**ABSTRACT**

Whilst there are countless recipe collection applications published on the Apple App Store and Google Play, none of them provides a complete interface with the capability to create, search, and save recipes as well as provide interactive tutorials. YURECEIP is a cross mobile application that lets people create, search for and share recipes all at once. The application provides users the ability to explore through a huge variety of recipes in the app.

In addition, the app provides the users with two distinct features such as a Shopping List where the users can select and add ingredients that need to be purchased. The second feature is about the Meal Planner, where the users can schedule meals. Also, this is a very handy application, in which every user can search for recipes, save recipes as favorites, and share recipes with friends on social media like WhatsApp. This app is a time saver providing recipes in a few clicks. Many Somali expats find it challenging to learn how to prepare traditional Somali foods and ultimately end up spending a lot of money on food that is still lacking in authenticity. This app will focus on Somali recipes and cuisine, allowing individuals from all over the world to learn about Somali cuisine.

By providing users with a simple method to learn how to prepare genuine Somali dishes, our app will address this issue. Recipes are generated and rated by both the user and recognized chefs in Somalia.

The YURECIPE app will always surprise its users with new recipes, using a rating system and personalizing the experience for each individual.

**Keywords**: Recipe, ingredients system, Flutter, Food system.

Catalog

[**ABSTRACT** 1](#_gjdgxs)

[CHAPTER 1 INTRODUCTION 4](#_30j0zll)

[1.1 Background 4](#_1fob9te)

[1.2 Motivation & Objectives 4](#_2et92p0)

[1.3 Problems to Solve 5](#_tyjcwt)

[1.4 Methodology 6](#_3rdcrjn)

[1.5 Thesis Outlines (needs checking) 6](#_35nkun2)

[Chapter 2 Literature Review 7](#_1ksv4uv)

[2.1 Overview 7](#_44sinio)

[2.2 Mobile development 7](#_2jxsxqh)

[2.3 An Introduction to Recipe 13](#_z337ya)

[2.4 Analysis of Recipe Applications 13](#_3j2qqm3)

[2.5 Whats YURECIPE App? 14](#_1y810tw)

[2.6 Related works 15](#_4i7ojhp)

[2.7 Short Comings of Related Works 15](#_2xcytpi)

[Chapter 3 User and system requirement documents 17](#_1ci93xb)

[3.1 Overview 17](#_3whwml4)

[3.2 Project Description 18](#_qsh70q)

[3.3 Software Development 18](#_3as4poj)

[**3.4 Development Technologies and Tools** 19](#_1pxezwc)

[3.5 External Interface Requirements: 26](#_23ckvvd)

[3.6 Requirements Specification 27](#_41mghml)

[3.7 Feasibility Analysis of the Application 30](#_vx1227)

[Chapter 4 System Analysis 31](#_3fwokq0)

[4.1 Overview 31](#_1v1yuxt)

[4.2 Mobile Operating Systems: 31](#_4f1mdlm)

[4.3 System Architecture 32](#_3tbugp1)

[4.4 Flowchart 33](#_28h4qwu)

[4.5 Sequence Diagram 34](#_nmf14n)

[4.6 Class Diagram 35](#_37m2jsg)

[4.7 Use Case Diagram 37](#_1mrcu09)

[4.8 Database Design – Cloud Firestore 38](#_46r0co2)

[4.9 User Interface Design for Somali Recipe App for Diaspora: 39](#_2lwamvv)

[Chapter 5 System Implementation 41](#_111kx3o)

[5.1 Overview 41](#_3l18frh)

[5.2 The Developed System 41](#_206ipza)

[Figure 5.1 Splash screen Figure 5.2 Introduction screen 41](#_4k668n3)

[5.3.1 Sign up Screen 42](#_2zbgiuw)

[Profile User 47](#_1egqt2p)

[View Recipe 48](#_3ygebqi)

[Create Recipe Screen 49](#_2dlolyb)

[Edit Profile Screen 51](#_sqyw64)

[Chapter 6 Conclusion, Challenges and Future Works (Needs Work) 53](#_3cqmetx)

[6.1 Conclusion 53](#_1rvwp1q)

[6.2 Challenges 54](#_4bvk7pj)

[6.3 Future works 54](#_2r0uhxc)

# CHAPTER 1 INTRODUCTION

## 1.1 Background

People have become used to the ease of online ordering and delivery services like seamless over the past few years. The ubiquity of social networking websites has also had a profound effect on how many people live their daily lives: it's hard for someone who checks Facebook multiple times per day not to "accidentally" reconnect with an old flame—on purpose or otherwise! [1].

This can merely assert that mobile applications have already made their ways to our lives. According to butler (2011), in recent years, the emergence of smartphones has changed the definition of mobile phones. The phone is no longer just a communication tool, but also an essential part of people's communication and daily life. Various applications added unlimited fun to people's lives. It is certain that the future of the network will be the mobile terminal.

Whether or not people enjoy cooking, they gather recipes and ideas for meals. The number one reason why people love their phones is because they can be used for entertainment. Many people use their phones to play games, listen to music and watch videos online. This is especially true for younger generations who are constantly on the go.

Maybe they have a collection of recipes passed down from a loved one. In either case, they certainly need to find a better way to organize their recipes for the long-term than using index cards in file folders—which is old and tedious. cooking with your phone will be tastier when you have the right recipes! The goal of this capstone project is to develop a mobile application that will improve your cooking skills and make grocery shopping easier by turning your phone into a virtual sous chef.

The YURECIPE app is a fantastic resource for anyone who enjoys cooking and experimenting with new recipes. Users can look for recipes, share them with friends, and save them as favorites. Users can also add recipes to and remove them from your own personal cookbook using the app. The application is very simple to use. Both beginners and professional chefs can use it because of its user-friendly layout. The app looks amazing on any mobile device thanks to its simple design, modern visuals, and animations. With only a few clicks, the user can find recipes and make their own cookbook by adding, viewing, or removing recipes as wanted. Users can find recipes easily because to the interface's simplicity and ease.

## 1.2 Motivation & Objectives

I was motivated to create this recipe app by a personal issue I had; I've always been curious in how technology might assist us in resolving issues. But I didn't start developing the app itself until I discovered a problem that existed in the real world and needed to be fixed.

I found that my lack of cooking abilities was costing me hundreds of dollars per month after keeping track of my daily activities and reflecting on the challenges I face in the kitchen.

People from all over the world, including Somalis, are looking for a means to learn how to cook, but they frequently lack the same resources.

Everyone will have access to this user-friendly platform through the recipe app I'm developing, which will enable them to share their recipes and learn from others'.

In order to better their own cooking abilities, users in this program will learn from one another's experiences and share their favorite recipes in an online community. While giving experienced cooks some advice on how to effectively make various foods, it will also teach users how to correctly use fundamental ingredients and tools.. In hopes of personalizing the user experience, it will also propose and suggest recipes to users.

The App will have a user-friendly design that makes it simple to use. Users of the app will also be able to rate and contribute their own recipes, which will improve their decision-making when cooking. The app will include simple instructions for the recipes so that users may prepare the meals themselves.

## 1.3 Problems to Solve

The current state of affairs in the world of cooking is challenging, with people facing countless challenges in their search for new and authentic dishes from different cultures. The situation is particularly challenging for those who are passionate about Somali cuisine. This is why building this recipe app is not only a personal mission to solve my own cooking problem but also an imperative to help thousands of others who face similar challenges.

**The three main problems that YURECIPE seeks to solve are as follows:**

1. Eating out is often expensive, and many people struggle to afford it regularly. This Recipe App provides a solution by allowing users to learn how to cook traditional Somali foods at home, thus reducing expenses and also broadening their cooking skills..
2. Despite the wide range of cooking and recipe resources available today, many people still feel disconnected and isolated in their pursuit of new dishes. This YURECIPE seeks to build a community where users can share their unique recipes and experiences, promoting cultural exchange and creating a supportive network for all.
3. YURECIPE is a crucial tool for solving real-world problems faced by those who love to cook and try new dishes. By tackling the problems listed above, it will have a real and positive effect on the lives of thousands of people around the world.

## 1.4 Methodology

I'll create an intuitive app that addresses the issues I raised above to find solutions. The app will serve as a repository for all potential fixes to issues and a platform for foodie communities to exchange recipes. The application will serve as a one-stop shop for all matters pertaining to food and cooking. Furthermore, it will make it simple for people to share their recipes with others. The app will keep data on various food items, recipes, ingredients, and so forth. The users can choose from these categories or create their own based on their preferences.

## 1.5 Thesis Outlines

The analysis of the thesis aims to assess whether or not the proposed capstone project is practical. The objective of this capstone, then, is to develop a “YURECIPE” mobile application that can improve users' cooking skills and streamline their food purchases Instead of searching through YouTube videos or flipping through cookbooks for hours, you can use an app to find just the recipe you're looking for.

**Chapter One**: Introduces the problem being addressed, explains the motivation and objectives, and outlines the methodology used.

**Chapter Two:** Discusses mobile applications and development, specifically focusing on recipe app, YURECIPE and related works in this field.

**Chapter Three:** Outlines the methodology used in the project, including the software development process and technology tools needed. Also, describes the requirements for the mobile application.

**Chapter Four:** Analyzes the system and provides a detailed description of the mobile application being developed.

**Chapter five**: Describe the system implementation of the mobile application

**Chapter Six**: The final chapter of the thesis is Chapter Six. The conclusion of the thesis, challenges, and future works are all discussed.

# Chapter 2 Literature Review

## 2.1 Overview

In this chapter, we will explore the area of developing mobile applications.

We'll go through the guidelines and tactics for developing mobile apps, as well as the various mobile and web application development approaches.

The YURECIPE app, which is presently being built as part of this project, will be the specific focus of our examination of the idea of recipes and recipe mobile applications. We will also look at other similar efforts in this area and point out their pros and cons.

Lastly, we'll look at developing native and cross-platform mobile applications.

## 2.2 Mobile development

Mobile application development is the set of processes and procedures involved in writing software for small, wireless computing devices, such as smartphones and other hand-held devices.

Like web application development, mobile application development has its roots in more traditional software development. One critical difference, however, is that mobile apps are often written specifically to take advantage of the unique features of a particular mobile device. For example, a gaming app might be written to take advantage of the iPhone's accelerometer or a mobile health app might be written to take advantage of a smartwatch's temperature sensor.

Today, the two most prominent mobile platforms are iOS from Apple and Android from Google. Phones and tablets from Apple come preload with essential applications, including a full web browser and the Apple App Store. Android devices also come preload with similar apps and you can install more using the Google Play Store. [4]

Mobile app development has grown rapidly with the increasing popularity of smartphones and other handheld devices. There are two major platforms for mobile app development: iOS by Apple and Android by Google. [21] Both platforms offer app stores for users to download and install additional apps, and come preload with essential apps such as a web browser. Mobile app development also differs from traditional software development by leveraging unique features of each device, like accelerometer or temperature sensors.

The mobile app development process typically involves the following steps: requirement gathering and analysis, designing, coding and testing, and deployment. It is important to choose the right development platform, tools, and approach for the specific requirements and goals of the app. To ensure the success of a mobile app, it is also important to consider factors such as user experience, performance, security, and compatibility with various devices and operating systems. The rapid growth in the use of mobile devices has created huge opportunities for businesses to reach their customers through mobile applications.

### 2.2.1 Approaches to Mobile Application Development: A Comprehensive Overview of Native, Hybrid, and Cross-Platform Development

The widespread use of mobile devices has driven the growth of the mobile app industry and has led to the development of new approaches to mobile application development. Mobile app development involves the creation of software applications that are designed to run on small wireless devices such as smartphones and tablets. This can range from simple utility apps to complex gaming apps, and everything in between. In this introduction, we will explore the different approaches to mobile app development and the factors that influence the choice of development method. These approaches include Native, Hybrid, and Cross-platform development, each with its own unique strengths and limitations. Understanding these approaches will help app developers make informed decisions when it comes to developing their own mobile apps. [22]

**There are several approaches to mobile application development, including:**

* Native app development: building apps specifically for one platform, such as iOS or Android, using the platform's development tools and programming languages.
* Hybrid app development: using HTML, CSS, and JavaScript to build apps that can run on multiple platforms. [22]
* Cross-platform app development: using tools that allow for the creation of a single code-base that can be deployed to multiple platforms.

Each approach has its own advantages and disadvantages, and the best approach depends on the specific requirements and goals of the app. [5]

Native app development involves writing code in a specific programming language for a particular platform. For example, an iOS app would be written in Swift or Objective-C, while an Android app would be written in Java or Kotlin. Native apps offer the best performance, as they are built specifically for the platform and can take full advantage of its features and capabilities. However, native app development can be more time-consuming and expensive, as it requires separate development for each platform.

Hybrid app development uses web technologies such as HTML, CSS, and JavaScript to build apps that can run on multiple platforms. These apps are essentially web pages that are wrapped in a native container, and they are often slower and less feature-rich than native apps. However, they are easier to develop and maintain, and they can be deployed to multiple platforms more easily.

Cross-platform app development uses tools and frameworks that allow developers to create a single code-base that can be deployed to multiple platforms. This approach offers a cost-effective and efficient way to build apps that can run on multiple devices. However, cross-platform development can result in apps that are less feature-rich than native apps, and performance may not be as good as native apps.

In summary, the best approach to mobile app development depends on the specific requirements and goals of the app, as well as factors such as budget, time frame, and desired level of performance.

#### 2.2.1.1 Native App development

Native app development refers to the process of creating software applications for mobile devices using a platform-specific programming language, such as Swift or Objective-C for iOS and Java for Android. The resulting applications are optimized for the specific device and operating system, providing a more seamless and efficient user experience. [19]

##### Advantages of Native App Development

**Performance and Speed:** Native apps are built specifically for the operating system and hardware of a specific device, resulting in a faster and more responsive user experience. This is particularly important for resource-intensive applications, such as gaming apps or augmented reality apps, which require quick and reliable performance[20].

**Access to Native Features:** Native apps have access to all the features of a device, such as the camera, GPS, accelerometer, and more. This allows developers to create truly unique and innovative applications that take advantage of the latest hardware and software capabilities. [20]

**Integration with the Operating System:** Native apps are integrated into the operating system, providing a seamless and integrated user experience. This includes access to device-specific features, such as notifications, multitasking, and more.

**Better User Experience:** Native apps provide a more natural and intuitive user experience, as they are designed specifically for the device and operating system. This results in a more efficient and enjoyable user experience, as users are familiar with the look and feel of the device[33].

##### Disadvantages of Native App Development

**Cost:** Developing a native app can be more expensive than developing a web or hybrid app, as it requires a separate development process for each platform. This can result in higher development costs and longer development times.

**Maintenance:** Maintaining a native app can also be more expensive and time-consuming, as updates must be released for each platform separately. This can result in longer development times and higher maintenance costs.

**Platform-Specific Development**: Native app development requires a platform-specific development process, which means that separate teams of developers may be needed for each platform. This can result in higher development costs and longer development times.

In conclusion, native app development offers a number of benefits, including better performance and speed, access to native features, and a more intuitive user experience. However, these benefits come at a cost, including higher development costs, longer development times, and higher maintenance costs.

As the mobile industry continues to evolve, new and innovative approaches to app development are emerging, including hybrid and cross-platform development. These approaches aim to provide the best of both worlds – the benefits of native app development with the cost-effectiveness and flexibility of web development[34].

#### 2.2.1.2 Hybrid app development

Hybrid app development is the process of building mobile apps that work across multiple platforms, including both iOS and Android. It involves using a combination of native code and web technologies, such as HTML, CSS, and JavaScript, to create a single code-base that can be deployed on different platforms. This approach offers many advantages over traditional native app development, which involves writing separate code for each platform. [16]

One of the primary benefits of hybrid app development is cost-effectiveness. Since a single code-base is used for multiple platforms, the development process is more streamlined and efficient, reducing the overall cost of development. Additionally, hybrid apps can be developed and maintained more easily than native apps, as updates can be made to the code-base without having to make separate updates for each platform.

Another advantage of hybrid app development is that it allows for a high degree of flexibility. Since the code-base is built using web technologies, it can be easily modified and updated, making it ideal for businesses that need to quickly respond to changing market conditions. Additionally, hybrid apps can be used on a variety of devices, including smartphones, tablets, and even desktop computers, making them ideal for businesses that need to reach a broad audience. [17]

Hybrid app development also offers a number of performance benefits. By leveraging native device capabilities, such as accelerometer, cameras, and Geo location services, hybrid apps can deliver high-performance experiences that are on par with native apps. Additionally, hybrid apps are generally faster to load and less resource-intensive than native apps, which can help improve the overall user experience.

However, it's important to note that hybrid app development is not without its challenges. One of the biggest challenges is ensuring that the app works correctly across all platforms, as well as ensuring that it takes advantage of the unique features of each platform. This can be a complex and time-consuming process, requiring significant expertise and experience to get right. Additionally, hybrid apps can sometimes be slower and less responsive than native apps, which can negatively impact the user experience.

To overcome these challenges, it's important to choose the right development tools and frameworks. For example, frameworks like Ionic and React Native can help streamline the development process and ensure that apps work correctly across all platforms. Additionally, it's important to partner with an experienced development team that has expertise in hybrid app development, as well as a deep understanding of the unique requirements of each platform.

In conclusion, hybrid app development is a powerful tool that can help businesses reach a broad audience, reduce development costs, and deliver high-performance experiences to users. However, it's important to carefully consider the challenges and choose the right development tools and frameworks to ensure that the development process is streamlined and the app delivers the desired results. With the right approach and the right partners, hybrid app development can be an effective way to bring your mobile vision to life and reach a wider audience.

#### 2.2.1.3 Cross-platform app development

Cross-platform app development refers to the creation of software applications that run seamlessly across multiple platforms, such as Android, iOS, and web. This type of development enables developers to write code once and deploy it on multiple platforms, thereby reducing development time, effort, and cost. [5]

##### Advantages of Cross-platform App Development

Reduced Development Costs: Cross-platform app development enables developers to write code once and deploy it on multiple platforms, thereby reducing development time, effort, and cost.

Consistent User Experience: Cross-platform development ensures a consistent user experience across all platforms, thereby increasing the app's user engagement and satisfaction.

**Improved Time to Market:** With cross-platform development, apps can be deployed to multiple platforms simultaneously, reducing the time to market.

Access to a Larger User Base: Cross-platform apps can reach a larger user base, including users on multiple platforms, thereby increasing the app's reach and potential revenue.

Cross-platform Development Tools

There are many tools available for cross-platform app development, including:

**React Native:** React Native is a popular cross-platform app development framework that enables developers to build native-like apps for Android and iOS using JavaScript and React.

Xamarin: Xamarin is a Microsoft-owned cross-platform development tool that enables developers to build apps for Android, iOS, and Windows using C#.

Flutter: Flutter is a UI toolkit for building beautiful and high-performance cross-platform apps using Dart.

PhoneGap: PhoneGap is a popular open-source cross-platform app development framework that enables developers to build apps using HTML, CSS, and JavaScript. [18]

##### Challenges in Cross-platform App Development

**Performance:** Cross-platform apps may not perform as well as native apps, as they rely on a layer of abstraction to work across multiple platforms.

**Limited Access to Native API:** Cross-platform app development tools may not provide full access to the native API of each platform, limiting the app's functionality.

**Fragmentation**: Cross-platform apps may not look and feel the same across different platforms, due to differences in screen size, aspect ratio, and platform-specific design elements.

In conclusion, cross-platform app development is a valuable approach for organizations that want to reduce development costs, ensure a consistent user experience, and reach a larger user base. By leveraging the right cross-platform development tools, organizations can build high-quality, native-like apps that deliver great user experiences across multiple platforms.

## 2.3 An Introduction to Recipe

A recipe is a set of instructions for preparing a particular dish or meal. It typically includes a list of ingredients and the specific amounts required, as well as step-by-step instructions for preparing the dish. Recipes can be for a variety of dishes such as appetizers, main courses, sides, and desserts and can come from various cuisines. Recipes can be found in cookbooks, on websites and food blogs, in newspapers and magazines, and in mobile apps. They can also be passed down from family members or shared among friends. Recipes often include information about the dish's origin, the skill level required to prepare it, and the number of servings it yields.

Additionally, recipes often include information about cooking time, temperature, and any special equipment or techniques needed to prepare the dish. Many recipes also include variations or substitutions for ingredients, as well as serving suggestions and tips for storage or leftovers.[ 15]

Recipes can vary in complexity, from simple, quick and easy meals to more complex and time-consuming dishes. Some recipes are designed for beginners, while others are intended for more experienced cooks. Many recipe sources include ratings and reviews from other users, which can be helpful in determining the quality and difficulty level of a recipe.

There are many different types of recipes, including traditional recipes passed down through generations, modern twists on classic dishes, and fusion cuisine combining elements of different cultural cuisine. Some recipes are also tailored to specific dietary restrictions, such as gluten-free, low-cab, or vegan recipes.

To summarize recipes are an essential tool for anyone who loves to cook or wants to learn how to cook. They provide a step-by-step guide to preparing a particular dish and give cooks the confidence to try new things in the kitchen. With the vast array of recipe sources available today, there are countless recipe options for cooks to explore, whether they are looking for a classic dish, or something new and exciting. [14]

## 2.4 Analysis of Recipe Applications

A recipe app is a mobile application that allows users to access, organize, and follow recipes on their smartphones or tablets. These apps provide a convenient and efficient way for individuals to discover new recipes, plan meals, and shop for ingredients.

Recipe apps typically offer a wide range of features to help users in their cooking journey, these features can include:

A database of recipes: Many recipe apps come with a vast collection of recipes from different cuisines and dietary restrictions. Users can search for recipes based on ingredients, dietary restrictions, and other criteria.

* **Meal planning**: Some apps allow users to plan out their meals for the week, and generate a shopping list of ingredients accordingly.
* **Ingredient substitution:** Some apps also have a feature that allows users to substitute ingredients with others that they may have on hand.
* **Step-by-step instructions:** Most recipe apps provide step-by-step instructions, along with photos and videos, to help users prepare the recipe.
* **Social features:** Some apps allow users to share recipes with friends, save their favorite recipes, or even connect with other food enthusiasts.
* **Nutritional information:** Some apps provide nutritional information for each recipe, including calorie counts, to help users make informed choices about their meals.

Many recipe apps are free to download and use, but some may require a subscription or in-app purchases to access certain features. Some apps are also available for smart watches, which allow users to access the recipes and cooking instructions on their wrist.

Recipe apps can be very useful for people who love to cook, or for those who are just starting to learn how to cook. They provide a convenient way to access a wide variety of recipes and to plan meals. They can also be a great tool for those who are trying to eat healthier or to cook with specific dietary restrictions in mind. However, it's important to remember that not all recipe apps are created equal, and it's important to do some research and read reviews before downloading one.

In conclusion, recipe apps are mobile applications that provide users with a convenient and efficient way to discover, organize and follow recipes on their smartphones or tablets. They come with a wide range of features such as a database of recipes, meal planning, ingredient substitution, step-by-step instructions, timers and alerts, social features and nutritional information, and can be very useful for people who love to cook and for those who are just starting to learn how to cook. However, it's important to remember that not all recipe apps are created equal, and it's important to do some research and read reviews before downloading one.

## 2.5 What’s YURECIPE App

YURECIPE application is a very useful app for Somalis who love to cook and try out Somali traditional recipes. It allows users to c search for recipes, share them among friends or save them as a favorite. The app also lets you create and delete your own personal cookbook of favorite dishes. The application is very easy to use. It has a simple and intuitive user interface that makes it suitable for both beginners and experienced chefs. The app has a clean design with modern graphics and animations that make it look great on any mobile device. This application provides recipes in a few clicks and allows the user to create their own cookbook by adding, viewing, or deleting recipes at will. The interface is clean and simple. so users can find recipes quickly.

In summary, YURECIPE application is a powerful and user-friendly recipe app that offers a wide range of features to help users discover and organize recipes, plan meals, and shop for ingredients. It’s simple and intuitive interface makes it suitable for both beginners and experienced cook.

## 2.6 Related works

### 2.6.1 YouTube Videos

One popular method of learning new recipes for Somali people is through watching cooking videos on YouTube. There are many channels dedicated to Somali cuisine, offering tutorials, cooking demonstrations, and recipe collections. Some popular channels include "Somali Kitchen", "Somali Recipes", and "Somali Cuisine". [6]

### 2.6.2 Facebook Channels

In addition to YouTube, Facebook is also a popular platform for Somali people to learn about new recipes. There are many Facebook pages and groups dedicated to Somali cuisine that offer recipe collections, cooking tips, and food-related discussion. Some popular pages include "Somali Food Recipes", "Somali Cooking", and "Somali Cuisine". [7]

### 2.6.3 Books

For those who prefer a more traditional approach to learning about new recipes, cookbooks dedicated to Somali cuisine are also widely available. These books offer a comprehensive collection of recipes, cooking tips, and information on the history and cultural significance of Somali food. Some popular cookbooks include "Somali Cuisine: Recipes and Traditions" by Xawaash and "The Somali Kitchen" by Hawa Hassan. [8],[9]

## 2.7 Short Comings of Related Works

#### 2.7.1 YouTube Videos

One of the major shortcomings of relying solely on YouTube videos for learning about new recipes is the lack of organization and structure. While there are many great cooking channels dedicated to Somali cuisine, it can be difficult for viewers to find the specific recipe or cooking demonstration they are looking for. Additionally, not all YouTube channels are created equal, and some may provide low-quality or inaccurate information. [10]

**Lack of Organization and Structure**

* One of the major shortcomings of relying solely on YouTube videos for learning about new recipes is the lack of organization and structure. While there are many great cooking channels dedicated to Somali cuisine, it can be difficult for viewers to find the specific recipe or cooking demonstration they are looking for.

**Quality and Accuracy of Information**

* Not all YouTube channels are created equal, and some may provide low-quality or inaccurate information. This can be a problem for those who are looking for accurate and reliable recipes. Additionally, some videos may not be well-produced or may be difficult to follow, making it difficult for viewers to learn how to make the recipe correctly.

**Incomplete or Missing Information**

* Another potential problem with YouTube videos is that some may not provide all of the information needed to make the recipe successfully. For example, a video may not include the complete list of ingredients or may not provide specific cooking times or temperatures.

**Time Consuming**

* While YouTube is a great resource for learning about new recipes, it can also be a time-consuming platform. With so many videos to choose from, it can take time to find the right one and even longer to watch it in its entirety.

### 2.7.2 Facebook Channels

While Facebook is a popular platform for learning about new recipes, it also has its own set of shortcomings. One major issue with Facebook is the amount of misinformation that can be spread on the platform, especially when it comes to recipes. Additionally, there is a lack of structure and organization on Facebook, making it difficult for users to find specific information or recipes. [11]

**Limited Content Availability**

One of the challenges with using Facebook channels for finding new recipes is that the availability of content may be limited. Many Facebook pages are created by individuals or small groups, and they may not have the resources to produce new content on a regular basis. This can result in a lack of new and interesting recipes for users to try.

**Quality and Accuracy of Information**

Just like with YouTube, not all Facebook pages are created equal, and some may provide low-quality or inaccurate information. Additionally, some pages may not have any verification or review process, leading to incorrect recipes or information.

**Difficulty in Searching and Finding Recipes**

Another challenge with using Facebook for recipe research is the difficulty in searching and finding recipes. With so much content available, it can be difficult to find what you are looking for and to filter out irrelevant content.

### 2.7.3 Books

One of the major shortcomings of relying on cookbooks is the cost. Many cookbooks can be quite expensive, especially if they are focused on a specific cuisine like Somali. Additionally, the information in cookbooks may not be as up-to-date as online resources, as it can take time for a cookbook to be published and distributed.[12]

**Outdated Information**

One of the challenges with using cookbooks and recipe books is that the information may be outdated. As new ingredients and cooking techniques are introduced, older cookbooks may no longer be relevant or up-to-date.

**Limited Selection**

Another challenge with using cookbooks and recipe books is the limited selection of recipes. While a single book may have a great selection of recipes, it is unlikely to cover all of the recipes or ingredients you are interested in.

**Inconvenient to Use**

Additionally, cookbooks and recipe books can be inconvenient to use. If you are looking for a recipe while in the kitchen, it may be difficult to flip through pages or to find the specific recipe you are looking for. [13]

# Chapter 3 User and system requirement documents

## 3.1 Overview

In this chapter, we will delve into the world of Software Development Methodology. We will discuss the technologies and tools used to conduct this project being built, including IDE’s and programming languages used.

Furthermore, we will examine the requirement specification for the functional and nonfunctional.

Finally, we will examine the feasibility analysis of YURECIPE Application.

## 3.2 Project Description

The goal of this step is to ensure that all requirements for the project are clearly defined and meet the desired outcome. This includes both functional and non-functional requirements.

Functional requirements describe the specific functions and features that the app or system should have. These are the requirements that users will directly interact with and that determine the overall functionality of the app. Examples of functional requirements may include features such as user registration, search functionality, and the ability to save and retrieve data.

Non-functional requirements, on the other hand, describe the constraints and properties of the app or system that are not directly related to its functionality. These requirements may include performance, security, usability, and scalability requirements. For example, a non-functional requirement may specify that the app must be able to handle a certain number of concurrent users, or that it must comply with specific security standards.

By ensuring that all requirements are consistent, precise, and complete, we can be confident that the final outcome of the project will meet the expectations of the stakeholders. This step is crucial for the success of the project as it lays the foundation for the design, development, and testing phases.

## 3.3 Software Development

The project has used the Agile methodology in software development projects. The Agile approach values customer collaboration, flexibility, and delivering working software in small iterations through sprints. This method allows for regular feedback and allows for changes in customer requirements to be easily incorporated into the development process

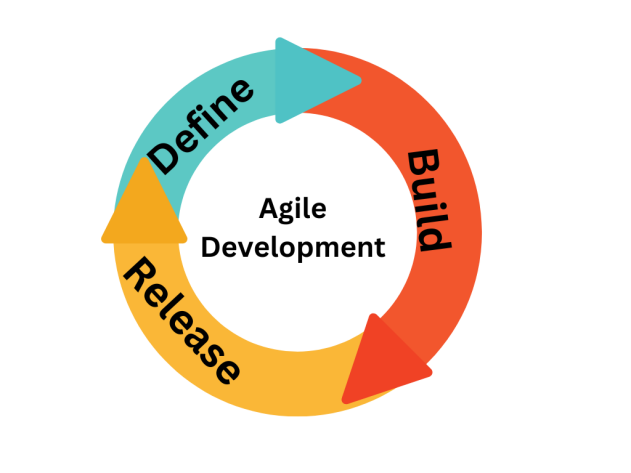


Figure 3.1 Agile Development Model

**Agile Development Model**

Agile development is an iterative approach to software development that values customer

collaboration, flexibility, and delivering working software frequently. Unlike the waterfall model,

the agile approach prioritizes customer involvement and allows for changes to be made

throughout the development process.

The Agile methodology breaks down the project into

smaller, manageable portions called sprints. Each sprint is a cycle of planning, development,

testing, and review, with the goal of delivering a usable product increment at the end of each

sprint. This allows for regular feedback from the customer, allowing for changes to be made

based on their needs. The Agile approach emphasizes face-to-face communication between team

members, regular check-ins with customers, and a willingness to adapt and change course as

needed.

**Highlights of Agile benefits:**

*  Customer involvement and regular feedback throughout the development process.
*  Flexibility to accommodate changes in customer requirements.
*  A focus on delivering working software quickly, allowing for early customer validation.
*  An iterative approach to development, allowing for regular reassessment and adjustment.
*  Emphasis on face-to-face communication and collaboration among team members.

**3.4 Development Technologies and Tools**

In today's fast-paced world, technology plays a significant role in every aspect of our lives, including cooking and meal planning. The development of mobile applications has made it easier for people to access and organize recipes, plan meals, and shop for ingredients. One such application is YURECIPE, a recipe app that is designed to make cooking and meal planning more convenient and enjoyable.

* Firebase
* Flutter
* dart
* Android Studio

we will delve into the technical details of the technologies used in the development of YURECIPE application. We will provide a comprehensive overview of each technology, including its definition, features, and how it is used in the development of the app.

### 3.4.1 Firebase

Firebase is a mobile and web application development platform created by Firebase, Inc. in 2011, and then acquired by Google in 2014. It is a comprehensive platform that provides a wide range of tools and services for developers to build, test, and grow their applications. Firebase offers a number of services including:

* Real-time Database: A cloud-hosted NoSQL database that allows developers to store and sync data across all connected devices in real-time.
* Authentication: Firebase Authentication provides a secure and easy way to authenticate users in your application. It supports various authentication methods including email and password, phone number, and social providers like Google, Facebook, and Twitter.
* Cloud Firestore: A flexible, scalable NoSQL cloud database that enables you to store, sync, and query data for your mobile and web apps.
* Cloud Storage: A scalable and secure object storage service that allows you to store and retrieve files, such as images and videos.
* Cloud Functions: A serverless environment that allows you to run your own Node.js code in response to certain events, such as a new user signing up or a new image being uploaded.
* Hosting: Firebase Hosting provides a simple and fast way to host web applications and static files. It also includes a free SSL certificate and global CDN.
* Machine Learning: Firebase offers a number of machine learning services such as Firebase ML Kit, which allows developers to easily integrate machine learning functionality into their apps.
* Crashlytics: A crash reporting service that allows you to track and fix crashes in your application.
* Performance Monitoring: A service that helps you monitor the performance of your app and identify potential issues.
* App Indexing: Enables Google to index your app content and surface it in the Google Search results.
* Dynamic Links: Allows you to create deep links that work across multiple platforms and devices.

One of the main benefits of Firebase is that it provides a comprehensive set of tools and services that can be easily integrated into a single application, allowing developers to focus on building their app rather than managing infrastructure. Additionally, Firebase is a fully managed service, which means that it takes care of the heavy lifting such as server management, scaling, and security. This allows developers to focus on building and growing their applications.

In conclusion Firebase is a comprehensive platform that provides a wide range of tools and services for developers to build, test, and grow their mobile and web applications. With its real-time database, authentication, cloud storage, hosting, machine learning, and other services, Firebase makes it easy for developers to add powerful functionality to their apps without having to worry about managing infrastructure.

### 3.4.2 Android Studio

Android Studio is the official Integrated Development Environment (IDE) for android application development. Android Studio provides more features that enhance our productivity while building Android apps.

Android Studio was announced on 16th May 2013 at the Google I/O conference as an official IDE for Android app development. It started its early access preview from version 0.1 in May 2013. The first stable built version was released in December 2014, starts from version 1.0.

Since 7th May 2019, Kotlin is Google's preferred language for Android application development. Besides this, other programming languages are supported by Android Studio.

Some of the key features of Android Studio are as follows [4]:

**Instant Run** – a feature that pushes code and resource changes to the running app.

It allows changes to be made to the app without the need to restart the app, or

rebuilding the APK, so that the effects can be seen instantly.

**An Emulator** – a virtual android device that can simulate variety of hardware

features such as GPS location, network latency, motion sensors, and multi-touch

input that can be used to run and install the app. It can then be used for testing

purposes.

**Testing Tools and Frameworks** – extensive testing tools such as, JUnit 4 and

functional UI test frameworks are included with Android Studio. Espresso Test

Recorder can generate UI test code by recording the developer’s interactions with

the app on a device or emulator.

The tests can be run on a device, an emulator, in

Firebase Test Lab, or on a continuous integration environment.

### 3.4.3 Introduction to Flutter

Flutter is a mobile app development platform created by Google. It allows developers to create web, desktop, and cross-platform apps that run on Android and iOS devices. Flutter uses a reactive programming language called Dart, making development faster and easier than traditional methods.

Flutter is a mobile app SDK ([software development kit](https://www.forbes.com/sites/forbestechcouncil/2021/04/07/how-the-api-and-sdk-revolutions-transform-software-development/?sh=59ca7b136563)) for building high-performance, high-fidelity apps for iOS and Android.

With powerful graphics and animation libraries, the Flutter framework makes it easy to build user interfaces that react smoothly in response to touch.

Flutter is built on the Dart programming language and provides a fast development workflow with hot reloading, so you can quickly iterate on your code.

Some of the top features of Flutter include:

* **Dart programming language:** Flutter uses the Dart programming language, which is easy to learn and allows you to develop high-quality apps.
* **Hot reload:**Flutter's "[hot reload](https://www.geeksforgeeks.org/difference-between-hot-reloading-and-live-reloading-in-react-native/)" feature lets you quickly and easily make changes to your app without restarting it.
* **Expressive and flexible UI:**Flutter's UI elements are built using the same principles as Google's Material Design guidelines, giving you an expressive and flexible way to create beautiful apps.
* **Native performance:** Flutter apps are compiled to native code, giving you the best possible performance on both iOS and Android.
* **Open source:** Flutter is an [open-source project](https://www.verizon.com/business/small-business-essentials/resources/in-recent-years-open-source-software-has-become-more/), which means you can use it for free and contribute to the platform's development.

#### 3.4.3.1 How does Flutter work?

Flutter is a mobile app SDK that allows you to create high-quality native apps on iOS and Android. It is also the primary way to develop cross-platform apps with Google's new Fuchsia operating system.

The Flutter framework makes it easy for you to build user interfaces that are beautiful, fast, and responsive. The framework is also extensible, so you can easily add new features and functionality.

When creating a Flutter app, you'll be working with what's called a "widget." Widgets are the basic building blocks of a Flutter app, and they're used to create both the visual components of an app (like buttons and text) and the functional elements (like Stateless Widgets).

There are two types of widgets: Stateless Widgets and Stateful Widgets. As the name suggests, Stateless Widgets are those with no internal state (or "state," for short). These are the most straightforward widgets and are often used for buttons or text.

On the other hand, Stateful Widgets have an internal state, and this state can be changed over time, and it will be reflected in how the widget looks and behaves. Stateful Widgets are often used for user input fields or animation controllers.

You can easily create both Stateless and Stateful Widgets using the Dart programming language. You can also use various other development tools to help with the development process, including the Dart Analyzer and the Flutter Inspector.

#### 3.4.3.2 Advantages of Flutter

There are several key benefits of using Flutter for developing mobile apps. They include:

* **Flutter is fast:**It uses a Dart programming language compiled into native code, meaning there is no need for a JavaScript bridge. This results in apps that are fast and responsive.
* **Flutter creates cross-platform applications:**The same code can be used to build apps for both iOS and Android devices from a single code-base rather than switching between different platforms. This can save a lot of time and effort when developing mobile apps. In addition, Flutter can be used for web development to create web applications.
* **Flutter has a rich set of widgets:** Widgets are the building blocks of Flutter apps, and a wide variety of them are available. This makes it easy to create beautiful and custom user interfaces.
* **Flutter is open source:**Anyone can contribute to the development of Flutter, and a growing community of developers is using it. In addition, many helpful docs/tutorials are available online, created by the Flutter community on sites like Github.
* **Flutter is free:** There are no licensing fees or charges for Flutter app development. This makes it an attractive option for startups and developers who want to create high-quality apps without spending much money.
* **Google backs Flutter:**As a Google product, it receives significant support from the tech giant, which constantly works to improve it. This means developers can be confident that Flutter will continue developing and supporting it.
* **Getting inspired by big successful apps built with Flutter:**Some well-known examples include Google Ads, Reflectly, Xianyu by Alibaba, and Postmuse. This is handy for developers who want to see what is possible with Flutter before committing to using it for their projects.
* **Easy debugging:**The Dart programming language has excellent tools for debugging, like the Dart Analyzer and the DevTools suite. This makes it easy to find and fix bugs in Flutter apps.
* **Automated testing:**The Dart programming language has good support for [automated testing](https://www.adservio.fr/post/automation-testing-types-frameworks-steps-benefits), and the Flutter framework also has its own set of tools for testing. This makes it easy to create a unit, widget, and integration tests for Flutter apps so developers can constantly optimize and improve the quality of their apps.
* **Hardware and software utilization:**Flutter apps can access the full range of hardware and software capabilities. This means they can take advantage of features such as the camera, GPS, and fingerprint scanner. They can also use platform-specific features, such as push notifications on Android apps or Face ID on Apple iOS.
* **Different screen adaptability:** Flutter applications can be designed to work on various screen sizes and aspect ratios. This makes it easy to create apps that look great on both phones and tablets.

### 3.4.4 Dart Programming Language

Dart is a client-oriented programming language that can be used to create quick apps on any platform. Its purpose is to provide the most productive programming language for cross-platform development, as well as a versatile runtime platform for app frameworks. Languages are defined by their technical envelopes, which are the decisions made during development to determine a language's capabilities and strengths. Dart is optimized for client development, prioritizing both development (sub-second stateful hot reload) and high-quality production experiences across a wide range of compilation targets (web, mobile, and desktop).

Flutter is likewise built on the Dart's base. Dart supports numerous basic developer jobs like formatting, analyzing, and testing code. Flutter apps offer the language and runtimes, but Dart also supports many fundamental developer tasks like formatting, analyzing, and testing code.

Dart is a type-safe language that uses static type checking to ensure that a variable's value always matches the static type of the variable. This is referred to as "sound typing" at times. Despite the

fact that types are required, type annotations are unnecessary due to type inference. The dart typing system is very versatile, allowing for the usage of a dynamic type in conjunction with runtime checks, which can be beneficial during experimentation or for code that requires a lot of dynamic behavior.

Dart is one of the languages that has a good null safety, which means that values can't be null unless you say so. Dart can protect you from null exceptions at runtime using good null safety and static code analysis [2].

Dart compiler technology let you run code in different ways: Native platform: for apps targeting mobile and desktop devices, Dart include both a Dart VM with just-in-time (JIT) compilation and ahead-of-time (AOT) compiler for producing machine code [2].

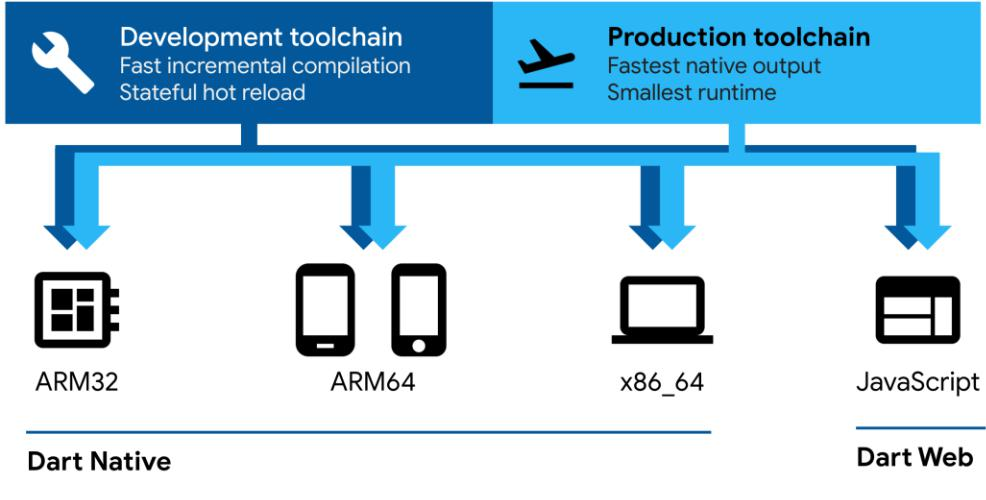


Figure 3.2 Dart compiler [2]

Figure 3.2 Shows the Dart compiler technology how it run codes from native platform to web platform.

## 3.5 External Interface Requirements:

### 3.5.1 Hardware

The mobile app will be operating on both Android and IOS Devices

### 3.5.2 Software

The software has been designed to be fully compatible across a variety of devices, including iPads, Android tablets, and iPhones. This is achieved through the use of Flutter, a mobile app development framework that allows for the creation of highly responsive user interfaces. Flutter's unique architecture allows for the development of a single code-base that can be used across multiple platforms, ensuring that the software will look and function consistently regardless of the device it is being used on. Additionally, Flutter's use of widgets allows for easy customization of the user interface, making it adaptable to different screen sizes and resolutions. This feature ensures that the software will be able to run smoothly on all devices, providing a seamless user experience.

### 3.5.3 Communication

The application utilizes Firebase Realtime, which allows for real-time data synchronization across all connected devices. This means that when you submit a query, you will immediately receive the corresponding output, without any delay. This feature ensures that the user experience is seamless and efficient. Additionally, Firebase Realtime allows for multiple users to access and update the same data simultaneously, making it a useful tool for collaborative applications.

### 3.5.4 Users

In order for users to access the more advanced features of the app, they will need to provide certain personal information such as their name, address, and phone number. This information will be recorded in a secure database and used to create their user account. Once they have an account, they will be able to log in and access additional functionalities such as creating, viewing, editing, and searching for recipes.

Additionally, users will have the option to create and manage a personal shopping list of ingredients for the recipes they want to make. This feature will allow them to easily keep track of the ingredients they need to purchase and ensure that they have everything they need on hand to make their desired recipes.

It is important to note that the personal information provided by users will be kept confidential and protected in accordance with relevant privacy laws and regulations. The app will have strict security measures in place to ensure that the information is protected from unauthorized access, use or disclosure.

## 3.6 Requirements Specification

The requirement specification describes the mobile application (YURECIPE) to be developed.

### 3.6.1 Functional Requirements

The recipe app system offers a range of features to enhance the user experience and provide users with access to a wide range of recipes and cooking-related information. These features include:

* **Login / Registration**: This feature allows users to create an account and log in to the app. This is necessary for accessing certain features and functions of the app, such as creating and saving recipes and using the meal plan feature.
* **Forgot Password**: This feature allows users to reset their password if they have forgotten it or are unable to log in to their account. It typically involves entering the email address associated with their account and following a set of instructions to reset the password.
* **Change Password**: This feature allows users to change their password if they wish to do so, either for security purposes or to simply update their login credentials.
* **Search for Recipe**: This feature allows users to search for recipes based on various criteria, such as ingredient, cuisine, dietary restrictions, and more. It is a useful tool for finding specific recipes or discovering new dishes to try.
* **Recipe Categories**: This feature allows users to browse recipes by category, such as appetizers, main dishes, desserts, and more. This can be a convenient way to find recipes based on the type of dish or the occasion for which it is intended.
* **Recipe Cuisine**: This feature allows users to browse recipes by cuisine, such as Italian, Mexican, Chinese, and more. This can be helpful for those looking to explore the flavors and techniques of different culinary traditions.
* **Complete Recipe Detail**: This feature provides users with detailed information about a specific recipe, including the recipe name, image, likes, rate, share, author, duration, ingredients, and preparation steps. This is a useful tool for understanding exactly what is involved in preparation particular dish.
* **Most Collected Recipes**: This feature showcases the most popular recipes on the app, as determined by the number of users who have saved them to their favorites list. It is a way for users to discover new recipes that are highly rated and recommended by other users.
* **Recent Recipes**: This feature displays the most recent recipes added to the app, providing users with a way to stay up-to-date on the latest food trends and discover new dishes to try. Meal Plan: This feature allows users to create and schedule a meal plan for the week, selecting specific recipes and scheduling them for specific days. It is a useful tool for those looking to plan their meals in advance and stay organized.
* **User Profile**: This feature allows users to view and edit their profile, including their name, email address, and other personal information. It is a way for users to manage their account and customize their experience on the app.
* **User Recipes**: This feature allows users to view and manage the recipes that they have saved or created on the app. It is a convenient way for users to access their favorite recipes and organize them for future reference.

Overall, these features provide users with a range of tools and resources to access and create recipes, stay up-to-date on the latest food trends, and plan and organize their meals. They offer a user-friendly and convenient platform for anyone looking to expand their culinary knowledge and skills.

The project scope of the YURECIPE platform is to develop an Android app that provides a collaborative and interactive process for accessing and creating food recipes. The main goal of the app is to empower Somali individuals living abroad by providing them with a convenient way to access traditional Somali recipes and discover new dishes to try. To achieve this goal, the app will be implemented using the DART programming language and will utilize Flutter for the interface design. DART is a modern programming language that is optimized for building mobile applications, and Flutter is a popular framework for building cross-platform apps with a native-like user experience. In addition to the app interface, the YURECIPE platform will also utilize Firebase as the backend technology. Firebase is a comprehensive mobile development platform that provides a range of tools and services for building, testing, and deploying mobile apps. It is particularly well-suited for building real-time, collaborative applications, such as a recipe app, as it allows for seamless synchronization of data across devices. Overall, the project scope of the YURECIPE platform is to develop a powerful Android app that helps Somali individuals living abroad access and create food recipes. The app will be implemented using the DART programming language and Flutter interface design, with Firebase serving as the backend technology. This will provide users with a collaborative and interactive platform for accessing and creating food recipes, empowering them to connect with their cultural heritage and expand their culinary

### 3.6.2 Non-Functional Requirements (needs check)

This section outlines the quality attributes that are crucial for the overall usability and effectiveness of the mobile application. These requirements are necessary to ensure that the application is able to meet the needs of its users.

**Scalability:** The app must be able to handle updates and new features without impacting performance.

**Security:** The app will use Firebase Firestore and Cloud Storage, which will provide robust security features.

**Speed:** The app must be fast enough to provide a good user experience.

**Compatibility:** The app must be compatible with different versions of mobile devices, including Android and iOS.

**Reliability**: The app must function 24/7 without any crashes or issues.

**System Requirements:**

Android 4.4

Internet connection

By implementing these non-functional requirements, the application will be highly reliable and feasible for users.

## 3.7 Feasibility Analysis of the Application

### A feasibility analysis is a critical examination of a proposed project, including an assessment of its technical and economic viability. In the context of the YURECIPE application, a feasibility analysis would consider the following factors:

### Technical Feasibility: The first factor to consider is the technical feasibility of the YURECIPE application. This includes an examination of the technical requirements necessary for the application to function, including the use of firebase as the backend and flutter for UI development. This analysis would consider the technical skills required for the development team, the hardware and software requirements for the app, and the compatibility with various mobile devices.

### Economic Feasibility: The second factor to consider is the economic feasibility of the YURECIPE application. This includes an assessment of the costs involved in developing the app, including hardware and software expenses, as well as ongoing maintenance costs. Additionally, the economic feasibility would consider the potential revenue streams, such as advertising and in-app purchases, and the overall return on investment.

### User Acceptance: The third factor to consider is user acceptance. A key aspect of the feasibility analysis is determining if users will be interested in the YURECIPE application and if it will meet their needs. This includes understanding the target audience and their cooking habits, as well as conducting market research to gather feedback and opinions from potential users.

### Operational Feasibility: The fourth factor to consider is operational feasibility. This includes an assessment of the day-to-day operations of the YURECIPE application and determining if it can be easily integrated into the daily routines of users. This analysis would consider the ease of use, the speed of the application, and the overall user experience.

### By conducting a comprehensive feasibility analysis, the developers of the YURECIPE application can ensure that the app meets the needs and expectations of users and that it is technically, economically, and operationally feasible to develop and maintain.

# Chapter 4 System Analysis

## 4.1 Overview

The chapter focuses on the architecture of mobile operating systems and the various types available. It delves into the concepts of flow charts, sequential diagrams, and class diagrams, providing an in-depth analysis of each one. The chapter also explores the use case diagrams, outlining their significance and use in software development. Additionally, it sheds light on the crucial aspects of database design and architecture, which play a significant role in ensuring the efficient functioning of an application. Lastly, the chapter highlights the importance of the application's user interface, which serves as the interface between the user and the application.

## 4.2 Mobile Operating Systems:

A mobile operating system, also known as mobile OS, is the software platform that powers smartphones, tablets, and other mobile devices. It provides a foundation for developing and running apps, managing memory and other resources, and communicating with the underlying hardware.

### 4.2.1 Android:

Android is a mobile operating system developed by Google. It is based on the Linux kernel and designed primarily for touchscreen mobile devices such as smartphones and tablets. Android is open-source software, meaning that it can be freely modified and distributed by device manufacturers, wireless carriers, and developers [30].

As of 2021, Android has a market share of around 71.8% of the global smartphone operating system market [29].

### 4.2.1 iOS:

iOS is a mobile operating system developed by Apple for its iPhone, iPad, and iPod Touch devices. It is a proprietary software, meaning that it can only be used on Apple devices. iOS is known for its sleek design, user-friendly interface, and strong security features [31].

As of 2021, iOS has a market share of around 27.6% of the global smartphone operating system market [29].

## 4.3 System Architecture

A system architecture is a theoretical framework that specifies a system's structure, behavior, and other aspects. It is a design description language that helps in the description of a system's architecture and behavior. Below in Figure 4.1 is the system architecture of the YURECIPE application.

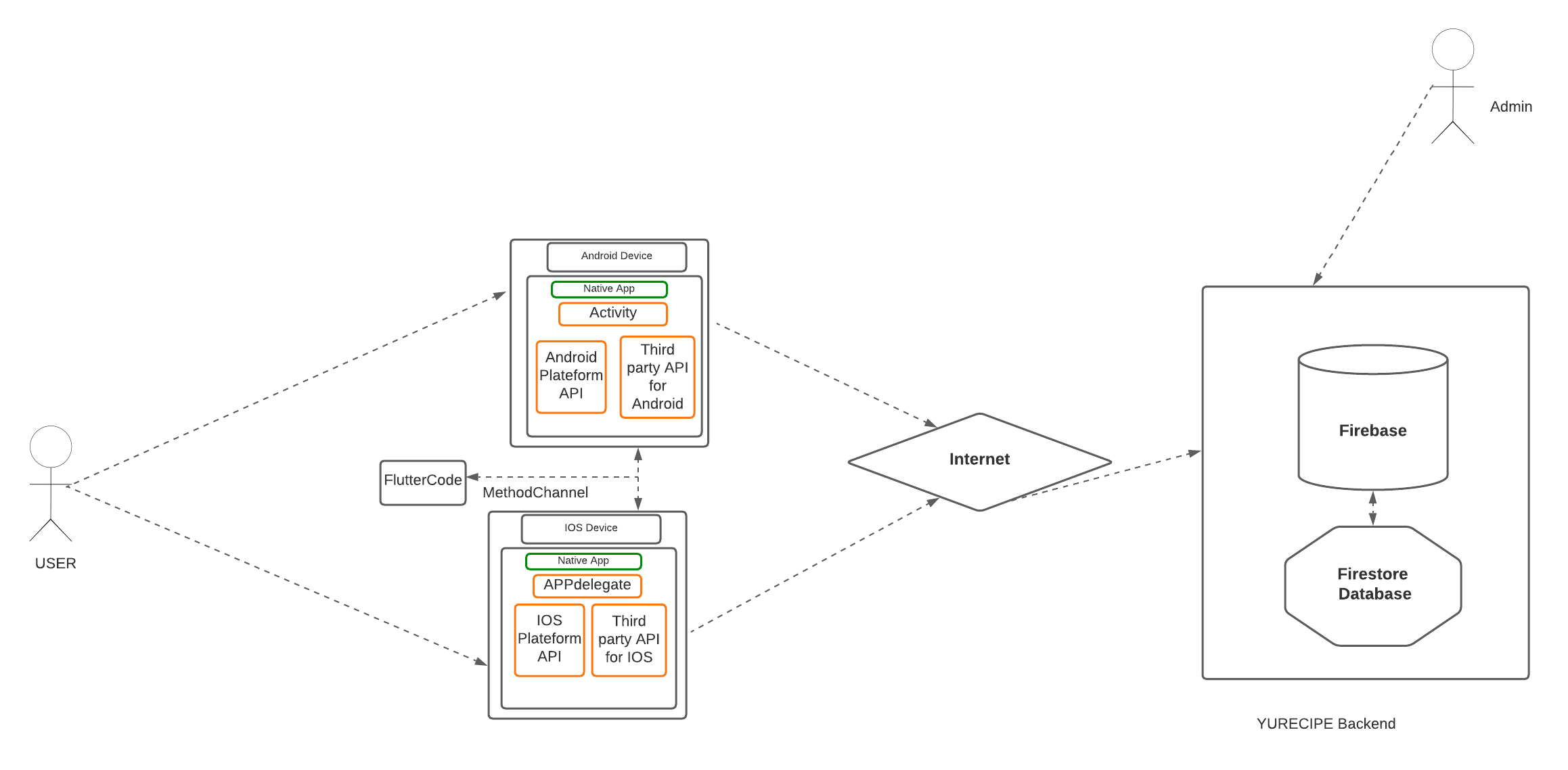


Figure 4.1 System Architecture of YURECIPE mobile application

The architecture of the YURECIPE Service mobile app, as depicted in Figure 4.1, utilizes the Flutter method channel to convert the code into native apps for both iOS and Android. An internet connection is necessary for users to access the backend services which include Firebase for authentication, crash analytics, performance monitoring and Cloud Firestore for the database and Cloud Storage for storing non-text files such as images.

## 4.4 Flowchart

This represents a workflow process in the application. The application algorithm is portrayed as a diagram with a staged process approach to developing the application. With arrows connecting in a logical way, the various steps are illustrated. The process of documenting, creating and managing the application process [23]. Because the Get Help Service mobile application has two types of users: experts and regular users. The flowchart below Figure 4.3 demonstrates each user's workflow in the application.

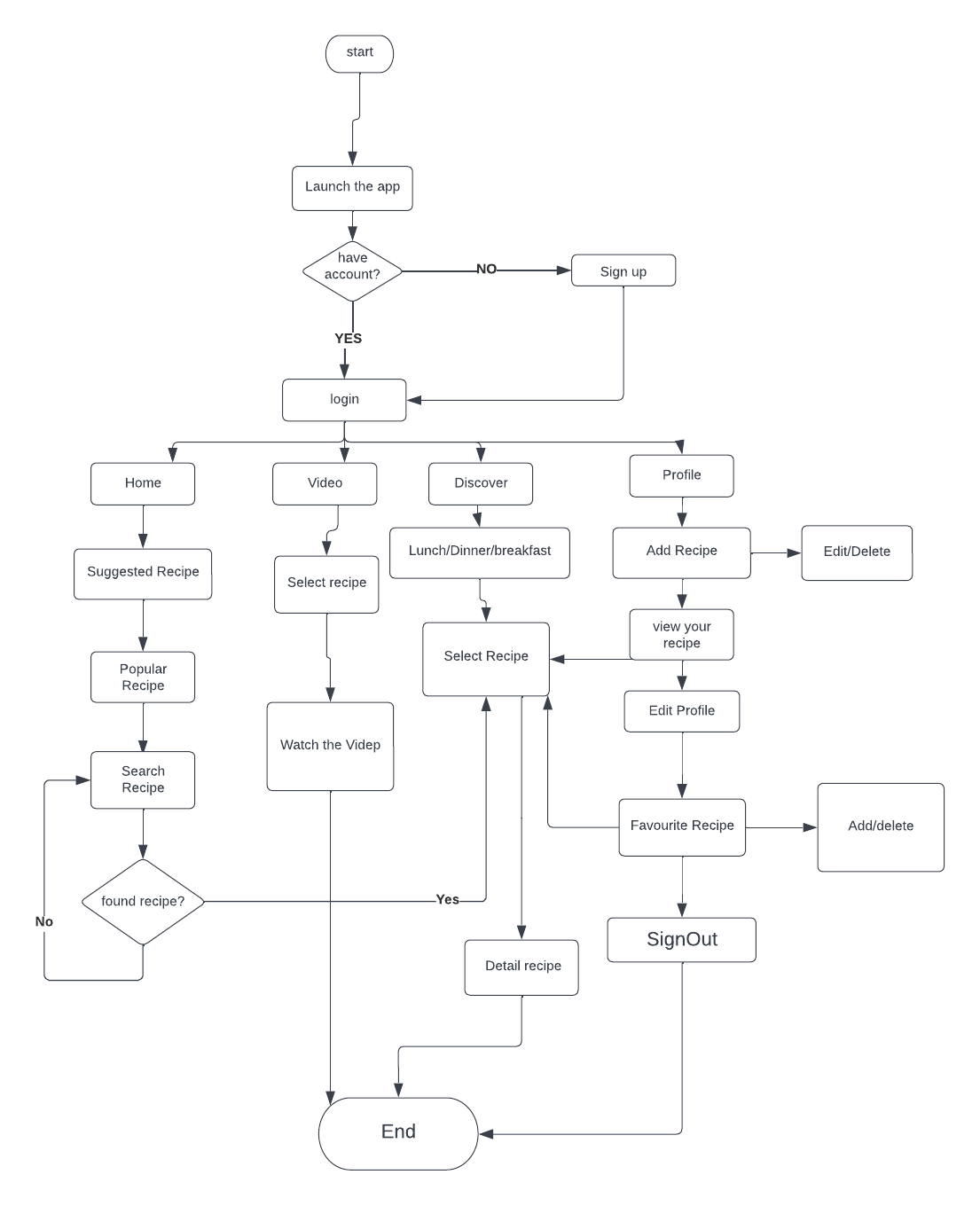


Figure 4.2 User flowchart diagram

## 4.5 Sequence Diagram

The sequence diagrams illustrate the interaction of events in the developed system of the YURECIPE mobile application. This is a commonly used tool among software developers and business professionals to understand the interactions between objects and the sequence of events in a new system being developed or an existing process[24]. The diagram (Figure 3.2) depicts the elements involved in the system to show the interactions and actions between users to carry out the functional scenarios. The sequence diagrams are often used in conjunction with use case implementations to provide a better understanding of how events and components communicate to achieve a process in the logical flow of the system under development.

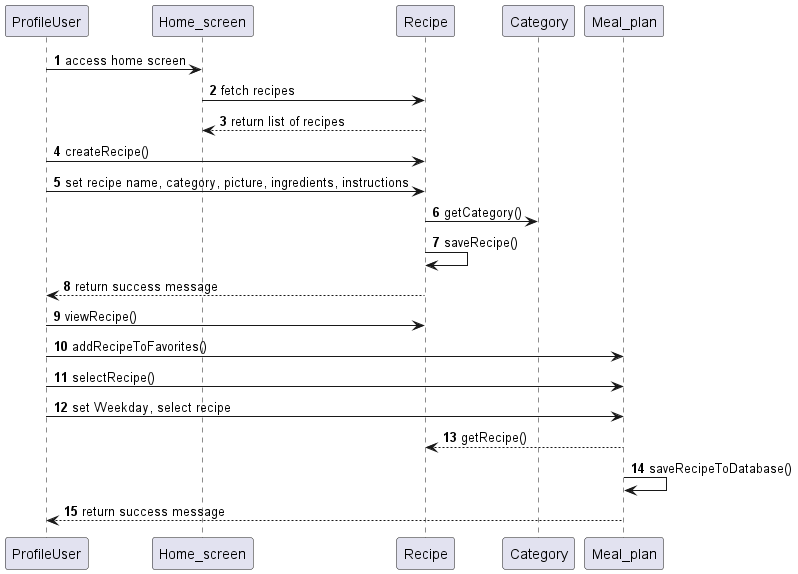


Figure 4.3 YURECIPE Sequence diagram

The diagrams presented above detail the various steps a user goes through when using the mobile application. The process starts with the user logging into the app, then moves on to browsing recipes, creating their own recipes, selecting their favorite recipes, and making meal plans. All of these actions are shown to demonstrate how the user interacts with the application to achieve their goals.

In addition to these steps, the diagrams also illustrate how the user's data is saved to the database. This includes information such as the recipes they have created, their favorite recipes, and any meal plans they have made. The diagrams provide a visual representation of how all of these elements work together to create a seamless and intuitive user experience.

## 4.6 Class Diagram

The diagram presented above provides a snapshot of the application, depicting various aspects and elements of the application in a visual manner. This snapshot serves as a means of describing, visualizing, and documenting the various parts of the application, which is an important aspect of the development process.

The diagram is also used to create the executable code for the application, which involves turning the design into a functional program that can be used by users. This process of creating the executable code helps to ensure that the features and activities of each class are accurately represented and functioning as intended.

The diagram showcases the different characteristics of the application, such as its features and activities, through the representation of each class. This provides a comprehensive view of how the various elements of the application work together to create a functional and user-friendly experience.

In summary, the diagram plays a crucial role in the development process by providing a visual representation of the application and helping to ensure that the features and activities are accurately represented and functioning as intended.

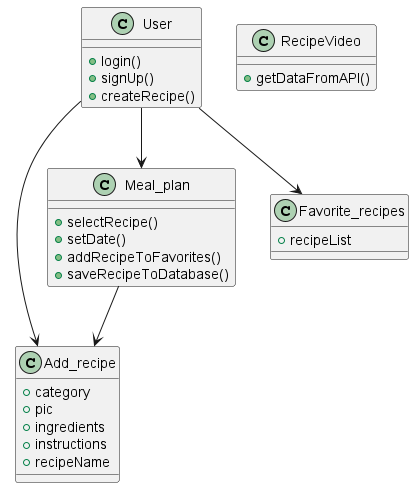


Figure 4.4 YURECIPE Class diagram

The above class diagram plays a crucial role in understanding the architecture and design of the mobile application. It shows how the different classes are structured, what responsibilities each class has, and how they work together to achieve the desired outcome. This information is valuable for both developers and users as it provides a clear understanding of the underlying logic and processes of the application.

## 4.7 Use Case Diagram

The user case diagram displays the interactions between the users and the system and outlines the different actions and scenarios that a user can perform within the system. It provides a visual representation of how the system functions from the user's perspective, including the different tasks and processes that a user can initiate within the system.

The user case diagram highlights the relationships between the user and the system, demonstrating how the two work together to accomplish specific tasks and carry out various scenarios. It provides a clear understanding of the user's role in the system, including what actions they can perform, and how the system responds to their actions.

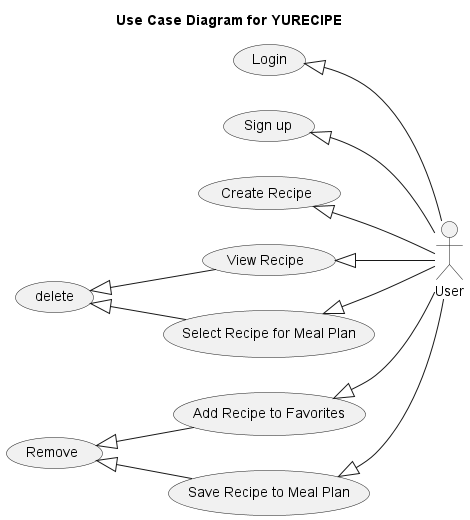


Figure 4.5 YURECIPE Use Case diagram

The above Figure 4.5 shows the user case diagram. The available attributes and functionalities for a user in the application are depicted in boxes for each case scenario.

## 4.8 Database Design – Cloud Firestore

Cloud Firestore is a flexible and scalable database solution offered by Firebase and Google Cloud that can be used for web, mobile, and server development. It provides real-time synchronization between client applications and enables offline support for mobile and web applications, ensuring that your app remains responsive regardless of internet connectivity or network latency[25].

With Cloud Firestore, you can store your data in documents that consist of field-to-value mappings, using a NoSQL data model. These documents are organized into collections, and you can use groups to further categorize and structure your data. The data model of Cloud Firestore supports a variety of data types, including simple strings and numbers, as well as complex nested structures[26].

Queries in Cloud Firestore are expressive, efficient, and adaptable, allowing you to retrieve data at the document level, sort and filter information, and add real-time listeners to keep your app's data up-to-date. For security, Cloud Firestore uses security rules that integrate with Firebase Authentication and Cloud Firestore security to provide secure access to your data, guarding against vulnerabilities.

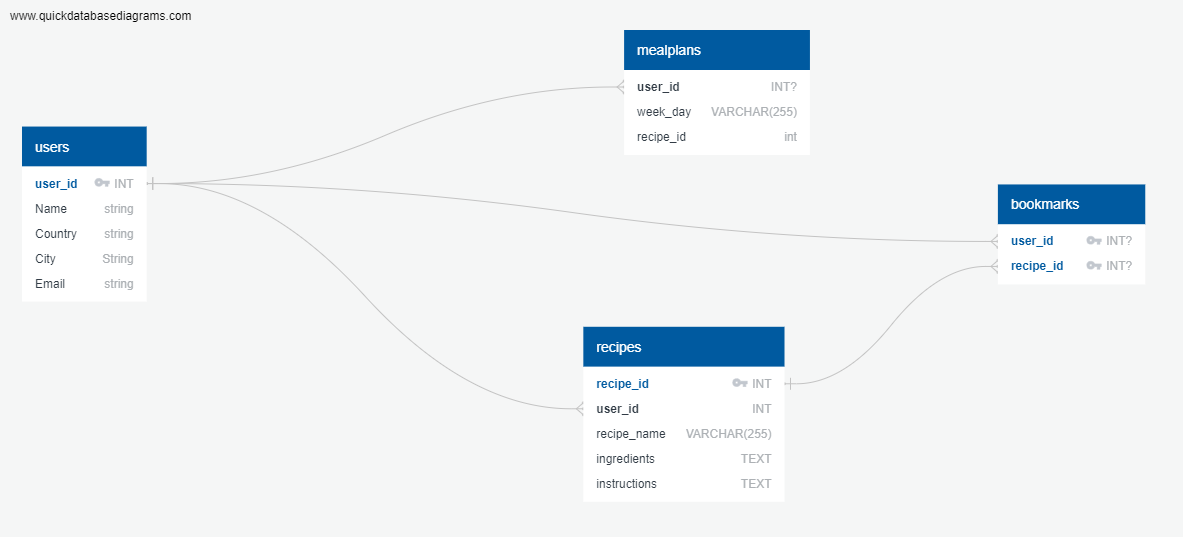


Figure 4.6 Entity relation diagram

## 4.9 User Interface Design for Somali Recipe App for Diaspora:

### Introduction:

The goal of this project is to create a mobile application that serves as a digital cookbook for Somali people living abroad. The main challenge faced by the Somali diaspora is the lack of access to authentic Somali ingredients and recipes, making it difficult to recreate the traditional dishes they love. The user interface design of this recipe app must cater to the needs of the Somali diaspora and provide a seamless experience while using the app[27].

### Design Considerations:

* Cultural Relevance: The design should reflect the rich cultural heritage of Somalia and provide a sense of familiarity to the users. This can be achieved through the use of traditional colors, patterns, and images [27].
* Ease of Use: The app should be intuitive and easy to use, allowing users to quickly find what they are looking for. This can be achieved through clear navigation, easy-to-understand icons, and an uncluttered layout [28].
* Personalization: The app should provide users with the option to personalize their experience by creating their own recipe collection, marking their favorite recipes, and setting reminders for meals.
* Accessibility: The app should be accessible to users with different abilities and take into consideration their specific needs, such as larger text size.

### Design Elements:

* Homepage: The homepage should feature a visual gallery of popular recipes and allow users to quickly navigate to different categories.
* Recipe Categories: The app should categorize recipes into different sections such as "Breakfast", "Lunch", "Dinner", etc. Users should be able to quickly switch between categories.
* Recipe Pages: Each recipe page should include a photo, a list of ingredients, instructions, and serving suggestions.
* Personalized Collection: Users should be able to create their own personalized collection of recipes and have quick access to it from the homepage.
* Meal Planner: The app should include a meal planner where users can schedule their meals for the week and create grocery lists based on the ingredients they need.

### Conclusion:

The user interface design of the Somali recipe app for the diaspora is crucial in providing a seamless and enjoyable experience for the users. The design must consider the cultural heritage, accessibility, and personalization needs of the users while also providing easy navigation and clear presentation of information [27].

# Chapter 5 System Implementation

## 5.1 Overview

The mobile app being developed is a Recipe Management system that enables users to create, view, and save recipes. Its aim is to provide a platform for people to create and share Somali recipes globally. This chapter details the features and functionality of the app, and provides screenshots for illustration. Additionally, the basics of mobile app testing and the techniques used to test this application are also discussed.

## 5.2 The Developed System

The newly developed mobile application has been installed on Vivo Android phone. The below figures are screenshots taken from the Vivo device, showcasing the app's splash screen and welcome screen. When the app is launched, it begins with the splash screen, which gives a quick introduction to the app's purpose and design. The splash screen is displayed for three seconds before transitioning to the welcome screen, as shown in Figure 5.1 and 5.2.



Figure 5.1Splash screen Figure 5.2 Introduction screen

As illustrated in Figure 5.1, the splash screen briefly appears for 2.5 seconds before moving on to the welcome screen, as depicted in Figure 5.2, which is displayed for new users.

### 5.3.1 Login Screen

Following the welcome screen, as seen in Figure 5.2, the user is presented with two buttons, one for logging in and another for signing up. New users can opt to sign up and be directed to the registration page, while returning users can click on the login button to access the login page. To log in, a user must enter their email and password, with placeholder text guiding them on where to input their information. Upon successful authentication, the user can then click on the Sign In button to access the home screen, as depicted in Figure 5.3, which shows the user login screen.

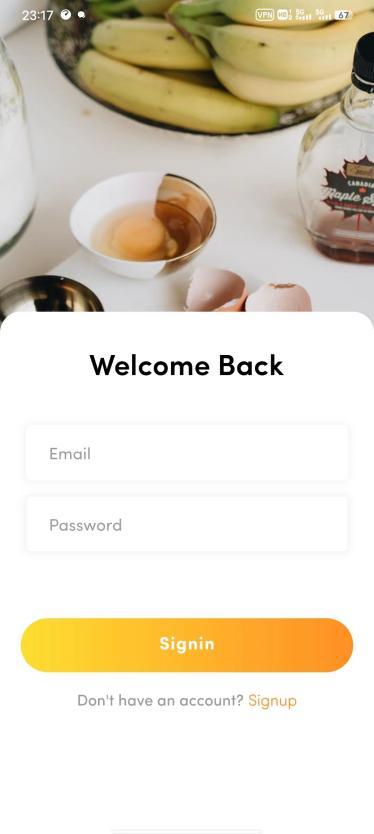


Figure 5.3 Login Screen

As shown figure 5.3 displays the Login page, which is intended for users who have already created an account.

## 5.3.1 Sign up Screen

To access the app, new users must first register for an account. The registration process requires the user's username, email address, and password. For password security, the password must include at least one number, upper and lowercase letters, a special character (! @ # $% & \*), and have a minimum length of six characters. Once the user has entered their information and pressed the 'Sign Up' button, the registration will either be successful and log the user in automatically, or return an error message if unsuccessful

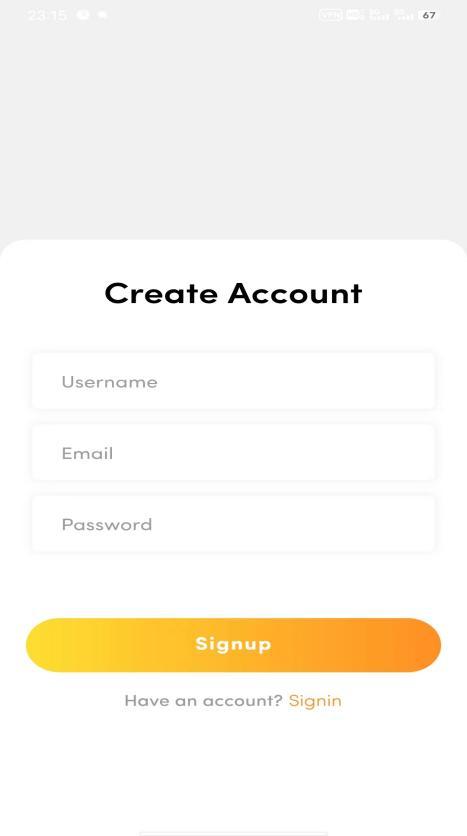


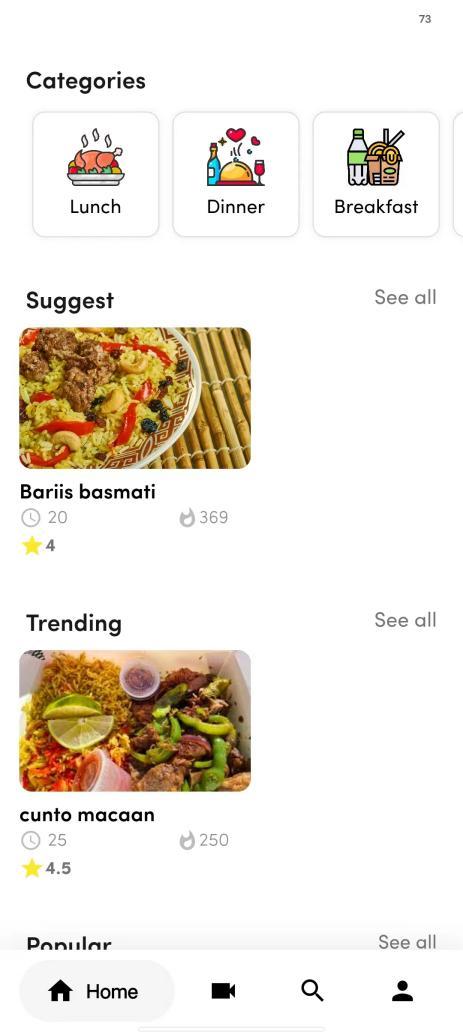
Figure 5.4 Sign In Screen

The above figure 5.4 displays the user sign-up screen, where new users can enter their required details to register for an account. At the bottom of the sign-up screen, there is a message reading "Have an account?" with a link to the login page for users who already have an existing account

### 5.3.1 Home Screen

The home screen is designed to provide users with a variety of recipe options for their various meal needs. It features several categories including breakfast, dinner, and lunch. These categories serve as a starting point for users to explore and find recipes that suit their taste preferences. In addition to these categories, the home screen also showcases suggested recipes from the administration. These recipes are handpicked by the admin and are meant to offer inspiration and new ideas for meal preparation.

the home screen also displays trends, suggested, and popular recipes, all of which are placed by the administration. The purpose of these sections is to provide users with a curated selection of recipes and meal ideas, making it easier for them to find what they are looking for

Figure 5.5 Home Screen

### 5.3.1 Search Screen

The system provides recipe searching capabilities. When the user clicks on the search bar, they are presented with options to search by breakfast, dinner, lunch categories as shown in figure 5.7 or by entering specific search terms. The search function is robust and returns results even if only partial recipe names are entered as shown in figure 5.8. In the case where the searched recipe cannot be found, an error screen is displayed as shown in Figure 5.9

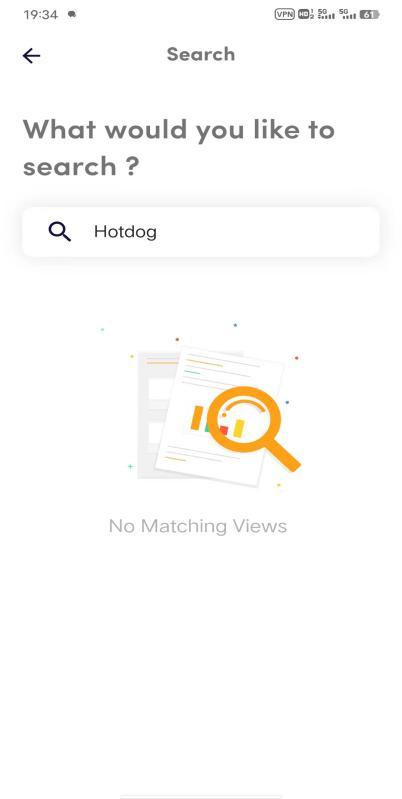
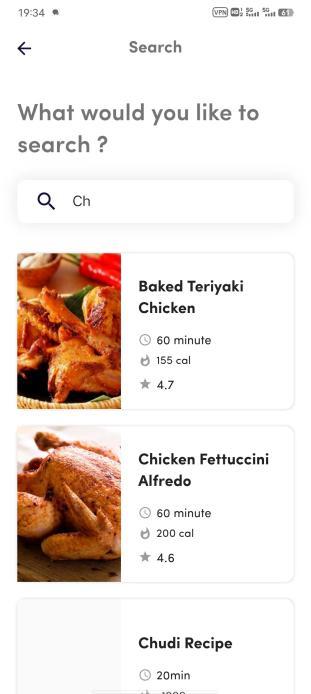
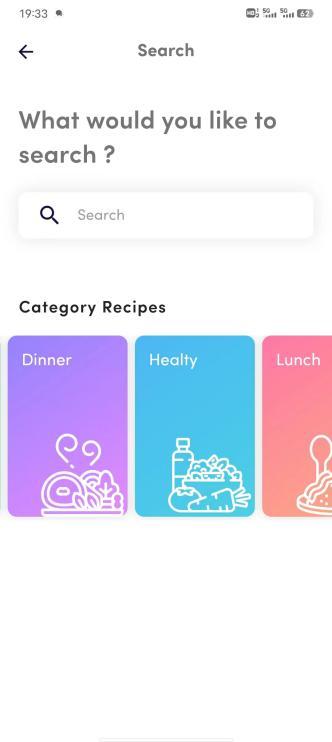
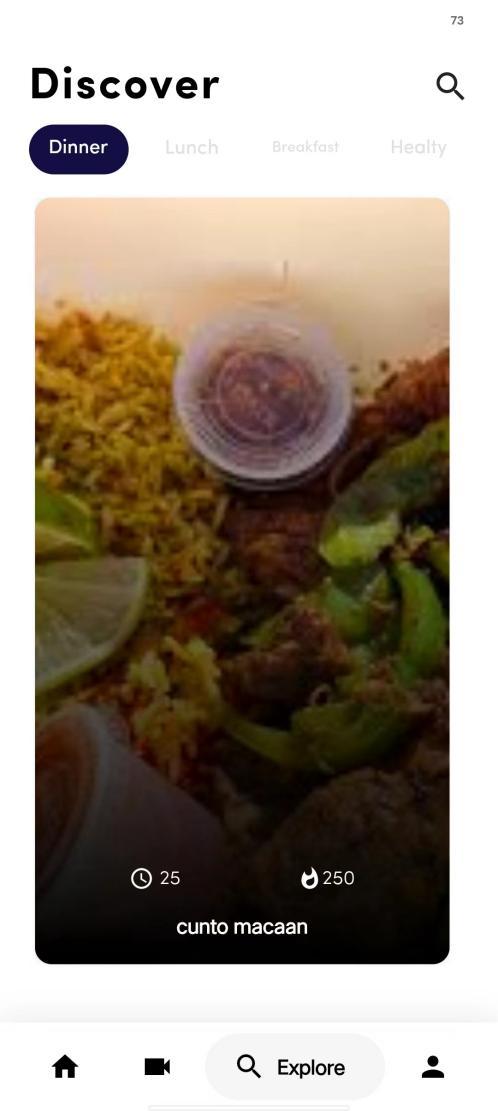


Figure 5.6 Search screen Figure 5.7 Search screen Figure 5.8 Search screen

### 5.3.1 Discover Screen

The "Discover" feature displays a list of recipes organized by category, providing an opportunity for users to discover new recipes, as shown in Figure 5.10.

Figure 5.9 Discover screen

### 5.3.1 Meal Plan Screen

The "Meal Plan" feature allows users to browse recipes and add them to their meal plan for each day of the week, as shown in Figure 5.11.

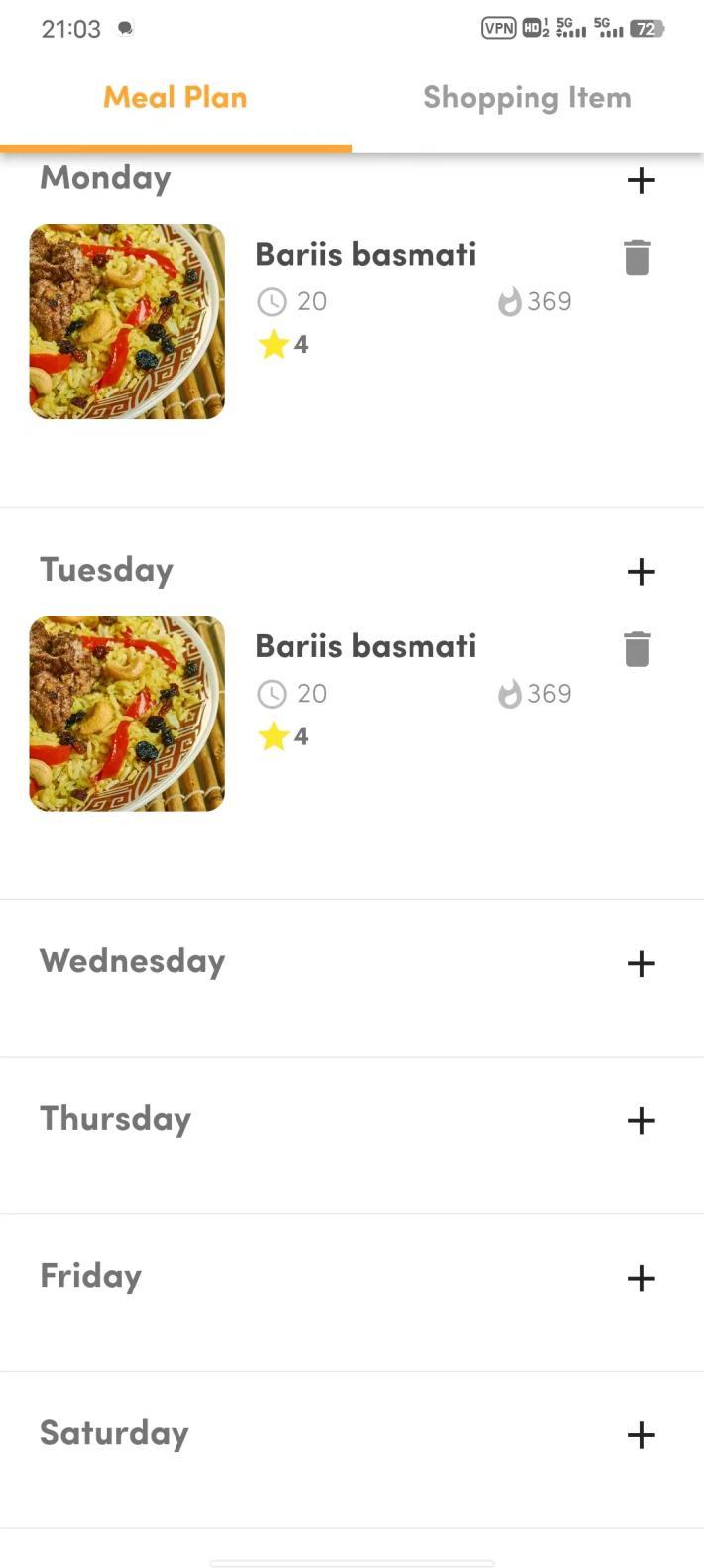


Figure 5.10 Meal Plan screen

## Profile User

The "Profile" section displays the user's information and provides easy navigation through icons, including options to view the meal plan, add recipes, view their own recipes, edit the profile, and log out

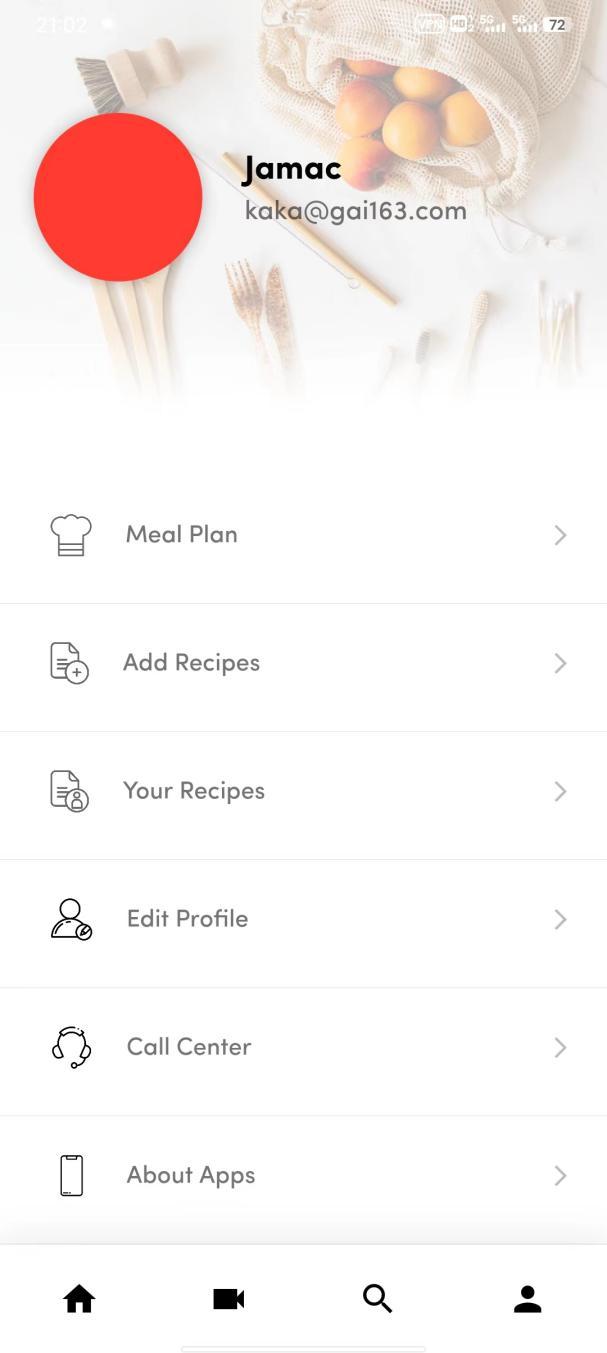


Figure 5.11 Profile Screen

## View Recipe

In this section, the user can view their own recipes and also delete them

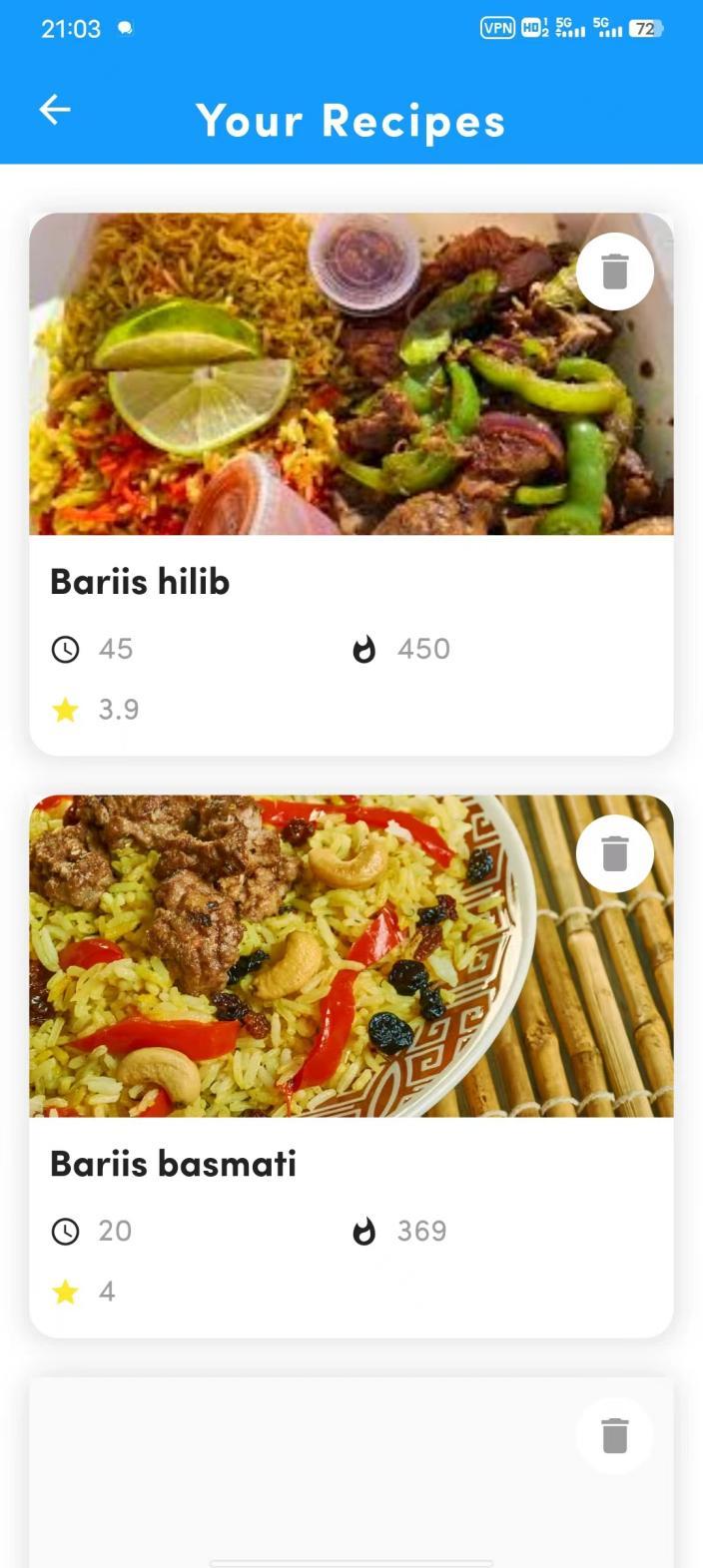


Figure 5.12 View user’s Recipe

## Create Recipe Screen

The "Create Recipe" feature allows users to create new recipes by entering recipe name, ingredients, directions, calories, cooking time, and rate the recipe. Ingredients are separated by commas, and directions are separated by commas as well. The user can rate the recipe on a scale of 1 to 5, based on their personal experience with the recipe then clicks the submit button to be successfully added.

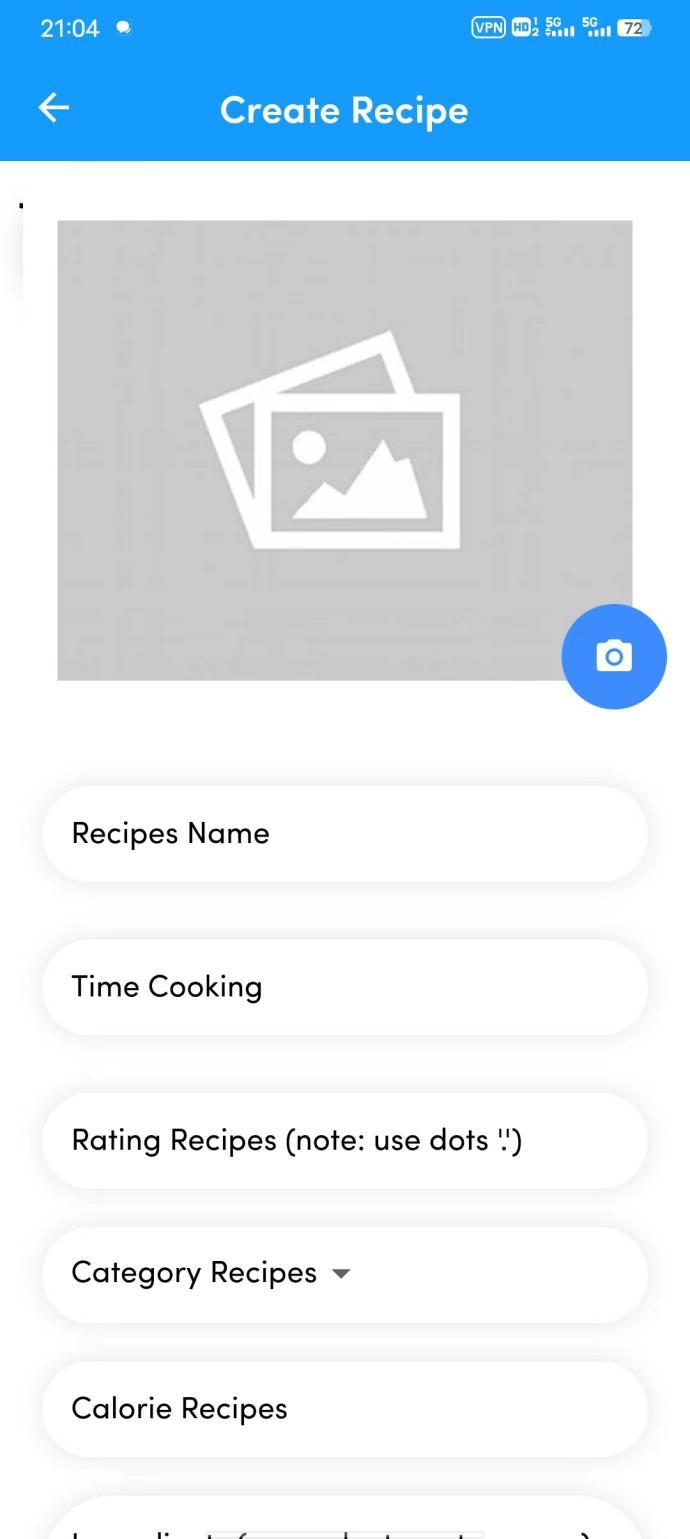


Figure 5.13 Create Recipe Screen

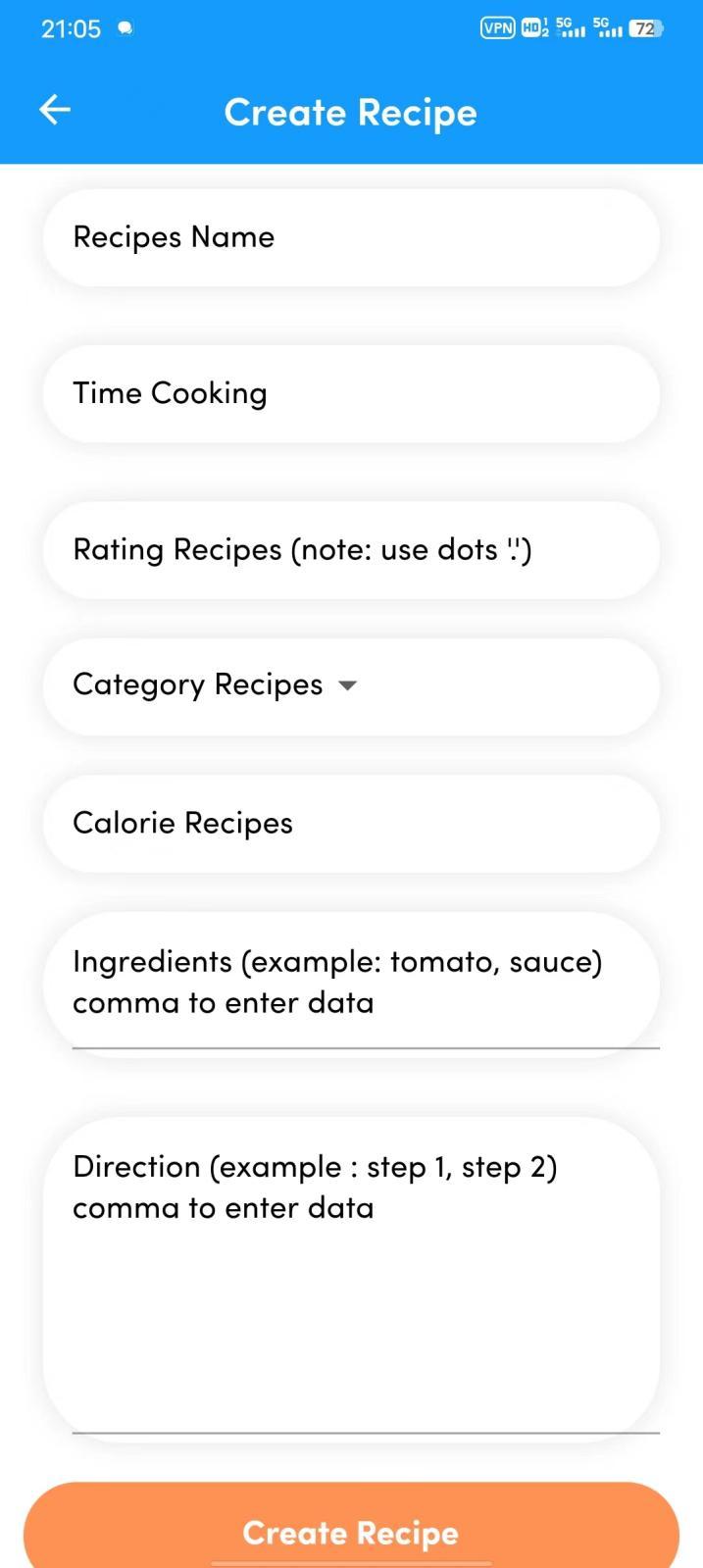


Figure 5.14 Create Recipe Screen

## Edit Profile Screen

The "Edit Profile" section displays the user's information and allows the user to change their name, country, and city.

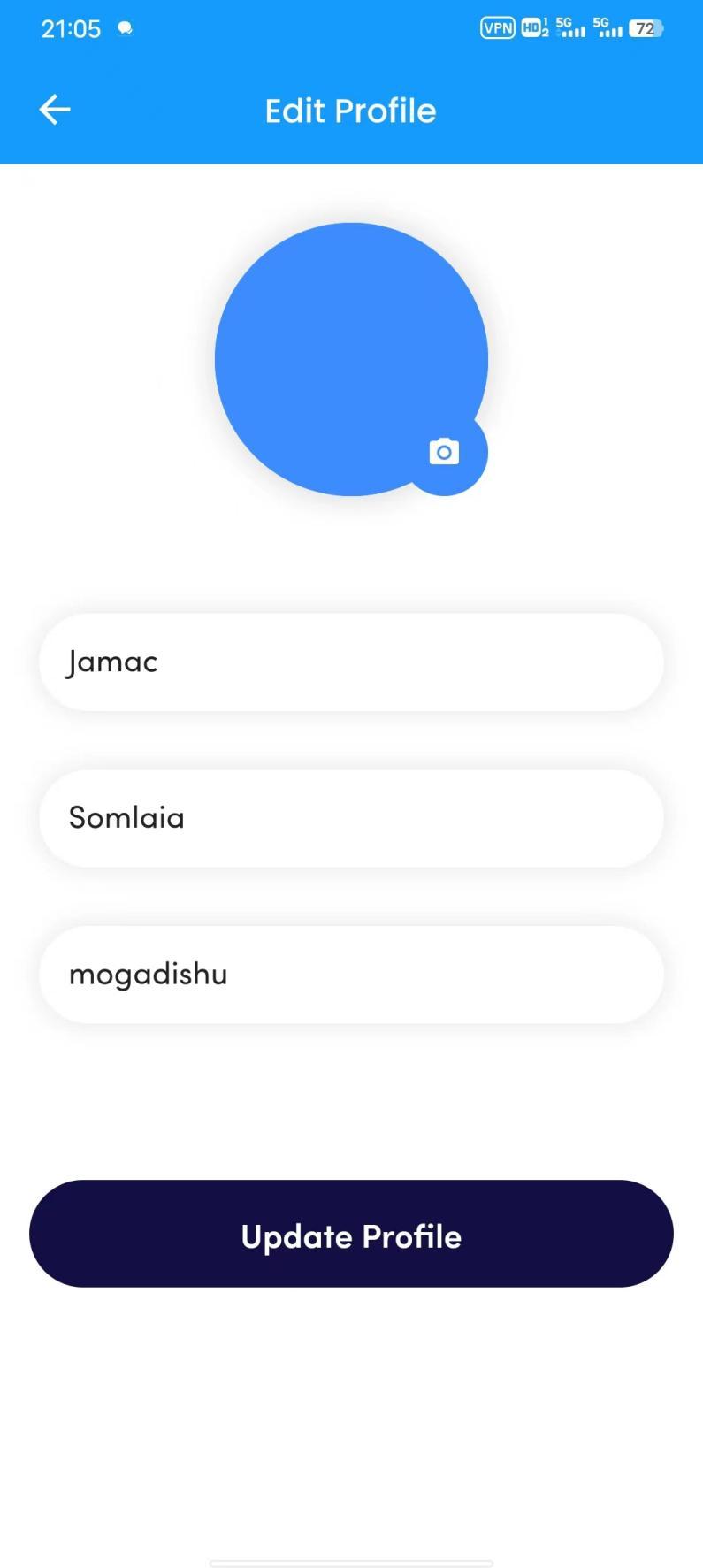


Figure 5.15 Edit Profile

**View Recipe**

The "View Recipe" feature is an essential component of the recipe platform. It allows users to view a recipe in detail, including its ingredients, cooking directions, cooking time and calories. This feature provides users with all the information they need to prepare a delicious dish

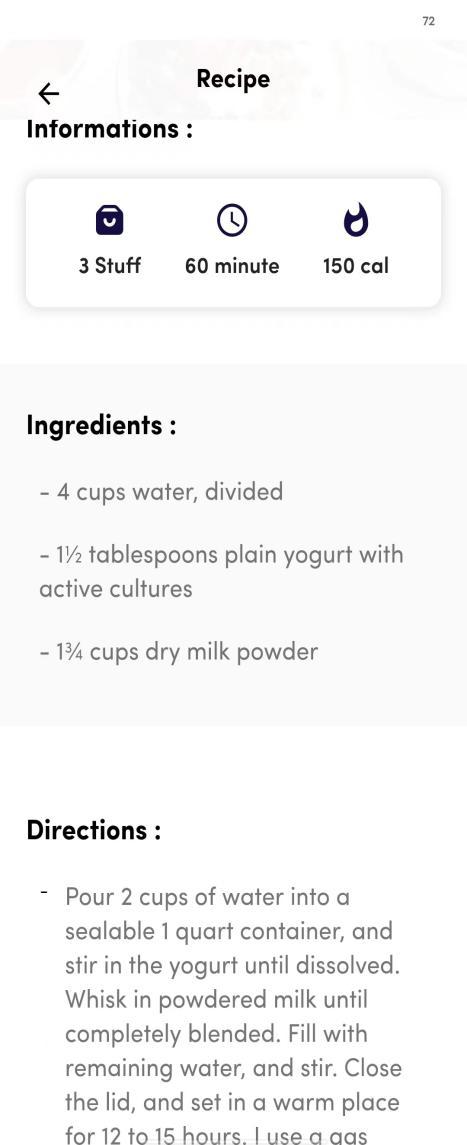


Figure 5.18 View Recipe

**Chapter 6 Conclusion, Challenges and Future Works (Needs Work)**

# 6.1 Chapter 2 Conclusion

The thesis aimed to develop a cross-platform mobile application, a prescription mobile application called "YURECIPE", a mobile application that helps people learn to cook Somali traditional food, The advantage of this application is that users can browse, create, view and save their favorite recipes. This application combines the features of cloud storage to avoid the storage space of the user her device and provide the user with better search results in the shortest way. This application is very useful for Somalis living abroad who want to learn how to cook delicious Somali food and share recipes with other Somali diasporas around the world. The application used Flutter, an easy-to-use framework for designing user interfaces for cross-platform mobile applications. The application backend is Firebase and Firestore cloud storage to improve app quality and performance. I used a Flutter package that manages the state of "Bloc" for the state management of the application. Decouple the application UI from the data model, track data status, and rebuild and update pages as needed. Additionally, testing your application in different ways to meet the minimum user requirements will result in a stable user experience and compatibility issues.

## 6.2 Challenges

The development of YURECIPE App was faced with several challenges, including:

1. Inadequate testing tools: One of the major challenges faced was the lack of tools to test the application on iOS platforms. Testing on iOS requires a MacBook with an iPhone, but the only available tools were a Windows operating system and an Android phone.
2. Time constraints: The tight deadline also made it difficult to finish certain features such as the rating system, Video tutorial and the shopping list.
3. Cross-platform limitations: Flutter, which is a cross-platform framework, was used in the development process, but this also presented its own set of challenges in terms of compatibility and functionality.

## 6.3 Future works

The current version of the application is still in its initial stage and has scope for further improvement. One aspect that cannot be overlooked is thorough testing on both Android and iOS platforms. Another important feature that needs to be implemented is the rating system, which should be based on user ratings instead of being self-rated by the person who posted the recipe. The shopping list should also be improved to display all the ingredients of the meal plan in an organized and convenient manner for the users.

# Refererces

[1] Washington Post https://www.washingtonpost.com/

1. What is Dart programming language, accessed from https://dart.dev/overview%20 on November 15, 2021.
2. References Beck, K. (2001). Agile manifesto. Retrieved from https://agilemanifesto.org/ Scrum Guide (2017). Scrum.org. Retrieved from https://www.scrum.org/scrum-guide Kanban Guide for Scrum Teams (2018). Scrum.org. Retrieved from <https://www.scrum.org/resources/kanban-guide-scrum-teams>
3. <https://www.techtarget.com/searchapparchitecture/definition/mobile-application-development>
4. Cross-Platform Mobile Development in Action" by Kerri Shotts.
5. "Somali Kitchen" YouTube channel, youtube.com/channel/UCt1cWV7vxdRbR6p7R0H0JWg
6. "Somali Food Recipes" Facebook page, facebook.com/SomaliFoodRecipes
7. "Somali Cuisine: Recipes and Traditions" cookbook by Xawaash, amazon.com/Somali-Cuisine-Recipes-Traditions-Xawaash/dp/0982654502
8. "The Somali Kitchen" cookbook by Hawa Hassan, amazon.com/Somali-Kitchen-Recipes-Staple-Ingredients/dp/0578485893
9. "The Limitations of Cooking Videos on YouTube" by Jacob Warren, thespruceeats.com/limitations-of-cooking-videos-on-youtube-480383
10. "The Pros and Cons of Using Facebook for Recipe Research" by Olivia Harvey, thespruceeats.com/pros-and-cons-of-using-facebook-for-recipe-research-481396
11. The Pros and Cons of Using Cookbooks for Recipe Research" by Olivia Harvey, thespruceeats.com/pros-and-cons-of-using-cookbooks-for-recipe-research-481394
12. "Advantages and Disadvantages of Cookbooks and Recipe Books" by Madeline Hurley, thespruceeats.com/advantages-and-disadvantages-of-cookbooks-and-recipe-books-480344
13. "The Complete Cooking for Two Cookbook" by America's Test Kitchen
14. "The Joy of Cooking" by Irma S. Rombauer and Marion Rombauer Becker
15. Ionic Framework: https://ionicframework.com/
16. PhoneGap: https://phonegap.com/
17. React Native: https://facebook.github.io/react-native/
18. <https://mdevelopers.com/blog/what-is-a-native-mobile-app-development->
19. <https://raygun.com/blog/native-app-development/>
20. A. Hammershøj, A. Sapuppo and R. Tadayoni, "Challenges for mobile application development," 2010 14th International Conference on Intelligence in Next Generation Networks, Berlin, Germany, 2010, pp. 1-8, doi: 10.1109/ICIN.2010.5640893.
21. A. Hammershøj, A. Sapuppo and R. Tadayoni, "Challenges for mobile application development," 2010 14th International Conference on Intelligence in Next Generation Networks, Berlin, Germany, 2010, pp. 1-8, doi: 10.1109/ICIN.2010.5640893.
22. What is Flowchart diagram? Accessed from https://en.wikipedia.org/wiki/Flowchart%20 on February 23, 2022.
23. Sequence Diagrams: An Introduction: <https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-sequence-diagram/>
24. "Introduction to Cloud Firestore: A NoSQL Database for Your Mobile Apps" by Mladen Janković, published on Telerik.com
25. "Firebase Cloud Firestore: A NoSQL Database for Your App" by Brandon Morelli, published on freeCodeCamp.org.
26. Don't Make Me Think: A Common Sense Approach to Web Usability by Steve Krug.
27. Designing for Interaction: Creating Smart Applications and Clever Devices by Dan Saffer.
28. Statista. (2022). Smartphone operating system market share worldwide from 2009 to 2022. Retrieved from https://www.statista.com/statistics/272698/global-market-share-held-by-mobile-operating-systems-since-2009/
29. Android. (2023). Android Developers. Retrieved from https://developer.android.com/
30. Apple Inc. (2023). iOS - Apple Developer. Retrieved from <https://developer.apple.com/ios/>
31. <https://agiletech.vn/traditional-sdlc-vs-agile-sdlc/> accessed 20230 feb
32. Android's Developer Guide (https://developer.android.com/guide)
33. Android Programming: The Big Nerd Ranch Guide" by Bill Phillips and Brian Hardy