**COMP 3059 – Capstone Project I**

**Software Requirements Analysis and Design Assignment**

This assignment is an overview to gather the software needs with requirements analysis and help to proceed with the design.

The requirements analysis helps to break down functional and non-functional requirements to a basic design view to provide a clear system development process framework. It involves various entities, including business, stakeholders and technology requirements.

The design is the activity following requirements specification and before programming. Software design usually involves problem solving and planning a software solution.

To work on this assignment you could use the references and a sample template given below. The sample template can be customised to suit the nature of your project.

Reference Readings/Example:

<http://www.uacg.bg/filebank/acadstaff/userfiles/publ_bg_397_SDP_activities_and_steps.pdf>

[www.cse.msu.edu/~chengb/RE-491/Papers/SRSExample-webapp.doc](http://www.cse.msu.edu/~chengb/RE-491/Papers/SRSExample-webapp.doc)

<https://nces.ed.gov/pubs2005/tech_suite/part_2.asp>

Reference template:

[www.tricity.wsu.edu/~mckinnon/cpts322/cpts322-srs-v1.doc](http://www.tricity.wsu.edu/~mckinnon/cpts322/cpts322-srs-v1.doc)

# 1.0 Introduction

## Purpose

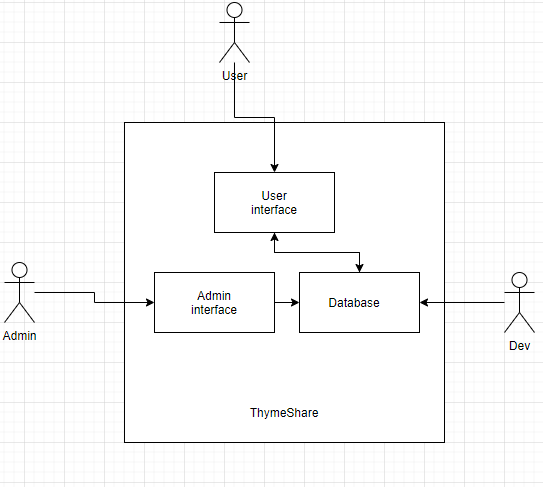
The purpose of this document is to present a detailed description of our ThymeShare project. It will explain the features, the software to be used for the project, the use cases for the application, and will have various data modeling and analysis diagrams. This document is intended for the development team.

# System Overview

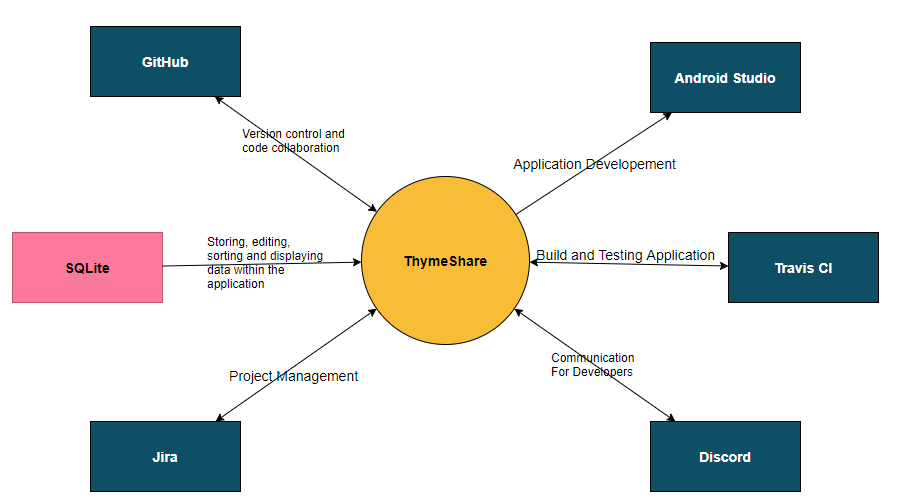
The System Overview section introduces the system context and design.

## Project Perspective

a new self-contained system.



## System Context



## General Constraints

* With handling personal information of users, we need to ensure that the security protocols of our program are up to date, and the data we are storing is encrypted. There is a possibility of malicious attacks to our servers and we must have security measures in place.
* Device limitations/differences (Screen size, sensors, camera, and interactions)
* Network issues
* Application will be limited to English

## Assumptions and Dependencies

**Dependencies:**

* API’s:
  + Google Mobile Vision API
  + Google Maps/GPS
* Hardware:
  + Android mobile device.
* Database
* User accounts

**Assumptions:**

* Users that enjoy making new recipes and sharing their own will become regular users.
* The application will be completed in the proposed timeframe.
* Users privacy will be protected.
* Removal of spam and posts against terms and service.
* Simple UI and efficient browsing.
* Suggestions based of recipes and ingredients user provides.

## 3.0 Functional Requirements

### 3.1 <Functional Requirement or Feature #1>

|  |  |
| --- | --- |
| **Feature #1** | |
| **Title** | Users sign up |
| **Description** | A Form that will add user information and check if any information is already used. Upon submission the user will be added to the applications users. |
| **Inputs** | Username, first name, last name, email, password, confirm password, country, accept terms of service. |
| **Processing** | |
| 1. User will input their information. 2. Upon submission the user will see if their username and email has already been used, the terms of service also must be checked. 3. If the username and email have not already been used user will be added to the application. | |
| **Outputs** | |
| 1. User should get a thank you message and proceed to the home page. | |

|  |  |
| --- | --- |
| **Feature #2** | |
| **Title** | Creating Recipe |
| **Description** | The user may create a recipe on the application. (Users will be able to edit their recipes) |
| **Inputs** | Category selection, recipe name, description, recipe steps, images (if desired) |
| **Processing** | |
| 1. Upon submission the page will be formatted and aske the user for confirmation 2. Upon confirmation the page will be posted to the application for other users to rate, review, and share. | |
| **Outputs** | |
| 1. The recipe will be added to the application and shared with other users for them to try, rate, review, and share. | |

|  |  |
| --- | --- |
| **Feature #3** | |
| **Title** | Sharing recipes |
| **Description** | The users will be allowed to share recipes via link, text message and other social media platforms. |
| **Inputs** | By clicking the appropriate button for the platform, they want to share on. |
| **Processing** | |
| 1. Application will generate the link/ open the platforms supported for sharing | |
| **Outputs** | |
| 1. The recipe will be shared. | |

|  |  |
| --- | --- |
| **Feature #3** | |
| **Title** | Adding ingredients to user. |
| **Description** | The user may upload the ingredients they have by scanning text (image to text) or by adding items from a selection process. |
| **Inputs** | Ingredient name. |
| **Processing** | |
| 1. User will add the ingredient and the application will add the item to list of user’s ingredients. | |
| **Outputs** | |
| 1. Open the user’s ingredient page. | |

|  |  |
| --- | --- |
| **Feature #4** | |
| **Title** | Create shopping list |
| **Description** | The user will be able to add items to a shopping list via user input, from recipes and from their ingredient page. |
| **Inputs** | Item name and amount. Store optional. |
| **Processing** | |
| 1. Create a list. | |
| **Outputs** | |
| 1. User will see the full list. 2. User will be able to check items off, and select sorting options (a-z, z-a, amount, stores) | |

|  |  |
| --- | --- |
| **Feature #5** | |
| **Title** | Liking recipes on the app |
| **Description** | The user can like/favourite recipes to save for later. |
| **Inputs** | The press of the like/favourite button will add it to your saved recipes |
| **Processing** | |
| 1. Upon the click of the save button the recipe will be added to the users saved list | |
| **Outputs** | |
| 1. “Recipe Saved” 2. User can remove saved recipes on the saved recipe page. | |

|  |  |
| --- | --- |
| **Feature #6** | |
| **Title** | User Setting |
| **Description** | When clicking on the settings page the user will be able to edit their information and profile. |
| **Inputs** | Edit buttons on private information that can be changed, delete account button, |
| **Processing** | |
| 1. Information changed will be authorized and check if any of the information changed is already in use or not correct format. 2. If authorization fails error message will occur. 3. The user’s information will be changed (to delete account you must request an admin) | |
| **Outputs** | |
| 1. Show the user a message depending on if the changes went though, an error occurred, or an admin request has been sent. | |

|  |  |
| --- | --- |
| **Feature #7** | |
| **Title** | Recipe Suggestion |
| **Description** | The user will be suggested recipes based of the ingredients they have and the recipes they have liked, saved, and created. |
| **Inputs** | The inputs would be based off the user’s recipe and ingredient information, the more information stored the better suggestions. |
| **Processing** | |
| 1. The application will pair recipes based of the categories they are placed in and create suggestions based off the ingredients the user has. The suggestion will also be based of the ingredients used in the saved, liked and created recipes from the user. | |
| **Outputs** | |
| 1. The application will display the suggested recipes for the user to browse. | |

|  |  |
| --- | --- |
| **Feature #8** | |
| **Title** | GPS |
| **Description** | The GPS will be used to display near by stores that may have ingredients you are looking for. |
| **Inputs** | Address, user location. |
| **Processing** | |
| 1. The app will mark near by stores to the location provided by the user. | |
| **Outputs** | |
| 1. A google maps GPS will display your location or provided address and show the stores near you. | |

## 3.2 Use Cases

### 3.2.1 Use Cases

**User:**

|  |  |
| --- | --- |
| Use Case 1: Add recipe to cookbook |  |
| Actor | User |
| Flow: | User wants to add a recipe to their cookbook. The user first enters the name of the recipe. Afterwards, the user adds the ingredients, and writes in preparation and cooking instructions. The user will then save the recipe and it will be saved into their cookbook. |

|  |  |
| --- | --- |
| Use Case 2: Add new recipes from others |  |
| Actor | User |
| Flow: | User wants to look for new recipes to add to their cookbook. They click browse and be given recipe recommendations based on what they have previously added. They will then browse through the recommendations and add the ones they like into their cookbook. |

|  |  |
| --- | --- |
| Use Case 3: Add an ingredient |  |
| Actor | User |
| Flow: | The User wants to add an ingredient that is not known to Thymeshare. The user submits a request to add the new ingredient to the platform, waiting for admin review. |

|  |  |
| --- | --- |
| Use Case 4: Look for where to get an ingredient |  |
| Actor | User |
| Flow: | The User wants to look for an ingredient they do not have. They will click on the ingredient and be redirected to the ingredient’s information page. From there, they will click the “Where to buy near me’ option and have a map open, showing them the closest locations that they can purchase the ingredient. |

|  |  |
| --- | --- |
| Use Case 5: Populate inventory |  |
| Actor | User |
| Flow: | The User wants to add ingredients they have to their inventory. They navigate to their inventory in the app, and select “Scan grocery receipt”. They then point their camera at the receipt until the app recognizes the receipt and populate a list of ingredients to add. They can then vet the list to ensure it is correct. Alternatively they can manually enter items they want to their inventory. |

**Admin:**

|  |  |
| --- | --- |
| Use Case 1: View users’ recipes. |  |
| Actor | Admin |
| Flow: | Admin wants to view short description of users’ recipes that might violate app’s rules in a list before they proceed to review them. |

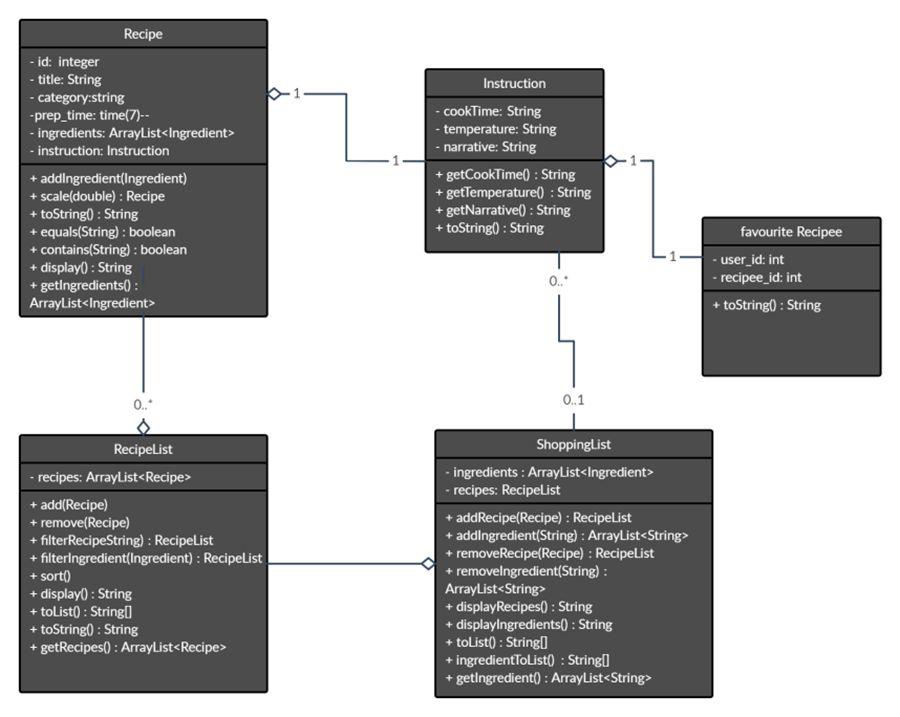
|  |  |
| --- | --- |
| Use Case 2: Review users’ recipes. |  |
| Actor | Admin |
| Flow: | Admin wants to review users’ recipes (that he chose from the list in use case 1) and edit/delete user’s recipe and/or message/ban the user. |

**Developer:**

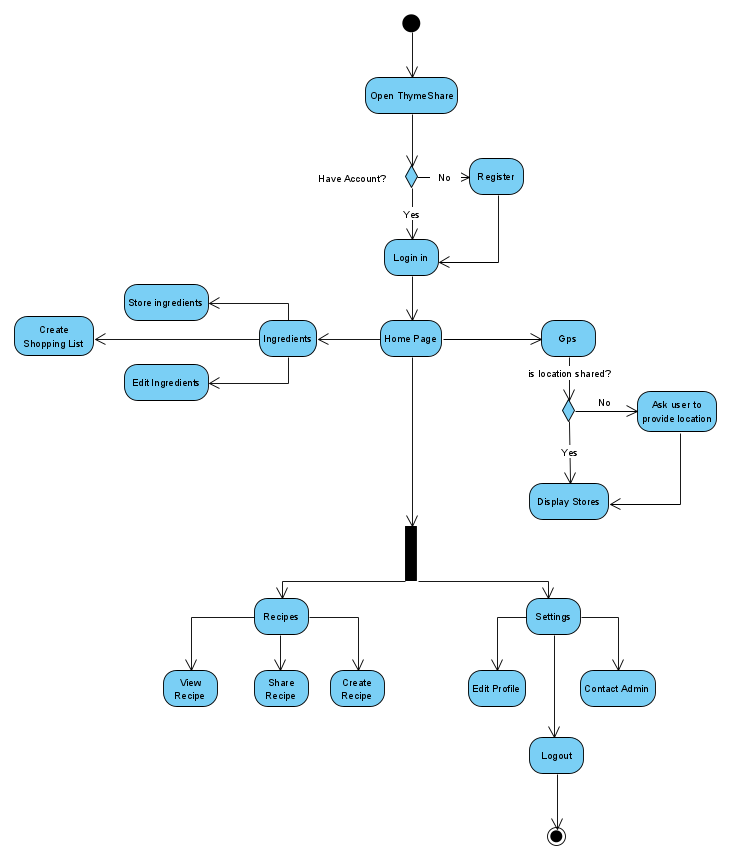
|  |  |
| --- | --- |
| Use Case 1: Track issues |  |
| Actor | Developer |
| Flow: | Developer wants easy way to track issues that arise in database/app. |

**3.3 Data Modelling and Analysis**

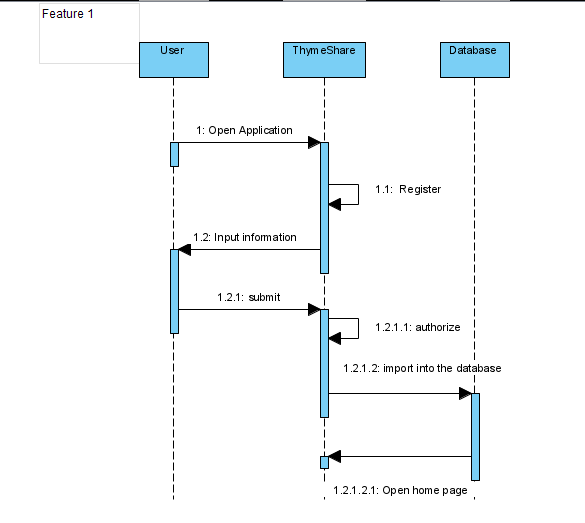
* **Normalized Data Model Diagram**

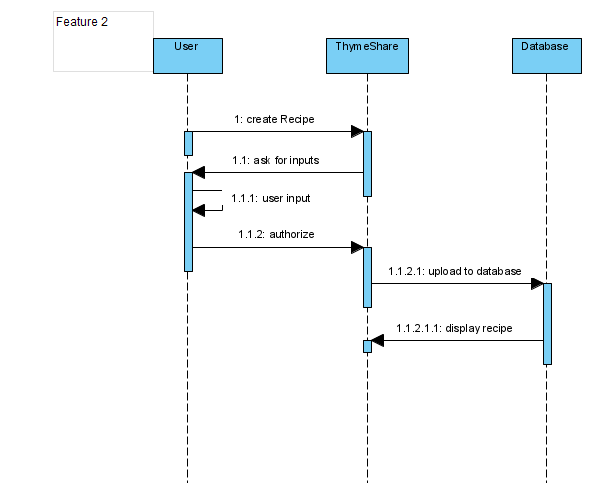
****

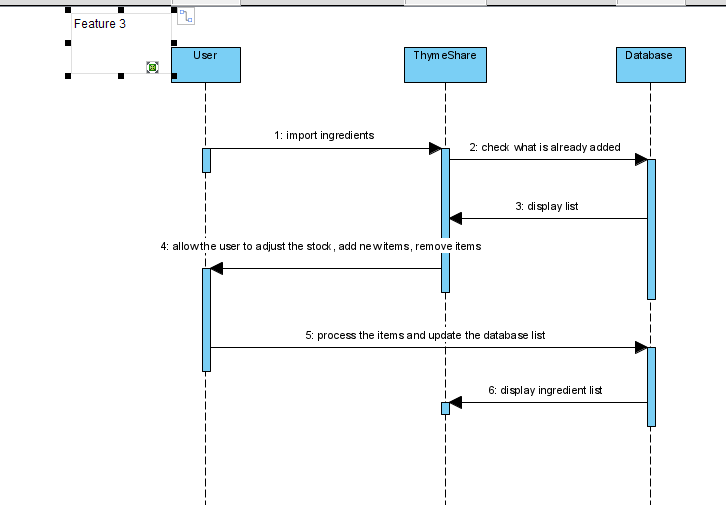
* **Activity Diagram**

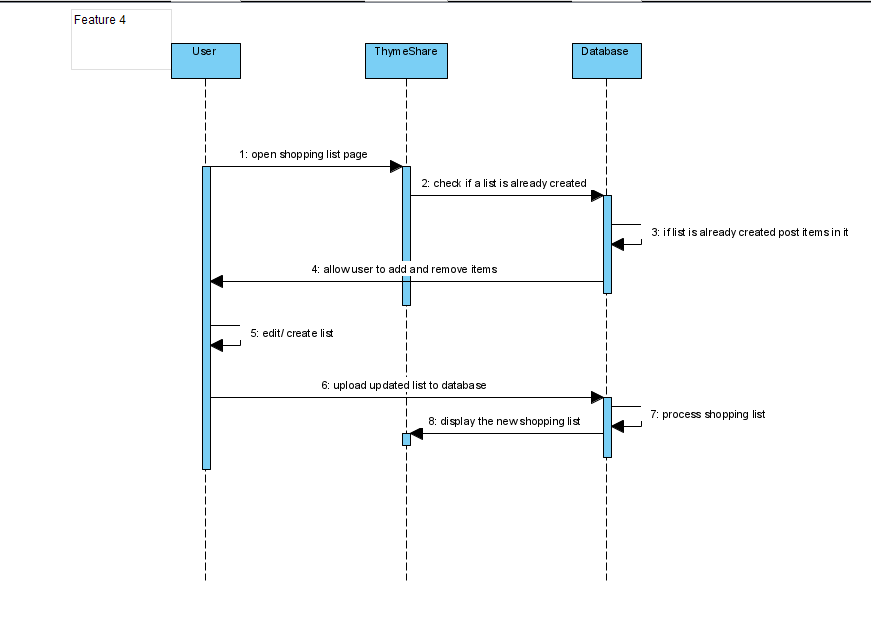
****

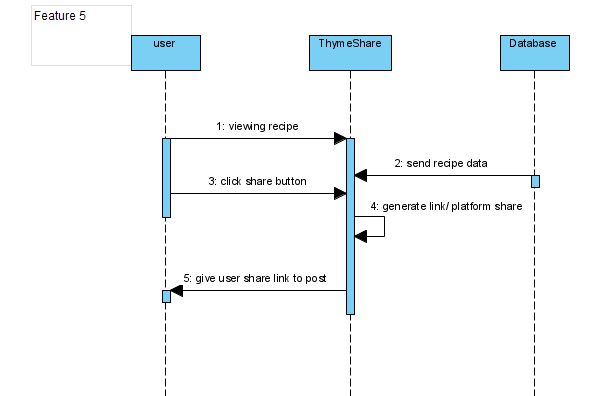
* **Sequence Diagrams**

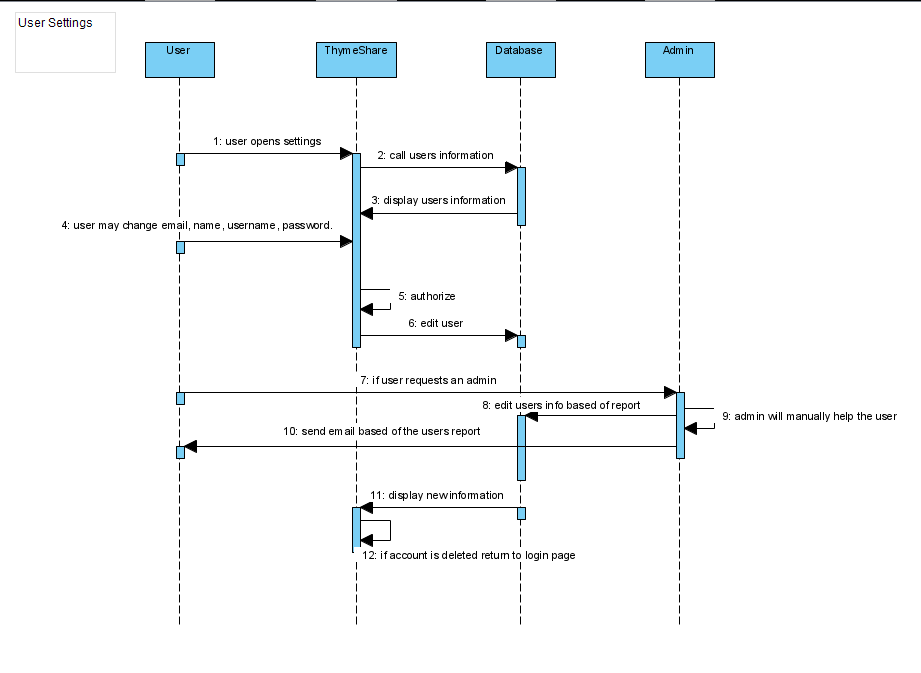
****

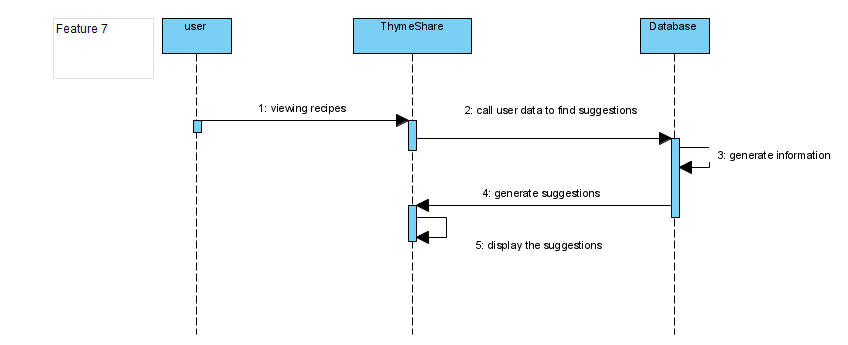
****

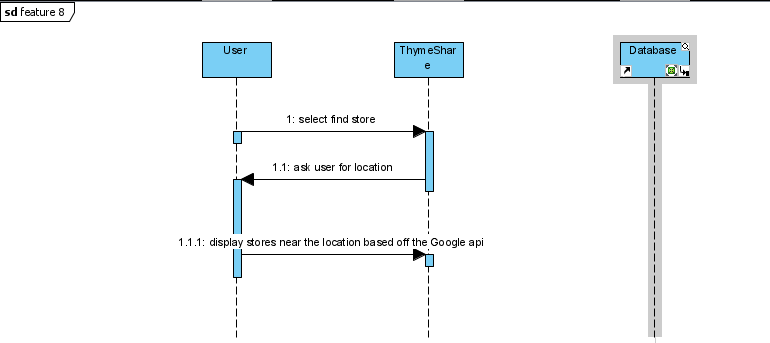
****

****

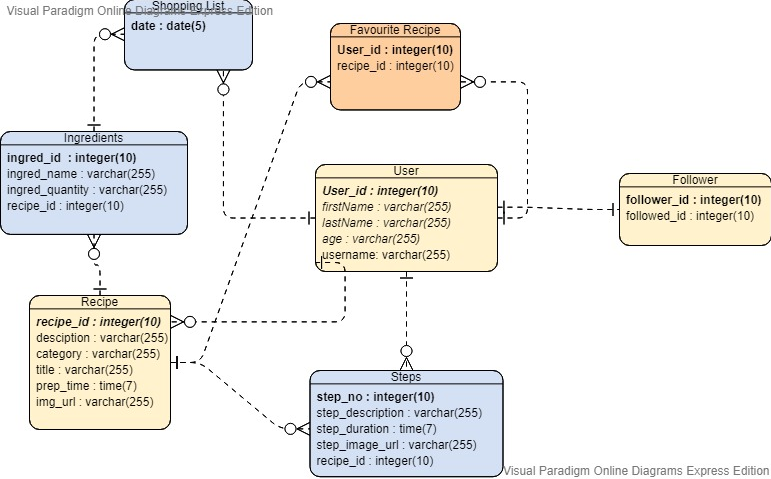




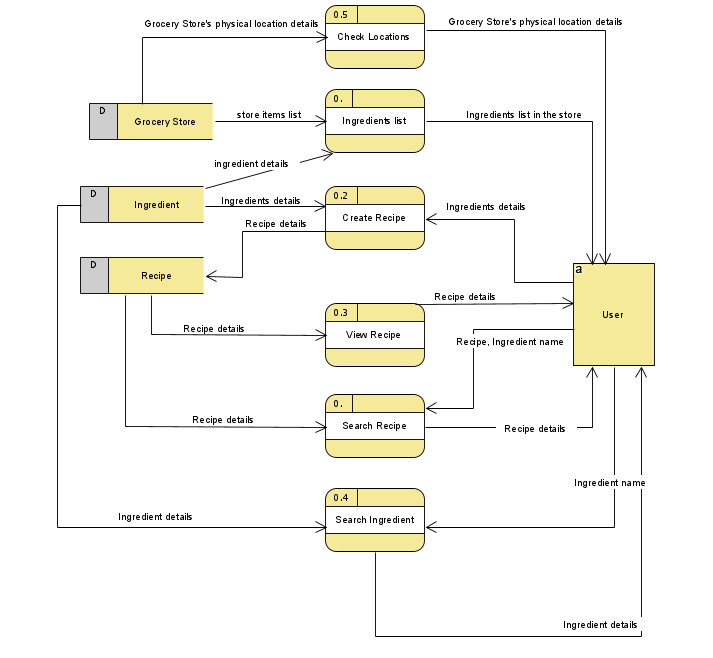




* **UML Class Diagram**



* 1. **Process Modelling**
* **Data Flow Diagram**

****

## 4.0 Non-Functional Requirements

* All personal information will be encrypted, and users will have the option of two factor authentication
* Admins cannot change a user’s personal information unless the user requests it, or if user information is deemed inappropriate
* The minimum version of android that will be able to run the app is android 15.
* Cloud servers will use the services of Amazon Web Service (AWS)
* Phone permissions required will include the phone camera, location tracking and GPS, data usage, and photos
* Data usage will be kept to a minimum, only using data when connecting to our servers
* Regulatory compliance with google play store must be considered when designing the functionality of the app
* Each user can hold 1000 recipes in their cookbook (subject to change if necessary)
* Initial radius of ingredient location will be 30km. The user can zoom out to look further if desired

## 5.0 Logical Database Requirements

Our ThymeShare app will be dynamic. It will interact with the users instantly according to the inputs that it receives. So, for it to be possible we will be using Logical Database MySQL. Users will need to enter their personal details as name, address, and phone to be recorded in the database and login to access to more elaborated functionalities as Create/View/Delete/Search a recipe. Also, if they want to keep a shopping list (the needed ingredients). This information will be kept in our database. The goal of logical database design is to create well-structured tables that properly reflect the company's business environment. All given hard disks and information files must be accessible from a single source. So, we will be taking integrated hosting i.e. Shared hosting for both our source code and our database.

**7.0 Approval**

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

|  |  |  |
| --- | --- | --- |
| Name | Signature | Date |
| Trevor Rocha | Trevor Rocha | 11/15/20 |
| Derek So | Derek So | 11/15/20 |
| Kartik | Kartik | 11/15/20 |
| Magomed Alimuradov | Magomed Alimuradov | 11/15/20 |