

**Tribhuvan University**

**Faculty of Humanities and Social Science**

**Orchid International College**

**E-COMMERCE MOBILE APPLICATION**

**A PROJECT REPORT**

**Submitted to**

**Department of Computer Application**

**Orchid International College**

***In partial fulfillment of the requirements for the Bachelors in Computer Application***

Submitted by

Smriti Khadka (93902090)

September 2024

Under the Supervision of

**Dhiraj Kumar Jha**



**Tribhuvan University**

**Faculty of Humanities and Social Science**

**Orchid International College**

# SUPERVISOR’S RECOMMENDATION

I hereby recommend that this project prepared under my supervision by Smriti Khadka entitled **“Recipe Recommendation Application”** in partial fulfillment of the requirements for the degree of Bachelor of Computer Application is recommended for the final evaluation.

**SIGNATURE**

Mr. Dhiraj Kumar Jha

**SUPERVISOR**

Head of Department

Faculty of Humanities and Social Science

Bijayachowk, Gaushala, Kathmandu



**Tribhuvan University**

**Faculty of Humanities and Social Science**

**Orchid International College**

# LETTER OF APPROVAL

This is to certify that this project prepared by Smriti Khadka entitled “**Recipe Recommendation Application”** in partial fulfillment of the requirements for the degree of Bachelor in Computer Application has been evaluated. In our opinion it is satisfactory in the scope and quality as a project for the required degree.

|  |  |
| --- | --- |
| **Signature of Supervisor**  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Mr. Dhiraj Kumar Jha  Head of Department  Department of IT  Orchid International College | **Signature of HOD/ Coordinator**  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Mr. Dhiraj Kumar Jha  Head of Department  Department of IT |
| **Signature of Internal Examiner**  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Dhiraj Kumar Jha  Head of Department | **Signature of External Examiner**  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |

# ABSTRACT

**Chef's Delights** is an innovative culinary application designed to provide users with personalized recipe recommendations based on their preferences and dietary needs. Developed using Java in Android Studio for the front end and PHP with MySQL for the backend, Chef's Delights utilizes advanced algorithms and user data to deliver a highly customized and engaging cooking experience. As the demand for convenient and user-friendly cooking aids grows, this app meets the need for accessible and enjoyable recipe discovery through mobile devices.

**Personalized Recipe Recommendations** allow users to discover new dishes tailored to their tastes, cooking skills, and available ingredients. **Video Tutorials** provide access to a curated library of cooking videos that guide users step by step through the preparation of various dishes, ensuring a seamless and educational experience in the kitchen.

The integration of PHP and MySQL in the backend ensures efficient data management and smooth user interactions, while the Java and Android Studio-based front end delivers an intuitive and responsive user interface. This project demonstrates how Chef's Delights successfully combines modern technology with culinary expertise to create an enjoyable and effective cooking environment, helping users enhance their culinary skills and discover new favorite dishes.

**Keywords:** Chef's Delights, Recipe Recommendation App, Personalized Recipes, Culinary Quiz, Video Tutorials, Java, Android Studio, PHP, MySQL

# ACKNOWLEDGEMENT

We have invested a significant amount of effort into the Chef's Delights project. However, completing this project would not have been possible without the unwavering support and guidance of numerous individuals. We would like to extend our sincere thanks to all of them. We are profoundly grateful to our supervisor, Mr. Dhiraj Kumar Jha, for his expert guidance and supervision. His valuable insights, along with his continuous encouragement and constructive feedback, have been pivotal in the successful completion of this project. We would also like to express our deepest appreciation to our fellow team members who have contributed their expertise and effort towards the development of Chef's Delights. Their cooperation and dedication have been essential in bringing this project to fruition.

Finally, we wish to acknowledge and thank everyone who has provided assistance, advice, and support throughout this journey. Your contributions have been indispensable.

**Smriti Khadka (93902090)**

# TABLE OF CONTENT

[SUPERVISOR’S RECOMMENDATION i](#_Toc177859901)

[LETTER OF APPROVAL ii](#_Toc177859902)

[ABSTRACT iii](#_Toc177859903)

[ACKNOWLEDGEMENT iv](#_Toc177859904)

[TABLE OF CONTENT v](#_Toc177859905)

[LIST OF FIGURES vii](#_Toc177859906)

[LIST OF TABLES viii](#_Toc177859907)

[LIST OF ABBREVIATION ix](#_Toc177859908)

[Chapter 1: Introduction 1](#_Toc177859909)

[1.1 Introduction 1](#_Toc177859910)

[1.2 Problem Statement 1](#_Toc177859911)

[1.3 Objectives 1](#_Toc177859912)

[1.4 Scope and limitation 2](#_Toc177859913)

[1.4.1 Scope of System 2](#_Toc177859914)

[1.4.2 Limitation of Existing System 3](#_Toc177859915)

[1.5 Report Organization 3](#_Toc177859916)

[Chapter 2: Background Study and Literature Review 4](#_Toc177859917)

[2.1 Background Study 4](#_Toc177859918)

[2.2 Literature Review 4](#_Toc177859919)

[Chapter 3: System Analysis and Design 5](#_Toc177859920)

[3.1 System Analysis 5](#_Toc177859921)

[3.1.1 Requirement Analysis 6](#_Toc177859922)

[3.1.2 Feasibility Study 8](#_Toc177859923)

[3.1.3 Process Modeling 11](#_Toc177859924)

[3.2 System Design 13](#_Toc177859925)

[3.2.1 Architectural Design 13](#_Toc177859926)

[3.2.2 System Flowchart 14](#_Toc177859927)

[3.2.3 JSON Parse Tree Structure 15](#_Toc177859928)

[3.2.4 Interface Design 18](#_Toc177859929)

[3.2.5 Physical DFD 21](#_Toc177859930)

[Chapter 4: Implementation and Testing 23](#_Toc177859931)

[4.1 Implementation 23](#_Toc177859932)

[4.1.1 Tools Used 23](#_Toc177859933)

[4.1.2 Implementation Detail of Modules 24](#_Toc177859934)

[4.2 Testing 24](#_Toc177859935)

[4.2.1 Test Cases for Unit Testing 24](#_Toc177859936)

[4.2.2 Test Cases for System Testing 27](#_Toc177859937)

[Chapter 5: Conclusion and Future Recommendation 31](#_Toc177859938)

[5.1 Lesson Learnt / Outcome 31](#_Toc177859939)

[5.2 Conclusion 31](#_Toc177859940)

[5.3 Future Recommendation 31](#_Toc177859941)

[Chapter 6: References 32](#_Toc177859942)

[APPENDICES 33](#_Toc177859943)

# LIST OF FIGURES

[Figure 3.1‑1 Agile Methodology 6](#_Toc177862209)

[Figure 3.1‑2 Use Case Diagram of Recipe Recommendation Application 8](#_Toc177862210)

[Figure 3.1‑3 Gantt Chart of Recipe Recommendation Application 10](#_Toc177862211)

[Figure 3.1‑4 Context Diagram of Recipe Recommendation Application 11](#_Toc177862212)

[Figure 3.2‑1 MVC Architecture of Recipe Recommendation Application 13](#_Toc177862213)

[Figure 3.2‑2 System Flowchart of Recipe Recommendation Application 15](#_Toc177862214)

[Figure 3.2‑3 Login Page UI 17](#_Toc177862215)

[Figure 3.2‑4 Signup Page UI 18](#_Toc177862216)

[Figure 3.2‑5 Home Page UI 19](#_Toc177862217)

[Figure 3.2‑6 Starting Page UI 19](#_Toc177862218)

[Figure 3.2‑9 Level 1 DFD of Recipe Recommendation Application 20](#_Toc177862219)

# LIST OF TABLES

[Table 4.1‑1 Implementation of Modules 21](#_Toc177824308)

[Table 4.2‑1 Test Case 001 - Signup page 22](#_Toc177824309)

[Table 4.2‑2 Test Case 002 - Login Page 23](#_Toc177824310)

[Table 4.2‑3 Test Case 003 - Payment Page 25](#_Toc177824311)

[Table 4.2‑4 Test Case 004 - Order Product 26](#_Toc177824312)

# LIST OF ABBREVIATION

DFD : Data Flow Diagram

ICCOINS : International Conference on Computer and Information Sciences

IDE : Integrated Development Environment

JSON : JavaScript Object Notation

MVC : Model View Controller

TC : Test Case

UI : User Interface

UX : User Experience

# Introduction

## Introduction

In today's busy world, finding time to prepare delicious and healthy meals can be a challenge. The Recipe recommendation application, Chef's Delight, is designed to make cooking easy, fun, and accessible for everyone. With an intuitive frontend developed in Android Studio using Java, the application ensures a seamless and enjoyable user experience. Local data storage is efficiently managed with SQLite, providing quick and reliable access to your favorite recipes. The backend, built with PHP and MySQL, guarantees secure and scalable data handling. Key features such as personalized recipe recommendations, meal planning empower users to take control of their culinary adventures and achieve their cooking goals with ease.

## Problem Statement

Cooking can be a challenging task for many individuals, especially when faced with the daunting task of deciding what to cook or finding recipes that match their dietary preferences. Traditional methods of recipe discovery, such as cookbooks or word of mouth, can be limiting and often fail to cater to specific needs like dietary restrictions or ingredient availability. There is a growing demand for a more personalized approach to cooking that simplifies the process and makes it enjoyable. Some challenges faced by individuals when trying to prepare meals are:

* Limited time and resources can make it difficult for people to plan and prepare healthy, balanced meals regularly.
* Lack of culinary skills or experience can make it intimidating for individuals to try new recipes or cooking techniques.
* Dietary restrictions or specific health conditions may require special ingredients or modifications to traditional recipes, which can be difficult to manage without proper guidance.

## Objectives

The main purpose of **"Chef's Delights"** is to achieve the following objectives:

* To provide personalized recipe recommendations tailored to individual preferences and dietary needs.
* To offer an engaging and enjoyable cooking experience through interactive culinary tasks.
* To save time and reduce the stress of meal planning by simplifying the process of discovering new recipes.

## Scope and limitation

The existing system for recipe discovery relies on traditional methods, such as cookbooks or manually searching online. Currently, users must visit various websites or reference multiple sources to find suitable recipes, which can be time-consuming and inconvenient. This approach makes it difficult to track individual cooking preferences and dietary requirements. To overcome these limitations, the Chef Delight’s application aims to streamline the recipe discovery process by providing users with personalized recommendations, all within a single platform. While the application offers a vast collection of recipes categorized by meal type, cuisine, and ingredients, its primary limitation is that it relies on the accuracy of user input for personalized suggestions. As such, future updates will focus on enhancing AI-driven suggestions and improving the user experience through better integration of dietary preferences and cooking history.

### Scope of System

To overcome the limitations of traditional recipe discovery, the Chef ‘s Delight application is designed to provide a comprehensive and personalized cooking experience. Users can explore a wide variety of recipes across multiple categories, all within a single platform, simplifying their meal planning process. By offering digital access to recipes, Chef ‘s Delight reduces the need for physical cookbooks and expensive meal subscriptions, making it more cost-effective for users. The app provides several key features, including browsing recipes by category, viewing ingredients with detailed quantities, and accessing personalized meal recommendations based on user preferences and dietary needs. This flexibility allows users to cook at their own pace without being constrained by time or location. Additionally, Chef ‘s Delight enhances the cooking experience through step-by-step instructions, nutritional insights, and meal planning assistance. Its scalable design ensures continuous updates, allowing the platform to adapt to changing user preferences and the latest culinary trends

### Limitation of Existing System

The limitations of the existing system are listed as follows:

1. Although the app offers a diverse range of recipes and video tutorials, it may lack the hands-on experience and direct guidance that in-person cooking classes provide.
2. The app might not fully accommodate all skill levels or dietary preferences, potentially limiting its usefulness for users with specific needs or advanced cooking skills.
3. Performance issues such as app crashes, slow loading times, or compatibility problems with certain devices could disrupt the user experience and hinder access to recipes and tutorials.
4. Over-reliance on digital platforms for culinary inspiration might reduce opportunities for hands-on cooking practice and real-world culinary exploration, impacting users' overall cooking skills development.

## Report Organization

This report document contains five chapters including this chapter. Chapter two defines and describes Background Study and Overview of related existing systems and their pros and cons. Chapter three presents the System Analysis and Design including Requirement Analysis and Feasibility Analysis. Chapter four presents the Implementation, Testing and debugging are explained. In chapter five, Conclusion, Limitations and Future enhancement are briefly explained. Overall, this report contains architecture of the system and tools and technologies that are used to build the system.

# Background Study and Literature Review

## Background Study

In traditional culinary resources, such as cookbooks and static recipe websites, users often face limitations in engagement and personalization. Cookbooks provide structured recipes but lack interactive features and adaptability to individual preferences. Static websites may offer a broad range of recipes but typically do not offer personalized recommendations or real-time updates. These traditional methods can also make it difficult for users to find recipes suited to their specific tastes or dietary needs, leading to a less dynamic cooking experience. **Chef's Delights** addresses these issues by providing a user-friendly, interactive digital platform that offers personalized recipe recommendations, video tutorials, and community engagement, meeting the need for a more tailored and engaging culinary experience in today’s fast-paced lifestyle.

## Literature Review

In extensive review of research papers and journals, particularly those published by IEEE, and through the study of existing systems, we have identified several critical factors that will contribute to the successful development of my recipe mobile application. Websites for cooking recipes, and there are also recipes regarded as “easy” to cook. However, those recipes are not estimated as “easy” by taking user’s conditions. [1] An Intelligent Recipe Management System is the ultimate intermediate system between the process control application and all other support systems that are used in respect to the application; it processes data from other work systems and provides feedback for them. [2] [1]One of the widely used recommended algorithms is based collaborative filtering algorithm. In the recent years, it was proved that using a hybrid model for recommendation worked much better than a single one in many years. In the hybrid recommender system, we use content-based algorithm and model-based CF algorithm. [3] These insights from the study of existing systems provide a robust framework for developing my secure, efficient, and user-friendly recipe mobile application that meets the dynamic demands of today's recipe’s landscape.

# System Analysis and Design

## System Analysis

Chef’s Delight is a modern recipe recommendation application that requires continuous updates and feature enhancements to provide an intuitive and personalized cooking experience. Given the evolving nature of user preferences and culinary trends, the Agile model is the ideal choice for our system, allowing for iterative development and rapid feedback cycles. Agile's flexibility enables the introduction and refinement of new features in short, manageable sprints. This approach allows the development team to address different components of the application concurrently, while integrating user feedback and adjusting priorities as needed. Regular updates ensure that RecipeHub stays relevant and user-friendly, enhancing both the functionality and the overall user experience.

The essential steps for system analysis are as follows:

1. Research and define the core components of the recipe recommendation system.
2. Analyze current recipe discovery methods and identify limitations.
3. Conduct interviews with users, culinary experts, and stakeholders to gather insights.
4. Write a detailed requirements document based on research and interviews.
5. Establish clear standards, policies, and procedures for system development.
6. Review the requirements with users and stakeholders to ensure accuracy.
7. Continuously update and expand the project plan to reflect evolving needs.

Figure ‑ Agile Methodology

### Requirement Analysis

Requirements analysis is a crucial step for determining the success of a system or

software project. Requirements are generally split into two types:

1. Functional Requirements
2. Non-Functional Requirements
3. **Functional Requirements**

* The system allows users to create accounts and log in securely.
* Administrators can create, edit, and manage recipe listings and categories.
* Users can browse available recipes and save their favorite recipes for future use.
* Users can search for recipes by name, ingredients, or specific categories efficiently.
* Users can view detailed recipe information, including ingredients, nutrition count, and cooking steps.
* Administrators can access a comprehensive dashboard to manage user accounts, recipe listings, and system settings.
* Administrators should be able to moderate recipe submissions, user reviews, and handle any user-related issues.

1. **Non-Functional Requirements**

**Security:**  
The system ensures secure login and registration using Firebase Authentication. User passwords are encrypted and stored securely via Firebase's built-in encryption mechanisms.

**Availability:**  
The system is accessible anytime, anywhere, as long as users have an internet connection and a mobile device. The application is cross-platform, functioning seamlessly on both Android and iOS devices. It provides a consistent user experience across various mobile browsers and network conditions.

**Reliability:**  
The system leverages Firebase's real-time database for reliable data synchronization. It ensures high availability with 24/7 access and automatic scaling to accommodate varying user loads, providing real-time updates when recipes or other content are added.

**Maintainability:**  
The system is designed with maintainability in mind, adhering to best practices in Android development. Firebase backend services, including Firestore Database and Firebase Storage, facilitate easy maintenance and updates. Regular backups and version control mechanisms are in place for quick recovery and updates by the development team.

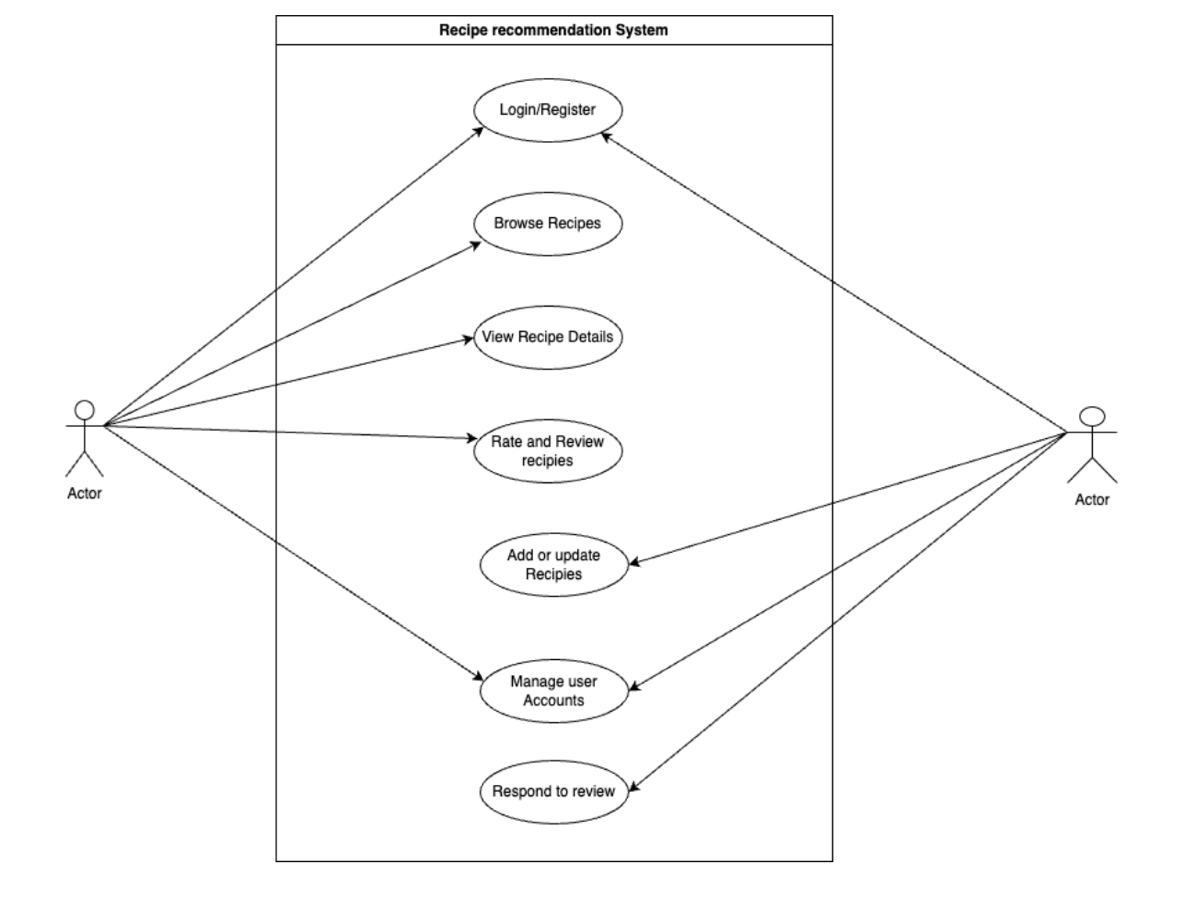


Figure ‑ Use Case Diagram of Recipe Recommendation Application

### Feasibility Study

The feasibility study is the important step in any software development process. This is

because it makes analysis of different aspects like cost required for developing and

executing the system, the time required for each phase of the system and so on. If these important factors are not analyzed then definitely it would have impact on the organization and the development and the system would be a total failure. So, for running the project and the organization successfully this step is a very important step in a software development life cycle process.

1. **Technical Feasibility**
2. The system has the technical capacity to store and manage data such as user details, recipe information, reviews, and ratings. The project is designed to be scalable, allowing for future expansions if needed. It involves the use of reliable hardware, software, and technologies. The application requires a device (such as a smartphone, tablet, or computer) and internet connectivity. The system's user interface is intuitive and user-friendly. It should be supported by hardware with adequate CPU, processing power, and memory to ensure smooth operation and responsiveness.
3. **Operational Feasibility**

The proposed system is operationally feasible, offering a reliable experience for all types of users, including those with minimal technical knowledge. The system is designed to cater to individual users and can be easily adapted to small or large-scale organizations. Its simplicity and straightforward navigation ensure that users can easily browse, manage, and interact with recipes without requiring extensive training or technical support.

1. **Economic Feasibility**

The application will be free to use, with potential monetization options such as premium features or services like personalized meal plans or exclusive content. The app is not resource-intensive, making it accessible on devices with basic specifications. Operating the application will require minimal human resources and hardware, making it cost-effective. The development of this application will involve a moderate financial investment, making it economically viable for implementation and maintenance

1. **Schedule Feasibility**

Schedule feasibility assesses the realism and achievability of the project timeline. The feasibility of the Recipe Recommendation Application is high, as it can be completed within a set deadline. The project plan is designed to ensure that all milestones, from development to testing and deployment, are met within the allocated time frame, ensuring timely delivery without compromising quality.

The schedule and duration of the project is shown in the Gantt chart below:

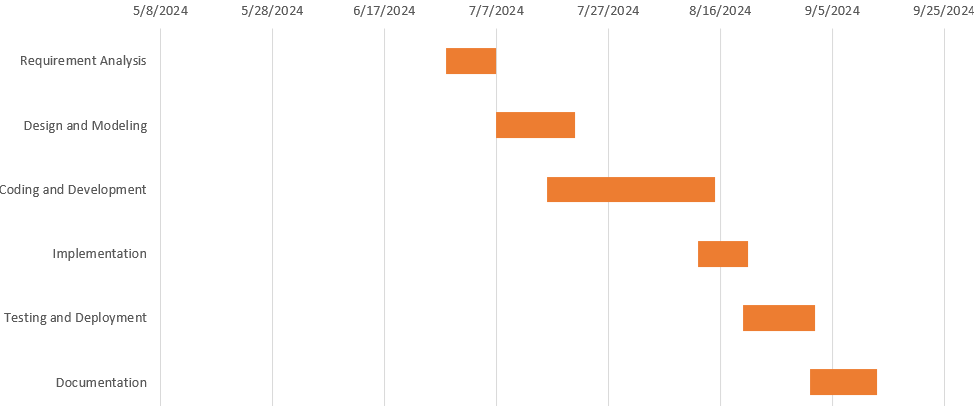


Figure ‑ Gantt Chart of Recipe Recommendation Application

The Gantt chart outlines a project timeline with six phases: Requirement Analysis (02/07/2024, 10 days), Design and Modeling (11/07/2024, 10 days), Coding and Development (22/07/2024, 27 days), Implementation (19/08/2024, 10 days), Testing and Deployment (01/09/2024, 5 days), and Documentation (07/09/2024, 8 days). Each phase includes its respective

### Process Modeling

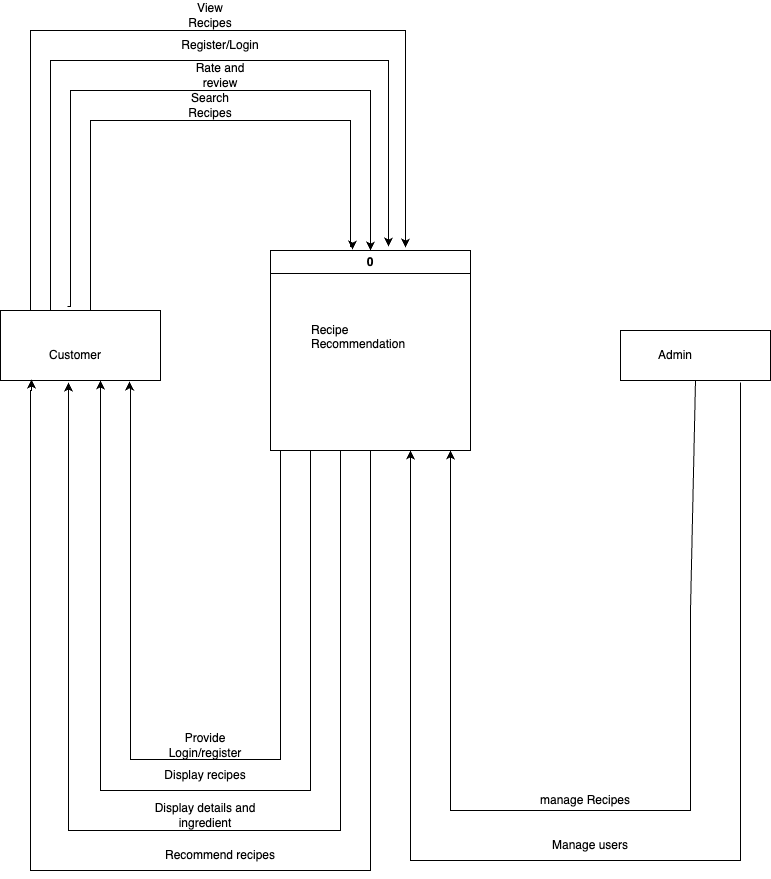


Figure ‑ Context Diagram of Recipe Recommendation Application

This context diagram represents the interaction between users, admins, and the Recipe Recommendation mobile application. Users can browse recipe categories, view subcategories, select recipes, and explore detailed recipe information including ingredients, steps, and nutrition details. They can also create accounts, log in, rate recipes, and save favorite recipes. The Recipe Recommendation app responds by displaying categorized recipes, fetching subcategory details, providing recipe steps and nutrition information, and managing user profiles.

Admins interact with the app to manage recipe categories, subcategories, individual recipes, ingredients, and ratings. They are responsible for ensuring that the recipe database is up-to-date, managing recipe submissions, and handling user feedback. The admin also oversees special content like featured recipes or promotions. The system ensures that both users and admins have a smooth and efficient experience with recipe discovery and management.

The Recipe Recommendation app acts as the central platform, managing interactions between users and admins to offer personalized recipe recommendations and effective recipe management.

## System Design

The system design of Recipe Recommendation Application that consists of architectural design, database design, user interface design and physical DFD are shown as follows:

### Architectural Design

The diagram illustrates the Model-View-Controller (MVC) architecture in a mobile application context. The process begins with the User interacting with the Mobile Client, which sends an App Request to the Controller. The Controller manages the application logic and communicates with both the Model and View components. When the state changes, the Controller informs the Model, which then interacts with the Database to retrieve or update data. The Model notifies the View of any changes, and the View queries the Model for updated state information. The Controller also selects the appropriate View to display, which then provides the Response back to the Mobile Client for the User to see. This architecture separates concerns, with the Model handling data and business logic, the View managing the user interface, and the Controller coordinating between them, allowing for a more organized and maintainable application structure.

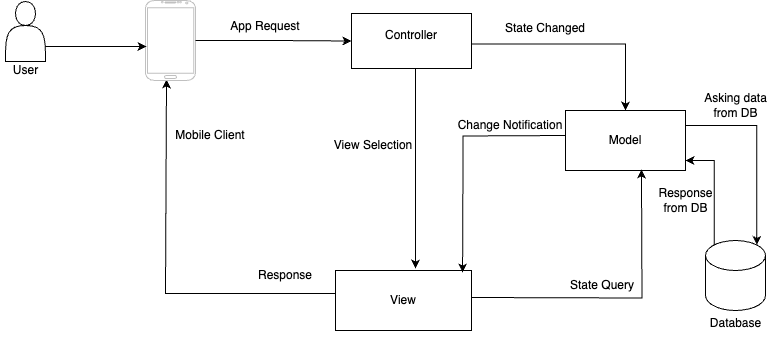
 This

Figure ‑ MVC Architecture of Recipe Recommendation Application

.

### System Flowchart

The flowchart below outlines the process flow of the Recipe Recommendation application. The process begins by checking whether the user is already registered. If the user is not registered, they are directed to complete the registration process. Once registered or if they were already registered, the user logs into the system.

After logging in, the user can browse available recipes, search for specific dishes, and view detailed information about each recipe, including ingredients, steps, and nutritional information. The user has the option to save their favorite recipes for easy access later.

Once a user has selected recipes they are interested in, they can choose to provide feedback or ratings for those recipes. If the user decides not to give feedback, they can return to browse more recipes or view their favorites. The process is completed when the user successfully engages with the application, whether by saving recipes or providing feedback, concluding their interaction.

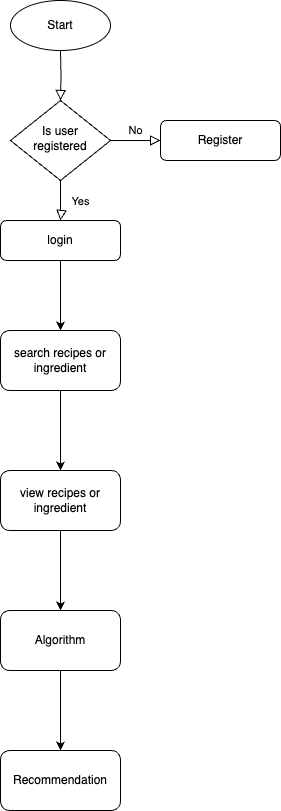


Figure 3.2‑2 System Flowchart of Recipe Recommendation Application

### JSON Parse Tree Structure

In this project, we structured our data using JSON, a lightweight data interchange format. The JSON structure below represents the key components of our Recipe Recommendation application, including user data management, recipe cataloging, favorite recipes, and feedback processing.

{

"users": [

{

"uid": "user123",

"email": "user@example.com",

"name": "John Doe",

"favoriteRecipes": [

{

"recipeId": "recipe456"

}

],

"feedback": [

{

"recipeId": "recipe123",

"rating": 5,

"comment": "Delicious and easy to make!"

}

]

}

],

"recipes": [

{

"recipeId": "recipe123",

"name": "Spaghetti Carbonara",

"description": "A classic Italian pasta dish.",

"ingredients": [

{

"name": "Spaghetti",

"quantity": "200g"

},

{

"name": "Eggs",

"quantity": "2"

},

{

"name": "Pancetta",

"quantity": "100g"

}

],

"steps": [

"Cook spaghetti according to package instructions.",

"In a separate pan, cook pancetta until crispy.",

"Whisk eggs and combine with spaghetti and pancetta."

],

"images": ["spaghetti.jpg"],

"category": "Italian",

"cookingTime": "30 minutes"

}

],

"categories": [

{

"categoryId": "category123",

"name": "Italian"

}

]

}

### Interface Design

The interface design for all the major pages of Recipe Recommendation Application is shown as follows:

1. ******Login Page UI**

Figure ‑ Login Page UI

1. **Signup Page UI**

Figure ‑ Signup Page UI

1. **Home Page UI**



Figure ‑ Home Page UI

1. **Starting page**

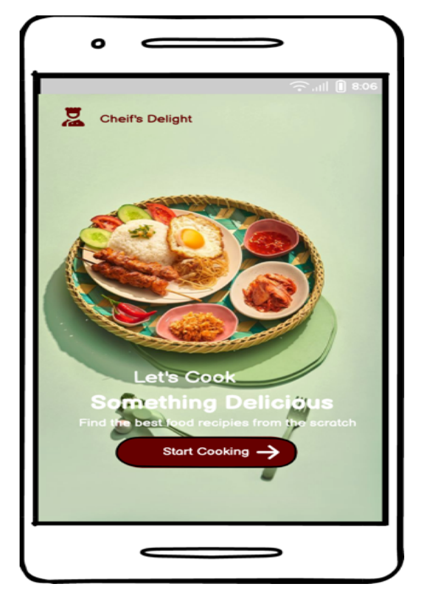


Figure ‑ Starting Page UI

### Physical DFD

**Level 1 DFD**

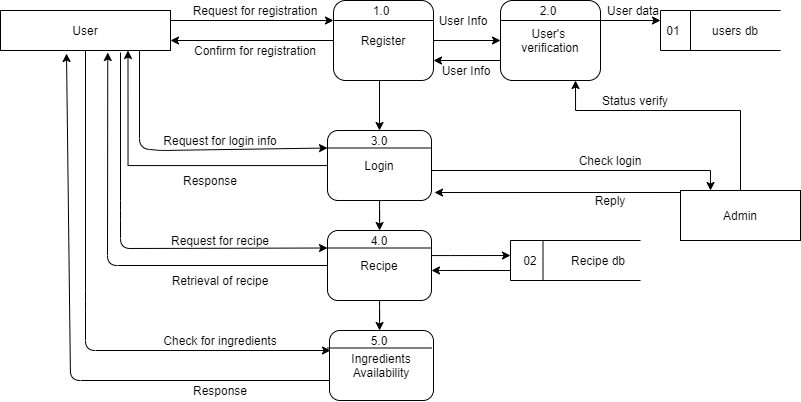


Figure ‑ Level 1 DFD of Recipe Recommendation Application

This Data Flow Diagram (DFD) illustrates the key processes and data flows in the Recipe Recommendation application from a user’s perspective. It outlines five main processes: Register/Login (1.0 – 3.0), Browse Recipes (4.0), and Ingredients Availability (5.0).

The diagram shows how users interact with these processes, starting from registering or logging in, browsing available recipes, and saving their favorite recipes for future access. Each user action triggers specific processes that interact with the underlying database to fetch or store data, with information returned to the user interface.

Additionally, there is an Admin entity that manages the recipe database, oversees user feedback, and updates recipes as necessary. This DFD provides a clear overview of the primary functions and data movements within the Recipe Recommendation application, highlighting the sequence of operations from user interaction to recipe management.

# Implementation and Testing

## Implementation

The Agile model is an iterative and adaptive approach to software development, ideal for managing the dynamic requirements of a Recipe Recommendation application. By incorporating ongoing feedback and continuously refining the product, it ensures that the application meets user needs at every phase. Planning, analysis, design, implementation, and testing are integrated throughout the lifecycle, allowing for rapid updates and flexibility.

In the **planning phase**, the system’s objectives and development strategy are defined. The **analysis phase** involves gathering user requirements, such as the ability to browse recipes, view detailed ingredients, and save favorite recipes. The **design phase** focuses on laying out how these features will be implemented in the system. The **implementation phase** involves coding and database setup, with a focus on creating a user-friendly interface and efficient backend. This is followed by **testing** to ensure the app performs as expected and delivers a smooth user experience before deployment.

### Tools Used

The tools and technologies used in the development of the Recipe Recommendation application include:

* **Android Studio**: The main integrated development environment (IDE) used for developing the Android app. It provides powerful tools for coding, debugging, and testing, enabling efficient app development.
* **Java**: The primary programming language used for building the Android application, allowing for seamless integration with the Android framework and providing strong performance.
* **PHP**: A server-side scripting language used for handling server-side logic, including the communication between the Android app and the SQL database.
* **SQL**: The database system used for storing recipes, ingredients, user data, and other relevant information, ensuring data persistence and quick retrieval.
* **Balsamiq**: A wireframing tool used for creating low-fidelity prototypes of the application to visualize and iterate on design concepts quickly.

### Implementation Detail of Modules

Table ‑ Implementation of Modules

|  |  |  |
| --- | --- | --- |
| Code | Module Name | Description |
| M1 | Login Module | Handles user authentication, verifying credentials for secure access to the system. |
| M2 | Register Module | This module has been used for Registering a new customer into the system |
| M3 | Search Module | Enables users to search and retrieve relevant information or products within the application or website. |
| M4 | Feedback Module | Collects and manages user feedback, providing insights for improving the system or services |
| M5 | Admin Module | Empowers administrators to manage users, content, and system settings for smooth platform operation. |

## Testing

The testing for Recipe Recommendation Application is done by testing the unit and system modules like login module, sign up module.

### Test Cases for Unit Testing

The test cases for the login and signup module are as follows:

Table ‑ Test Case 001 - Signup page

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Project Name:** **Recipe Mobile Application** | | | | | | | |
| **Test Case** | | | | | | | |
| **Test Case ID:** **TC\_Signup\_001** | | | | **Test Design By:** **Smriti** | | | |
| **Module Name:** **Sign Up Page** | | | | **Test Design Date: 2024-09-12** | | | |
| **Test Title**: **Register New User** | | | | **Test Executed By: Smriti** | | | |
| **Description:** **Test the sign-up page of customer** | | | | **Test execution date: 2024-09-12** | | | |
| **Pre-Conditions:** **User has all necessary details** | | | | | | | |
| **Dependencies:** | | | | | | | |
| **Step** | **Test Step** | **Test Data** | **Expected Result** | | **Actual Result** | **Status**  **(Pass/Fail)** | **Notes** |
| 1 | Navigate to sign up page |  | Sign up page should open | | As Expected, i.e., User is navigated to sign up page | Pass |  |
| 2 | Provide all required information | Email: [smriti@gmail.com](mailto:smriti@gmail.com)  Password: 123smriti | Credential can be entered | | As Expected, | Pass |  |
| 3 | Click on Sign UP button |  | Customer should be able to sign up to the system | | As Expected, i.e. Customer is able to register to the system | Pass |  |
| **Post-conditions:**  **User credentials are validated and successfully registered to Recipe Recommendation Application** | | | | | | | |

Table ‑ Test Case 002 - Login Page

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Project Name:** **Recipe Mobile Application** | | | | | | | |
| **Test Case** | | | | | | | |
| **Test Case ID:** **TC\_Login\_002** | | | | **Test Design By:** **Smriti** | | | |
| **Module Name:** **Login Page** | | | | **Test Design Date: 2024-09-12** | | | |
| **Test Title**: **Login existing Customers** | | | | **Test Executed By: Smriti** | | | |
| **Description:** **Test the login page of customer** | | | | **Test execution date: 2024-09-12** | | | |
| **Pre-Conditions:** **User has all necessary details** | | | | | | | |
| **Dependencies:** | | | | | | | |
| **Step** | **Test Step** | **Test Data** | **Expected Result** | | **Actual Result** | **Status**  **(Pass/Fail)** | **Notes** |
| 1 | Navigate to login page |  | Login page should open | | As Expected, i.e., User is navigated to log in page | Pass |  |
| 2 | Provide all required information | Email: [smriti@gmail.com](mailto:smriti@gmail.com)  Password: 123smriti | Credential can be entered | | As Expected, | Pass |  |
| 3 | Click on Login button |  | Customer should be able to log in to the system | | As Expected, i.e., Customer is able to login to the system | Pass |  |
| **Post-conditions:**  **User is validated with database and successfully logged in to Recipe Recommendation Application.**  **The account system details are logged in to the database.** | | | | | | | |

### Test Cases for System Testing

Software is just one component of a larger Recipe Recommendation system. It must integrate seamlessly with other elements, such as user data and backend services. To ensure successful integration, thorough system integration and validation tests are essential. The steps taken during the design and testing phases can significantly enhance the likelihood of successful software integration within the broader system.

A common issue in system testing is “finger pointing,” which occurs when a defect is discovered, and one developer blames another. To mitigate this, developers should proactively anticipate potential integration issues and design error handling paths to validate all information received from other system components. Conducting tests that simulate erroneous data or potential failures at the software interface is crucial, and documenting test results provides “evidence” in case of disputes. Active participation in planning and designing system tests ensures that the application is adequately evaluated.

Several types of system tests are particularly valuable for the Recipe Recommendation application:

* **Usability Testing**: Assesses the app's user-friendliness from the perspective of users, ensuring a seamless and intuitive recipe browsing experience.
* **Functionality Testing**: Tests all features of the app against specified requirements, ensuring functionalities like recipe search, saving favorites, and providing feedback work as intended.
* **Performance Testing**: Evaluates the app's performance under typical user conditions, ensuring quick and efficient responses, even during peak usage times.
* **Security Testing**: Verifies that security measures protect user data and interactions, ensuring transactions and personal information are secure from unauthorized access.
* **Stress Testing**: Simulates high-traffic scenarios to challenge the app, ensuring it can handle significant loads without crashing or degrading in performance.

Table ‑ Test Case 003 - Payment Page

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Project Name:** **Recipe Mobile Application** | | | | | | | |
| **Test Case** | | | | | | | |
| **Test Case ID:** **TC\_Search\_003** | | | | | **Test Design By:** **Smriti** | | |
| **Module Name:** **Search** | | | | | **Test Design Date: 2024-09-16** | | |
| **Test Title**: **Search recipe** | | | | | **Test Executed By: Smriti** | | |
| **Description:** **Test the search functionality** | | | | | **Test execution date: 2023-07-21** | | |
| **Pre-Conditions:** **User has all necessary details** | | | | | | | |
| **Dependencies:** **Firebase services (Firestore, Firebase Auth), internet connection** | | | | | | | |
| **Step** | **Test Step** | **Test Data** | **Expected Result** | **Actual Result** | | **Status**  **(Pass/Fail)** | **Notes** |
| 1 | Navigate to search |  | Search should open | As Expected, i.e., User is navigated to search | | Pass |  |
| 2 | Provide all required information | Momo | Ingredients for the searched items should display | As Expected, Ingredients are displayed | | Pass |  |
| 3 | Start the recipe |  | Recipe session should start | As Expected, recipe session is started | | Pass |  |
| **Post-conditions:**  **The user’s information is validated and stored in Firebase Firestore. User should be able to view the recipe and the available ingredients, stored in Firebase.** | | | | | | | |

Table ‑ Test Case 004 - Order Product

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Project Name:** **Recipe Mobile Application** | | | | | | | |
| **Test Case** | | | | | | | |
| **Test Case ID:** **TC\_Order\_Product\_004** | | | | **Test Design By:** **Smriti** | | | |
| **Module Name:** **Order** | | | | **Test Design Date: 2024-09-16** | | | |
| **Test Title**: **Order Product** | | | | **Test Executed By: Smriti** | | | |
| **Description:** **Test the order product functionality** | | | | **Test execution date: 2023-07-21** | | | |
| **Pre-Conditions:** **User has all necessary details, including a valid account and items in the cart.** | | | | | | | |
| **Dependencies: Internet Connectivity is required.** | | | | | | | |
| **Step** | **Test Step** | **Test Data** | **Expected Result** | | **Actual Result** | **Status**  **(Pass/Fail)** | **Notes** |
| 1 | Navigate to cart page |  | Cart page should open | | As Expected, i.e., User is navigated to cart page | Pass |  |
| 2 | Review product details | Product items in cart | User should see a summary of products in the cart with prices. | | As Expected, i.e., all products and prices are displayed. | Pass |  |
| 3 | Click on "Checkout" button |  | User should be directed to the checkout page. | | As Expected, i.e., User is navigated to the checkout page | Pass |  |
| 4 | Provide shipping information and payment detail | Valid address and payment detail | User can select shipping address and payment method | | As Expected, i.e., shipping address and payment options are displayed. | Pass |  |
| 5 | Click on "Place Order" button |  | Order should be processed and confirmed. | | As Expected, i.e., Order confirmation message appears. | Pass |  |
| 6 | Verify order confirmation message |  | Confirmation message should contain order details. | | As Expected, i.e., message shows order number and summary. | Pass |  |
| 7 | Check order history |  | The newly purchased product should appear in the order history. | | As Expected, i.e., product appears in order history. | Pass |  |
| **Post-conditions:**  **User is validated with database and successfully logged in to Recipe Recommendation Application.** | | | | | | | |

# Conclusion and Future Recommendation

## Lesson Learnt / Outcome

This project has been instrumental in developing a Recipe Recommendation application using Android Studio, Java, PHP, and SQL for data management. It has enhanced my familiarity with both client-side and server-side programming, allowing me to effectively manage the application’s functionality and database interactions. We learned how users can browse recipes, save favorites, and provide feedback, gaining hands-on experience with user interface design and backend integration.

The use of SQL for data management streamlined the organization of recipes, user profiles, and interactions. Throughout this project, I further developed my problem-solving skills and project management abilities, as I had to oversee all aspects of the project within the timeline.

Overall, this experience allowed me to apply theoretical knowledge from various subjects to practical scenarios, significantly enhancing my software development skills and preparing me for future projects in mobile application development.

## Conclusion

This document outlines the rationale and methodology behind building the Recipe Recommendation application. Designed for food enthusiasts and home cooks, this application aims to simplify the process of discovering and preparing recipes. With a user-friendly interface, users can easily search for recipes based on ingredients, dietary preferences, or meal types.

The app features a secure login system, allowing users to rate dishes, and provide feedback. Real-time updates keep users informed about new recipes and trending dishes. Our experience in developing this application highlights the potential of mobile technology to enhance culinary experiences and foster a community of cooking enthusiasts.

Ultimately, this application empowers users to explore diverse recipes, access detailed cooking instructions, and enjoy a more organized approach to meal planning, making cooking more enjoyable and efficient.

## Future Recommendation

The following enhancements are planned for future releases of the Recipe Recommendation application:

* **User-Generated Content**: Allow users to submit their own recipes, enhancing community engagement and expanding the recipe database.
* **Integration with Grocery Delivery Services**: Partner with grocery delivery services to enable users to order ingredients directly from the app, streamlining meal preparation.
* **Personalized Recommendations**: Implement algorithms to provide tailored recipe suggestions based on user preferences, dietary restrictions, and past interactions.
* **Meal Planning Features**: Introduce meal planning tools that help users organize their weekly meals, including shopping lists based on selected recipes.
* **Video Tutorials**: Add video tutorials for selected recipes, providing users with step-by-step guidance and enhancing the cooking experience.

# References

|  |  |
| --- | --- |
| [1] | I. K. Asmi Yajima, ""Easy" Cooking Recipe Recommendation," *"Easy" Cooking Recipe Recommendation,* 2009. |
| [2] | M. J. D de Kerf, "The Benefits of Intelligent Recipe Mangament," *The Benefits of Intelligent Recipe Mangament,* 2018. |
| [3] | J. H. ZhengXian Li, "A scalable recipe recommendation system for mobile application," *A scalable recipe recommendation system for mobile application,* 2016. |

# APPENDICES