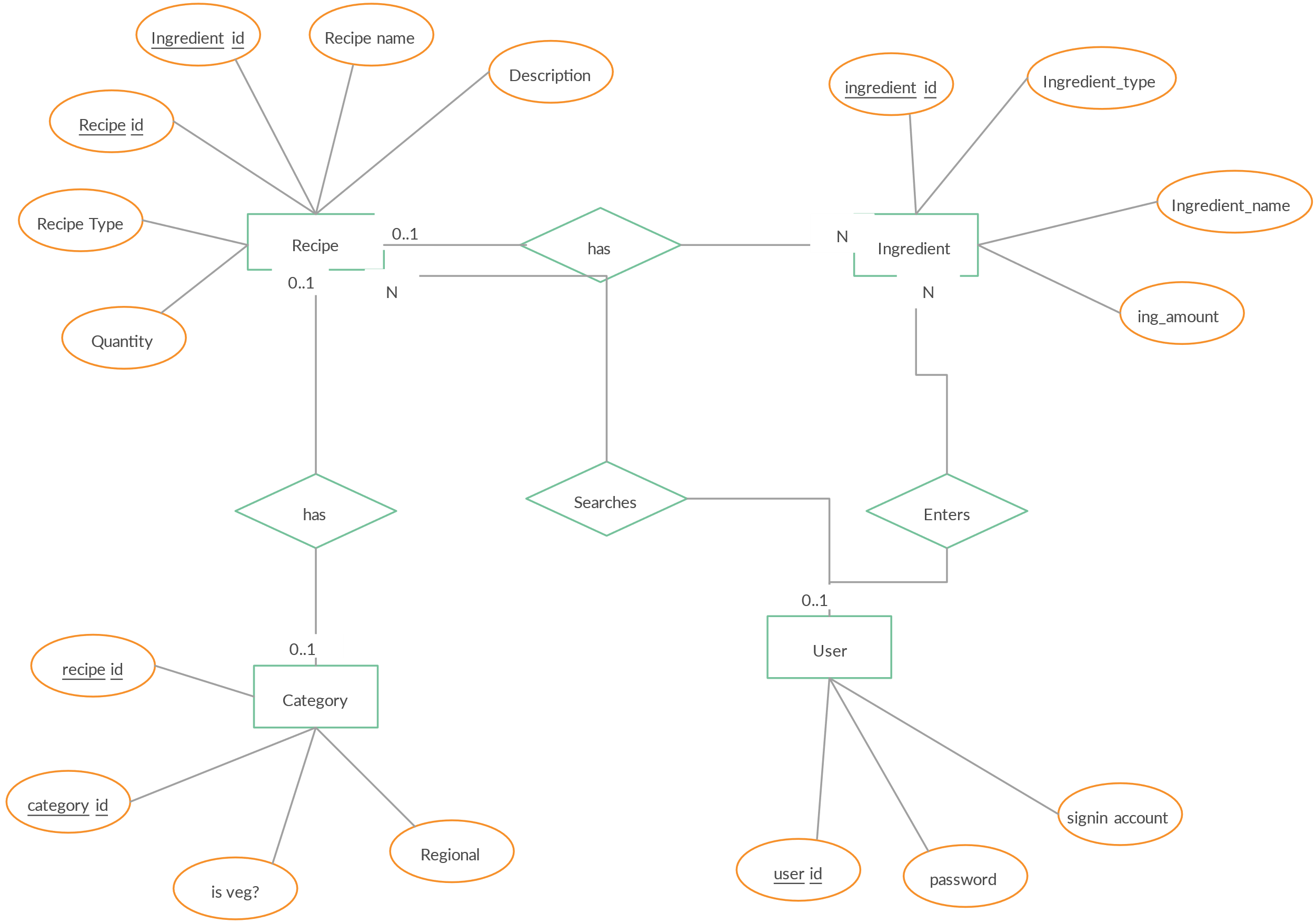
IDE-> Integrated Development Environment

SRS-> Software Requirement Specification

**2. Overall Descriptions**

### 2.1 Product Perspective

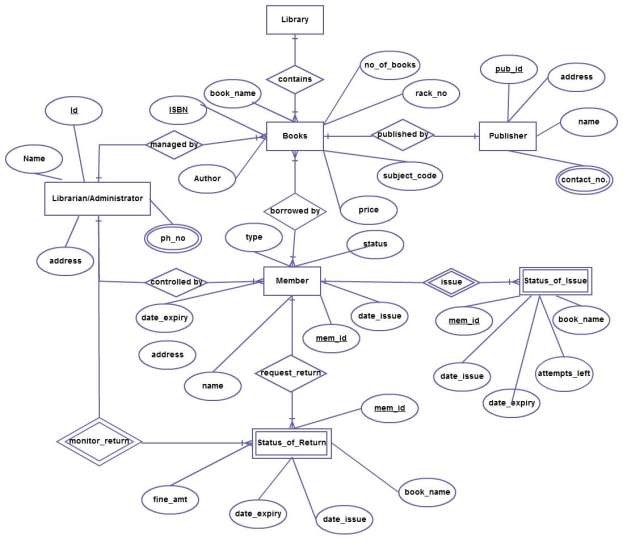
Use Case Diagram of Library Management System



This project's high-level schematic provides a fundamental overview. Staff members and students are both possible users. To make finding recipes easier, this system will have a feature that recommends recipes. This suggestion will be based on a number of factors, including the dish name, ingredients, and cuisine. Additionally, users of the system have the ability to create and change recipes, and they can seek recommendations—recommendations that must meet specific requirements—from the system.

### 2.2 Product Function

Entity Relationship Diagram of Library Management System



The Online Library System provides online real time information about the books available in the Library and the user information. The main purpose of this project is to reduce the manual work. This software is capable of managing Book Issues, Returns, Calculating/Managing Fine, Generating various Reports for Record-Keeping according to end user requirements. The Librarian will act as the administrator to control members and manage books. The member’s status of issue/return is maintained in the library database. The member’s details can be fetched by the librarian from the database as and when required. The valid members are also allowed to view their account information.

### 2.3 User Classes and Characteristics

Various services are offered by the Recipe Recommendation System depending on the kind of user. With all the rights of an administrator, the Chef will serve as the controller. The person using the system online could be a professional chef or a home cook.

The features that are available to the Chef are:-

* Create recipes: Add new recipes to the system, including ingredients, instructions, and dietary information.
* Edit recipes: Modify existing recipes to update ingredients, instructions, or dietary information.
* Delete recipes: Remove recipes from the system that are no longer relevant or appropriate.
* Manage user accounts: Create, edit, and delete user accounts, as well as assign user roles.
* Generate recipe recommendations: Recommend recipes to users based on their preferences, dietary restrictions, and available ingredients.
* View system reports: Analyse usage data and generate reports on user activity, recipe popularity, and ingredient trends.

The features that are available to the Users are:-

* Create an account: Register with the system to create a personal account for saving recipes and preferences.
* Search for recipes: Search for recipes using keywords, ingredients, cuisines, or dietary restrictions.
* View recipes: Access detailed information about recipes, including ingredients, instructions, nutritional information, and user reviews.
* Save recipes: Add recipes to their personal collection for easy access and reference.
* Rate and review recipes: Provide feedback on recipes by rating them and writing reviews.
* Receive personalized recommendations: Get recipe suggestions based on their saved recipes, past searches, and dietary preferences.

**2.4 Operating Environment**

The product will be operating in windows environment. The Recipe Recommending System is a website and shall operate in all famous browsers. Also it will be compatible with the IE 6.0. The only requirement to use this online product would be the internet connection.

The hardware configuration include Hard Disk: 40 GB, Monitor: 15” Color monitor, Keyboard: 122 keys. The basic input devices required are keyboard, mouse and output devices are monitor, printer etc.

### 2.5 Assumptions and Dependencies

**Assumptions:**

* The algorithm for recipe recommendations is error-free and provides accurate suggestions based on user preferences.
* The user interface of the system is designed to be intuitive and user-friendly, ensuring ease of use for individuals seeking recipe recommendations.
* All relevant information about users, recipes, and ingredients is stored in a database accessible by the recommendation system.
* The system is equipped with sufficient storage capacity and ensures fast access to the recipe database to enhance the user experience.
* The system supports efficient search functionality, allowing users to quickly find recipes based on their preferences and dietary restrictions.
* The Recipe Recommendation System operates 24 hours a day, providing users with access at any time.
* Users can access the system from any device with internet browsing capabilities and a stable internet connection.
* Users are required to have correct login credentials to access their accounts and perform actions within the recommendation system.

**Dependencies:**

* The proper functioning of the recommendation system relies on specific hardware and software configurations to ensure optimal performance.
* The development and execution of the project depend on the accurate listing of requirements and specifications to create an effective recipe recommendation algorithm.
* End users, particularly those with administrative roles, should possess a proper understanding of how the recommendation system works to manage and optimize its features.
* The system relies on maintaining a comprehensive database of recipes, user preferences, and ingredient information to provide relevant and accurate recommendations.
* Any updates or modifications to the recipe database must be recorded accurately to maintain the integrity of the data and enhance the quality of recommendations.
* The success of the project is contingent on the availability of general reports that offer insights into user preferences and system performance.

### 2.6 Requirement

Software Configuration:-

This software package is developed using java as front end which is supported by sun micro system. Microsoft SQL Server as the back end to store the database. Operating System: Windows 10, Windows 11

Language: Java Runtime Environment, Net beans 7.0.1 (front end)

Database: MS SQL Server (back end)

Hardware Configuration:-

Processor: Pentium(R)Dual-core CPU

Hard Disk: 40GB

RAM: 256 MB or more

### 2.7 Data Requirement

The inputs consist of the query to the database and the output consists of the solutions for the query. The output also includes the user receiving the details of their accounts. In this project the inputs will be the queries as fired by the users like create an account, picking their diet plans, health preferences and putting into account. As for output, the Users get a customised summary that includes information about eating plans, deadlines, and real-time health status.

## 3. External Interface Requirement

### 3.1 GUI

The software should offer a visually appealing and efficient graphical interface for both users and administrators to interact with the recipe recommendation system. Key features include:

**User and Administrator Functionality:**

* The GUI should facilitate user-friendly interactions for both regular users and administrators. It should support tasks such as recipe exploration, preference setting, and system management.

**Quick Reports:**

* Users and administrators should have the ability to view quick reports, such as recently recommended recipes or popular recipes within a specific time period.

**Search Facility:**

* The system must provide a robust search functionality that allows users to search for recipes based on various criteria, such as cuisine, ingredients, or dietary preferences.

**Customizable Interface:**

* Administrators should be able to customize the user interface to tailor it to the specific needs of the users. This may include adjusting layouts, themes, and other visual elements.

**Module Integration:**

* The GUI should seamlessly integrate with all modules of the system, including the recommendation algorithm, user management, and preference tracking. All modules must be accessible through the graphical interface.

**Standardized Design:**

* Different interfaces within the system should follow a standardized design template, ensuring consistency and a cohesive user experience across all functionalities.

**User Registration and Login Interface:**

1. **Registration:**

* Users can create accounts by providing necessary details, including preferences and dietary restrictions.

1. **Login:**

* Once registered, users can log in using a username and password. Incorrect login attempts trigger an error message.

**Recipe Search:**

* Users can enter criteria such as cuisine type, ingredients, or dish name to search for recipes.

**Categories View:**

* The system should display categories of recipes, allowing administrators to add, edit, or delete categories.

**Administrator Control Panel:**

* The control panel enables administrators to manage users, edit or remove recipes, and configure recommendation options.

## 4. System Features

The users of the system should be provided the surety that their account is secure. This is possible by providing: -

* User authentication and validation of members using their unique User ID
* Search engine working based on the user preferences
* Allows the user to confirm the calorie count, micronutrients and macronutrients of the recipe

## 5. Other Non-functional Requirements

### 5.1 Performance Requirement

The proposed recipe recommendation system that we are going to develop will serve as the primary tool for users seeking culinary guidance. It will interact with users to provide accurate and personalized recipe suggestions. Therefore, the system is expected to perform all the functionalities specified for optimal user experience.

* The performance of the system should be fast and accurate, ensuring quick response times when users search for recipes or explore recommendations.

* The Recipe Recommendation System shall handle expected and non-expected errors effectively to prevent loss of information and minimize downtime. It should incorporate robust error testing mechanisms to identify and address issues, ensuring a seamless user experience.

* The system should be able to handle a large amount of data efficiently. It should accommodate a diverse range of recipes and user preferences without any degradation in performance.

### 5.2 Safety Requirement

For the recipe recommendation system, safeguarding data integrity is paramount to ensure a secure and reliable user experience.

To mitigate the risk of data loss, the system will implement regular and automated database backups. These backups will be performed at scheduled intervals to capture the latest recipe and user preference data, preventing substantial losses in the event of a database crash resulting from factors such as viruses or operating system failures.

Moreover, the system infrastructure will include a robust Uninterruptible Power Supply (UPS) or inverter facility to address power supply failures. This ensures that the recipe recommendation system remains operational even during unexpected power outages, providing uninterrupted service to users. The UPS/inverter facility will offer a seamless transition to backup power, minimizing the risk of service disruptions and potential data corruption.

### 5.3 Security Requirement

* The system will utilize a secure database to safeguard sensitive user information, recipe data, and system configurations from unauthorized access.
* Normal users will have read-only access, limiting their ability to edit or modify any data, except for their personal information and certain specified details. This access restriction helps prevent unintended alterations to the recipe database.
* The system will support various user types, each with specific access constraints. Differentiated access levels will be established to control the actions users can perform within the system, ensuring that only authorized actions are executed.
* Robust user authentication mechanisms will be implemented to verify the identity of users. This involves secure login processes, possibly incorporating multi-factor authentication, to prevent unauthorized access.
* Stringent measures will be in place to ensure the security of user passwords. Encryption and hashing techniques will be applied to protect user credentials, minimizing the risk of password hacking.
* The system will have separate accounts for administrators and regular members. Admin accounts will possess exclusive rights to update the recipe database, while member accounts will be restricted from such database modifications. This hierarchical access ensures that critical database management functions are limited to authorized administrators only.

### 5.4 Requirement attributes

* Multiple administrators will have the right to introduce changes to the system. This collaborative approach between administrators ensures efficient management and updates of the recipe recommendation system.
* The recipe recommendation system will be developed as an open-source project. This openness promotes collaboration, fosters community involvement, and allows users to contribute to improving the system. The open-source nature encourages transparency and innovation within the development community.
* The database will be designed and maintained with a focus on high quality to ensure user friendliness. This includes efficient data organization, easy navigation and a user interface that improves the overall user experience. To maintain the quality standards of the database, regular testing and optimization will take place.
* The system interface will prioritize user-friendliness and ensure that all users, regardless of their technical expertise, can easily navigate and interact with the application. Intuitive design and clear instructions will contribute to a positive user experience.
* Users will be able to easily download and install the recipe recommendation system. Clear installation instructions and supporting documentation will be provided to streamline the integration process, enabling users to quickly set up and access the system

### 5.5 Business Rules

In the recipe recommendation system, users, whether administrators or members, are required to adhere to established business rules governing their conduct within the platform. These rules encompass clear communication of the project's cost structure, and any discount offers, emphasizing compliance with specified terms. Users must abstain from engaging in illegal activities and respect local, national, and international laws. Adherence to system protocols, including ethical use of features and respectful interaction, is expected from both administrators and members. Violations of these rules may result in appropriate actions, such as account suspension or termination, to maintain a secure, lawful, and user-friendly environment for all participants.

### 5.6 User Requirement

The users of the recipe recommendation system consist of members, who are individuals seeking culinary inspiration, and administrators, akin to chefs or culinary experts responsible for system maintenance. Members are assumed to possess basic computer and internet browsing knowledge, while administrators are expected to have a deeper understanding of the system's internals, capable of addressing technical issues arising from disk crashes, power failures, and other unforeseen events. The system will provide an intuitive user interface, comprehensive user manual, online help, and installation guides to ensure users can navigate and utilize the platform seamlessly.

The admin provides certain facilities to the users in the form of: -

* Administrators
* Serving as culinary guides
* Offer facilities such as backup and recovery
* Password retrieval
* Data migration
* Data replication
* Auto-recovery
* File organization to enhance the user experience
* Regular server maintenance and timely updates are imperative to keep the system running smoothly and up to date with evolving culinary trends.

**6. Other Requirements**

### 6.1 Data and Category Requirement

The recipe recommendation system will categorize users into distinct roles, including home cooks, culinary enthusiasts, administrators, and others. Access rights will be determined based on the user's category, granting administrators the capability to modify, delete, and append recipe data, while all other users, except for designated administrators, will have rights limited to retrieving information from the database. Similarly, the recipe database will be organized into various categories, reflecting different culinary styles, dietary preferences, and meal types. Each category will be coded in a specific format to facilitate efficient retrieval and presentation of relevant recipe data. This systematic categorization ensures that users can easily explore and discover recipes aligned with their preferences, enhancing the overall user experience.

### 6.2 Appendix

A: Admin, Abbreviation, Acronym, Assumptions; B: Browse, Business Rules; C: Cuisine, Client, Conventions; D: Data Requirement, Dependencies; G: GUI (Graphical User Interface); K: Key Ingredients, Instruction; M: Member; N: Non-functional Requirement; O: Operating Environment; P: Performance, Perspective, Purpose; R: Recipe, Requirement, Requirement Attributes; S: Safety, Scope, Security, System Features; U: User, User Class and Characteristics, User Requirement;

**6.3 Glossary**

The following are the list of conventions and acronyms used in this document and the project as well:

* Administrator: A login ID representing a user with administrative privileges for managing the recipe recommendation software.
* User: A general login ID assigned to most users interacting with the recipe recommendation system.
* Client: Intended users for the recipe recommendation software, including home cooks, culinary enthusiasts, and administrators.
* SQL: Structured Query Language; used in the recipe recommendation system to retrieve information from a database.
* SQL Server: A server employed in the recipe recommendation system to store data in an organized format.
* Layer: Represents a section of the recipe recommendation system.
* User Interface Layer: The section of the recipe recommendation system referring to what the user directly interacts with, facilitating recipe browsing and selection.
* Application Logic Layer: The section of the recipe recommendation system referring to the Web Server. This is where computations related to recipe recommendations are completed.
* Data Storage Layer: The section of the recipe recommendation system referring to where all recipe data is recorded and stored.
* Use Case: A broad-level diagram in the recipe recommendation system illustrating a basic overview of user interactions and system functionalities.
* Class Diagram: In the recipe recommendation system, it is a type of static structure diagram describing the structure of the system by illustrating recipe categories, their attributes, and the relationships between them.
* Interface: In the recipe recommendation system, it refers to elements used to facilitate communication across different mediums.
* Unique Key: In the recipe recommendation system, it is used to differentiate entries in the recipe database, ensuring the distinct identification of recipes and user data.

### 6.4 Class Diagram

A class is an abstract, user-defined description of a type of data. It identifies the attributes of the data and the operations that can be performed on instances (i.e. objects) of the data. A class of data has a name, a set of attributes that describes its characteristics, and a set of operations that can be performed on the objects of that class. The classes’ structure and their relationships to each other frozen in time represent the static model. In this project there are certain main classes which are related to other classes required for their working. There are different kinds of relationships between the classes as shown in the diagram like normal association, aggregation, and generalization. The relationships are depicted using a role name and multiplicities. Here ‘Librarian’, ‘Member’ and ‘Books’ are the most important classes which are related to other classes.

