Deakin University

SIT737 - Service Oriented Architectures and Technologies

Assignment 1 – Project Diary

Topic

Food Recommenders System

Group A1 18

* Elvina Zheng – 216241884
* Predit Regmi – 217652071
* Sochan Mao – 217176271

Table of Contents

[Team Members 3](#_Toc511934977)

[Introduction 3](#_Toc511934978)

[Project Goal 3](#_Toc511934979)

[Project Value 4](#_Toc511934980)

[Value Hypothesis 4](#_Toc511934981)

[Growth Hypothesis 4](#_Toc511934982)

[Metrics 5](#_Toc511934983)

[Project Plan 6](#_Toc511934984)

[Communication Strategy 6](#_Toc511934985)

[Task Management Strategy 6](#_Toc511934986)

[Iteration Plans 6](#_Toc511934987)

[Design Thinking Process 9](#_Toc511934988)

[Architecture 9](#_Toc511934989)

[Overview 9](#_Toc511934990)

[Architectural Decisions 11](#_Toc511934991)

[Retrospective 14](#_Toc511934992)

[Appendix 17](#_Toc511934993)

[Annex A: Design thinking questionnaires 17](#_Toc511934994)

## Team Members

* Elvina Zheng – 216241884
* Predit Regmi – 217652071
* Sochan Mao – 217176271

# Introduction

Nowadays, eating has become a kind of culture in a lot of countries, so cooking has become an important skill for our life. Many people get higher standards for food than before with the life quality gets better, like some of them focus on the taste of the food, some of them require the healthy food. However, a lot of people just know what kind of ingredients they want, no ideas about how to cook with the sauces and seasonings. In the case, we create an application and recommend users how to cook by searching their ideal ingredients. This project report is including the parts of project goal, project value, project plan, design thinking process, architecture, and retrospective.

# Project Goal

The project of recipes application aims to guide users to cook their favourite food through the recipes recommendation application. The end user only need to input the key word of ingredients and the application will give them instant results about the relevant recipes and all the detailed steps of cooking, which help users to develop their cooking skills. Initially, this application will require users categorise their favourite types of food, like Chinese food, Vietnamese food, Malaysia food and so on. After the categorised, the system will tend to offer their favourite types of food based on the combination of ingredients from user input and types of food chosen. In addition, this application is easy for the user who is picky with food and some of them has special diets. For example, some people prefer to cook healthy food, so the recommended recipes will be with light oil and included some high protein food recommendation. Also, the user can save these recipes by login our application as a member. Our system will be designed to process the considerable amount of data from mobile application, which involved in relational database to boost the recommendation of recipes effectiveness.

# Project Value

## Value Hypothesis

All the business project needs to create the value for users first and then gain the returns from it, which same as the benchmark of our recipe application service. The standards of a group of people who choose to cook by their own instead of eating in restaurants may think about food hygiene and safety, money saving, and better quality and quantity. Hence, this type of people more likely to be our users.

Our project value is based on satisfying the user needs for cooking. With running the project of recipes recommended application, the system will catch and store the users detailed information by registering our application, such as the age, name, location, and preference of food, which help us complete the system. Compared to open a restaurant, we provide the way to cook rather than offer cooked food directly by the restaurant. Moreover, the model running through the mobile application to reduce the cost and simplify the process of opening a physical restaurant, such as reducing the labour cost, rent cost, decoration cost and so on.

Additionally, the mobile application is much convenience than an actual book by carrying and more flexible than electronic recipe book via searching the ingredients and offering multiple recipes. It is also reduced the cost by publishing book and saving time.

## Growth Hypothesis

Further developments of the process are applying more functions for the application and combining this application with some business projects. For example, selling the popular electronic books about food on the application, some users may like to read the book about healthy diet, nutrition, food culture and relational books. In addition, if we get supported by other business investors, we can build a studio for cooking some special dishes and open a course about cooking step by step like taking an online-course for users. In this way, the user can just watch the gourmet show through our application. Also, it can sell bundled and related kitchen supplies on the application, such as chef’s knife, cutting board, measuring cups, mixing bowls, vegetable peeler, and so on. Sometimes, users do not have enough time to buy a lot of different ingredients and seasonings, we can sell the food package to apply for the same model as instant noodles that just need to put all the packed seasonings and ingredients by the cooking steps. Furthermore, the application is free for the user to upload their own recipes and communicate with other users, which enrich our material library and let uses learn from each other.

Exploring more users is critical to the operation of the application. Social media is a powerful tool to explore a new user group. In the beginning, targeting a specific group that prefer to cook by themselves and interested in gourmet cuisine, and collecting users’ information through Facebook, twitter, Instagram, and other social software. It is necessary to post advertisement about our application on the social media, let more people to know about our application and download it.

## Metrics

To test whether we are successful, we will assess our achievement by comparing the sales performance and the increase in the number of users in the next quarter and collecting the user’s feedback to monitor the user satisfaction. To collect the user’s feedback about our application, we can use many different ways of telephone survey, questionnaires, and email complaint to understand what users need. This is an important process to analyse the user behaviour and improve our application for the further development. Moreover, we can also observe the user’s satisfaction from the frequency of the interactions between users and the numbers of logging on our application. Since we have connected to the social media, so the comments from the social media is important for our reputation, and it saves the time to do the survey from users.

# Project Plan

## Communication Strategy

We primarily used the mobile application WhatsApp to communicate with the group on a daily basis. Similarly, we set up meetings to discuss and share ideas face to face. Additionally, we setup a google drive folder to share the work and see the progress. Google drive was also used to work on the assignment files together and to give feedback/comments.

## Task Management Strategy

Everyone had their share of idea on each part of the assignment. The ideas were then combined according to the requirement. Google drive was updated for each new task completed and hence it was used to track on the completion of the task.

## Iteration Plans

Our iterations during the planning phase is given below

|  |  |  |  |
| --- | --- | --- | --- |
| **Iteration 1** | Planning | | |
| Start Date | 02/04/2018 | End Date | 05/04/2018 |
| Description | We went through a list of features to finalise which features to include in the service. | | |
| Result | We recognised several features which might not be feasible or might not be used by others. An idea of the service with useful features was produced. | | |
| Follow Up | Recheck the plan | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Iteration 2** | Add a Chatbot for users to ask common frequently asked questions. | | |
| Start Date | 05/04/2018 | End Date | 07/042018 |
| Description | We added a feature to out project where users can ask frequently asked questions about the services provided. | | |
| Result | A change was done in the project plan to include a new feature. | | |
| Follow Up | Create a Project Proposal | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Iteration 3** | Create a Project Proposal | | |
| Start Date | 08/04/2018 | End Date | 12/04/2018 |
| Description | A document was created describing the architecture of the service and its details. | | |
| Result | We went through few versions of architectural design to figure out what might work the best. A document with the architecture of the service and its details was created. | | |
| Follow Up | Recheck the proposal | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Iteration 4** | Add reviews and comments feature | | |
| Start Date | 13/04/2018 | End Date | 15/042018 |
| Description | We added a feature to out project where users can review and add comments to the recipes. | | |
| Result | A change was done on the architecture. Additional information was added in the document about the feature | | |
| Follow Up | Recheck the proposal | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Iteration 5** | Recheck the project proposal | | |
| Start Date | 16/04/2018 | End Date | 17/042018 |
| Description | We went through the project proposal to check if anything was missing. Or to check if any idea needed any refinement. | | |
| Result | A change was done on the architecture. | | |
| Follow Up | Proofreading and final editing. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Iteration 6** | Recheck the project proposal | | |
| Start Date | 17/04/2018 | End Date | 18/042018 |
| Description | The document was reformatted, and few errors were corrected. | | |
| Result | Final project proposal prepared. | | |
| Follow Up | Start the implementation process. | | |

# Design Thinking Process

We have done an interview with our group members referred to the hobbies experiences. As we mentioned in our project plan about the task management strategy, one of our members is the product owner. Therefore, we had interviewed her for several questionnaires following step by step beginning with Interview, dig deeper, Capture findings, Define problem statement, five draft layout sketches for our customer’s needs. She prefers the mobile web application due to many people are owning their smartphones with features of asking system about how to cook a recipe, asking system to recommend the recipes based on their existed ingredients and people can share their new recipes into the systems. More detail questionnaires see ANNEX A.

# Architecture

## Overview

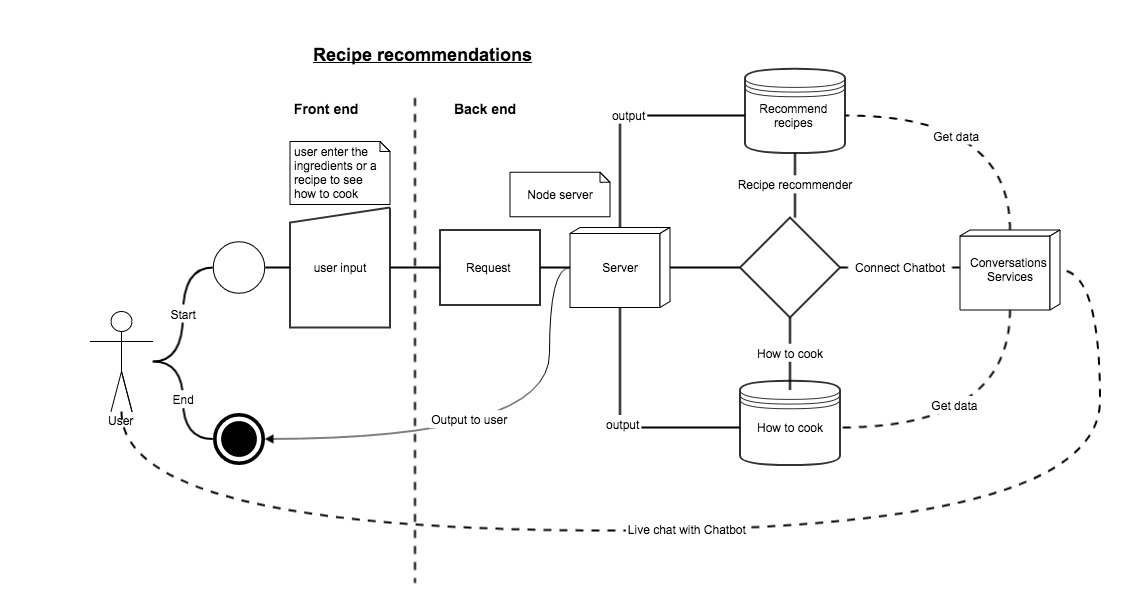


Image1: Architectural Diagram.

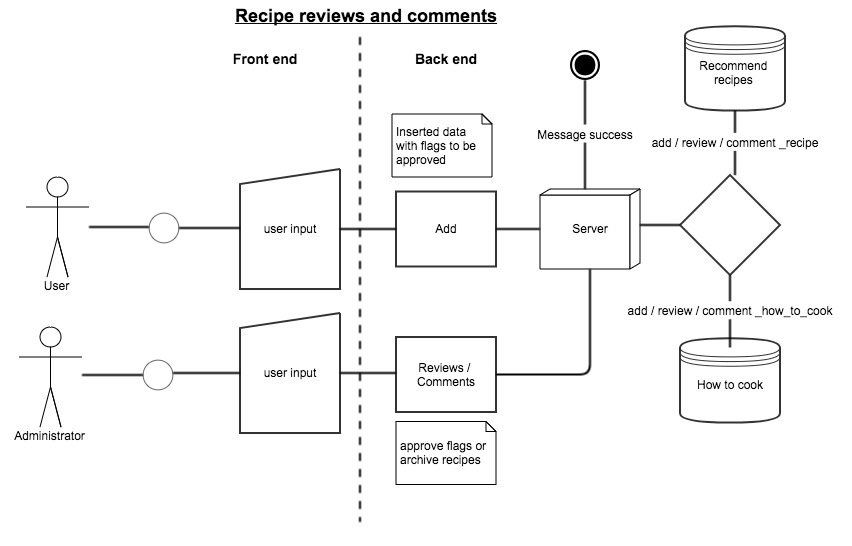


Image2: Recipe reviews and comments.

After the design thinking questionnaires, we get a system architecture as described in above figure. It is a Food Recommender System; the application consists of two parts of Front end and Back end. There will be two main objectives of this application. Users are able to request three types to server as described in the Image1.

Firstly, if the request is how-to-cook then server get data from How-to-cook data, then server response to user the instructions related to that specific recipe based on user input. Secondly, if the request is Recommend-recipe then server get data from Recommend-recipes data and server response back to user the list of appropriated recipes based on the input ingredients.

Thirdly, if user request is live\_chat then server starts to connect user to the Conversations service, we will use external service from Conversations to create Chatbots to response users about foods.

Furthermore, there will be one more part about Recipe Reviews and Comments. System will allow users to enter their own recipes into the cloud systems with the flags to-be-reviewed. Administrators can review and comment to all unapproved recipes as shown in Image2.

## Architectural Decisions

This stage is the most important for every system designs. It can affect the quality of the systems and its functionalities. Furthermore, they may face some financial issues while wrong decision had been made. There are several important parts of Architecture Decisions such as storage technology and model, platforms, tools, language etc, below table will showing some of them.

|  |  |
| --- | --- |
| **AD001** | **Choice of the storage technology and model** |
| **Problem Statement** | Our application seems do not require big storage due to there are few *operations* will be made. However, in the future if it needs to upgrade to real time communication based-system, for example user can chat with the system about recipes or whatever related to food to get some recommendation from the system, so it required to store a lot of information for each dialogue. |
| **Available Options** | |
| **Option 1.** | **Utilise a Relational Database**  Description:  Relational database has been commonly used for decades, the system need to have some ready *designed* of entities records into tables. Furthermore, some features have been used such as views, stored procedures, SQL functions, replicate etc.  Pros:   * Can use query language to work on data. * Very simple model can be read by all stakeholders including the customers who do not have IT background.   Cons:   * Can be facing SQL Injection while not well implementation with orientation programming. |
| **Option 2.** | **Utilise a NoSQL / Document Oriented Database**  Description:  According to Wikipedia.org (2018), this solution has been used since the late 1960s. It is storing data in document oriented in various *format* such as XML, JSon, BSon and YAML. Data structure can be changed over time like MongoDB. Database with fast clustering, for example Terrastore and may other powerful databases such as CouchDB, RavenDB, OrientDB, ThruDB, SisoDB, RatorDB, CloudKit, Perservere, Jackrabbit etc.  Pros:   * Data structure More flex * Very simple in term of architecture * Schema-less implementation. * Easy to find the cluster database, it means if we install it in many machines, it will be working over all of those machine as using the only one machine.   Cons:   * Poor documentation * Not support with the universal language of SQL * Hard to find support even though in high pay * Lack of knowledge of this type of storage model within the team. |
| **Option 3.** | **Utilise a File based System**  Description:  This solution recommends using file to store data.  Pros:   * Easy to use model, very simple   Cons:   * Cannot apply for big data, it will be hard to search for data * No performance of data management. * Do not have some backup features |
| **Decision** | |
| **Selected Option** | The selected option is Option 1. |
| **Justification** | Option 1 is more scalable, and it is easily to explain our customer about her needed products. This is because, user can understand the database schema and confident on development team.  Furthermore, everyone knows well about relational databases vs NoSQL. NoSQL may be needed in big data or for some flexible data that the model can be changed day to day, it is not a basic data storage that every IT programmer has ever learned. |
| **Implications** |  |

|  |  |
| --- | --- |
| **AD002** | **Platform, language** |
| **Problem Statement** | Based on our design thinking and feedback from our customer on our five draft sketch layouts, we are thinking about which platform should produce the quality of mobile web application. |
| **Available Options** | |
| **Option 1.** | **NodeJS, Html, CSS, JavaScript**  Description:  This solution is using NodeJS to create API to query data from database based on user inputs.  Pros:   * Scalable network operations * Running as a non-blocking I/O calls that can support tens of thousands concurrent connections   Cons:   * Node does not provide scalability to not take advantage of today server-class hardware * It is a pain to work with relational databases * It is suitable for I/O staffs only |
| **Option 2.** | **JEE (Java Platform, Enterprise Edition)**  Description:  JEE stands for Java Platform, Enterprise Edition, used to create computing distributed and web services.  Pros:   * Can run on all environment of operating systems * Proven track records * High volume, handle complex   Cons:   * Complex application development application * Difficulty to use tool * Limitation of Java Swing * Expensive for built, deploy and application managements * There is no build-in support web services standard |
| **Option 3.** | **.Net Framework**  Description:  .Net Framework constitutes of Framework Class Library (FCL) and Common Language Runtime (CLR). It has been used to create user interface, data access, database connectivity, web application development, etc. .Net Core is being very powerful for Universal Windows Platform (UWP), cross-platform and cloud computing workloads.  Pros:   * Easy to use for programmers * Strong framework to build graphic user interface * Give many choice of programming languages * Cost less, less expensive tools   Cons:   * Framework run only windows * Need to learn every Microsoft tools * Limitations of integrated development environment |
| **Decision** | |
| **Selected Option** | The selected option is Option 1. |
| **Justification** | Option 1 NodeJS is becoming more powerful for back end. Furthermore, our course of web service is practicing with NodeJS. Consequently, it is the right time for us to use NodeJS in our project development environment. |
| **Implications** | Due to some articles state that NodeJS is a pain with relational database due to AD001 we choose Relational Databases as our model. So, we will do some more researches about the challenges to use NodeJS with relational databases. |

# 

# Retrospective

We as a team of three started with an idea similar to the current one but had to change the original idea as there was loopholes in the original idea. The original idea was to let users choose ingredients and a chef prepares a dish with those. Since we formed a group we have chosen a topic about “Our own food” that user can use the system to choose their recipes of a plate online then system sends to the chef to cook and deliver it to the user. However, it seems complicated to make it happen because there are a lot of stakeholders such as restaurants to join, need delivery service, check service to make sure it is completed tasks, etc. In addition, we realised that the user might not like the dish if the mix of ingredients he chose was not suitable and the chef is not happy with the customize of ingredients. Therefore, it will be complicated to implement for assignment 2 and maybe not be easy to implement it individually in a very short period. So, we are thinking about another small service “Food Recommenders System”.

“Food Recommenders System” is a service to recommend users about how to cook their food and how to make an appropriate food if they have resources. There will be two main functions from cloud service. First it tells user if they request about how to make a plate (buy ingredients and how to make it step by step). Another function, if a user got some resources, for example, a Salmon fish, onions, carrots but s/he does not know what recipes they can make from this fish, they can ask our “Food Recommenders System”. The only thing we would have liked to do it differently was that we did not go to the details of the original idea when we thought of it. There were few problems that we missed since we did not think of the details. This resulted in us modifying the original idea.

Talking about architecture decisions, technologies are developing significantly, there are many frameworks and technologies that allow us to create services on cloud. As a team, we have discussed about how to store data and technologies that we will use to implement it in stage 2. For data storage, we decided to use relational databases with NodeJS as Backend.

Moreover, as a teamwork we are thinking about all user needs for this Cook Your Own Food. We will use Chatbots from Conversations service of IBM Cloud to implement the live chat for user to ask everything about food. It will be more user attractive, especially it is a real time response that should make users happy with the results on their food questions. We also include a recipe management, user can add their own recipes, but they are needed to be approved before appearing in food recommender systems.

To sum up, this project diary is crucial helpful for project development. This is because, it is telling us or ask the team members to do discussion and research in detail to select technologies to produce the needed products for customers. Furthermore, we have learned a lot about existed services on the cloud, for instance, IBM cloud that we can use to create new service such as Food Recommenders System.

# Appendix

## Annex A: Design thinking questionnaires

