TRILINGUAL MOBILE DICTIONARY FOR MALAYSIAN SCHOOL STUDENTS

WONG SZEE YUNG

A project report submitted in partial fulfilment of the requirements for the award of Bachelor of Science (Honours) Software Engineering

Lee Kong Chian Faculty of Engineering and Science
Universiti Tunku Abdul Rahman

May 2021

DECLARATION

I hereby declare that this project report is based on my original work except for citations and quotations which have been duly acknowledged. I also declare that it has not been previously and concurrently submitted for any other degree or award at UTAR or other institutions.

Signature : 

Name : Wong Szee Yung

ID No. : 1701638

Date : 3/5/2021

APPROVAL FOR SUBMISSION

I certify that this project report entitled **"TRILINGUAL MOBILE DICTIONARY FOR MALAYSIAN SCHOOL STUDENTS"** was prepared by **WONG SZEE YUNG** has met the required standard for submission in partial fulfilment of the requirements for the award of Bachelor of Science (Honours) Software Engineering at Universiti Tunku Abdul Rahman.

Approved by,

Signature : 

Supervisor : Ms. Beh Hooi Ching

Date : 3th May 2021

Signature :

Co-Supervisor :

Date :

ACKNOWLEDGEMENTS

First of all, I would like to express my special thanks of gratitude to my project supervisor, Ms. Beh Hooi Ching, for giving me the golden opportunity to do this wonderful project on the topic of trilingual mobile dictionary application. Also, she has provided a lot of insightful advice, encouragement and extraordinary patience throughout the whole project.

Secondly, I would like to express my gratitude to my loving parents for supporting me during the process of preparing and completing this report.

Lastly, I would also like to express my gratitude to my friends who have supported me with guidance, encouragement and comments throughout the development of this project.

ABSTRACT

In Malaysia, most of the school students are required to study three languages which are Malay, Chinese and English. To some extent, the relationship among the three languages may help those school students to understand the words they have come up with or obtained. As we know, dictionary is one of the essential mediums for learning any language. Along with the rapid growth of mobile technologies at this era, numerous mobile applications were extensively developed as a media for learning. In fact, dictionaries on handheld devices such as smartphones and tablets are more realistic and convenient than the conventional dictionaries as it would not restrict by the time and space limitation, where this allows users to easily retrieve the information at anytime and anywhere. Moreover, the procedures of flipping and searching a particular vocabulary among three languages from the conventional dictionaries is challenging, time consuming and not effective at all. Thus, the aim of this project is to assist Malaysia school students in their learning by simplifying the process of accessing the dictionaries data. At the same time, the interactive tools that are available in the mobile application might be able to boost up the performance of Malaysian school students in learning vocabularies. In order to evaluate the common features that are important for Malaysian school students and to figure out any valuable core functions that are not supported by any other existing applications, a literature review on the latest comparable mobile applications was undertaken. On the other hand, evolutionary prototyping development methodology is adopted to continually cater with the version upgrades and user feedback. Furthermore, user acceptance testing was carried out with five invited participants, where these participants are the mixture of primary school students and secondary school students. As a result, most of the students are able to search and understand the information shown in the mobile dictionary application without spending much time. In short, the project has successfully met the main goal which is allowing Malaysian school students to translate and learn the vocabularies in an effective and convenient way.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS / ABBREVIATIONS

ACID	Atomicity, Consistency, Isolation, Durability
ANSI	American National Standards Institute
API	Application Programming Interface
BLOC	Business Logic Component
CAP	Consistency, Availability, Partition-Tolerance
COC	Convention on Configuration
CRUD	Create, Read, Update, Delete
DRY	Don’t Repeat Yourself
ERD	Entity Relationship Diagram
GRE	Graduate Record Examinations
IDE	Integrated Development Environment
IELTS	International English Language Testing System
ISO	International Organization for Standardization
KISAP	Keep It Simple as Possible
MCMC	Malaysia Communications and Multimedia Commission
NoSQL	Not Only SQL
OS	Operating System
RAD	Rapid Application Development
RDBMS	Relational Database Management System
REST	Representational State Transfer
SDLC	Software Development Life Cycle
SQL	Structured Query Language
SUS	System Usability Scale
TOEFL	Test of English as a Foreign Language
TOEIC	Test of English for International Communication
UAT	User Acceptance Testing

LIST OF APPENDICES

CHAPTER 1

INTRODUCTION

1.1 Introduction

Background, problem statement, project goal, project objectives, proposed solution, proposed approach and project scope are discussed in this chapter.

1.2 Background

Twenty-first century learning and teaching environments are changing rapidly due to unprecedented opportunities created for education by the advancement of information and communication technologies. After approximately half a century of machines being embedded into teaching, the exponential development of smart apps is opening up a whole universe of digital technology learning experiences. According to Kukulska-Hulme and Shield (2008), mobile learning is a form of learning that happens with the aid of smartphones and literally implies studying without any consideration of location and time.

Although mobile learning initially centred on the role of mobile technology and applications in education, mobile learning has been distinguished in recent years by consumer accessibility and informal learning that occurs out of the classroom (Sharples, 2006). In this situation, mobile learning is supported by all handheld devices, such as tablets and smartphones. Pecherzewska and Knots (2007) have stated that smartphones are the most widely used learning devices among the portable technological instruments, this is because their usability and portability has been recognized and recommended by many scholars for the purpose of studying and teaching in the educational settings. Moreover, research also demonstrates smartphones can be used to optimize teaching, motivate and improve place-based learning.

Vocabulary as one of the primary components of a language has dominated a growing proportion of second or third language acquisition. Schmitt (2008) has once mentioned that indirect learning with clear follow-up was a very successful form of learning vocabulary, yet reading comprehension and dictionary analysis are a common example of this. Over about a hundred

years, dictionaries play an essential role in encouraging word learning and it has also become the focus of some scholars. Numerous studies regarding this topic have been undertaken and discussed, such as the facilitated impact of dictionary consultation and the difficulties on method choosing faced by students when they are consulting the dictionaries (Chan, 2011). Thus, this report is mainly focused on finding a solution and approach to create the trilingual mobile dictionary application and solve the stated problems.

1.3 Problem Statement

In Malaysia, most of the school students are required to study three languages which are Malay, Chinese and English. To some extent, the relationship among the three languages may help those school students to understand the words they have come up with or obtained, but there are several other terms which involve specific working procedures to search for the concrete explanation of context in the target language with their corresponding equivalents. However, the process of flipping and checking a particular vocabulary among three languages from paper dictionaries is time consuming, challenging and not effective at all.

Besides that, even though the finding (HAND PHONE USERS SURVEY) done by MCMC from 2014 to 2017 has clearly shown that the smartphone accessibility among school students have increased steadily, but the trilingual mobile dictionary is still not available in any mobile or web platform (Suruhanjaya Komunikasi dan Multimedia Malaysia Malaysian Communications and Multimedia Commission HAND PHONE USERS SURVEY 2017 STATISTICAL BRIEF NUMBER SEVENTEEN, 2017). Therefore, students have to perform twice on the translation by switching two different bilingual dictionary applications in order to understand a certain word.

According to the research on the existing bilingual mobile dictionary applications, school students can only get superficial understanding from the mobile dictionary which only provides limited information about words. Hence, school students always feel challenged to pick the right interpretation for an unfamiliar term in the dictionary. In fact, definition of words in dictionary that may be precise for adults are frequently too difficult to grasps for youngsters, and simple descriptions in school glossaries also do not explain

the word's context accurately (Some Obstacles to Vocabulary Development | Adolescent Literacy Topics A-Z, 2020).

1.4 Project Goal

The project's goal is to offer the primary and secondary school students an effective and efficient way to check the appropriate meaning and translate the words among English, Chinese and Malay at the same time.

1.5 Project Objectives

1. To decrease at least 10% of time usage in searching a particular vocabulary.
2. To develop a functionable trilingual mobile dictionary application that consists of interactive tools such as searching, favourite list, related words, word properties and add note.
3. To compile and expand a comprehensive corpus of English, Mandarin and Malay vocabularies in the trilingual mobile dictionary application.

1.6 Proposed Solution

In order to understand the meaning and translate certain words to other languages in an efficient and effective way, a trilingual dictionary in a mobile module is chosen due to the high level of mobility and accessibility by the school students.

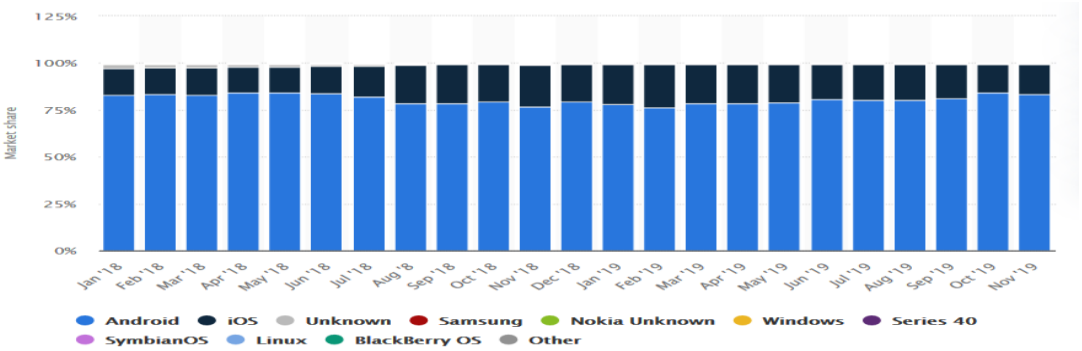


Figure 1.1: Market Share of Mobile Operating System in Malaysia from January 2018 to November 2019

(Malaysia: mobile OS market share 2019 | Statista, 2020)

According to statistic result shared by Statista Cooperation, Android operating system has occupied a greater market share of mobile operating system in Malaysia from January 2018 to November 2019 (Malaysia: mobile OS market share 2019 | Statista, 2020). Since the majority of smartphone users in Malaysia are using the Android operating system, hence, this trilingual mobile dictionary application will only be developed in the Android operating system.

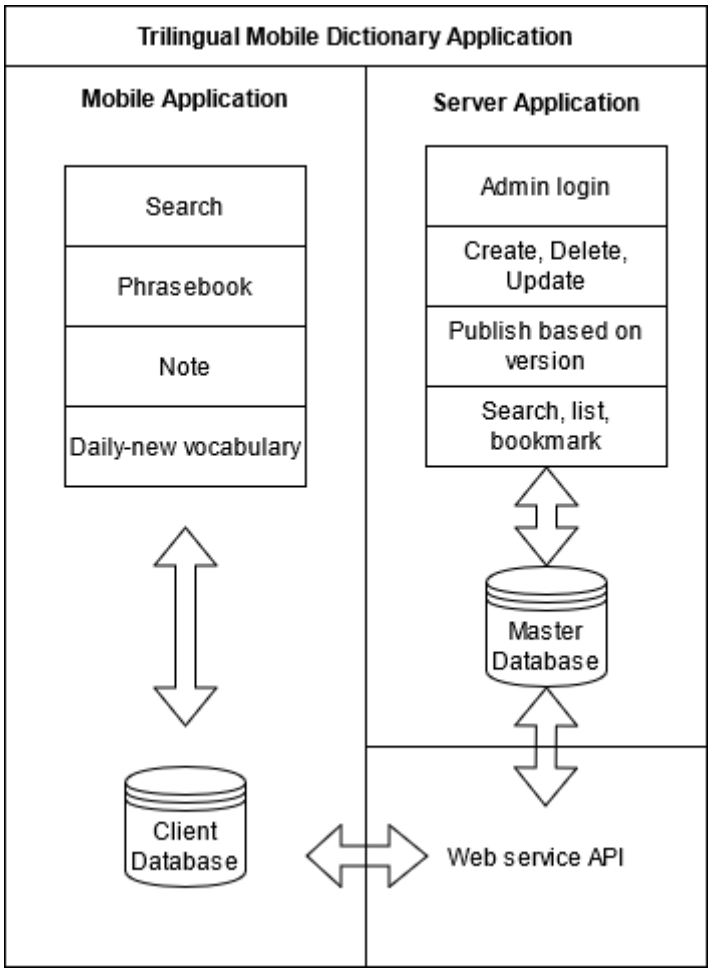


Figure 1.2: Architecture of the Application

Trilingual mobile dictionary application includes two components which are mobile application and web server. In order to have a better understanding of these two components, web server can be interpreted as playing the role of "collector", while mobile application is playing the role as a "presenter". In other words, mobile application serves as a platform for displaying or presenting the information that is collected by the web server, however, both mobile application and web server are interdependent. In this project, there are

a total of five technologies used for the development, management and servicing of web server and mobile application. First of all, the web server will include JavaScript, Ruby on Rails and PostgreSQL. The reason why JavaScript, Ruby on Rails and PostgreSQL are selected is because JavaScript creates dynamic images with its own language on application and web pages; Ruby on Rails offers the default frameworks for a web service, database and web pages; meanwhile PostgreSQL safely manages data. On the other hand, Flutter acts as a programming kit for open-source software and SQLite acts as a database driver, thus, Flutter and SQLite are selected and used in the development of trilingual mobile dictionary application. In brief, these five technologies are capable of enabling the developer to define some roles such as terms collection, authentication, uploading of documents, processing of documents, terms checking, terms managing, list terms, search terms and bookmark the desired terms.

1.7 Proposed Approach

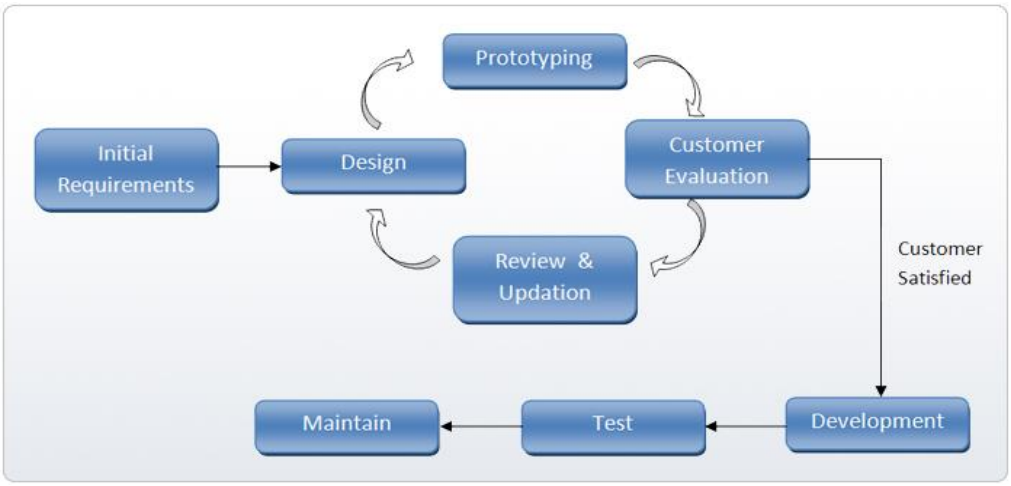


Figure 1.3: Evolutionary Prototyping Model

(Evolutionary Prototyping Methodology - Welcome to Shopper's Mart, 2020)

The evolutionary prototyping approach is used as the methodology to develop this trilingual mobile dictionary application. Prototyping is one of the most commonly used Software Development Life Cycle Model and it is used where the users do not realize the precise specifications of the project beforehand. Furthermore, the system is partly introduced in this process model before or after the review period, thereby offering the users the ability to see the

commodity early in the development cycle. The cycle begins with survey and the creation of the unfinished high-level paper model where this paper can be used to build the original prototype that incorporates only the essential functions that the user needs. Next, if the users have defined the problems, the prototype is further optimized to remove them. The cycle is then proceeding until the version is accepted and satisfied by the user. In brief, throughout this model, an end product prototype is first produced, tested and improved as per users feedback repeatedly before a final appropriate version is achieved and become the foundation for the actual product (Software Engineering | Prototyping Model, 2020).

There are some reasons that evolutionary prototyping is chosen besides the other three types of prototyping when developing the trilingual mobile dictionary application. Firstly, in the early life cycle, the users get to see the partial product, while this can definitely guarantee a greater degree of satisfaction from the users. Moreover, evolutionary prototyping also provides the developer a room for improvement whenever there is any new requirement or criteria added by the users. Lastly, missing functionalities and errors can be found out earlier when evolutionary prototyping is implemented, this can eventually help the developer to save a lot of effort and cost.

1.8 Project Scope

In this section, the project scope can be categorized into target users and modules of the project. Ultimately, a trilingual mobile dictionary application is developed to solve the stated problems.

1.8.1 Target Users

The target users of this trilingual mobile dictionary application are purely focused on Malaysian primary and secondary school students who have a strong desire to learn more unfamiliar vocabularies in different languages such as English, Mandarin and Malay.

1.8.2 Modules Covered

The components and functionalities of the trilingual mobile dictionary application are discussed in the section below.

1. **Searching vocabularies**

   The trilingual mobile dictionary application allows the users to search the vocabularies in English and it will display the word properties, related words, sentences and inter-translation of Malay and Mandarin at the same time.

2. **Adding vocabularies into the phrasebook**

   Nowadays, school students rarely use a notebook to record down the meaning of the word. Therefore, the trilingual mobile dictionary application provides a function such as phrasebook in order to let the users collect and recap the terms they have learnt earlier.

3. **Adding note to a word**

   Students always have some special or specific skills in learning new vocabularies, for example, identify the word nuances, create word associations or learn the roots of the word. Thus, users are capable of jotting down their interpretations or own ideas to certain words in the note function that is provided in the trilingual mobile dictionary application.

4. **Displaying daily new vocabulary**

   The trilingual mobile dictionary application updates and displays new vocabulary every day, so that users could learn and accumulate more phrases in a shorter period. Besides that, users can also see this function as a daily learning guideline or mission.

1.8.3 Modules Not Covered

Firstly, the dictionary application will only be available on the mobile platform, this is because smartphones have a higher portability and accessibility to teenagers compared to computers. Secondly, functions such as history list and sharing vocabularies in social media are also unavailable, it is because this function will not contribute to the stated objectives. Nevertheless, all these functions are really helpful with this application and it is used with potential enhancement in the future.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

Mobile dictionaries can be downloaded and accessed easily in the current digital market. In this chapter, the functionalities of three existing mobile dictionaries are studied and discussed. Besides that, several development methodologies have been explored, discussed and compared. On the other hand, the comparison between the types of database and the characteristics of dictionary APIs are also further elaborated.

2.2 Study and Assess on the Similar Existing Application

A series of analysis regarding current mobile dictionaries available in markets such as App Store or Play Store are carried out in order to establish a mobile dictionary that can fulfil and cover all the specific usage criteria and functionalities. In this section, three mobile dictionaries developed by foreign software developers are studied and assessed, while these three mobile dictionaries are Dictionary & Translator Free from Bravolol, Dict Box from XungLe, Malay-English Dictionary from AllDict.

2.2.1 Dictionary & Translator Free

Dictionary & Translator Free is a free bilingual mobile dictionary application that supports translation and detailed word definitions between English and Mandarin. It was released on 4 February 2012 and published by a Hong Kong company named Bravolol.

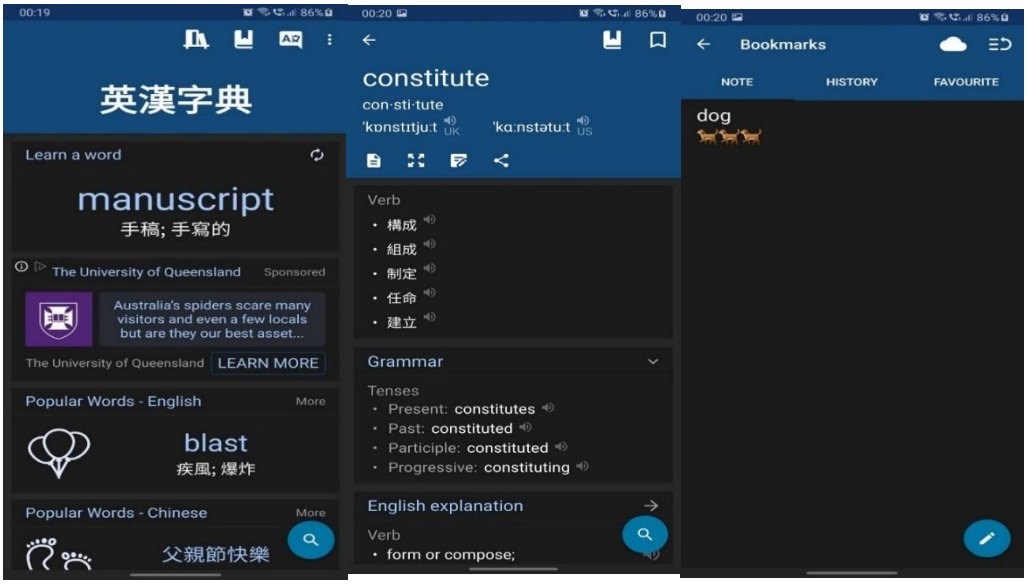


Figure 2.1: Interface of Dictionary & Translator Free

2.2.1.1 Main Features of Dictionary & Translator Free

- **Search and translate vocabulary**

  Dictionary & Translator Free application allowed users to insert either English or Chinese into the search bar and perform searching on the particular word. Moreover, the search features also provided text prediction in order to predict what is the word that is keyed in by the users.

- **Present the details of a vocabulary**

  Dictionary & Translator Free application provided pronunciation of Chinese and English words (UK and US). Furthermore, users can also retrieve the information such as word properties, grammar, English explanation and sentences about the particular vocabulary.

- **Add note to a vocabulary**

  Dictionary & Translator Free application allowed users to add their own understanding as a note to a vocabulary, moreover, besides words or letters, emoji can also be added into the note.

- **Add vocabulary into the phrasebook**

  Dictionary & Translator Free application allowed users to bookmark the vocabulary that users are desired to learn. Users can also remove the bookmarked vocabulary from the phrasebook if it is necessary.

- **Check history**

  Dictionary & Translator Free application allowed users to look back the vocabularies that they have searched before.

- **Share vocabulary to social media**

  Dictionary & Translator Free application allowed users to share the vocabulary that they have learned to the social media sites. The eligible social media sites are Facebook, Wechat, Whatsapp, Instagram, Dropbox, Google drive and etc.

- **Quiz**

  Dictionary & Translator Free application provided users an environment to learn the translation between Chinese and English. In that learning environment, users are given the corresponding scores and chances for answering the quiz so that users can know what level of translating skills they possess.

- **Display daily new vocabulary**

  Dictionary & Translator Free application displayed different vocabularies on a daily basis so that users are capable of exploring new words every day.

- **Translate paragraphs or sentences**

  Dictionary & Translator Free application allowed users to translate paragraphs or sentences that contained 1000 characters from and to 72 languages that were prepared in the application. Moreover, users can also perform translation by using image processing techniques to process the paragraphs or sentences that are captured by the mobile device.

2.2.1.2 Evaluation on Dictionary & Translator Free Mobile Application

Wide usage of smartphone and wireless devices in education has contributed to innovative improvements in teaching and learning for teachers and students. Dictionary & Translator Free is an offline dictionary application that allows users to translate and check the meaning of Chinese and English words, not only that, it is also considered as a highly useful tool for study. Users are able to access this mobile dictionary application without any registration or linking

any social media to the application. Dictionary & Translator Free application has actually provided some features that can encourage users to learn more vocabularies while using this application.

First of all, one of the best features in this mobile dictionary application is definitely the quiz function. Since the quiz questions are typically focused on translating words, the solution will be either right or wrong. This method of testing is far simpler than testing the oral abilities and compositional competencies of the users especially students. By using this function, users can easily understand the culture of the target language and they are capable of gaining more motivation on learning different languages. Nevertheless, this quiz feature should be designed in a level-based concept, for an instance, beginner level, intermediate level and advanced level. This is because level-based learning allows users to know which level they belong to and also, they can feel a sense of achievement when they reach a higher level.

Moreover, another good feature that has been explored in this mobile dictionary application is the presentation of vocabulary's details. Instead of understanding the superficial meaning of a vocabulary, this mobile dictionary application has also provided information such as word properties, grammar and some sample sentences for the word searched. Without questioning, these series of details can provide users a clearer picture on how to implement the particular word. In addition, Dictionary & Translator Free application also allows users to add note to a vocabulary, while the note format can be either letter or emoji. According to studies, our brain is basically an image processor rather than a text processor, this is because visual learning is much more efficient and effective than some other learning form. Thus, users especially primary school students find it easier to remember and understand the vocabulary when they add the emoji to refer to the word. However, since there are limited emoji available in smartphones, it will be better if the Dictionary & Translator Free application allows users to add pictures as a note.

In short, beside general features such as translating and searching words, adding note to a vocabulary feature can also be included in the development of the trilingual mobile dictionary application. This is because students always have a different style for remembering vocabulary, while this

feature is able to provide students a space to jot down their ideas and understanding to a particular vocabulary. Ultimately, this feature can help students to learn and memorize the vocabulary in an easier and efficient way.

2.2.2 Dict Box

Dict Box is a free bilingual mobile dictionary application that supports translation and detailed word definitions between English and Mandarin. It was released on 30 December 2012 and published by a China company named Xung Le.

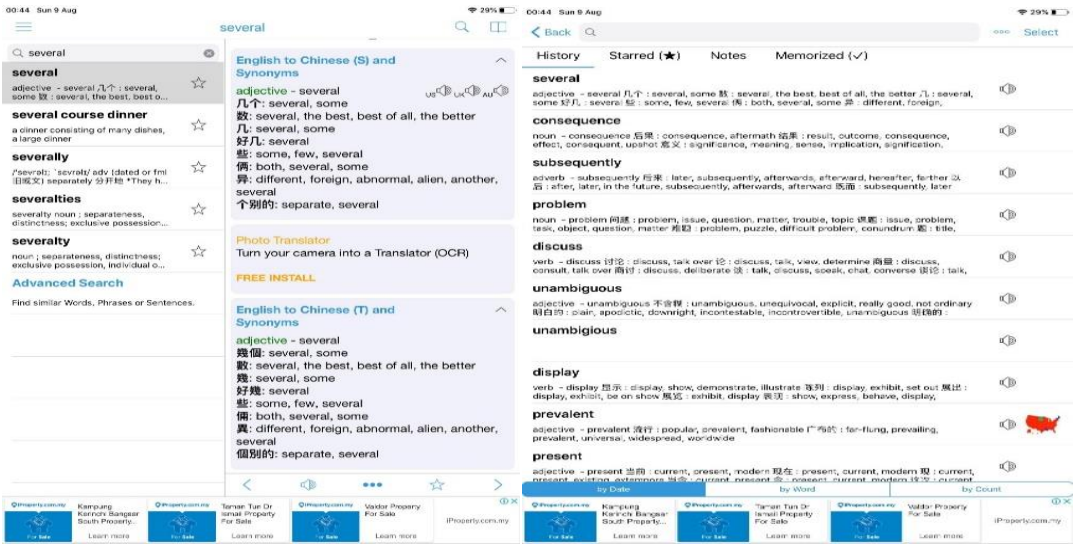


Figure 2.2: Interface of Dict Box

2.2.2.1 Main Features of Dict Box

- **Search and translate vocabulary**

  Users are able translate and look up the meaning of the word after he or she has typed in the vocabulary into the search bar. Dict Box used Wikipedia information as a source to explain the word searched by the users.

- **Display the details of a vocabulary**

  Users can get the information such as word properties, sample sentences and pronunciation of UK, US and AUS on a particular word. Besides that, Dict Box also collaborated with Oxford Dictionary, hence,

users can also seek for the vocabulary's details from another publisher by using this integrated mobile dictionary application.

- **Attach note to a word**

  Users are able to attach their own thoughts and interpretation on a word to the note, while the note format is not restricted to only letters but emoji are also accepted.

- **Manage vocabularies**

  In this mobile dictionary application, users can add the vocabulary that they are interested to learn into the bookmark section. Then, users can also categorize the vocabulary that they have already remembered in a memorized section. Not only that, users are also allowed to remove or delete the vocabularies in the bookmark and memorized section when it is necessary.

- **Check history**

  The history section in the Dict Box was designed for the users who wished to look up the vocabularies that he or she has searched before.

- **Flashcards**

  Users can prepare their English tests such as IELTS, TOEFL, GRE and TOEIC by studying the vocabulary flashcards available in the Dict Box. Besides that, users can also add the flashcard into the bookmark section and memorized section.

- **Word Reminder**

  Users are allowed to set a reminder regarding the target number of words to learn per day for themselves. This feature will integrate with the mobile device and pop out the notification bar which contains a vocabulary and its meaning, meanwhile, it also boosts up the progression of learning.

2.2.2.2 Evaluation on Dict Box Mobile Application

Mobile dictionary application has moved into the digital realm and has incorporated other features rather than just performing a classic translation. Dict Box is an offline mobile application that has provided more than just a check and translate vocabulary features. Users are not required to register this

mobile dictionary application before using it, however, users are given freedom to update their membership to premium in order to utilize premium features in this application. After reviewing the Dict Box application, there are some fascinating features that can be discussed at the section below.

Firstly, one of the best features that has been recognized is definitely the flashcards feature. Flashcards are one of the popular study tools as they facilitate learning through active recall, which is one of the most powerful practices that our brains do. On the other hand, the flashcards feature in this mobile dictionary application also acts as a revision station for those people who intend to take English tests such as IELTS, TOEFL, GRE and etc. This is because the flashcards feature has categorized the vocabularies that are frequently used in those exams. Nevertheless, the flashcards for MUET exam are currently not available in this application as MUET is not considered as an international certificate.

Furthermore, Dict Box application also provided authority to their users to manage the vocabularies. For example, when users are interested in learning new vocabularies, they can actually add those vocabularies into a bookmark section and read those vocabularies when they have leisure time. This section has indirectly eased the users by saving their time to look back at the search history. In addition, users can also label those words that they have remembered in the memorized section, while this section has provided users an opportunity to monitor their learning progression. Besides that, Dict Box application also allowed users to remove the words from the bookmark and memorized section.

Next, besides managing vocabularies, word reminder is also a superb development idea in this application. Many people reject it to get proper coordination. Regardless about what sort of crazy chaos, their lives are on a regular basis and they know themselves well enough to realize that they would inevitably forget to carry their lists with them when it matters most. Hence, this feature has allowed users to set their target number of words to learn per day in order to prevent users from forgetting and ensure the user’s learning progression is on the right path. However, the pop out notification can be cumbersome as it only provides a button to save the vocabulary to the

bookmark section or memorized section instead of providing a cancel button to turn down the notification.

In brief, manage vocabularies is one of the good features that can be considered to implement in the development of trilingual mobile dictionary application. This is because it can provide a systematic environment for primary or secondary school students to learn new vocabularies, ultimately, it can also encourage or motivate the school students to fully utilize the mobile dictionary.

2.2.3 Malay-English Dictionary

Malay-English Dictionary is a free bilingual mobile dictionary application that supports translation and detailed word definitions between Malay and English. It was released on 24 September 2015 and published by a Russia company named AllDict.

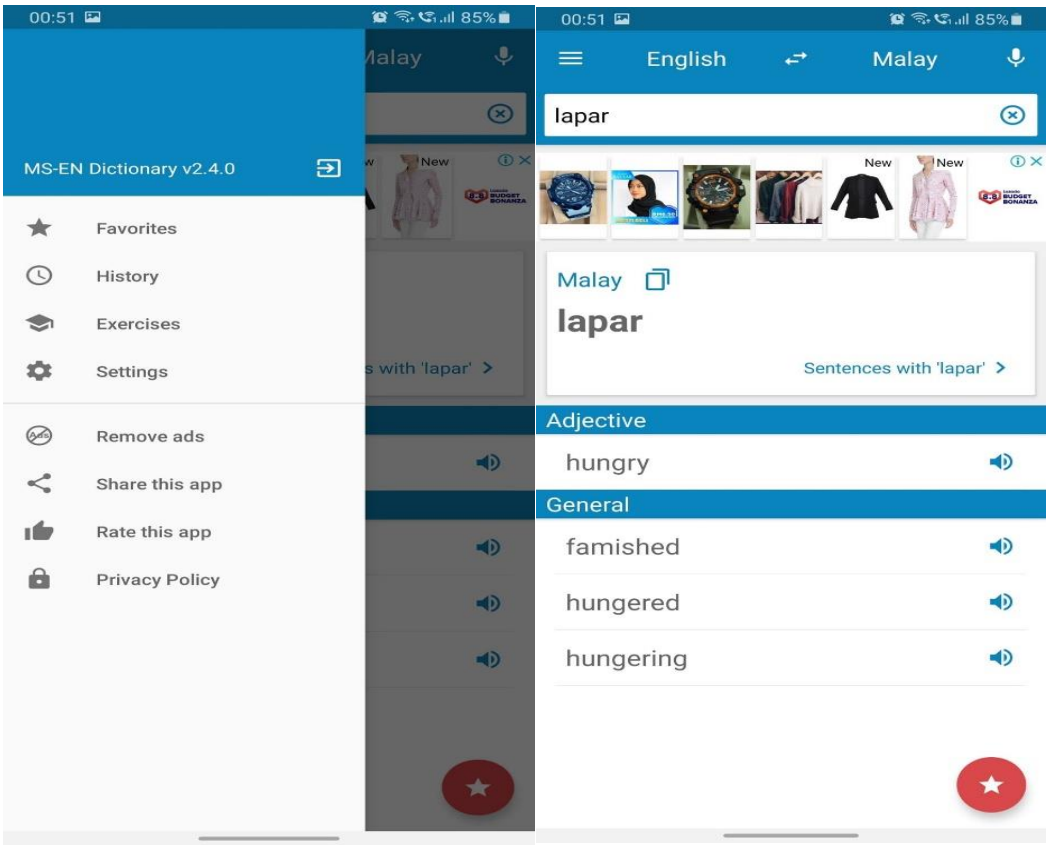


Figure 2.3: Interface of Malay-English Dictionary

2.2.3.1 Main Features of Malay-English Dictionary Mobile Application

- **Search and translate vocabularies**

  Users are capable of searching and translating the vocabularies between Malay and English. Besides that, Malay-English Dictionary application also allowed voice input from the users in order to translate the respective vocabulary.

- **Search history**

  When users are looking up their search history, Malay-English Dictionary application allows users to sort the history of searching by date, name and popularity in both ascending and descending order.

- **Exercise integrated with favourite list**

  Users are allowed to create different modules in the favourite list feature and add vocabularies to the modules that it belongs to. Not only that, a series of exercises regarding listening, writing and memorizing also provided and integrated with the favourite list that was created by the user.

- **Illustrate the details of a vocabulary**

  Malay-English Dictionary provided users the information such as word properties, pronunciation and sample sentences of a word.

2.2.3.2 Evaluation on Malay-English Dictionary Mobile Application

The emerging mobile application technology enables communities to access different learning resources at everywhere and any time. Malay-English Dictionary is a simple offline mobile dictionary application which provides some basic functions that normally see in most of the dictionary applications. Users of this application are not required to register and login the account before accessing the data of the dictionary. Next, users can choose to upgrade their membership if they wish to remove the pop-up advertisement that occurred in the application.

Besides the basic functions, developers have integrated exercises with their favourite list in this mobile dictionary application. When a user is interested in certain vocabulary, he or she can add that vocabulary to the modules in the favourite list, where the purpose of these modules which

created by the user is to categorize the favourited vocabulary into a more systematic and organized view. Instead of doing revision by looking at the vocabularies in the favourite list, users can actually choose to have listening, writing and memorizing exercises that are related to the vocabularies in the favourite list. As a result, this feature has definitely reduced the time of revision and it also eased the users to revise the vocabularies that they are intended to learn.

In addition, the general functions such as search history have been well improved by the developers of Malay-English Dictionary application. This is because Malay-English Dictionary application provided a search history feature with the help of sortation. Users may not always look up their history by following the date, sometimes they might follow the impression toward the vocabulary's spelling. In short, the sortation of date, popularity and name in this feature is very helpful for the users when they look back at the history, moreover, it could also fasten the process of checking history.

On the other hand, illustrating the details of a vocabulary is considered one of the good features in this application. However, in this mobile dictionary application, they only provide details such as word properties, pronunciation and translation between Malay and English, where these details are not enough for users to deeply understand the word. Furthermore, there are also mistakes in the generated sample sentences in this application, this error can easily mislead the users when they are learning new vocabularies. Ideally, this feature should contain extra data such as the meaning of the vocabulary, grammars and correct sample sentences in order to ensure that the users have better insight on the usage of a vocabulary.

In conclusion, although this mobile dictionary application does not consist plenty of functions, a modified feature such as illustrating the details of a vocabulary can be considered to implement in the trilingual mobile dictionary application development. This is because rich information allows users especially primary or secondary school students to have a strong foundation while learning new vocabulary. At the same time, it is important to ensure that students do not only understand the superficial meaning of the word.

2.2.4 Comparison Among the Reviewed Mobile Dictionary Application

Table 2.1: Comparison Among the Reviewed Mobile Dictionary Application

Mobile Application / **Features**	**Dictionary& Translator Free**	**Dict Box**	**Malay-English Dictionary**
Search and translate vocabulary	Yes	Yes	Yes
Present vocabularies' details	Yes	Yes	No
Attach note to a vocabulary	Yes	Yes	No
Add vocabulary into a phrasebook	Yes	Yes	Yes
Check history	Yes	Yes	Yes
Share vocabulary to social media	Yes	No	No
Quiz or exercise	Yes	No	Yes
Display daily new vocabulary	Yes	No	No
Translate paragraphs or sentences	Yes	No	No
Manage vocabularies	Yes	Yes	Yes
Flashcards	No	Yes	No
Word reminder	No	Yes	No
Free download	Yes	Yes	Yes
Web version	No	No	No

2.2.5 Conclusion

In brief, each mobile dictionary has brought up some unique and good features, where these features are playing different roles and integrating with each other. After a series of evaluation, clearly, Dictionary& Translator Free is considered as the most complete version of mobile dictionary. This is because the overall interface is detailed and user friendly where this component has eventually eased the users in looking up information. Besides general features, Dictionary& Translator Free has provided more features compared to Dict Box and Malay-English Dictionary, for example, features such as displaying daily new vocabulary, translating paragraphs or sentences and sharing vocabulary to social media.

Lastly, some features are chosen for the development of trilingual mobile dictionary application in order to encourage and motivate primary or secondary school students in learning new vocabulary and enhance their learning progression at the same time, while these features are shown as below:

1. Search and translate vocabulary
2. Add vocabulary into the phrasebook
3. Attach note to a word
4. Display daily new vocabulary

2.3 Review on Software Development Methodologies

In order to handle a project successfully, the developer has to select a software development methodology that can fit better with the project at hand. All software development methodologies have various strengths and limitations which contribute for specific purposes. The four software development models will be discussed in this section are waterfall methodology, rapid application development methodology, prototype methodology and agile methodology.

2.3.1 Waterfall Development

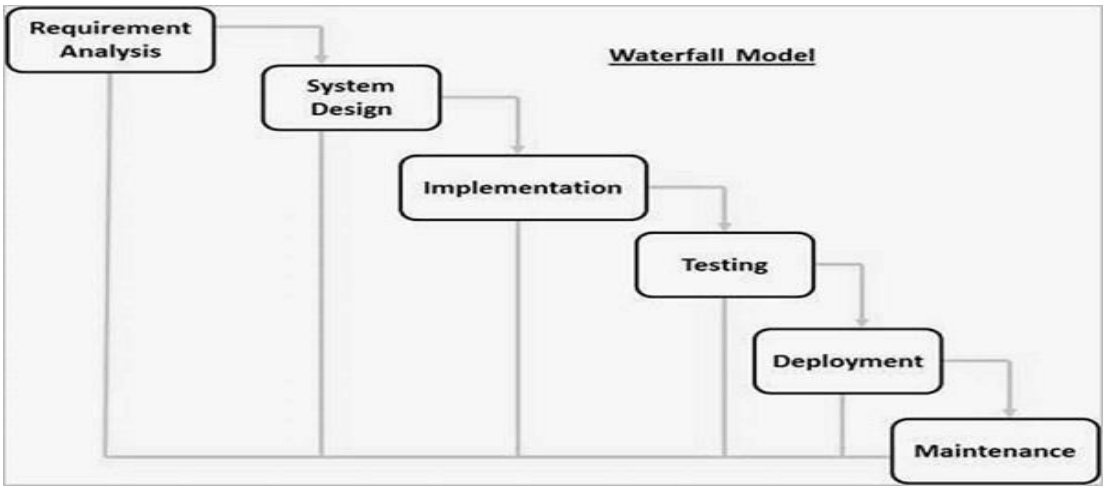


Figure 2.4: Waterfall Methodology
(SDLC - Waterfall Model - Tutorialspoint, 2020)

The first implemented software development process was waterfall methodology, it is often considered as a paradigm of linear-sequential life-cycle (Dutta, 2018). In waterfall methodology, the main focus will be centred on the sequence of phases such as criteria identification, system design,

solution developing, software testing, implementation and maintenance. However, waterfall methodology is usually defined as ‘distinct and independent for specification and growth process’, which means each step must be confirmed that it is completed before the next step can begin in order to ensure that there is no overlapping of phases happening when using waterfall methodology (Eason, 2016). Besides that, at the end of each phase, there would be a review session to evaluate if the project is on the right track and determine whether the project should proceed or scrap. Although waterfall methodology is not popular in software development environments, it is still useful when the requirements are fixed and unambiguous. There are several strengths and weaknesses when developers using waterfall methodology:

Table 2.2: Strengths and Weaknesses of Waterfall Methodology

Strengths	**Weaknesses**
It is simple, understandable and easy to use as the phases in waterfall methodology is straight-forward and overlapping between phases is also prohibited.	Phases in waterfall methodology is not iterative, which means when a client wants to change requirements after requirement phase the whole project has to be redo again. Also, it will lead to severe waste of time and cost to the development team.
Despite the model’s rigidity, the project will become simpler to handle. For example, there is always a review session at the end of each phase in order to make sure that the project is on the right path.	Waterfall methodology is not appropriate for the project which contained moderate to high changing probabilities on specification, this is because the waterfall model cannot adapt to changing demands.
Waterfall methodology plays best with smaller projects where requirements are well understood.	Integration in waterfall methodology is achieved as a “big-bang” at the very end, hence, developers are unable to detect any technical and market bottlenecks or obstacles in the early phase.

2.3.2 Rapid Application Development Methodology

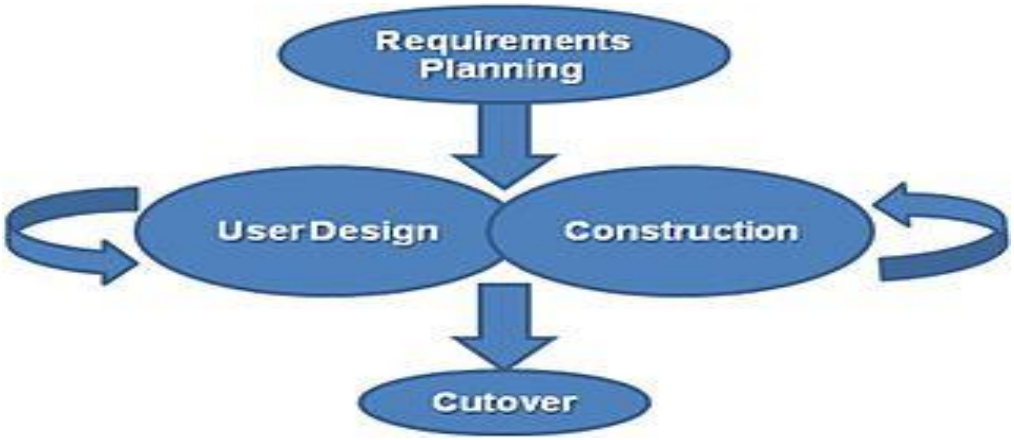


Figure 2.5: Rapid Application Development Methodology
(Rapid Application Development (RAD) - Simplifier, 2020)

Rapid Application Development (RAD) is a technique for designing software which only involves minimal prototyping preparation. The planning for building applications is performed by using RAD and it is fitted with all the program writing itself (Hirschberg, 2016). In rapid application development methodology, the main focus will be centred on several stages such as requirement planning stage, user design stage, construction stage and cutover stage. Furthermore, RAD promotes quicker development in software and aims to reduce the implementation expenses and improve product consistency at the same time. This technique is achieved through a chain of verified methods designed to develop the software in a distinct line (Naz and Khan, 2015). Since, detailed pre-planning is not required, RAD has significantly shortened the time of software development compared to structured or traditional methodologies. In brief, RAD is the methodology that speeds up the time of software development cycle and allows developers to produce quality software at the same time. In addition, there are several strengths and weaknesses when developers using rapid application development methodology:

Table 2.3: Strengths and Weaknesses of RAD Methodology

Strengths	Weaknesses
RAD methodology is able to accommodate changes as iteration of	The developing software has to be broad enough in order to split into

phases often occurred when this methodology is applied. In addition, clients are allowed to change their mind or idea towards the requirement of the software from time to time in order to achieve their satisfactory level.	multiple modules. If the software modules are little and fixed, the RAD methodology will not be the prior choice for such software development.
All the project prototypes generated can be stored in a repository for future use. The characteristic of re-usability of the materials able to increase the efficiency of the software development process.	In RAD methodology, development and accustomed issues are difficult to track because there is only little evidence or documentation to show what was accomplished.
In RAD methodology, the participants such as the project sponsor have a strong and continuous involvement that keeps providing feedback throughout the entire software development process. Therefore, the degree of satisfaction and acceptance of end users is higher when the end product is generated.	RAD methodology is not an ideal methodology when the technical risks are high. This could be arising as the new project is utilizing the latest technology or when a new project demands a strong degree of interoperability with an established framework.

2.3.3 Prototyping Methodology

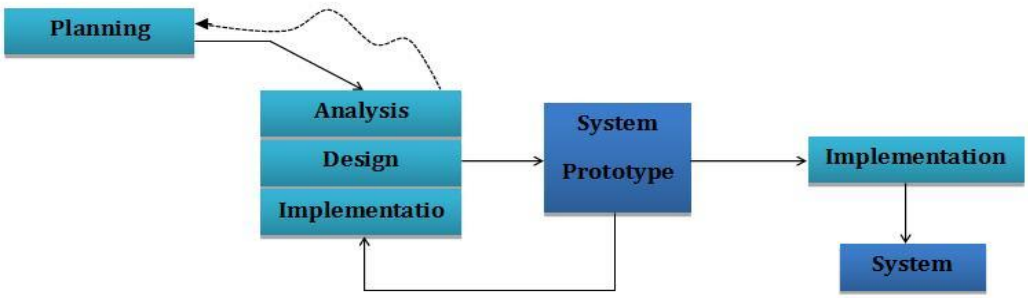


Figure 2.6: Prototyping Methodology

(Prototyping Methodology – The WritePass Journal : The WritePass Journal, 2020)

One of the most commonly used SDLC models is the prototyping methodology. Normally, this methodology will be implemented when clients themselves do not realize the precise project specification beforehand (Software Engineering | Prototyping Model, 2018). In prototyping methodology, the main focus will be centred on several phases such as requirement planning phase, iteration phase which includes develop, refine and demonstrate product, testing phase and implementation phase. Moreover, prototyping can also describe the process of building a practical replication of an application or software that has to be developed. In other word, prototyping methodology foster efficient software development by breaking down a complicated trouble and problem that sometimes labelled as an unknown issue into many detailed but smaller and simpler pieces, then, develop and subsequently optimize the software until it is able to fulfil end-users' preferences (Carr and Verner, 1997). Next, there are several strengths and weaknesses when developers using prototyping methodology:

Table 2.4: Strengths and Weaknesses of Prototyping Methodology

Strengths	**Weaknesses**
End-users are given opportunities to test the developed software and provide valuable feedback throughout the entire development process. By doing this, user requirements towards the software are always up to date and the development process will be on the correct path.	In prototyping methodology, documentation regarding software development can sometimes be missing or incomplete, this is because the primary objective is on the developing the prototype of the application.
Prototyping methodology helps to spot the defects and omissions at the early stage, especially when developers and stakeholder test and refine the software together.	There might not be any formal end-of-phase reviews in the prototyping methodology. Hence, it is very difficult to implement the scopes of the prototype and the project never seems to finish. Practically, this model may raise the software's

	complexity, this is because scope of the application can expand beyond the initial plan.
The iterative process in prototyping methodology has allowed absence of detailed requirements from clients. This is because extra features or functions can be added into the system in the middle of software development. Besides that, with solutions evolving, clients are becoming increasingly optimistic about the progression and the outcomes.	Efficiency, security, backup and recovery of the system might be ignored due to the reason of hasting in developing the prototype of the application. Ultimately, the consequences could be costly for the organization.

2.3.4 Agile Methodology

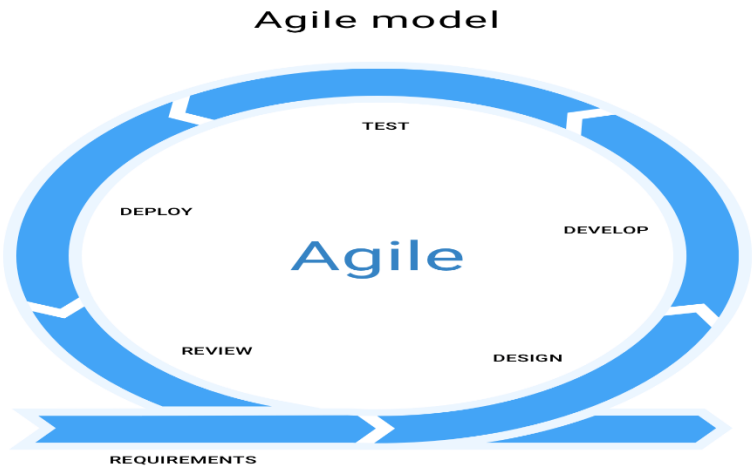


Figure 2.7: Agile Methodology

(Feoktistov, 2019)

Agile methodology is mainly an alternative response to waterfall methodology, where traditional methodologies encounter challenges when developing the software. The main objective of agile methodology is to reduce the overheads such as documentation in the development cycle, not only that, it also aimed to focus on programming rather than designing. One of the reasons that agile

methodology may not need to retain comprehensive paperwork to hand over at the deliver time is because the potential end-users and stakeholders have already experienced their system functionalities by involving themselves at the development cycle and having good communication between the developer team (Khalil and Kotaiah, 2018). In addition, customer interaction, gradual implementation, developers' ability to implement innovative operating practices, change management and simplicity are the basic core of agile development (Sunner, 2017). In agile methodology, the main focus will be centred on several phases, where these phases are exactly the same as the one in waterfall methodology, however, iteration of phases is adopted in the agile model instead of supporting only one-direction path as in waterfall methodology. On the other hand, there are several strengths and weaknesses when developers using agile methodology:

Table 2.5: Strengths and Weaknesses of Agile Methodology

Strengths	**Weaknesses**
Agile methodology makes early and predictable delivery possible. New updates with strong predictability are introduced rapidly and regularly if time-boxed, fixed schedule Sprints of week1 to week 4 are being utilized. Besides that, it also offers an opportunity for sooner launching or beta-test if there is sufficient market interest.	Agile methodology required more professional and qualified developers as it is an implementation-focused model. Experienced developers capable of handling the adjustments from time to time. In contrast, novice developers will require more resources in finding the solutions for a particular amendment.
Quality of the project can be well improved by using agile methodology. This is because the agile model divides the project into several manageable units in order to enable team members to concentrate on testing and high-quality	Agile methodology is not suitable for those customers who follow strictly on their own defined budget or timeline, otherwise, it will be cumbersome in the delivery process. This is because the organization would not know how much the

development. In fact, by working on regular builds and performing testing and evaluations after each cycle, quality of the software can be enhanced by identifying and correcting the existing errors.	development actually costs and what is the specific date of delivering the product at the initial stage.
In agile methodology, the high level of interaction between customers and project team can lead to greater satisfaction from clients. This is because the iterative characteristic in agile models enable users to change their desired requirements from time to time, therefore, clients are most likely to accept the established product when the development cycle comes to an end.	In agile methodology, documentation seems to get side tracked, therefore, it is more challenging for novice participants when they are trying to understand the progression.

2.3.5 Comparison Among the Studied Software Development Methodologies

Table 2.6: Comparison Among the Studied Software Development Methodology

Methodology / **Characteristic**	**Waterfall Methodology**	**Rapid Application Development Methodology**	**Prototyping methodology**	**Agile methodology**
Comprehensibility	Easy	Intermediate	Intermediate	Difficult
Detailed requirement	Initial stage	Middle stage	Middle stage	Middle stage
Adaptive to changes regularly	No	Yes	Yes	Yes

User involvement	Involved at initial stage	Highly involved at all stages	Highly involved at all stages	Highly involved at all stages
Development cost	Low	High	High	Very high
Iterative development	No	Yes	Yes	Yes

2.3.6 Conclusion

Each software development model has their own unique characteristics, strengths and weaknesses, it is important to implement the right software development methodology in order to strive the project objective before the deadline. After evaluating the four software development methodologies as stated above, prototyping methodology has chosen to be the approach in developing the trilingual mobile dictionary application. This is because there are limited requirements from users in the early phase, as time goes by, the prototype of the mobile application can play a role to inspire potential users in suggesting improvement on the existing features. Besides that, constant feedback from potential end-users is also an essential element when developing the mobile dictionary application as primary school students may have different perspectives from the secondary school students. In addition, user involvement can also accelerate the detection of defects and omission in an early stage rather than solving all the errors in the testing stage. Lastly, prototyping methodology offers high adaptation to requirement changing, where this property will definitely lead the developer to build an application that is able to strive for end-user's satisfaction, however, developer must make sure the development is on the right path from time to time in order to prevent any delay.

2.4 Evaluation on SQL and NoSQL Database

Traditional relational database management system (RDBMS) for storage is built upon the relational paradigm, it is also commonly recognized as SQL databases (Fotache, 2013). However, non-relational databases have grown in prevalence significantly in the past few years. These databases are commonly

referred to as NoSQL databases which clearly distinguish them from the SQL databases.

2.4.1 Advantages and Disadvantages of SQL and NoSQL Database

The tables below will demonstrate the pros and cons of SQL and NoSQL:

Table 2.7: Advantages of SQL Database

SQL	
Advantages	
Characteristics	**Explanation**
Query Processing	SQL provides a fast and unbeatable speed of query processing where this makes it capable to retrieve large amounts of data effectively and efficiently. Queries such as insertion, deletion, update and creation are often performed with virtually no time (Feuerlicht, Pokorný and Richta, 2009).
Coding Skills	Huge number of lines of code are not necessary as no coding skills are needed for data management. Only some simple keywords such as INSERT, UPDATE, INTO and SELECT are used and even the syntactic rules in SQL are not complicated. Hence, people often call SQL a user-friendly language. Its code-free design offers trouble-free operations.
Standardized Language	Unlike NoSQL, there is no standardization problem in SQL. This is because SQL suits the ISO and ANSI guidelines which are internationally accepted (Eisenberg and Melton, 2000). Also, due to the documentation across the years since establishment, SQL offers a common forum worldwide for all its consumers.

Table 2.8: Disadvantages of SQL Database

SQL	
Disadvantages	
Characteristics	**Explanation**
Complex interface	Although there is no complex coding in SQL, SQL has a daunting interface that causes some users uneasy for interacting with the database (Charu, 2014).
Cost	Expenses are too costly for some SQL operating versions, where this has also made it difficult to bring the vendor-in(Parker, Poe and Vrbsky, 2013).
Partial Control	A complete control over the database is not given to the programmer due to some implicit rules and requirements.

Table 2.9: Advantages of NoSQL Database

NoSQL	
Advantages	
Characteristics	**Explanation**
High Scalability	NoSQL database allows horizontal scaling for almost unlimited growth, while horizontal scaling means adding more servers to process the data. Due to the high scalability, NoSQL is easy to implement and able to handle massive amounts of data, furthermore, it also scales itself to manage the data which grows efficiently over time.
Big Data	As the transaction rates have increased dramatically, there has also been a significant rise in the amount of data being processed. RDBMS capability has improved due to these increases, but the limitations of data volumes that can be effectively handled by a single RDBMS are becoming unsustainable for certain organizations as with transaction rates (Raut, 2017). In contrast, NoSQL databases are allowed to

	store and manage huge volumes of data due to its high scalability and flexibility. Moreover, data can also be stored and handled easily nowadays with the simple design of NoSQL models. Today, the amounts of massive data that NoSQL structures can manage has outstripped even the biggest RDBMS can accommodate.
Economical	NoSQL databases usually use inexpensive generic server clusters to handle the overflowing amounts of data and transactions, whereas RDBMS prefers to focus on costly specialized servers and storage systems. The consequences are that the costs of transaction per second in NoSQL may be several times cheaper than RDBMS costs, meaning people can store and process more data at a far lower price.

Table 2.10: Disadvantages of NoSQL Database

NoSQL	
Disadvantages	
Characteristics	**Explanation**
Less Support	As NoSQL is still new in this era, there is very little community support (Binani, Gutti, & Upadhyay, 2016). Although it is open source, some considerations might be brought up when people are using the NoSQL databases. For example, whether the unlimited competent support is available for 24 hours if the database system failed. However, this is not guaranteed by the NoSQL vendors.
Management Issue	The data management in NoSQL is more complex than RDBMS as it lacks a standardized platform like SQL. NoSQL has limited query and absence of joins as all data are denormalized. In addition, NoSQL stores unstructured data, thus, it will be more

	challenging and hectic in order to handle it on a daily basis.
Integrity problems	Ensuring the integrity of the data requires extensive programming manually and it is not a DBMS feature (María, 2020). While, the "integrity" here is referring to the point of view of the restrictions from domain, referential of null value etc.

2.4.2 Conclusion

There are big variations between SQL and NoSQL databases as they are used on the basis of their features and characteristics in various kinds of systems. For example, SQL database is used when the data consistency is strongly emphasized. Also, SQL database often manages the data in an organized or well-structured way, hence, it is best adapted to a system that has detailed and interrelated details. In contrast, a NoSQL database will be utilized when the system is large and consists of a lot of 'null' data where it can also be recognized as unstructured data. In comparison, NoSQL databases have stronger performance and scalability than SQL databases. On the other hand, SQL databases support ACID property instead of CAP theorem by NoSQL database, where the 'C' for both ACID and CAP indicates the term 'consistency'. In SQL databases, consistency is important and also seen as a prior criterion in order to perform transactions, this also means that the transaction of data will be terminated if the data consistency is not achieved. However, in NoSQL databases, consistency may not be the prior requirement for continuing the data transaction as CAP theorem accepts temporary inconsistency of data when the data transaction is ongoing. In brief, consistency is vital for the trilingual mobile dictionary application as end-users will not want to retrieve the wrong translation or definition of a word, besides that, a structured database is required in order to handle the relationship of data effectively. Therefore, SQL database will be selected for storing all the data due to its hierarchical, linear and ordered and ACID properties.

2.5 Dictionary APIs

The API is commonly known as “Application Programming Interface”, it is an environment for technical advancement that requires access to the application or platform of another entity. In order to exchange information between device and API, a series of processes such as machine-to-machine interface for receiving and sending data via HTTP requests must be carried out. For example, in mobile dictionary development, an application programming interface is presented in another form to allow users to interact with the dictionaries regardless of the operating machine, hardware or software being used. In addition, it is possible to provide the API’s with libraries of tools that are widely known in the art of dictionary creation. In this way, the capacity of a given sub-language is built and accumulated over time based on the words observed and interests defined by individual users, user classes, regions or areas. Hence, the dictionary development interface of the current innovation will promote the peer-to-peer networking of advanced dictionary resources (Robert, 2003).

There are a lot of famous dictionary APIs nowadays such as Oxford Dictionary API, Merriam-Webster API, WordsAPI, Urban Dictionary API and more. However, most of the dictionary APIs that are available in the market have several basic features such as allowing users to integrate word definitions, synonyms, translations, audio pronunciation and more. In fact, these APIs are the digital glue which links the dictionary apps and services or access over the internet. Based on the aspects from fares, technologies and features, Oxford Dictionary API and Words API have been chosen to be implemented in the development of trilingual mobile dictionary application. This is mainly because the Oxford Dictionary API is the only API that is able to provide the get translation method which covers Malay, English and Mandarin while the Words API is the largest lexical database for English. On the other hand, Oxford Dictionary API and Words API are developed by following the RESTful principles where this principle will provide some advantages in terms of scalability, flexibility, portability and independency.

From the perspective of fares, the charges that Oxford offered is reasonable when comparing to the other dictionary APIs. Besides making the

developer easily and rapidly integrate the lexical data into their applications via a single API, Oxford Dictionary API also pre-processed the data and continually updates the information so that the users is able to retrieve the accurate linguistic resources. Last but not least, Oxford Dictionary API is the gateway to the radical new lexical engine and platform (LEAP) that optimizes and semantically connects the Oxford dictionaries data. The semantic relation also helps users to quickly cross reference lexical knowledge, opening up countless possibilities for using the information, from basic word searches to the processing of natural language and artificial intelligence (Oxford Dictionaries API, 2021).

In the Words API, WordNet is used to retrieve the word meaning and relationship while the CMU Pronouncing Dictionary is used to determine the syllable and rhymes of the word. WordNet is a broad lexical resource for English that operates in parallel with the Cambridge Dictionary. To be specific, it is made up of cognitive synonyms (synsets) which are the collections of verbs, nouns, adverbs and adjectives that each express a certain meaning. Synsets are often connected through logical, textual and lexical relationships. Furthermore, WordNet is somewhat similar to a thesaurus that categorize words together depending on their meanings. However, some distinction must be made before the classification may begin. In a nutshell, Words API is an intelligent platform for filtering the critical details that the users need.

CHAPTER 3

METHODOLOGY AND WORK PLAN

3.1 Introduction

There are four main discussions in this chapter which included detailed workflows in each software development methodology phase, research method to elicit requirements from users, development tools used for developing the trilingual mobile dictionary application and project plan.

3.2 Software Development Methodology

The implementation of a software development model has often been seen as a valuable and supportive framework that allows the developer to organize, schedule and manage the development of IT operation, services and product. Evolutionary prototyping methodology has been selected to implement in this project as the characteristics of this methodology cope better with the development of the application.

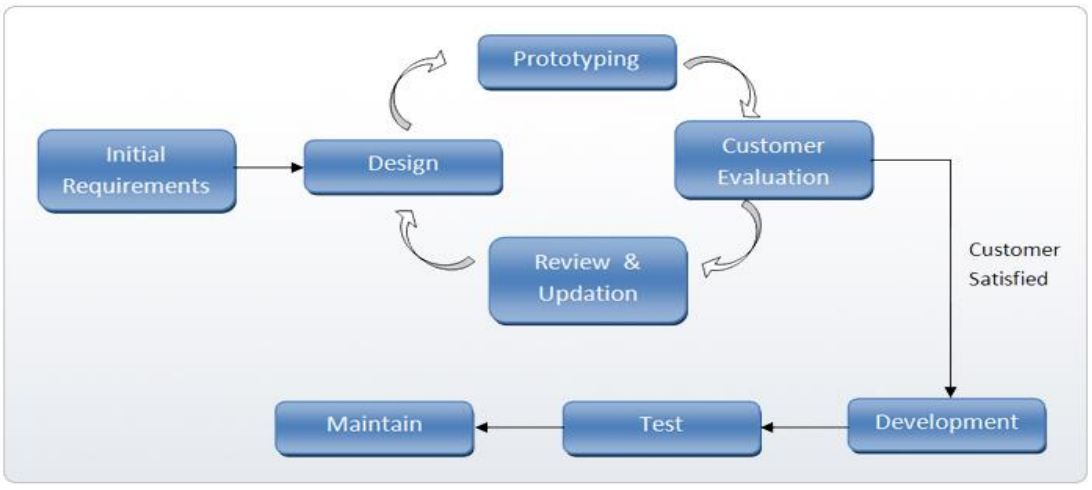


Figure 3.1: Evolutionary Prototyping Methodology

(Evolutionary Prototyping Methodology - Welcome to Shopper's Mart, 2020)

3.2.1 Initial Requirement

In this phase, the requirements are collected or gathered by conducting a series of studies and evaluations on the similar applications available in the market.

Throughout the evaluation and literature review, general features that are offered by similar applications can be recognized, moreover, weaknesses and innovative ideas can be improved and implemented into the development of trilingual mobile dictionary application. Apart from literature review, quantitative research methodology such as surveys are also carried out by distributing the questionnaire in order to understand the primary and secondary school students' preferences and problems they met when they are using paper dictionaries or mobile dictionaries that are currently available in the market. Next, functional and non-functional requirements are identified, and the supervisor is then reviewed and signed off. Last but not least, use case modelling is also utilized for requirement elicitation, this is because use case diagram is capable to present a graphical view of the software system's requirements while use case description is able to further elaborate or discuss the graphical representation shown in use case diagram.

3.2.2 Development

During the development of the trilingual mobile dictionary application, there is a set of phases required to iterate three times before the product is considered as completed. These phases have included design, develop, user validation and review.

First Iteration

During the first iteration, the attention is focused on the development of general functionality in a trilingual mobile dictionary application, where the basic functionality is search and translate vocabularies. The search feature will also provide information such as word properties, grammar and sample sentences rather than just showing the meaning of the vocabulary. In this iteration, it has included four phases that shown as below:

1. **Design Phase**

   The interface of the application and the design of search and translate function is sketched in draw.io. Once the interface for general function has been done, a screen flow for the sketches is linked so that it could provide a clearer picture when we are building the prototype.

2. **Develop Prototype Phase**

The prototype is built with the aid of Flutter framework. Sketches of interface and screen flows have become a reference when developing the prototype of the trilingual mobile dictionary application. Once the front end for search and translate function has done, the process will proceed to develop the back-end of the application.

3. **Evaluation Phase**

   After the first version of the prototype has been developed, a mixture of five primary and secondary school students are invited to assess the prototype. During the evaluation session, a short survey form is distributed to each potential end-user so that they can write down the weaknesses of the application. Feedback from the users who tested the system are recorded, organized and prepared to review on the next phase.

4. **Review Phase**

   After reviewing and analysing the comments from evaluators, the respective amendments have been performed on the current prototype in order to meet customer's expectations. Once the refined prototype is accepted by the evaluators, it will proceed to the second iteration of the development phase.

Second Iteration

After the refinement on the first iteration, the second iteration is heeded to the necessary function in a trilingual mobile dictionary application, where this necessary function is bookmark vocabularies. The bookmark feature is always manageable (add, edit, delete) by the user. In this iteration, it has involved four phases that shown as below:

1. **Design Phase**

   The bookmark feature is designed with the support of draw.io. After sketching the interface, the screen flows for the related interface are produced and act as an illustrator for the developer to develop the prototype.

2. **Develop Prototype Phase**

   Flutter framework is used to develop the prototype. Meanwhile, the prototype is built based on the GUI and interface movement drawings

that were designed during the previous phase. When the front-end of both features are completed, it will begin the back-end development process.

3. **Evaluation Phase**

   Another five primary and secondary school students are invited to evaluate the prototype. During the evaluation session, a feedback form is distributed to each potential end-user so that they can write down the limitations of the application. Feedback from the evaluators are recorded, organized and prepared to review on the next phase.

4. **Review Phase**

   Upon evaluating and examining the evaluators' feedback, the respective changes were carried out on the current prototype to satisfy the requirements of the potential end user. Then, it will proceed to the last iteration when the prototype is accepted by the evaluators.

Third iteration

Continuing from the second iteration, the attention now is centred on the development of additional functionalities in a trilingual mobile dictionary application, where these additional functionalities are attaching notes to a word and displaying daily new vocabulary. In this iteration, it has involved four phases that shown as below:

1. **Design Phase**

   After the enhancement of the previous iteration, two additional features are introduced in this phase. While, these features are attaching note to a word and displaying daily new vocabulary. The interface and screen flows of the additional features are sketched by using draw.io.

2. **Develop Prototype Phase**

   The prototype is developed using Flutter framework. In the meantime, the prototype is built based on the interface and screen flows of the sketches developed during the preceding phase. Back-end development of the mobile dictionary application will begin if the front-end development of both additional features is completed.

3. **Evaluation Phase**

Similar to the first and second iteration, five primary and secondary school students are selected to evaluate the prototype. They are requested to jot down the vulnerabilities of the prototype in a survey form that has been distributed by the developer. Comments from the evaluators are recorded, organized and prepared to review on the next phase.

4. **Review Phase**

   After reviewing and assessing the evaluators' feedback, the corresponding changes were made to the current prototype in order to fulfil the requirements of the potential end-user. Lastly, after the completion of the third version prototype, it will be refined into the final system and proceed to the testing phase.

3.2.3 Testing

Before delivering the product, various testing approaches have been carried out to ensure that the developed mobile dictionary application is able to accomplish the goal, objectives and scopes that have been defined earlier. Firstly, unit testing is conducted to test the components in the mobile application one after another with the help of frameworks such as stubs, driver, and mocks. The primary aim of this test is to verify that each feature of the mobile application performs as expected. Next, integration testing is carried out when individual components are integrated and tested as a combination by using test stubs and test driver. This degree of testing has the intention of identifying faults or errors in the interaction between integrated components. When the integration testing is completed, system testing is performed to test the completed and integrated software, where the objective of this test is to determine the conformity of the software with the stated specifications. Lastly, in order to examine the system's compatibility with market criteria and to determine its acceptability from users, usability testing and user acceptance testing are executed. A total of ten potential end-users are invited to test the mobile dictionary application and they are requested to rate and fill up the weaknesses or limitations of the application in the evaluation form. Once all

kinds of testing are done, the system is ready to deploy or deliver to the end-user.

3.3 Research Methodology

In this project, questionnaire is the quantitative research methodology that has been implemented to collect statistically relevant knowledge, feedback and ideas from the potential end-users. Normally, questionnaires are widely utilized in quantitative marketing and social analysis. It is the cheapest and critical tool to collect a wide spectrum of data from a vast number of respondents. Additionally, pre-testing within a smaller group of respondents is a helpful approach for testing a questionnaire and ensuring that the intended data is correctly collected. Therefore, a questionnaire which consists a total of 12 questions are prepared and 35 questionnaires were distributed to the target users which are primary and secondary school students.

3.3.1 Result of Facts Finding and Data Analysis

Questionnaire is one of the approaches which easily help the developer to understand the ideas and expectations from the potential end user. Therefore, the outcomes of the facts finding and data analysis are further analysed and discussed in the Section 4.4 which is allocated in the Chapter 4.

3.4 Development Tools

Since this project is related to mobile application development, thus, mobile application development tools are required in order to build the trilingual mobile dictionary application.

3.4.1 Programming Languages

A programming language is a vocabulary and collection of grammatical rules or instruction to execute certain tasks on a computer or smartphone, moreover, it is also an essential element for building any system. There are some considerations that need to be taken when a developer is deciding what programming languages are suitable for the dictionary mobile application development. For example, the support of documentation, degree of familiarity

toward the particular programming language, degree of difficulty to learn the particular programming language and also the performance of the programming language.

3.4.1.1 Dart

Dart is a client-oriented programming language that can be used to create quick applications on both mobile and web platforms. Its mission is to provide the productive programming language for cross-platform development as well as a versatile runtime platform for the applications framework. On the other hand, Dart is also the backbone of Flutter. Besides powering the Flutter applications with its language and runtime, it also helps developers with formatting, evaluating and testing the source codes.

3.4.1.2 SQL

SQL is also known as Structured Query Language, it is a fourth-generation language or basic language used to store, manage and extract data in the relational database management system (RDBMS). In other words, SQL is widely used whenever there is a requirement for creating, requesting, updating and deleting the data in the software's functions.

3.4.1.3 Ruby

Ruby is an object-oriented programming language that offers the flexible and descriptive characteristics (Ruby Programming Language, 2020). Everything in Ruby is an entity excluding the blocks, however, there are also substitutes for it too. Although it may not be the quickest language to build a web API but it is still a good option to use. This is because apart from it being fantastic to code in Ruby, there are also tons of excellent tools to specifically build the web APIs from testing to routing.

3.4.2 Integrated Development Environment (IDE)

An Integrated Development Environment (IDE) permits programmers to combine numerous aspects of coding. It improves programmer efficiency by integrating common software programming tasks into one application or

software, for example, programmers can edit the source code, build executable and debug errors within an IDE.

3.4.2.1 Visual Studio Code

There are several IDEs available in the market such as IntelliJ IDEA, Microsoft Visual Studio, NetBeans and more, however, visual studio code is chosen to implement in this project due to it eases programmer in edit, build and debug the codes. Additionally, visual studio code blends the flexibility of a source code with the efficient developer tool such as IntelliSense in order to support code completion and debugging, code refactoring and rich semantic source code understanding and navigation.

3.4.2.2 Framework

A software framework is a platform to facilitate the development of the software. Besides that, it is also a software library that offers a foundational structure to enable the developer to build the application for a particular environment.

3.4.2.3 Flutter

Flutter is an open-source framework that is responsible for the front-end of mobile application development. Moreover, it allows developers to build unique and entertaining smartphone applications by using Dart only. Most of the time, developers have to learn Swift and Android Studio in order to build an application on both platforms, however, Flutter framework enabled developer to code once and get two products at the same time, while this cross-platform usage has definitely helped developer to save a lot of time and resources for the development.

3.4.2.4 Ruby on Rails

Ruby on Rails is an open-source framework that is responsible for the back-end of mobile application development. It follows the concepts or rules such as CRUD (Create, Read, Update, Delete), KISAP (Keep It Simple as Possible), REST (Representational State Transfer), DRY (Don’t Repeat Yourself) and

COC (Convention on Configuration), these adoptions made developer to work on lesser codes but achieve more tasks within a short period (Ruby on Rails - Introduction - Tutorialspoint, 2020). With Ruby and Rails, making improvements to the existing code or introducing new functionalities to software is easy. Furthermore, amendments on data models can also be performed rapidly and easily even after the application has been launched.

3.4.3 Database

Typically, a database is managed by Database Management System (DBMS) and a database is a set of structured records while these records can be conveniently accessed, maintained and manipulated from time to time.

3.4.3.1 SQLite

SQLite is a program library that provides a relational database management system (RDBMS). The “lite” has indicated lightweight in terms of the setup or configuration, database administration and the required resources (SQLite Home Page, 2020). Furthermore, there are also some significant and noticeable characteristics in SQLite, for example, serverless, transactional, zero-configuration and self-contained.

3.4.3.2 Application Programming Interface (API)

An application programming interface (API) is a way to communicate programmatically with a specific feature or tool for the application.

3.4.3.3 Dictionaries API

Dictionaries API enables developers to browse some of the world’s largest word-data databases in all languages. While this API allows developers to integrate the word definition, translations, word properties and even audio pronunciation in the mobile dictionary application files (Top 6 Best Dictionary APIs (for Developers in 2020) [38+ Reviewed], 2020).

3.4.4 Version Control System

Version control is also known as source control, it is a maintenance of text revisions, programming applications and data collection. A timestamp is correlated with increasing modification and these modifications can be compared, updated and merged with others.

3.4.4.1 Git

A version control system such as Git is a type of software equipment that supports a developer to handle the source code modification from time to time. Since there will be a lot of amendments or versions of source code throughout the development of the mobile application, Git helps track all modifications to the source code in a special kind of database. If an error occurs, developers are able to turn back the clock and refer to older versions of the source codes in order to perform a better correction on the mistakes while mitigating disruptions to the developer. In short, the source code is a collection of useful information and comprehension regarding the issue domain gathered and optimized with diligent effort by the developer, thus, the version control system plays a role to preserve the source codes from both devastation and accidental loss due to human errors as well as unintentional outcomes.

3.5 Workflow

3.5.1 Work Breakdown Structure

1.0 Final Year Project

- 1.1 Preliminary Planning
  - 1.1.1 Background Research
  - 1.1.2 Identify Problem Statements
  - 1.1.3 Identify Project Goal
  - 1.1.4 Identify Project Objectives
  - 1.1.5 Identify Solution for Development
  - 1.1.6 Identify Approach for Development
  - 1.1.7 Identify Project Scopes
- 1.2 Requirement Gathering
  - 1.2.1 Literature Review

1.2.1.1 Review on similar existing applications

1.2.1.2 Review on software methodologies

1.2.1.3 Review on databases

1.2.2 Methodology

1.2.2.1 Explain Implementation of Methodology

1.2.2.2 Determine Research Methodology

1.2.2.3 Determine Development Tools

1.2.2.4 Create Project Plan

1.2.3 Project Specification

1.2.3.1 Determine Software Requirements

1.2.3.2 Develop Use Case Diagram

1.2.3.3 Develop Use Case Description

1.2.3.4 Facts Finding and Data Analysis

1.3 Prototype Development

1.3.1 First Iteration

1.3.1.1 Design

1.3.1.2 Prototyping

1.3.1.3 Customer Evaluation

1.3.1.4 Review and Updation

1.3.2 Second Iteration

1.3.2.1 Design

1.3.2.2 Prototyping

1.3.2.3 Customer Evaluation

1.3.2.4 Review and Updation

1.3.3 First Iteration

1.3.3.1 Design

1.3.3.2 Prototyping

1.3.3.3 Customer Evaluation

1.3.3.4 Review and Updation

1.4 Development

1.4.1 Integrate Prototype to Final System

1.5 Testing

1.5.1 Unit Testing

1.5.2 Integration Testing

1.5.3 System Testing

1.5.4 User Acceptance Testing

1.6 Deployment Phase

1.6.1 System Deployment

3.5.2 Work Planning

Table 3.1: Work Planning for Final Year Project.

Task Name	**Duration**	**Start**	**Finish**
Preliminary Phase	**21 days**	**Monday 15/06/2020**	**Sunday 05/07/2020**
Background Research	4 days	Monday 15/06/2020	Thursday 18/06/2020
Identify Problem Statements	3 days	Friday 19/06/2020	Sunday 21/06/2020
Identify Project Goal	2 days	Monday 22/06/2020	Tuesday 23/06/2020
Identify Project Objectives	2 days	Wednesday 24/06/2020	Thursday 25/06/2020
Identify Solution for Development	4 days	Friday 26/06/2020	Monday 29/06/2020
Identify Approach for Development	3 days	Tuesday 30/06/2020	Thursday 02/07/2020
Identify Project Scopes	3 days	Friday 03/07/2020	Sunday 05/07/2020
Requirement Gathering Phase	**56 days**	**Monday 06/07/2020**	**Sunday 30/08/2020**
Literature Review	**20 days**	**Monday 06/07/2020**	**Saturday 25/07/2020**
Review on similar existing applications	8 days	Monday 06/07/2020	Monday 13/07/2020
Review on software methodologies	8 days	Tuesday 14/07/2020	Tuesday 21/07/2020
Review on databases	4 days	Wednesday 22/07/2020	Saturday 25/07/2020
Methodology	**16 days**	**Sunday 26/07/2020**	**Monday 10/08/2020**
Explain Methodology Implementation	4 days	Sunday 26/07/2020	Wednesday 29/07/2020
Determine Research Methodology	4 days	Thursday 30/07/2020	Sunday 02/08/2020
Determine Development Tools	4 days	Monday 03/08/2020	Thursday 06/08/2020
Create Project Plan	4 days	Friday 07/08/2020	Monday 10/08/2020
Project Specification	**20 days**	**Tuesday 11/08/2020**	**Sunday 30/08/2020**
Determine Software Requirements	7 days	Tuesday 11/08/2020	Monday 17/08/2020
Develop Use Case Diagram	7 days	Tuesday 18/08/2020	Monday 24/08/2020
Develop Use Case Description	3 days	Tuesday 25/08/2020	Thursday 27/08/2020

Facts Finding and Data Analysis	3 days	Friday 28/08/2020	Sunday 30/08/2020
Prototype Development Phase	**196 days**	**Monday 31/08/2020**	**Sunday 14/03/2021**
First Iteration	**32 days**	**Monday 31/08/2020**	**Monday 18/01/2021**
Design Phase	7 days	Monday 31/08/2020	Sunday 06/09/2020
Prototyping Phase	16 days	Friday 01/01/2021	Saturday 16/01/2021
Customer Evaluation Phase	3 days	Sunday 17/01/2021	Tuesday 19/01/2021
Review and Updation Phase	6 days	Wednesday 20/01/2021	Monday 25/01/2021
Second Iteration	**31 days**	**Monday 31/08/2020**	**Thursday 18/02/2021**
Design Phase	7 days	Monday 31/08/2020	Sunday 06/09/2020
Prototyping Phase	15 days	Tuesday 26/01/2021	Tuesday 09/02/2021
Customer Evaluation Phase	3 days	Wednesday 10/02/2021	Friday 12/02/2021
Review and Updation Phase	6 days	Saturday 13/02/2020	Thursday 18/02/2021
Third Iteration	**31 days**	**Monday 31/08/2020**	**Sunday 14/03/2021**
Design Phase	7 days	Monday 31/08/2020	Sunday 06/09/2020
Prototyping Phase	15 days	Friday 19/02/2021	Friday 05/03/2021
Customer Evaluation Phase	3 days	Saturday 06/03/2021	Monday 08/03/2021
Review and Updation Phase	6 days	Tuesday 09/03/2021	Sunday 14/03/2021
Development Phase	**7 days**	**Monday 15/03/2021**	**Sunday 21/03/2021**
Integrate Prototype to Final System	7 days	Monday 15/03/2021	Sunday 21/03/2021
Testing Phase	**21 days**	**Monday 22/03/2021**	**Sunday 11/04/2021**
Unit Testing	7 days	Monday 22/03/2021	Sunday 28/03/2021
Integration Testing	7 days	Monday 29/03/2021	Sunday 04/04/2021
System Testing	4 days	Monday 05/04/2021	Thursday 08/04/2021
User Acceptance Testing	3 days	Monday 09/04/2021	Sunday 11/04/2021
Deployment Phase	**7 days**	**Monday 12/04/2021**	**Sunday 18/04/2021**
System Deployment	7 days	Monday 12/04/2021	Sunday 18/04/2021

3.5.3 Gantt Chart

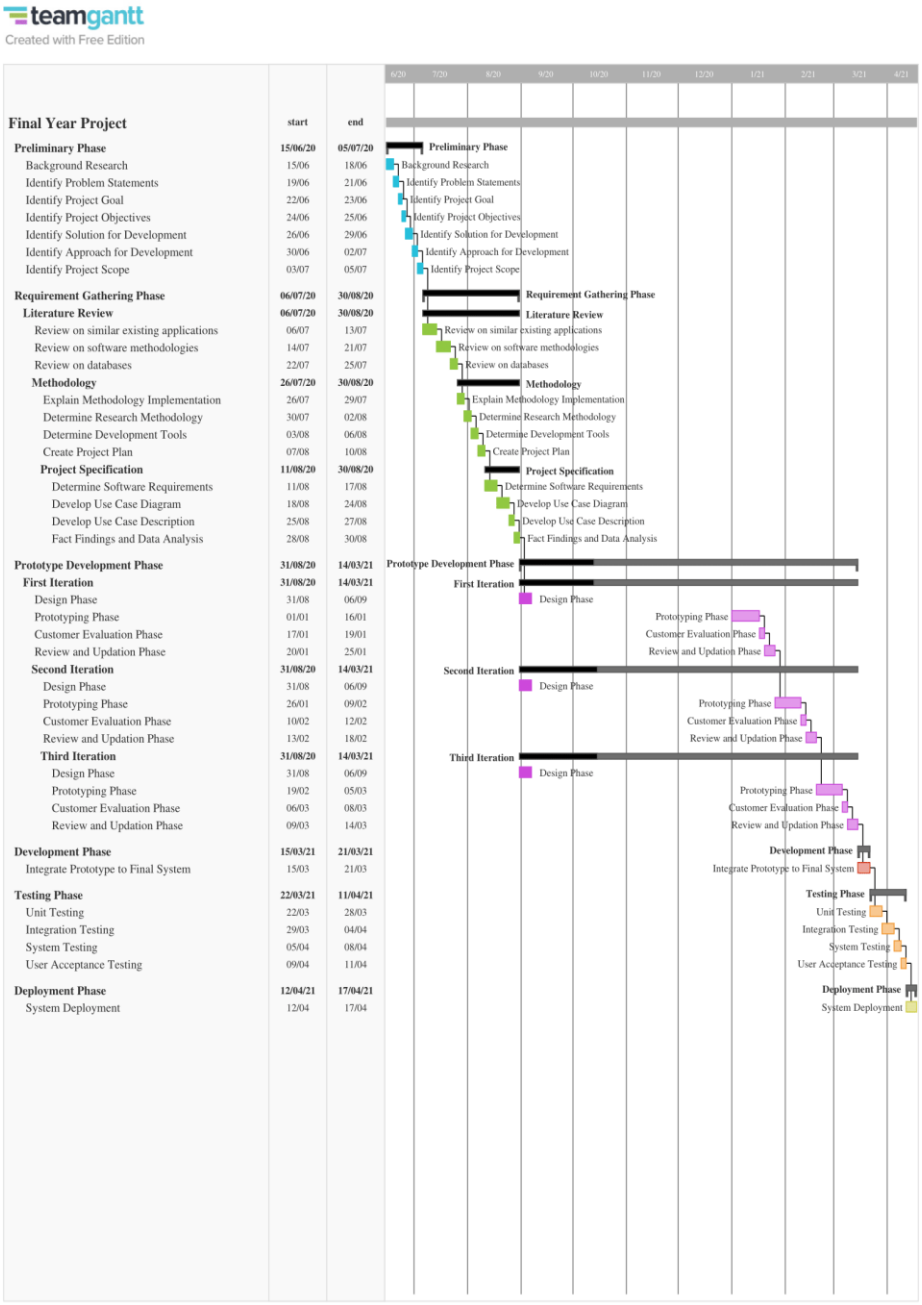


Figure 3.2: Gantt Chart Diagram for Final Year Project.

CHAPTER 4

PROJECT SPECIFICATION

4.1 Introduction

In this chapter, five sections are discussed and introduced based on the software requirements elicited and analysed from the public. Functional and non-functional requirements will first be determined in Section 4.2 while use case modelling of this project is introduced in Section 4.3. Next, Section 4.4 is the facts finding and data analysis based on the questionnaire that has been distributed earlier. Lastly, a simple sketch of the prototype is demonstrated in Section 4.5.

4.2 Project Requirement Specification

4.2.1 Functional Requirements

Functional requirements can also be interpreted as the support statements of services that the system will offer, as well as demonstrating how the system will respond to certain specific inputs and how the system will perform in such specific circumstances. In this project, the functional requirements of trilingual mobile dictionary application are as below:

Table 4.1: Functional Requirements

Functional Requirement ID	Functional Requirement Statement
FR01	The system shall allow users to search the vocabulary in English and translate it to Mandarin and Malay.
FR02	The system shall allow users to view the detailed information of the vocabulary such as word properties, grammars, related words and sample sentences.
FR03	The system shall allow users to add more than one vocabulary into the phrasebook.

FR04	The system shall allow users to edit the vocabulary's content or delete the vocabulary from the phrasebook.
FR05	The system shall allow users to attach notes or examples to vocabulary.
FR06	The system shall allow users to edit or delete the note or example that is attached along with the vocabulary.
FR07	The system shall allow users to view the daily new vocabulary on the main page of the application.
FR08	The system shall allow users to slide the daily new vocabulary on the main page of the application.
FR09	System administrator shall be able to update and maintain the vocabulary version.

4.2.2 Non-Functional Requirements

Non-functional requirements illustrate the limitations of the features or services offered by the system, where these limitations include time limits, restrictions on implementation method or specification and more. In this project, the non-functional requirements of trilingual mobile dictionary application are as below:

Table 4.2: Non-Functional Requirements

Non-Functional Requirement ID	Non-Functional Requirement Statement
NFR01	The system shall be built for Android platform.
NFR02	The system shall be operated in Android platform.
NFR03	The user shall be able to use the system functions without training.
NFR04	The system shall be operated 24 hours a day

	with the exception of downtime for the system to undergo maintenance or patching.
NFR05	The system shall allow users to use the system with internet connection.

4.3 Use Case Modelling

A use case modelling is done to provide a better explanation on the system's functionalities and also demonstrate the specific details regarding each feature on the system.

4.3.1 Use Case Diagram

Use case diagram is generated to display the functions offered for the system administrators and users.

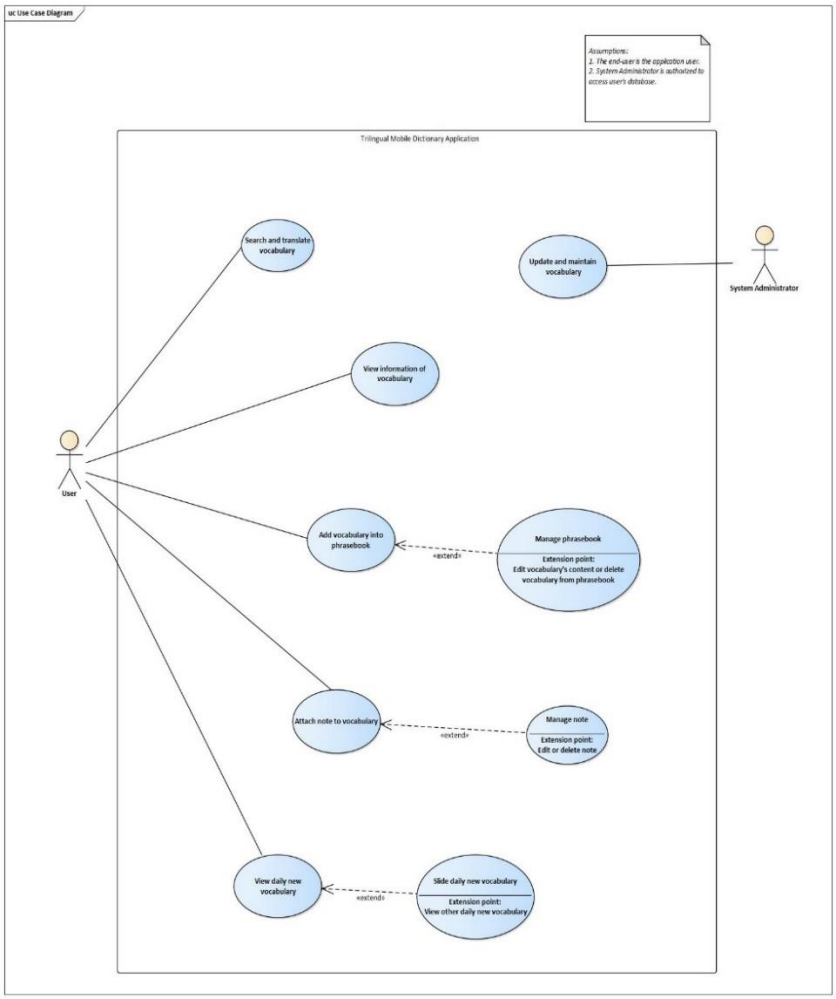


Figure 4.1: Use Case Diagram for Trilingual Mobile Dictionary Application

4.3.2 Use Case Description

4.3.2.1 Search and Translate Vocabulary

Use Case Name: Search and translate vocabulary	ID: UC01	Importance Level: High
Primary Actor: User	Use Case Type: Details, Essential	
Stakeholders and Interests: User- He/ she wants to search and translate vocabulary in the mobile dictionary application.		
Brief Description: Search and translate vocabulary use case describes how a user explores the vocabulary.		
Trigger: User wants to learn the meaning and translation of the particular vocabulary via the dictionary application.		
Relationships: Association: User Include: N/A Extend: N/A Generalization: N/A		
Normal Flow of Events: 1. User types the vocabulary in the search bar. 2. User presses “Enter” to search and translate the vocabulary. 3. System displays the meaning of the vocabulary in three languages. 4. System displays translated vocabulary in three languages.		
SubFlows: N/A		
Alternate/ Exceptional Flows: N/A		

4.3.2.2 View Information of Vocabulary

Use Case Name: View information of vocabulary	ID: UC02	Importance Level: High
Primary Actor: User	Use Case Type: Details, Essential	
Stakeholders and Interests: User- He/ she wants to know the detailed information regarding the particular		

vocabulary.
Brief Description: View information of vocabulary use case describes how a user explores the details of a vocabulary.
Trigger: User wants to learn the detailed information of the particular vocabulary via the dictionary application.
Relationships: Association: User Include: N/A Extend: N/A Generalization: N/A
Normal Flow of Events: 1. User types the vocabulary in the search bar. 2. User presses "Enter" to search the vocabulary. 3. System displays the word properties of the vocabulary. 4. System displays the sample sentences of the vocabulary. 5. System displays the grammars of the vocabulary. 6. System displays the related words of the vocabulary. 7. User presses "Cambridge Dictionary" to see different details of the word.
SubFlows: N/A
Alternate/ Exceptional Flows: N/A

4.3.2.3 Add Vocabulary into Phrasebook

Use Case Name: Add vocabulary into phrasebook	ID: UC03	Importance Level: High
Primary Actor: User	Use Case Type: Details, Essential	
Stakeholders and Interests: User- He/ she wants to bookmark a vocabulary.		
Brief Description: Add vocabulary into phrasebook use case describes how a user bookmarks the vocabulary.		

Trigger: User wants to bookmark the vocabulary for future reference.
Relationships: Association: User Include: N/A Extend: Manage phrasebook Generalization: N/A
Normal Flow of Events: 1. User key in the vocabulary in the search bar. 2. User presses "Enter" to search the vocabulary. 3. User bookmark the vocabulary into the phrasebook by pressing the disc logo. 4. System displays a message regarding the vocabulary has successfully been added to the phrasebook. Perform exceptional flow of 4a: Failed to bookmark. 5. User manages the vocabulary in the phrasebook. Perform sub flow of 5a: Remove vocabulary from the phrasebook. Perform sub flow of 6a: Edit the contents of the vocabulary in the phrasebook.
SubFlows: 5a. User is allowed to remove the unwanted vocabulary from the phrasebook. 6a. User is allowed to edit or modify the details of a word. Then, the disc icon must be pressed in order to save all changes.
Alternate/ Exceptional Flows: N/A 4a. System displays a message regarding the vocabulary has failed to be added into the phrasebook.

4.3.2.4 Attach Note to Vocabulary

Use Case Name: Attach note to vocabulary	ID: UC04	Importance Level: High
Primary Actor: User	Use Case Type: Details, Essential	
Stakeholders and Interests: User- He/ she wants to attach a note to a vocabulary.		

Brief Description: Attach note to vocabulary use case describes how a user attaches a note to the vocabulary.
Trigger: User wants to attach the note to a vocabulary in order to improve their understanding of the word.
Relationships: Association: User Include: N/A Extend: Manage note Generalization: N/A
Normal Flow of Events: 1. User key in the vocabulary in the search bar. 2. User presses "Enter" to search the vocabulary. 3. User press disc icon to save the word into phrasebook. 4. User find the word in the phrasebook and press the pencil icon in order to add a note. 5. User creates a note by typing the words or inserting the emoji at the "Example" field. 6. User presses the tick icon to save the note. 7. System displays a message regarding the note is successfully added. Perform exceptional flow of 6a: Failed to add note. 8. User manages the note attached along with the vocabulary. Perform sub flow of 7a: Manage note that has been attached along with the vocabulary.
SubFlows: 7a. User can edit and remove the note attached to the vocabulary from time to time.
Alternate/ Exceptional Flows: N/A 6a. System displays a message regarding the note has failed to be added.

4.3.2.5 View Daily New Vocabulary

Use Case Name: View daily new	ID: UC05	Importance

vocabulary		Level: Moderate
Primary Actor: User	Use Case Type: Details, Essential	
Stakeholders and Interests: User- He/ she wants to learn new vocabulary on a daily basis.		
Brief Description: View daily new vocabulary use case describes how a user views daily new vocabulary.		
Trigger: User wants to learn new vocabulary on a daily basis.		
Relationships: Association: User Include: N/A Extend: Refresh daily new vocabulary Generalization: N/A		
Normal Flow of Events: 1. User enters the main page of the application. 2. System recommends daily new vocabulary to the user. 3. User views the daily new vocabulary in the main page. Perform sub flow of 3a: User wants to view other vocabulary.		
SubFlows: 3a. User can view more daily new vocabulary by sliding in order to let the system display another vocabulary.		
Alternate/ Exceptional Flows: N/A		

4.3.2.6 Update and Maintain Vocabulary

Use Case Name: Update and maintain vocabulary	ID: UC06	Importance Level: High
Primary Actor: System administrator	Use Case Type: Details, Essential	
Stakeholders and Interests: System administrator- He wants to update and maintain the vocabularies.		
Brief Description: Update and maintain vocabulary use case describes how a system		

administrator updates and maintains the vocabulary.
Trigger: System administrator wants to update and maintain the latest version of vocabulary in the application.
Relationships: Association: System administrator Include: N/A Extend: N/A Generalization: N/A
Normal Flow of Events: 1. System administrator login to the server application. 2. System administrator performs checking on the version of the dictionary. 3. System administrator updates the version of the vocabulary. Perform sub flow of 3a: Update vocabulary 4. System administrator maintains the version of the vocabulary. 5. System administrator publishes the latest version of update.
SubFlows: 3a. System administrator create, delete or edit the content of the vocabulary.
Alternate/ Exceptional Flows: N/A

4.4 Fact Findings and Data Analysis

In this project, questionnaire is used as an approach to elicit and understand the requirements from the public user especially primary and secondary school students. The questionnaires have been distributed to the respondents via Google Form as it is cost effective and efficient in collecting the responses from the public. A total of 35 respondents from different age and educational backgrounds were involved and provided responses while these responses are used to assess the benefits of mobile dictionaries that have been brought into their life and also preference of users toward the mobile dictionary application.

4.4.1 Responses on Questions

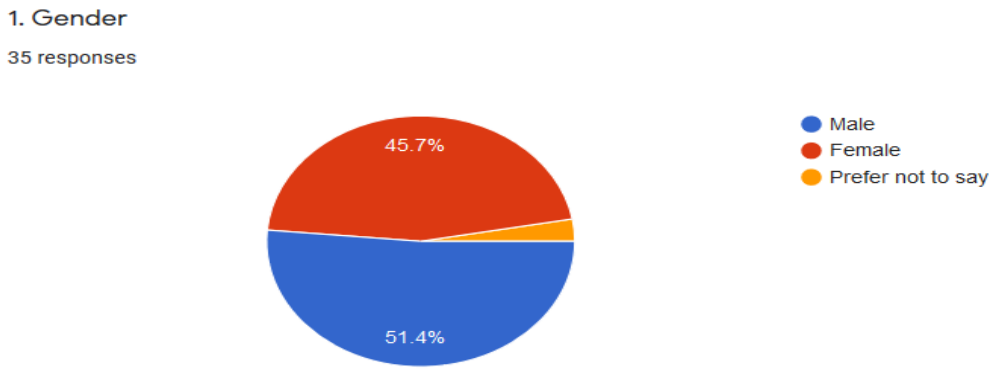


Figure 4.2: Respondents' Gender

Based on the pie chart above, 45.7% of respondents (16 people) are female and 51.4% of respondents (18 people) are male. Also, one respondent would not like to state his or her gender.

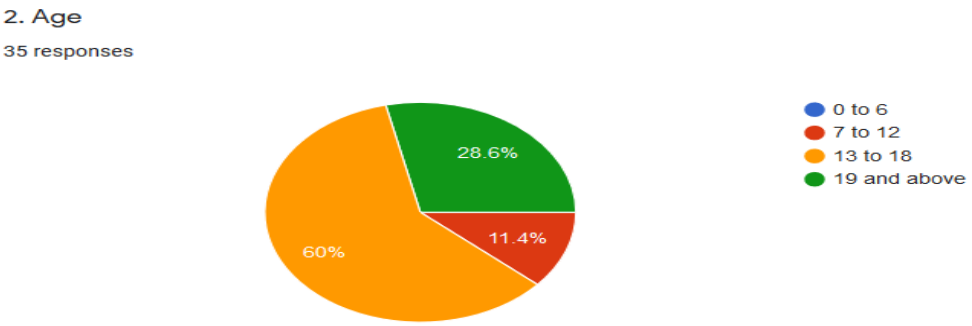


Figure 4.3: Respondents' Age

According to the pie chart above, respondents' age is split down into few groups, for example 0 to 6 for kindergarten students, 7 to 12 for primary school students, 13 to 18 for secondary school students, 19 and above for those who already graduated from high school. Out of 35 respondents, 60% of respondents (21 people) are in the range of 13 years old to 18 years old. 28.6% of respondents (10 people) are having the age of 19 years old and above. Then, it is followed by 11.4% of respondents (4 people) which range from 0-year-old to 6 years old. Survey form is rarely distributed to primary school students and kindergarten students due to their comprehension level toward the questions.

In short, the main focus of this questionnaire is more focused on secondary school students although primary school students will also be the main target user for the trilingual mobile dictionary application, but their understanding level toward the questions is too shallow.

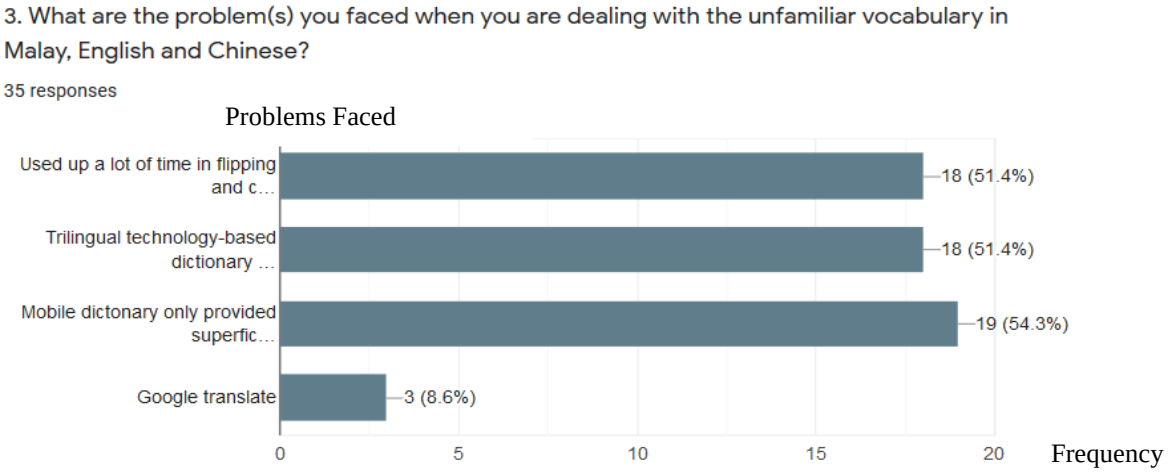


Figure 4.4: Problems Faced by Respondents

The histogram illustrated as above has demonstrated the problems that people usually face when they are dealing with the unfamiliar vocabulary in Malay, English and Chinese. Most of the respondents have agreed with the problem statements that have been stated earlier in this project. In fact, **"mobile dictionary only provided superficial information of a word**" is selected as the biggest challenge they face when they are consulting the mobile dictionary. Then, it is followed by another two problems which obtained the same number of votes from the respondents, where these problems are "**used up a lot of time in flipping and checking a particular vocabulary among three languages from paper dictionary**" and "**trilingual technology-based dictionary is not available in the market**". Besides that, three respondents have provided another option which is "**Google translate**", while the meaning of this option might be telling us that the Google translate does not provide a good translation between multiple languages and it never provides the rich details of the vocabulary.

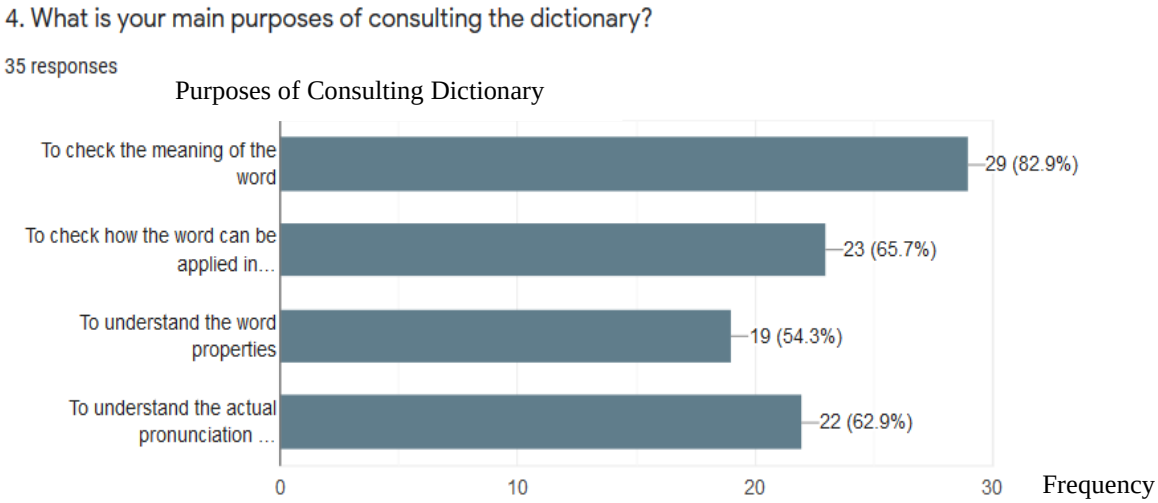


Figure 4.5: Purposes of Respondents Consult Dictionary

Based on the histogram shown above, we can interpret that **checking the meaning of the word** is the major objective for a user to consult the dictionary as 82.9% of respondents (29 people) have chosen this option. Next, more than half of the respondents defined the main purposes of consulting the dictionary are to **check how the word can be applied in a sentence**, **understand the actual pronunciation of the word** and **understand the word properties**. This has concluded that users want to have a mobile dictionary that contains rich information of the vocabularies where this also met the objective of this project which is to compile and expand a comprehensive corpus of English, Mandarin and Malay vocabularies in the trilingual mobile dictionary application.

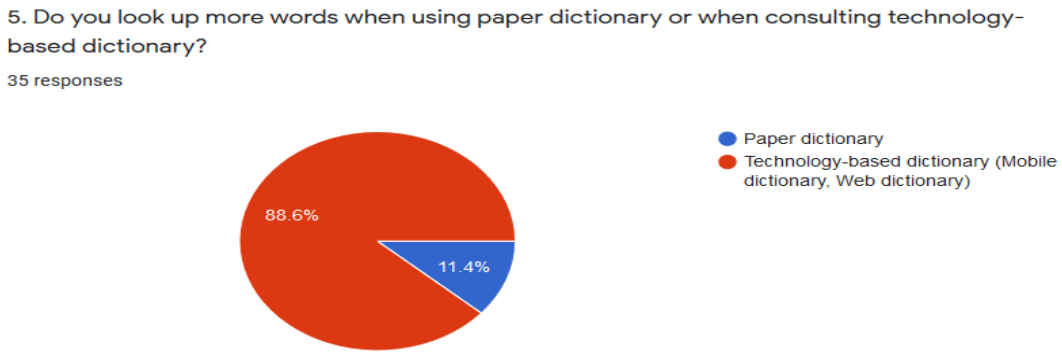


Figure 4.6: Respondents' Option for Types of Dictionary

The pie chart above shows that 88.6% of the respondents (31 people) prefer technology-based dictionary which is also commonly known as mobile dictionary and web dictionary. Meanwhile, 11.4% of respondents (4 people) are prone to use paper dictionaries when they want to loop up details of the vocabulary. Apparently, the demand for trilingual mobile dictionary application is high from the market as most of the respondents would like to use technology-based dictionaries when they are performing search or translate a vocabulary.

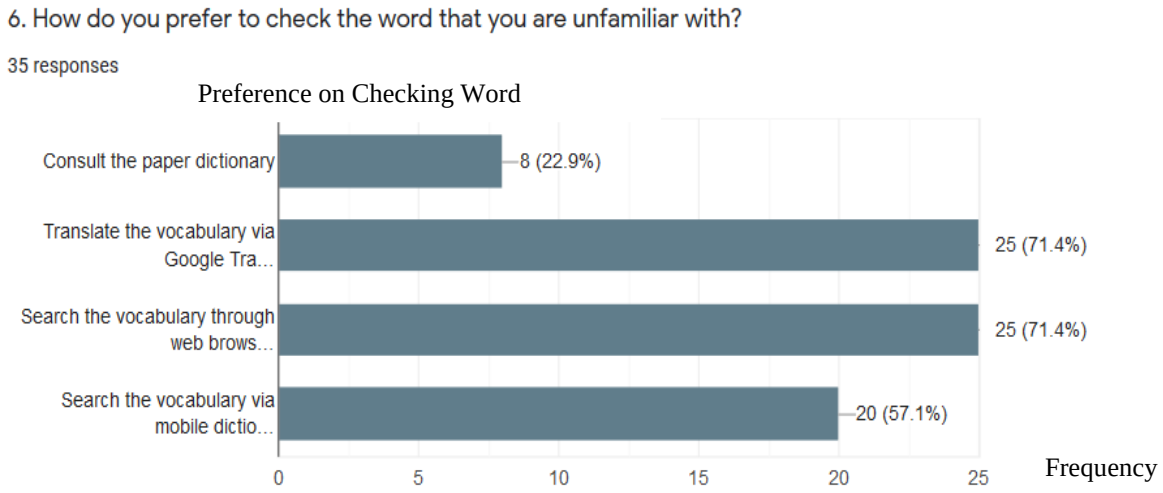


Figure 4.7: Respondents' Approaches for Checking Vocabulary

The histogram above has illustrated the opinions regarding the method used by the respondents when they are checking the unfamiliar word. Obviously, 71.4% of respondents (25 people) have conducted **searching and translation of vocabulary via Google Translate** and **some other web-based dictionary**. Then, 51.7% of the respondents (20 people) like to **search the vocabulary via mobile dictionary**, the result is followed by 22.9% of respondents (8 people) prefer to **consult the paper dictionary** when they want to check an unfamiliar word. The reason Google Translate and web-based dictionaries are popularly used may be due to the absence of trilingual mobile dictionary application in the market. Thus, users have to switch between different bilingual mobile dictionary applications in order to search and translate the vocabulary in Malay, English and Mandarin. On the other hand, lesser respondents prefer to

consult the paper dictionary as it is time consuming to flip and check a particular vocabulary among 3 languages from the paper dictionaries.

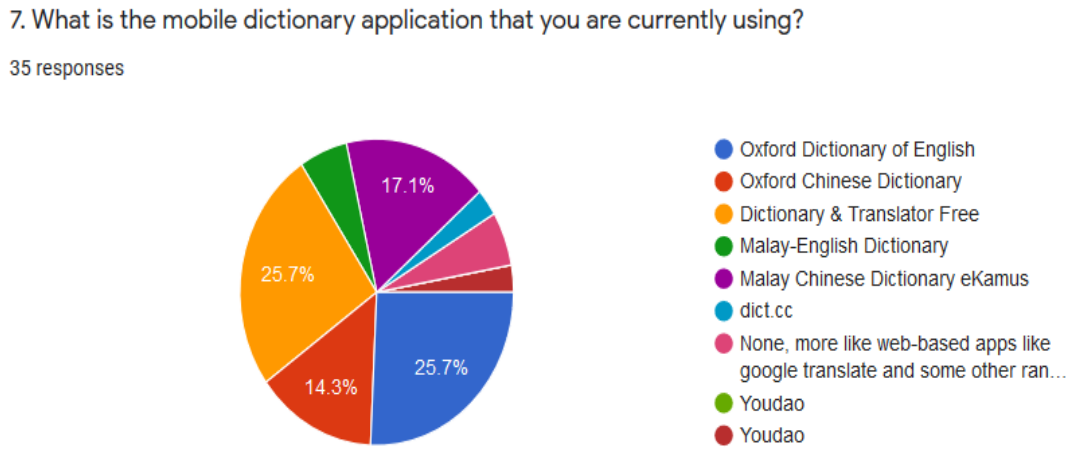


Figure 4.8: Mobile Dictionary Application Used by Respondents

According to the pie chart, each Dictionary & Translator Free and Oxford Dictionary of English are utilized by 25.7% of the respondents (9 people). Next, 17.1% of respondents (6 people) prefer Malay Chinese Dictionary eKamus, while 14.3% of respondents (5 people) prone to use Oxford Chinese Dictionary. Out of 100%, 5.7% of respondents (2 people) are using Malay-English Dictionary and another 5.7% of respondents (2 people) are using Google Translate and some random website. Besides that, both Youdao and dict.cc are used by only one respondent out of 35 respondents. In short, Dictionary & Translator Free and Oxford Dictionary of English have gained more respondents' interest in using it, therefore, we can refer to the design and features offered by both applications when we are developing the trilingual mobile dictionary application.

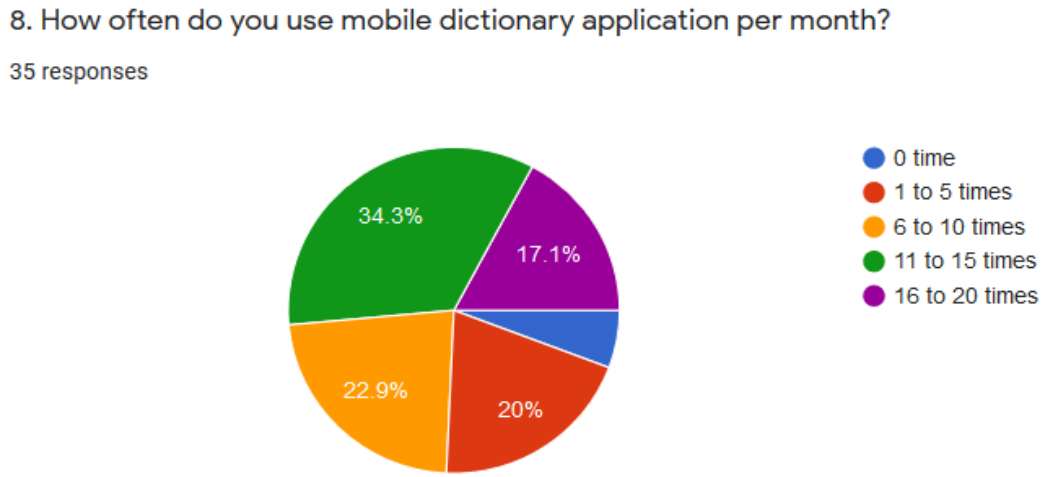


Figure 4.9: Frequency of Usage on Mobile Dictionary Application

Based on the pie chart shown as above, 34.3% of respondents (12 people) consult 11 to 15 times on mobile dictionary application per month. While, 22.9% of respondents (8 people) use 6 to 10 times of mobile dictionary application per month. Next, 20% of respondents (7 people) utilize mobile dictionary applications for 1 to 5 times per month. Then, the statistic is followed by 17.1% of respondents (6 people) where they consult 16 to 20 times on mobile dictionary application per month. There are also 2 respondents who have never used any mobile dictionary application before. In conclusion, most of the respondents consult mobile dictionary application more than 5 times per month, therefore, mobile dictionary application plays an essential role in helping them on their learning path.

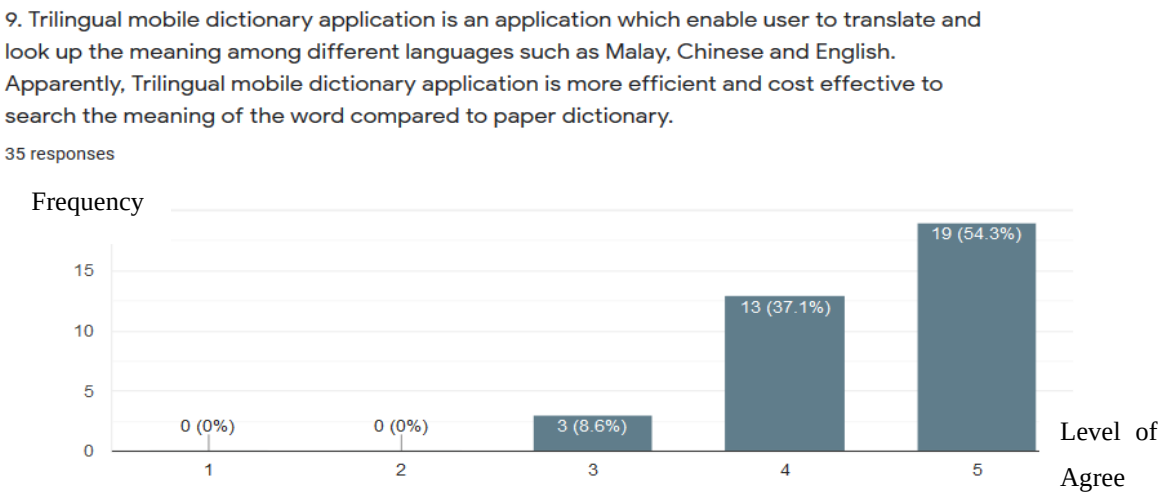


Figure 4.10: Respondents' Level of Agree on the Statement

Based on the histogram, 54.3% of the respondents (19 people) have strongly agreed to the statement, meanwhile, 37.1% of respondents (13 people) have agreed to the statement followed by 8.6% of the respondents (3 people) feel neutral to the statement. Overall, most of the respondents have agreed to the statement regarding the trilingual mobile dictionary is more efficient and cost effective to search the meaning of the word compared to paper dictionary. Also, this outcome has proven the problem statement (the process of flipping and checking a particular vocabulary among 3 languages from the paper dictionaries is time consuming, challenging and not effective at all) that proposed earlier is correct.

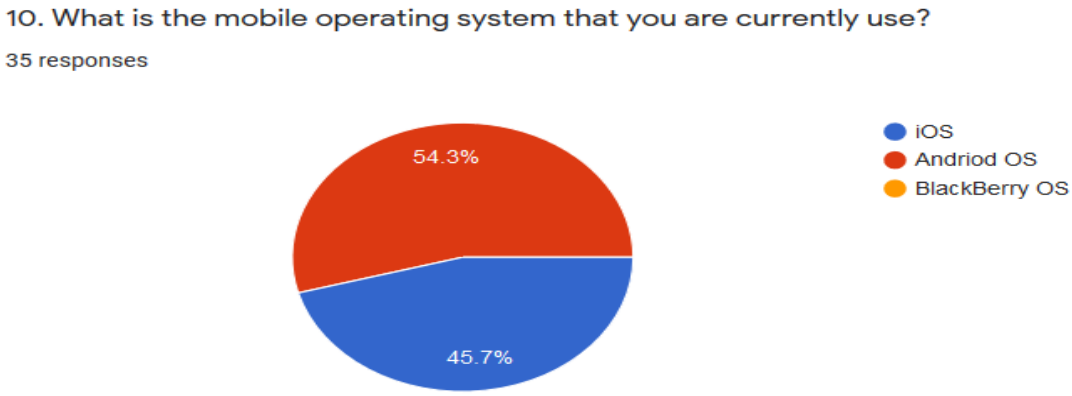


Figure 4.11: Mobile Operating System Used by Respondents

Out of 35 respondents, 54.3% of the respondents (19 people) are using Android phones while 45.7% of respondents (16 people) are using iPhone. Hence, the Android version of trilingual mobile dictionary application will be first developed and published on the Android market.

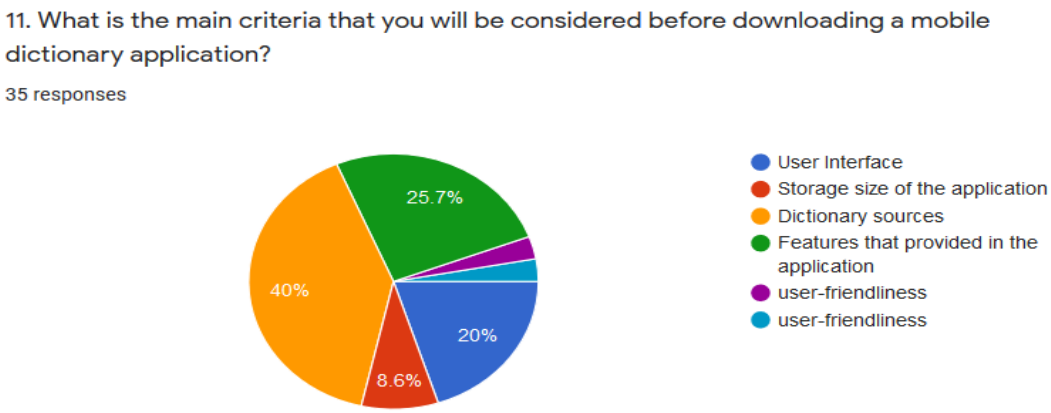


Figure 4.12: Respondents' Consideration Before Downloading a Mobile Dictionary Application

Out of 35 respondents, 40% of the respondents (14 people) emphasized that the dictionary source is the main criteria to look at before they download a mobile dictionary application. 25.7% of the respondents (9 people) are concerned about the features provided in the application, and 20% of the respondents (7 people) are concerned about the user interface of the application. On the other hand, 8.6% of respondents (3 people) will consider the storage size of the application due to the limited storage offered by the smartphone. Besides that, two respondents also suggested that user-friendliness of the application may become their primary consideration when they are downloading a mobile dictionary application. As a result, merely half of the respondents would like to have a mobile dictionary application which implements consistent and rich dictionary sources.

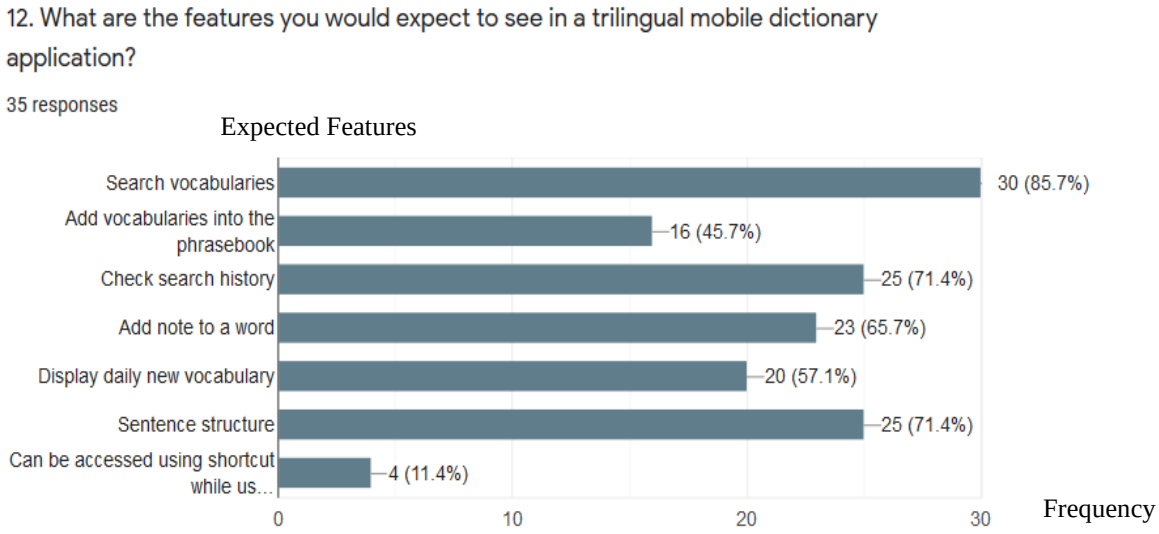


Figure 4.13: Respondents' Expectations

According to the histogram, **search vocabularies** is the feature that is expected by most of the respondents (85.7%). Moreover, there are four features such as **search history**, **note**, **daily new vocabulary** and **sentence structure** expected by more than 50% of the respondents. However, the **phrasebook** feature is expected by less than 50% of the respondents. Furthermore, four respondents have provided the opinion that they are expecting to see the application **can be accessed using a shortcut while using other applications and close it without leaving other applications**. Overall, the expectations from the respondents will be met as the project scope has defined most of the features that are required by the respondents.

4.5 Low Fidelity Prototype

In this section, some essential interface of the trilingual mobile dictionary application has been sketched out in order to provide the developer a better vision when he or she is developing the application. Moreover, this prototype is also allowing potential end-users to suggest possible enhancements and modifications on the features of the application. Apparently, the benefit of this prototype is enabling end-users to feel more comfortable when recommending the improvements by using the rough sketches. The user interface of the trilingual mobile dictionary application is shown as below:

Figure 4.14: UI – Display Daily New Vocabulary

Figure 4.15: UI - Search Vocabulary

Figure 4.16: UI – Display Detailed Information of the Vocabulary

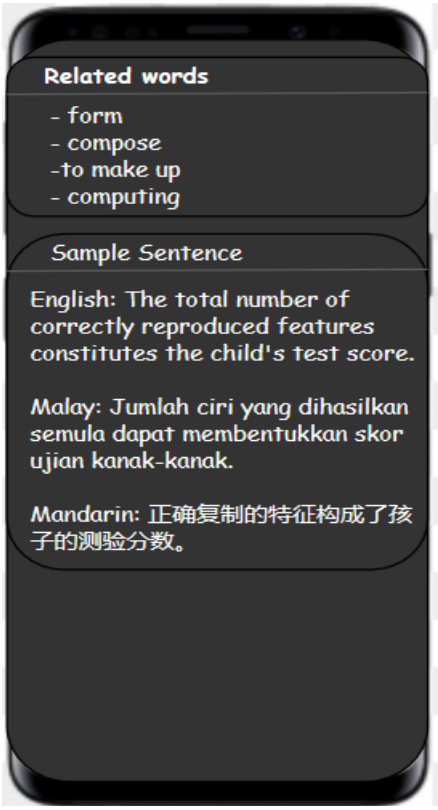


Figure 4.17: UI – Display Detailed Information of the Vocabulary

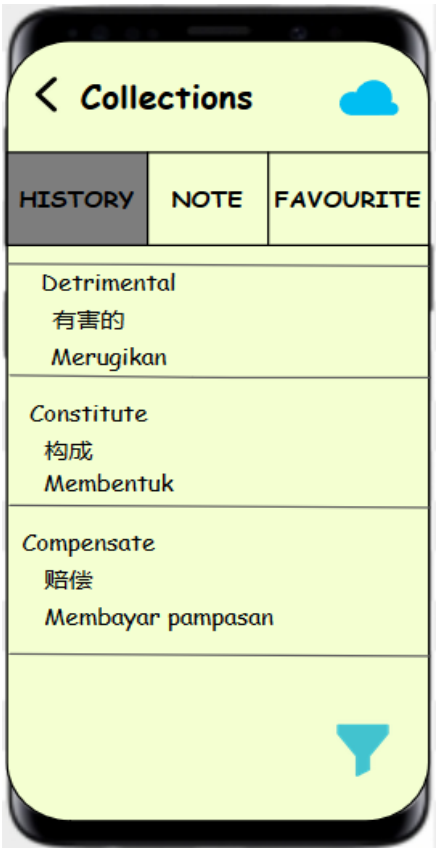


Figure 4.18: UI – Collections of History

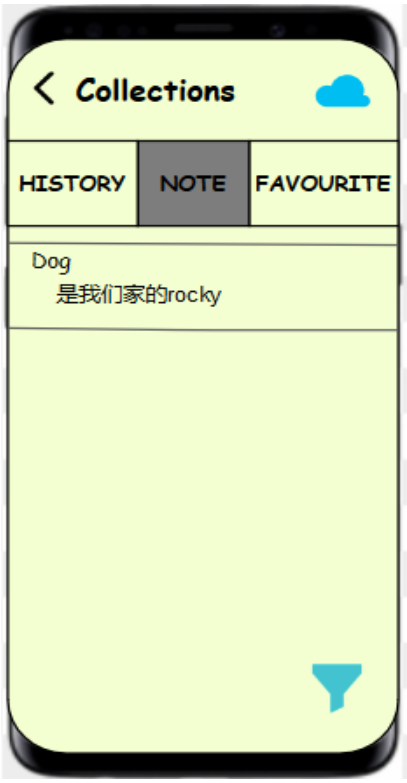


Figure 4.19: UI – Collections of Note

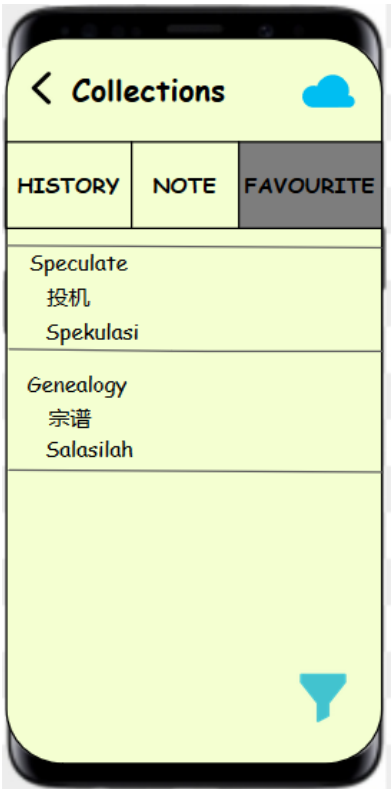


Figure 4.20: UI – Collection of Favourite

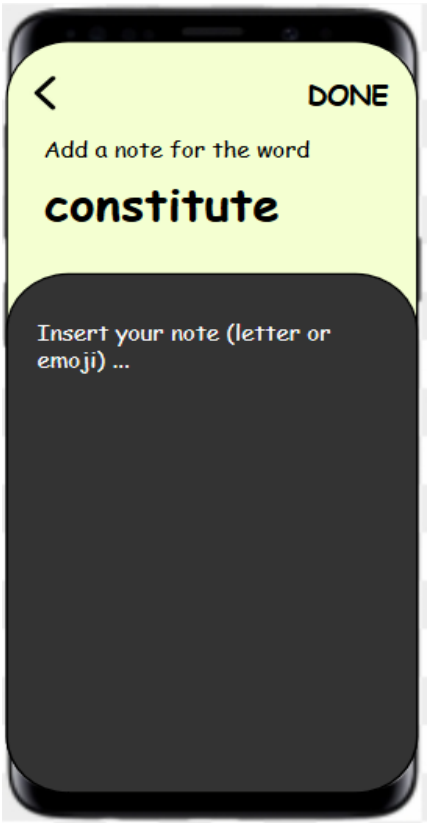


Figure 4.21: UI – Add Note to a Vocabulary

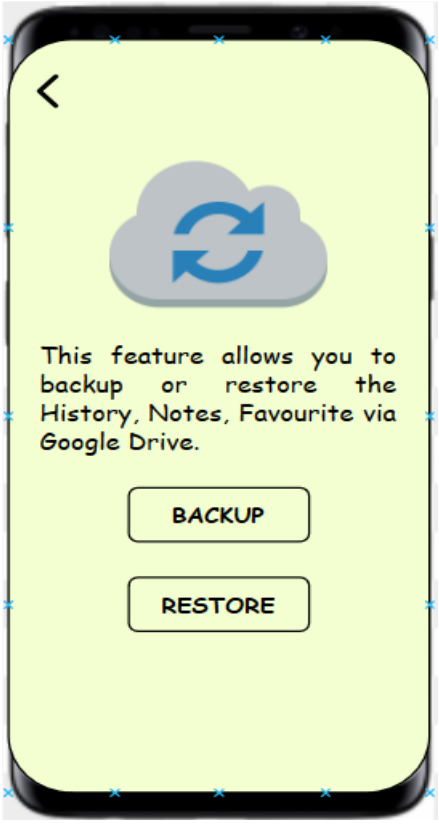


Figure 4.22: UI – Backup or Restore Data

CHAPTER 5

SYSTEM DESIGN

5.1 Introduction

In this chapter, modelling diagrams for the whole implementation are illustrated and explained. In addition, the construction of the system architecture will also be shown to prove the connectivity between front end application, back-end server and instances. Moreover, designs of user interface and database are depicted and described.

5.2 User Interface Design

Since the end product of this project is a trilingual mobile dictionary application, therefore all the people especially Malaysia school students are entitled to use this application. As a result, all the end users will be using the same user interfaces on the dictionary application.

5.2.1 UI Design Navigation Model

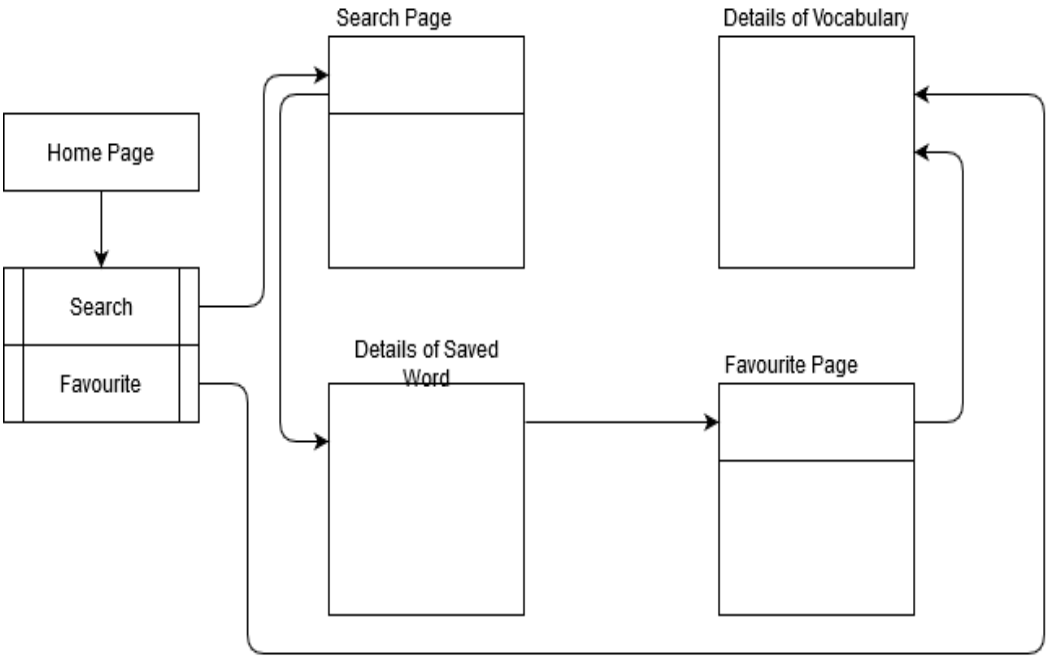


Figure 5.1: User Interface Design Navigation Model

The navigation model above illustrates the directory of the overall system for all the end users. Considering a dictionary application will not have any security problems, hence the users will be directly landed on the home page instead of the user login interface. As we can see, the navigation drawer is actually establishing various connections to different pages after the user lands on the home page.

5.2.2 UI for Trilingual Mobile Dictionary Application

5.2.2.1 Home Page

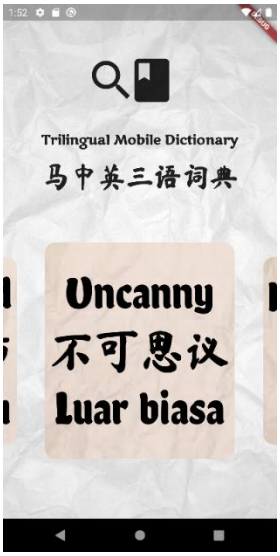


Figure 5.2: Home Page with Daily New Vocabulary Feature

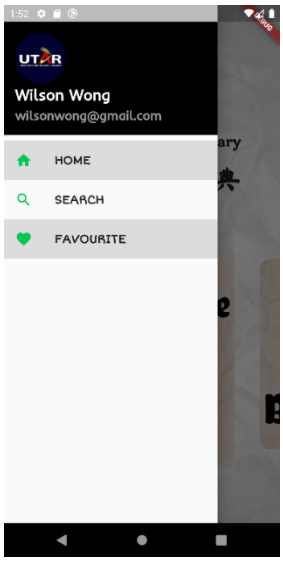


Figure 5.3: Home Page with Side Drawer

5.2.2.2 Search Page and Save Word Page

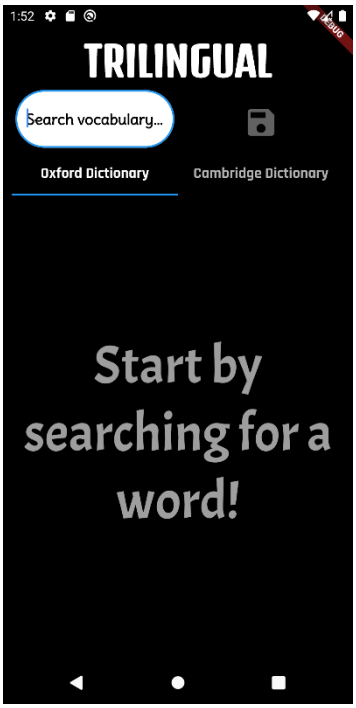


Figure 5.4: Search Page with Empty Input

Figure 5.5: Search Result from Google Translation API and Oxford Dictionary API (1)

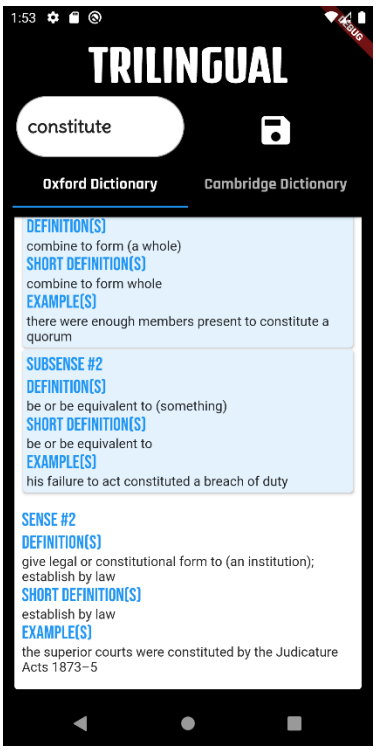


Figure 5.6: Search Result from Google Translation API and Oxford Dictionary API (2)

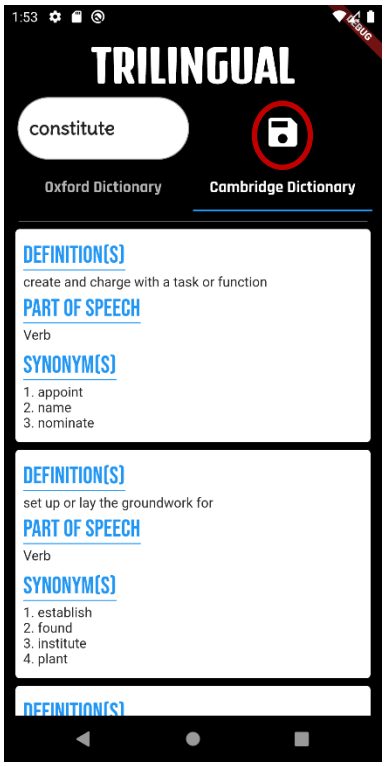


Figure 5.7: Search Result from Words API

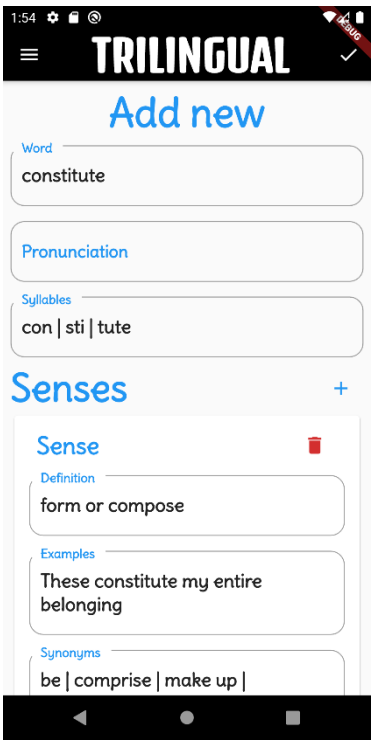


Figure 5.8: Save Word Page When the Disc Icon was Pressed (1)

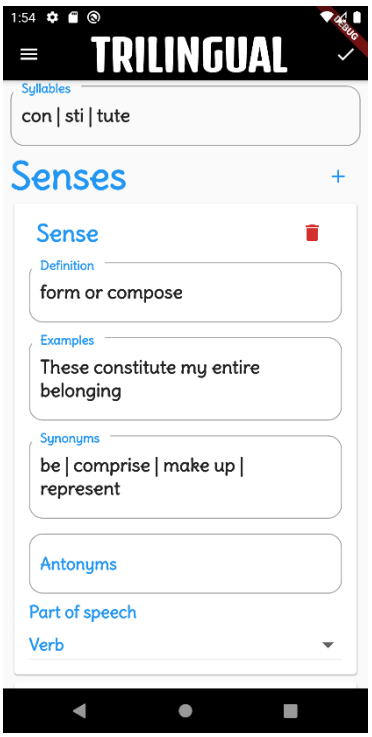


Figure 5.9: Save Word Page When the Disc Icon was Pressed (2)

5.2.2.3 Phrasebook Page and Save Word Page

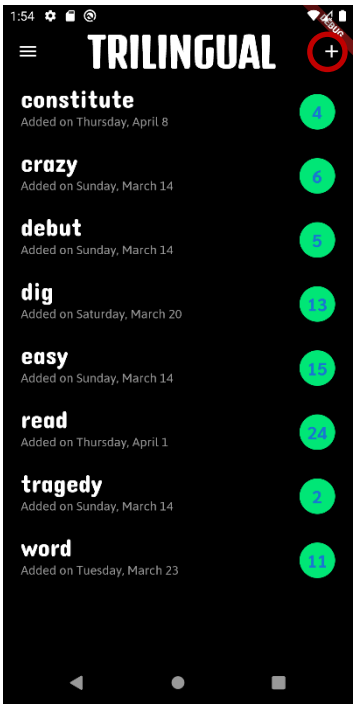


Figure 5.10: Phrasebook Page with List of Saved Word

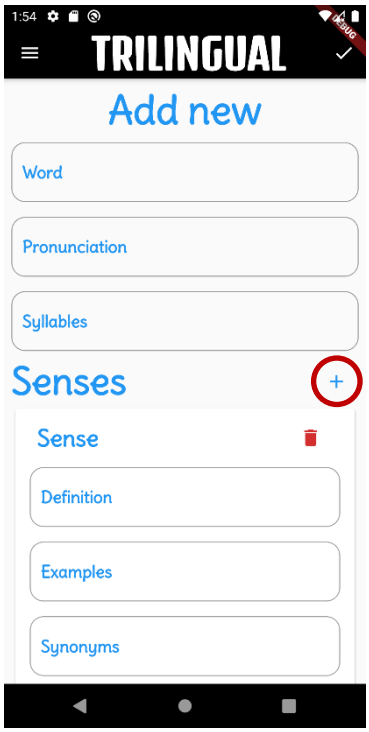


Figure 5.11: Save Word Page When the Add Icon was Pressed

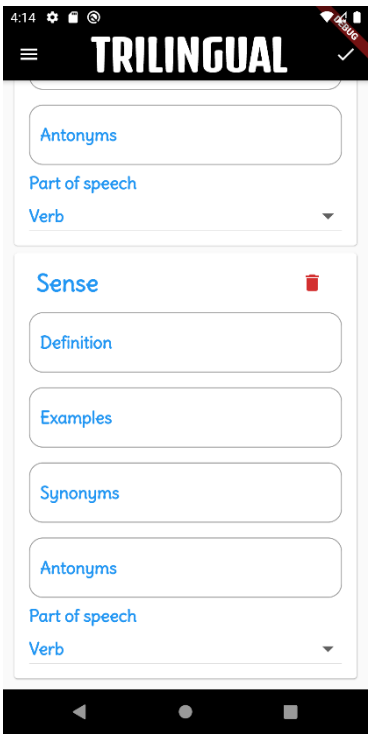


Figure 5.12: Save Word Page with More Senses When Add Icon was Pressed

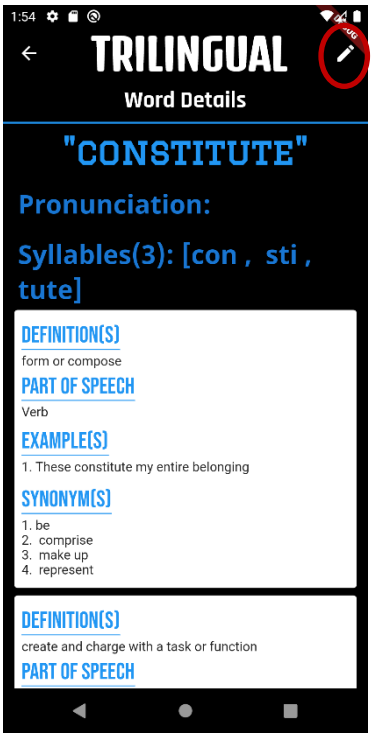


Figure 5.13: Phrasebook Page with Word Details

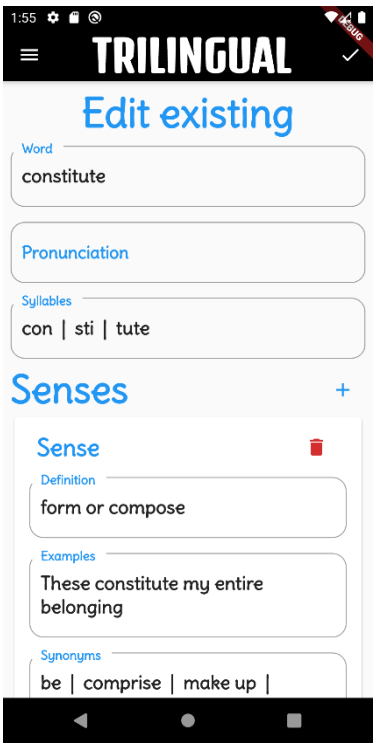


Figure 5.14: Save Word Page with Synchronized Details

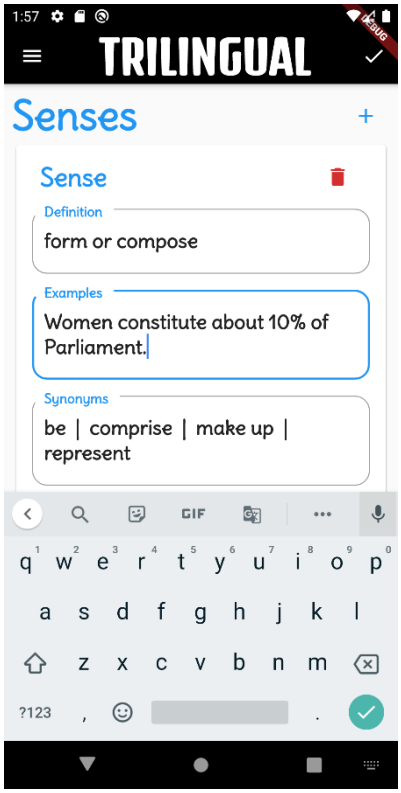


Figure 5.15: Save Word Page with Self-Defined Note

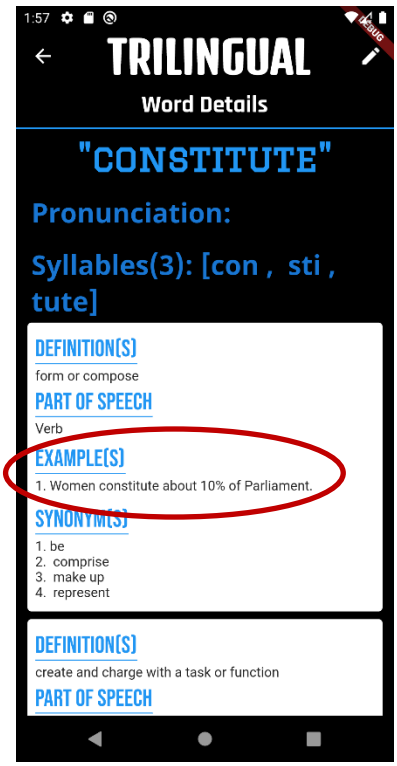


Figure 5.16: Phrasebook Page with Updated Note

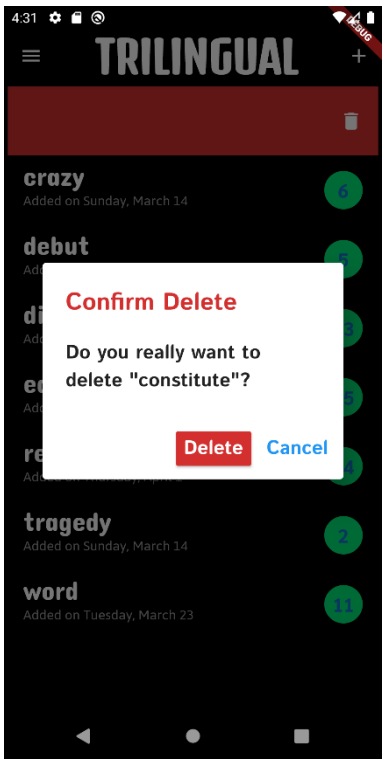


Figure 5.17: Phrasebook Page with Delete Feature

5.3 System Design Model

This section consists of the data flow diagram for the entire system. Besides that, system architecture design and database design are discussed in order to demonstrate the relationship between the system and the end users.

5.3.1 Data Flow Diagram

Data flow diagrams are usually used to reflect the data flow in a business information system graphically. In other words, it defines the procedures which are involved in a program that passes data from input to file storage and eventually generates the necessary reports. In this section, there are 3 layers of data flow diagrams built to display the data circulation. The layers that included are context layer, level 0 and level 1. For your information, the more comprehensive flow of information between the modules and data stores has been exposed as the layer is going deeper.

5.3.1.1 Context Diagram

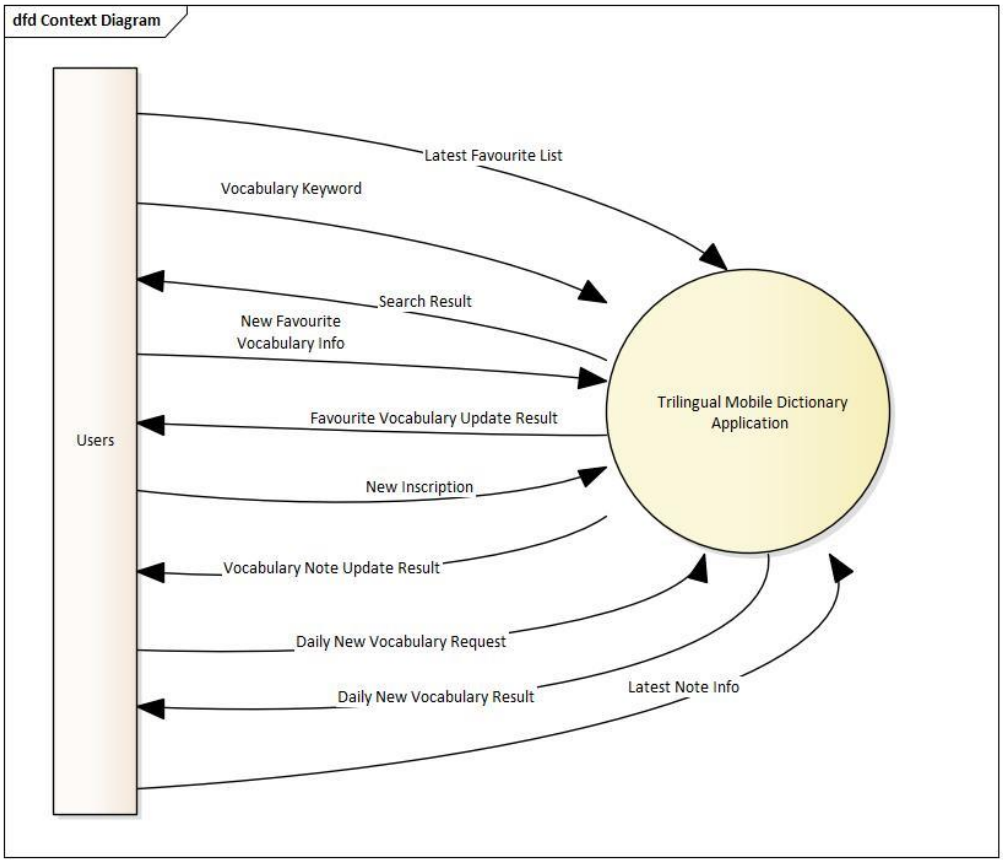


Figure 5.18: Context Diagram of Trilingual Mobile Dictionary Application

5.3.1.2 Level 0 Diagram

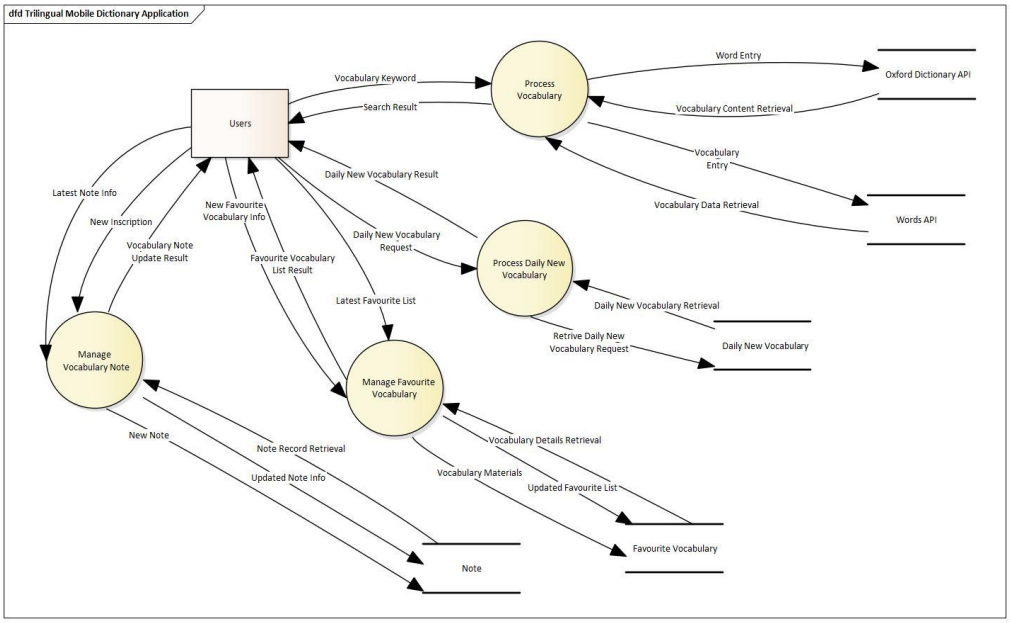


Figure 5.19: Level 0 Data Flow Diagram

5.3.1.3 Level 1 Diagram

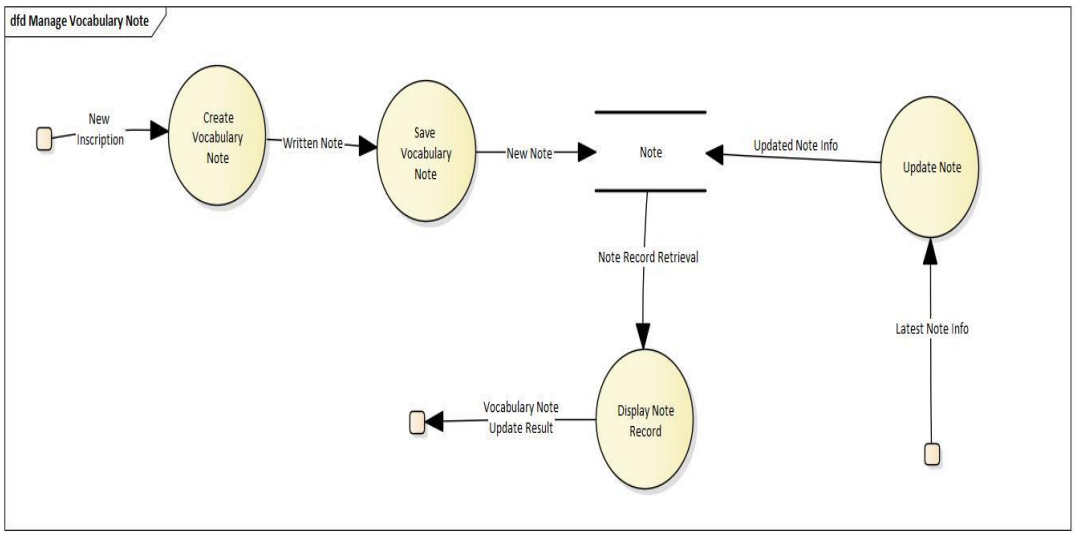


Figure 5.20: Level 1 Diagram for “Manage Vocabulary Note” Process

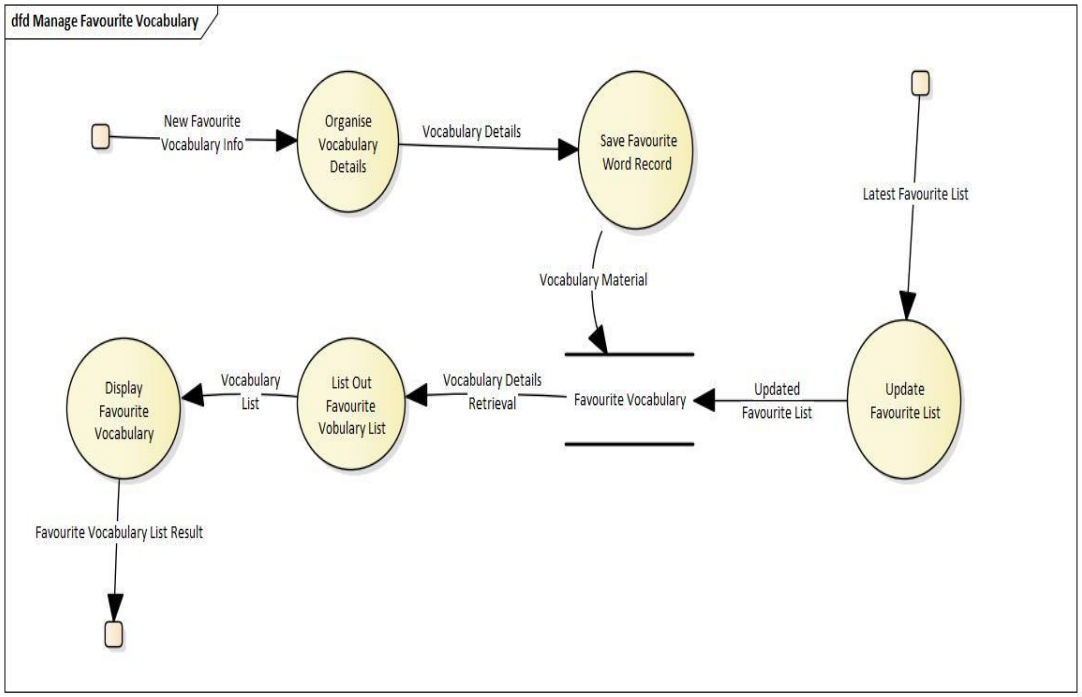


Figure 5.21: Level 1 Diagram for "Manage Favourite Vocabulary" Process

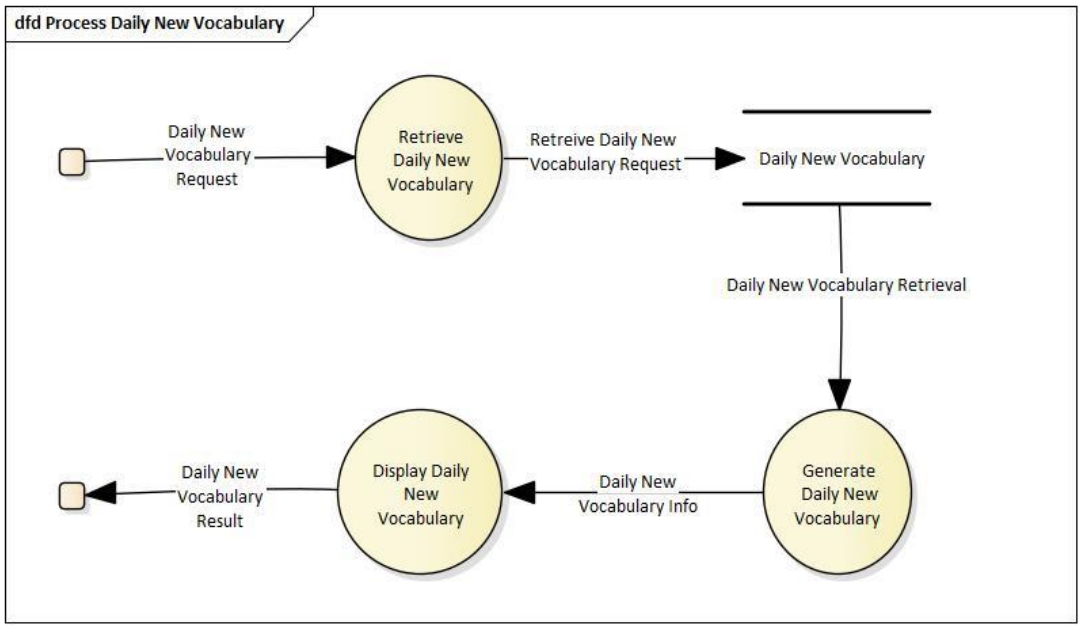


Figure 5.22: Level 1 Diagram for "Process Daily New Vocabulary" Process

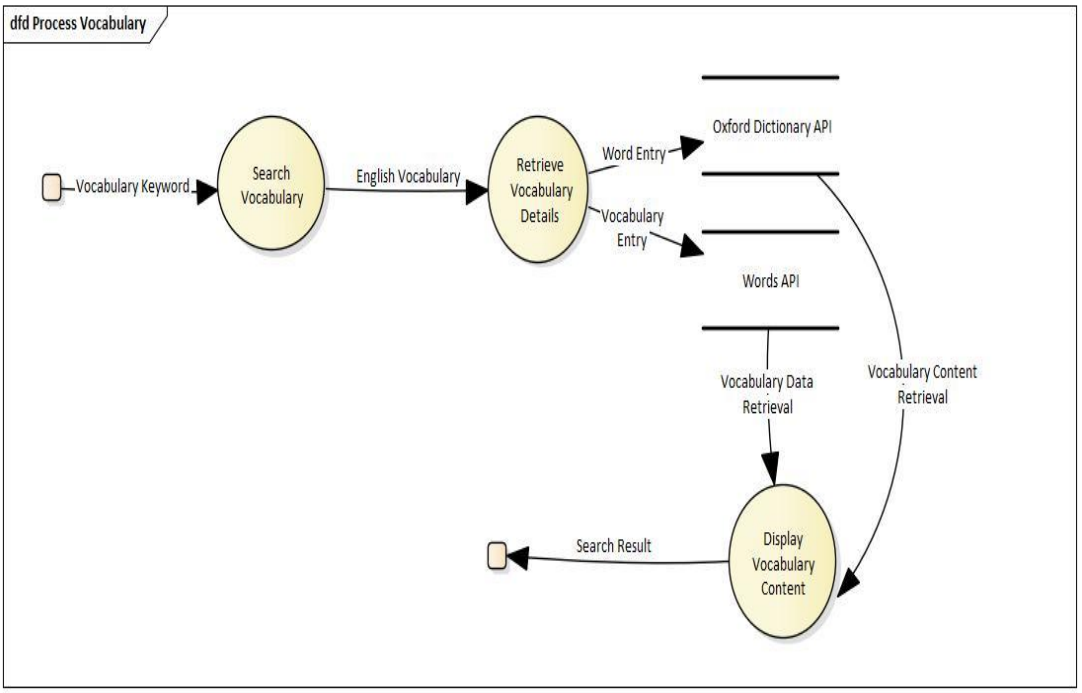


Figure 5.23: Level 1 Diagram for "Process Vocabulary" Process

5.3.2 System Architecture Design

The implemented system architecture design in the trilingual mobile dictionary application is business logic component while it is also commonly known as BLoC. The reason BLoC is selected to become system architecture design in this mobile application is because it is an architecture style which is based on various components that consist only of business logic. Based on this characteristic, it can easily be exchanged between multiple Dart applications where this advantage will be definitely benefiting our previous selection on the development framework – Flutter. Throughout the system development, BLoC makes it possible to isolate the presentation layer from the business logic. As a result, the developer can actually code faster, carry out efficient testing and reuse the code from time to time. In addition, BLoC was built around with three fundamental purposes such as simplicity, testability and power. These fundamental purposes have enabled Flutter developer to recognize the state of the application in any moment, build application in a faster and responsive way, capture each of the user interaction in the application so that data driven decisions can be made, and test each scenario efficiently to ensure the application reacts accordingly. What makes the BLoC architecture powerful is the ability to construct a larger application from simpler parts. Due to the small

distinct simpler parts, every feature of an application is often easy to evaluate and analyse.

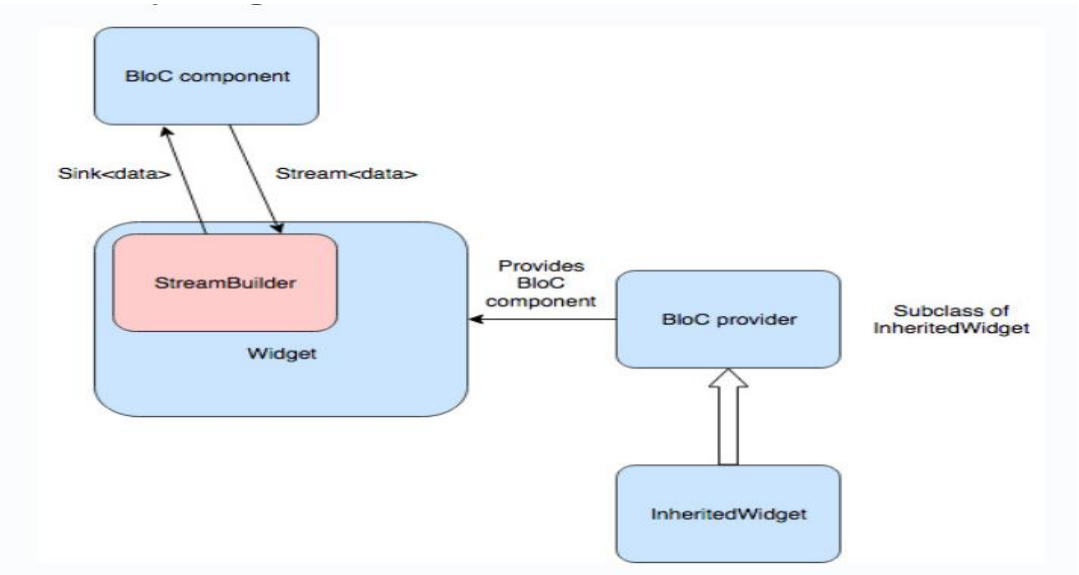


Figure 5.24: BLoC Architecture Design

Based on the figure above, it illustrates the workflow of BLoC architecture. The main BLoC concepts were coming from the components of “event” and “state”. In other words, “event” plays the role of getting inputs to the BLoC architecture whereas “state” acts as an output of the BLoC architecture. For example, when a user performs an action with the user interface, it will then send an event to the BLoC module (bloc | Dart Package, 2021). Next, the work pertaining to the business logic such as fetching information from the internet and processing the data has to be done by the BLoC module. After it does, the primary duty of the BLoC module is to identify the action, manage it and display the new or changed state to the user interface of the application. In the development of trilingual mobile dictionary application, searching information of the keyword or vocabulary can be known as an event to the BLoC component while fetching the data from Oxford Dictionary API and Words API to display on the user interface is considered as a state to the BLoC module.

5.3.3 Database Design

Entity Relationship Diagram (ERD) is commonly recognized as the ER Diagram or ER model. It is a type of structural diagram for the purpose of

designing database architecture. In other words, the main entities within the system scope as well as the inter-relationships among these entities are visualized using symbols and connectors in an ERD.

5.3.3.1 Physical Entry Relationship Diagram

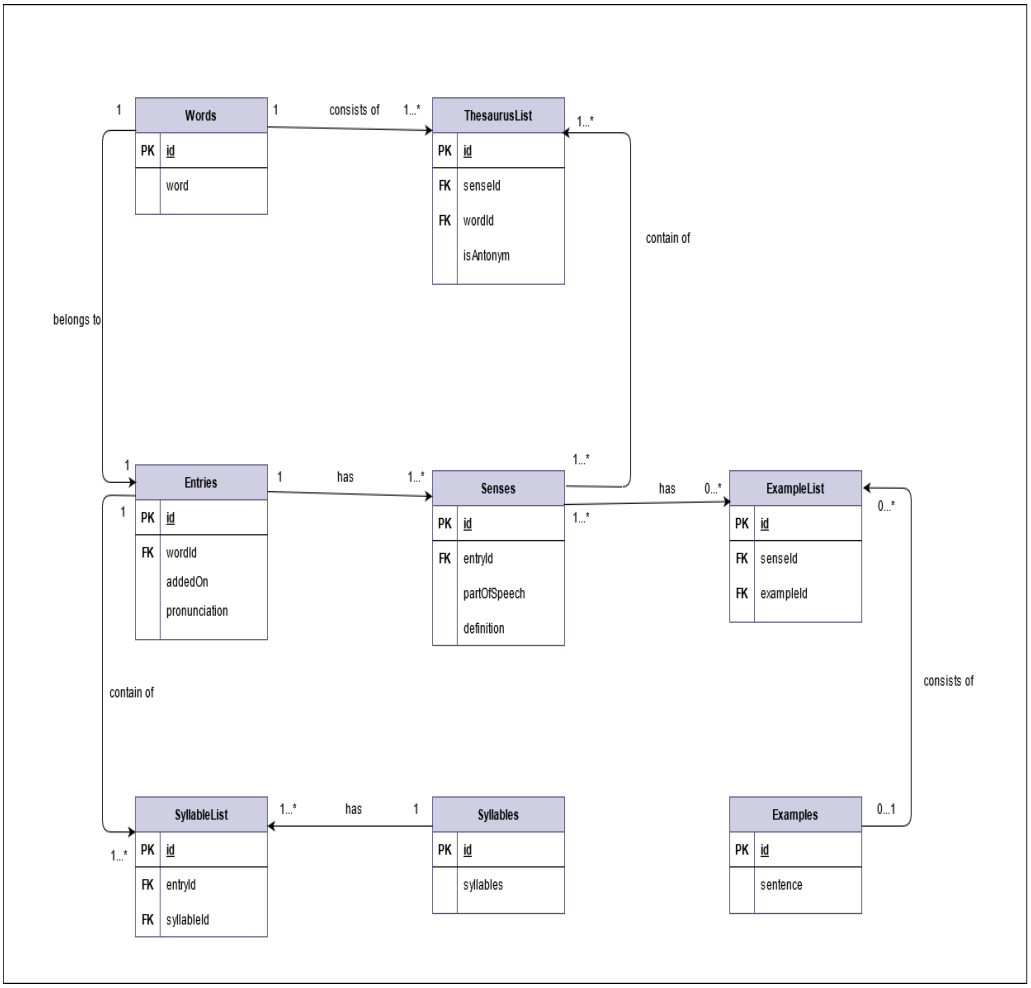


Figure 5.25: Physical Entity Relationship Diagram

5.3.3.2 Logical Entity Relationship Diagram

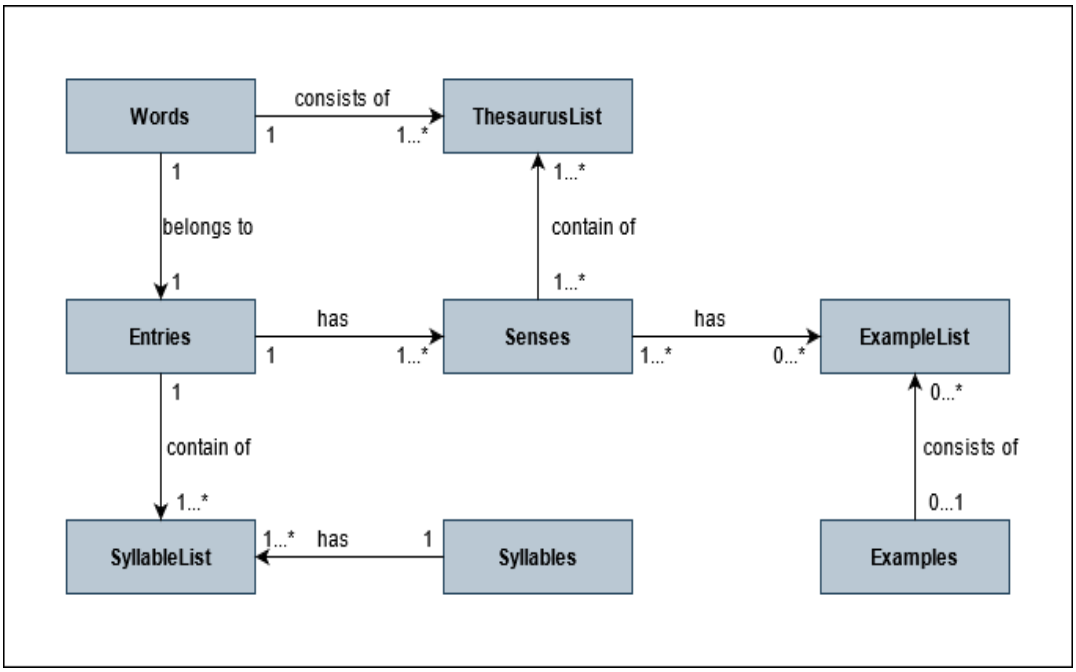


Figure 5.26: Logical Entity Diagram

5.3.4 Data Dictionary

Table 5.1: Description for Database Tables

Table Name	Description
Words	Containing the information for word
Entries	Containing the information for every input from the user
Senses	Containing the information for each entry
Syllables	Containing the information for syllable
Examples	Containing the sentence for example
SyllableList	Joining the syllables table and entries table
ThesaurusList	Containing the details for every entry
ExampleList	Joining the senses table and examples table

Table name: Words

Table 5.2: Data Dictionary for Words Entity

Attribute	**Description**	**Datatype**	**PK/FK**	**FK Referenced Table**
id	Unique identification for words	Int	PK	-
word	Vocabulary	Varchar	-	-

Table name: Entries

Table 5.3: Data Dictionary for Entries Entity

Attribute	**Description**	**Datatype**	**PK/FK**	**FK Referenced Table**
id	Unique identification for entries	Int	PK	-
wordId	Unique identification for words	Int	FK	Words
addedOn	Datetime when the particular entry has been saved	DateTime	-	-
pronunciation	English pronunciation of the vocabulary	Varchar	-	-

Table name: Senses

Table 5.4: Data Dictionary for Senses Entity

Attribute	**Description**	**Datatype**	**PK/FK**	**FK**

				Referenced Table
id	Unique identification for senses	Int	PK	-
entryId	Unique identification for entries	Int	FK	Entries
partOfSpeech	Word property of the entry	Int	-	-
definition	Explanation of the entry	Varchar	-	-

Table name: Syllables

Table 5.5: Data Dictionary for Syllables Entity

Attribute	**Description**	**Datatype**	**PK/FK**	**FK Referenced Table**
id	Unique identification for syllables	Int	PK	-
syllables	Syllable of the entry	Int	-	-

Table name: Examples

Table 5.6: Data Dictionary for Examples Entity

Attribute	**Description**	**Datatype**	**PK/FK**	**FK Referenced Table**
id	Unique identification for examples	Int	PK	-
sentence	Sample	Varchar	-	-

	sentence of the entry			

Table name: SyllableList

Table 5.7: Data Dictionary for SyllableList Entity

Attribute	**Description**	**Datatype**	**PK/FK**	**FK Referenced Table**
id	Unique identification for syllable list	Int	PK	-
entryId	Unique identification for entries	Int	FK	Entries
syllableId	Unique identification for syllables	Int	FK	Syllables

Table name: ExampleList

Table 5.8: Data Dictionary for ExampleList Entity

Attribute	**Description**	**Datatype**	**PK/FK**	**FK Referenced Table**
id	Unique identification for example list	Int	PK	-
senseId	Unique identification for senses	Int	FK	Senses
exampleId	Unique identification for examples	Int	FK	Examples

Table name: ThesaurusList

Table 5.9: Data Dictionary for ThesaurusList Entity

Attribute	**Description**	**Datatype**	**PK/FK**	**FK Referenced Table**
id	Unique identification for thesaurus list	Int	PK	-
senseId	Unique identification for senses	Int	FK	Senses
wordId	Unique identification for words	Int	FK	Words
isAntonym	To check whether the antonym of the entry exist	Boolean	-	-

CHAPTER 6

SYSTEM IMPLEMENTATION

6.1 Introduction

This chapter depicts the whole system implementation according to the system design. On the other hand, in-depth analysis of the development execution process will also be included in this chapter. Additionally, certain good practices were introduced during the development in order to make the whole project more understandable.

6.2 System Modules

Based on the nature of the trilingual mobile dictionary application, it can be utilized by anyone who is interested to learn English, Malay and Mandarin even though the target consumer of this dictionary application is Malaysian school students only. Overall, the user interface architecture was improved and upgraded from time to time during the system development. Hence, the modifications and the user manual of the dictionary application are explained in the sections below in order to instruct and assist the users of the system and reduce the likelihood of misunderstanding. Furthermore, in order to visualize a better picture on the system's API, the used APIs and its information such as background, functionality and parameters were also recorded in the section below.

6.2.1 Listing of the System Modules

Since the dictionary application does not include the role such as admin, therefore, the guests or the users have the full authority to access all the modules in the application. The modules for every user were stated as below:

Table 6.1: System Modules Listing

No	Modules
1	Daily new vocabulary

2	Search vocabulary
3	Phrasebook management
4	Note management

6.2.2 Daily New Vocabulary

Figure 6.1: Prototype for Home Page

Figure 6.2: Actual Implementation for Home Page

In the designed prototype, a refresh button is required to trigger another daily new vocabulary for the user to view it. However, the idea from Flutter's component named carousel slider was implemented in order to achieve a better user experience as it will automatically slide to the new vocabulary for every four thousand milliseconds without any trigger from the user.

Figure 6.4: Home Page (1)

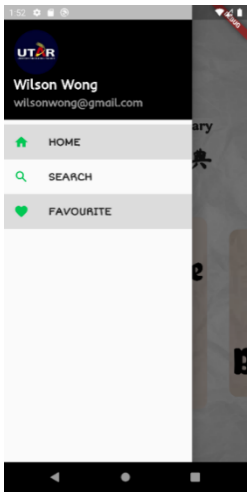


Figure 6.3: Home Page (2)

As we can see from the diagram above, the user interface for the home page is relatively easy to understand. This is because the users can directly view the daily new vocabulary without giving any input to the dictionary application. On the other hand, users are given two choices to navigate to the search page and phrasebook page where the options are either pressing the magnifying glass icon and book icon or slide left for dragging out the side drawer to choose the desired directory.

6.2.3 Search Vocabulary

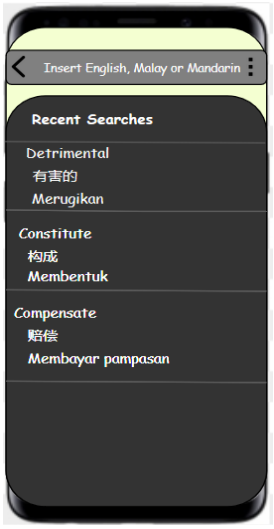


Figure 6.8: Prototype for Search Page (1)

Figure 6.7: Prototype for Search Page (2)

Figure 6.6: Actual Implementation for Search Page (1)

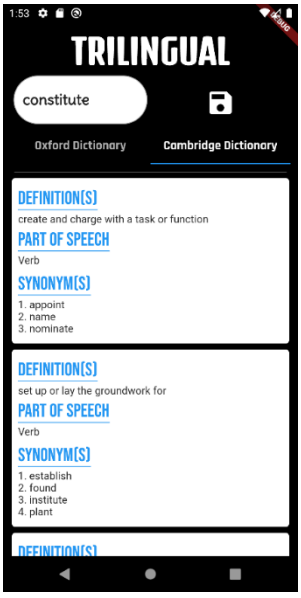


Figure 6.5: Actual Implementation for Search Page (2)

In the sketched prototype, the search page and the search result page were separated in a different view. Nevertheless, the actual implementation has combined the design of these two views due to the reference of two dictionaries. If dictionary application was built according designed prototype, the search result page will contain too much of word's information and it will eventually make the users feel hectic when using the trilingual mobile dictionary application. Thus, the Flutter's component named tab controller is used to help the users to distinguish and arrange the information from different sources.

Figure 6.10: Search Page (1)

Figure 6.9: Search Page (2)

In order to use the search feature, the users must ensure their device has established the connection to the network. Otherwise, the user will receive an error message regarding the reminder of the network connection. Based on the project scope, users are only allowed to make an English input at the search bar. If other characters such as digit, symbol or other languages are detected at the search bar, an error message of invalid input will be prompted to the user.

Based on figure 6.9 and 6.10, the users can look up the word's details from Oxford dictionary and Cambridge dictionary when they are pressed the dictionary name which is allocated at the tab controller. Furthermore, users can save the vocabulary and its details by pressing the disc icon whenever they

wish to favourite that information in their own phrasebook for their future reference. Similar to the home page, a side drawer is always ready for the users to switch their view.

6.2.4 Manage Phrasebook/ Favourite List

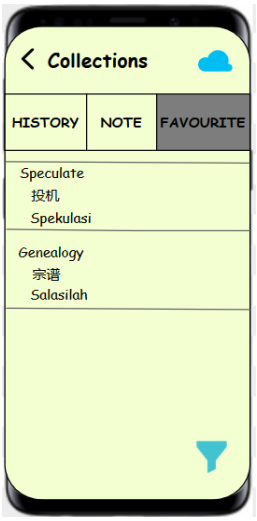


Figure 6.12: Prototype for Phrasebook Page

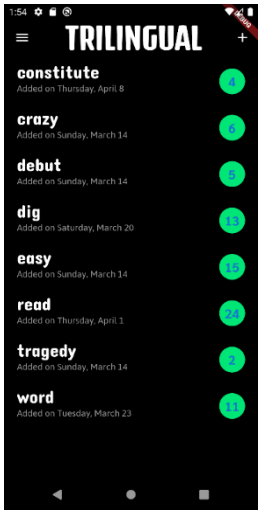


Figure 6.11: Actual Implementation for Phrasebook Page

Overall, there are no major changes on the actual implementation compared to the designed prototype. However, the minor changes on the actual implementation have included a circular avatar at the most right of every single row and it also replaced the translated text to word's added time.

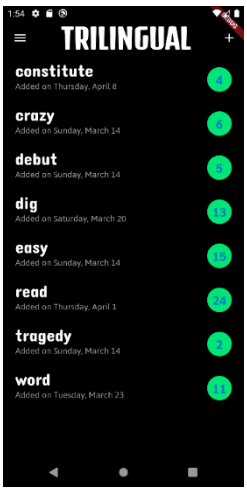


Figure 6.15: Phrasebook Page (1)

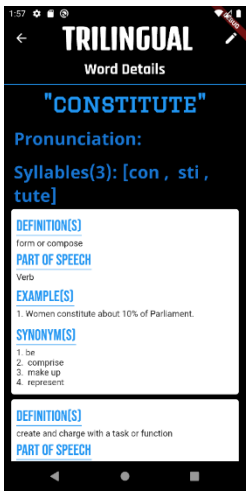


Figure 6.14: Phrasebook Page (2)

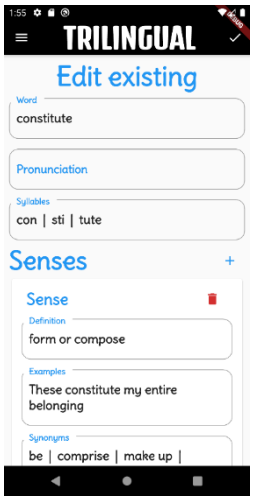


Figure 6.13: Save Word Page

According to the diagrams above, users are able to view a full list of saved words in the phrasebook page. As we can see, the circular avatar which is allocated at the most right of every single row is indicating the total number of senses it has. In order to ease the users to find the vocabulary, the list of saved words is sorted according to alphabetical order. On the other hand, there are two ways that the users can save a word to the phrasebook. The first option is pressing the add icon on the top right corner in the phrasebook page where this method is prepared for those users who want to record the word and its details that is not available in both Oxford dictionary and Cambridge dictionary. Whereas the second option is pressing the disc icon in the search page.

After the vocabulary has been saved, users can amend any of the saved details by pressing the pencil icon in the word's details page. Besides that, users are allowed to add more senses to the word's details by pressing the add icon and they can also remove the unnecessary senses by pressing the bin icon. Last but not least, the tick icon at the top right corner must be pressed in order to save all the changes.

6.2.5 Manage Note

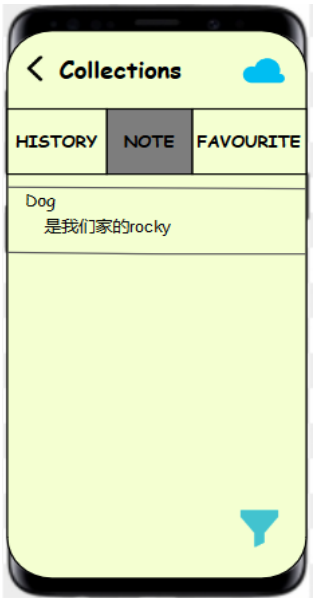


Figure 6.18: Prototype for Note Page (1)

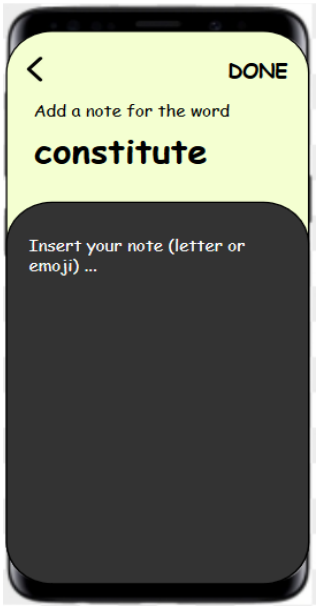


Figure 6.17: Prototype for Note Page (2)

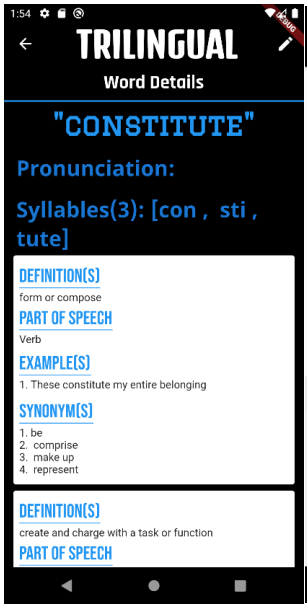


Figure 6.19: Actual Implementation for Note Page (1)

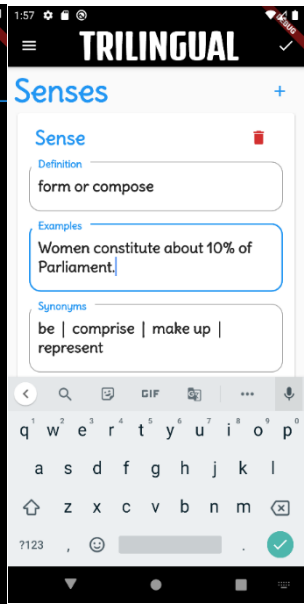


Figure 6.16: Actual Implementation for Note Page (2)

The concept of combination between phrasebook and note was chosen to implement in the application development rather than creating a distinct view to show the note that was saved by the users. This is because the new approach has successfully saved the users' time from checking the other word's details in the phrasebook. Instead, users can directly define their own note or example when they are saving or editing the word's details in the phrasebook.

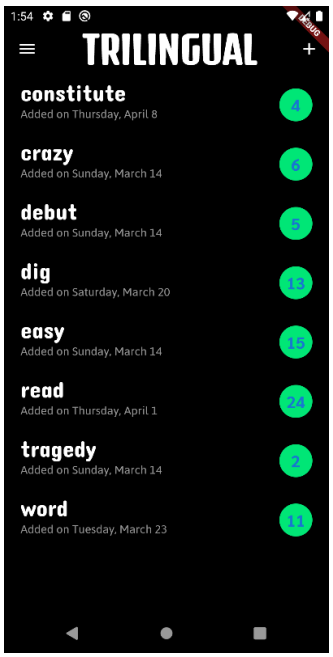


Figure 6.21: Phrasebook Page (1)

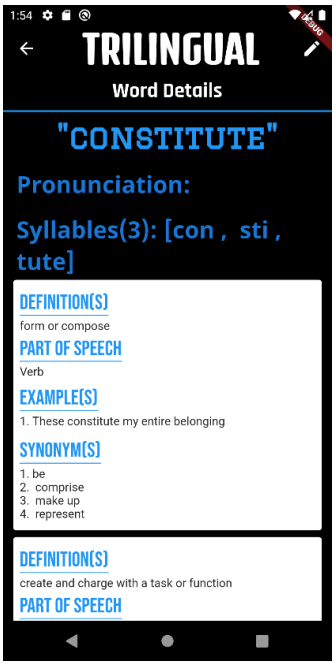


Figure 6.20: Phrasebook Page (2)

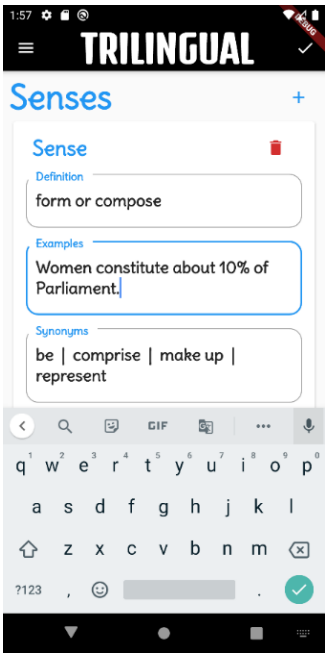


Figure 6.22: Note Page

The process of managing the note is fairly easy to execute as most of the steps are repeating the procedures in managing the phrasebook. Given that a user is on the phrasebook page and he wants to change the note for "constitute". First of all, he needs to press the pencil icon after he has reached the word's details page. Then, he can either edit the note at the example field or remove the note from the example field. Once the configuration has been done, the user is required to press the tick icon in order to save all the changes and a message will pop out to indicate whether the save process has completed successfully. In addition, the steps for adding notes are actually similar to the steps for changing notes except that the add icon has to be pressed in order to add a new sense so that the user can insert their note in the example field.

6.2.6 Relevant API

Table 6.2: API List

No	Endpoint	Description	Parameters
1	https://od-api.oxforddictionaries.com:443/api/v2/entries/en-gb/	Retrieve the information of the word input from the Oxford dictionary	Word input, appID, appKey
2	`/v3/{parent=projects/*}:translateText`	Translate the word input to Malay and Mandarin	Word input
3	https://wordsapiv1.p.rapidapi.com/words/	Get the details of the word input from the Cambridge dictionary	Word input, host, key

There are only three APIs involved in this system implementation and the trilingual mobile dictionary application would not be able to work without the existing of these three APIs. Based on Table 6.2, the first endpoint is used to establish a connection between dictionary application and the source of Oxford dictionary. Due to the budget constraint, the trilingual mobile dictionary application does not authorize access to the translation feature that offered by the Oxford Dictionary API. Therefore, the alternative solution which is Google Translation API was implemented to solve the translation issues and achieve the project scope. Apart from the translation feature, Oxford Dictionary API is well-performed in providing the word's details such as etymology, word property, definition, examples and more. As a result, the varieties of information have provided the users a comprehensive understanding of the particular word.

Furthermore, the purpose of implementing the third endpoint is to expand the corpus of the English vocabulary so that the users are able to refer and compare the word's details from Oxford Dictionary API and Words API. In the Words API, WordNet is used to extract the word definition and relationship while CMU Pronouncing Dictionary is used to derive the word's syllable and rhymes. WordNet is an extensive English lexical database that collaborates with the Cambridge dictionary. It consists of the cognitive synonyms (synsets) which are the arrays of verbs, nouns, adverbs and adjectives that each express a particular meaning. Additionally, the conceptual

semantic and lexical relationships bind synsets together. Moreover, WordNet is similar to a thesaurus that groups words together according to their definitions. However, certain distinctions have to be made before the grouping has started. Firstly, WordNet connects not just word forms – letter strings – but even senses of the vocabularies (WordNet | A Lexical Database for English, 2021). As a consequence, words in the network that are similar to each other are semantically disambiguated. Secondly, WordNet marks semantic relationships between words while the thesaurus groupings do not follow any clear pattern other than the context similarity. In short, the Words API is an intelligent tool to filter the important information that is needed by the users.

6.3 Good Practices

Software development best practices is referring to commercially proven approaches to software development that hit at the root causes of software development problems when applied in tandem. Furthermore, since a standard has been developed and accepted upon by the developers, good standards or habits make communication between them easier. The activities mentioned below have aided the software development process and made the process more effective as the probability of the run-time errors have decreased simultaneously.

6.3.1 Code Simplicity

word_list_bloc_state.dart

Vocab > lib > features > show_saved_words > presentation > bloc > word_list_bloc_state.dart > LoadedWordListBlocState

```dart
import 'package:equatable/equatable.dart';
import 'package:vocab/features/show_saved_words/domain/entity/word_details_summary.dart';

abstract class WordListBlocState extends Equatable {
  const WordListBlocState();
}

class InitialWordListBlocState extends WordListBlocState {
  @override
  List<Object> get props => [];
}

class LoadingWordListBlocState extends WordListBlocState {
  @override
  List<Object> get props => [];
}

class LoadedWordListBlocState extends WordListBlocState {
  final List<WordDetailsSummary> wordList;

  LoadedWordListBlocState(this.wordList);

  @override
  List<Object> get props => [wordList];
}

class ErrorWordListBlocState extends WordListBlocState {
  final String errorMessage;

  ErrorWordListBlocState(this.errorMessage);

  @override
  List<Object> get props => [errorMessage];
}
```

Figure 6.23: Example of Code Simplicity from BLoC

Simple code is the product from successful software development. One of the most significant barriers to a tech company's development and sustainability is the code complexity. This is because production costs escalate exponentially with the rising of the code complexity. Additionally, this exponential relationship arises since more developers are needed to expand and sustain a software application as it gets more complex. In this project, the BLoC pattern was implemented in the source code as it is simple to distinguish the presentation layer from the business logic and resulting the code that is quick to write, test and reuse. In other words, the developer does not need to pass the parameters or data from one class to many classes in order to establish a bridge between them. By using this architecture, the developer is able to:

1. Be aware of the dictionary application's actual state at any point of time.

2. Work as effectively as possible and reusing the elements both within and through the systems.
3. Easily evaluate or test each test case in order to ensure that the dictionary application responds correctly.

6.3.2 Readable Code

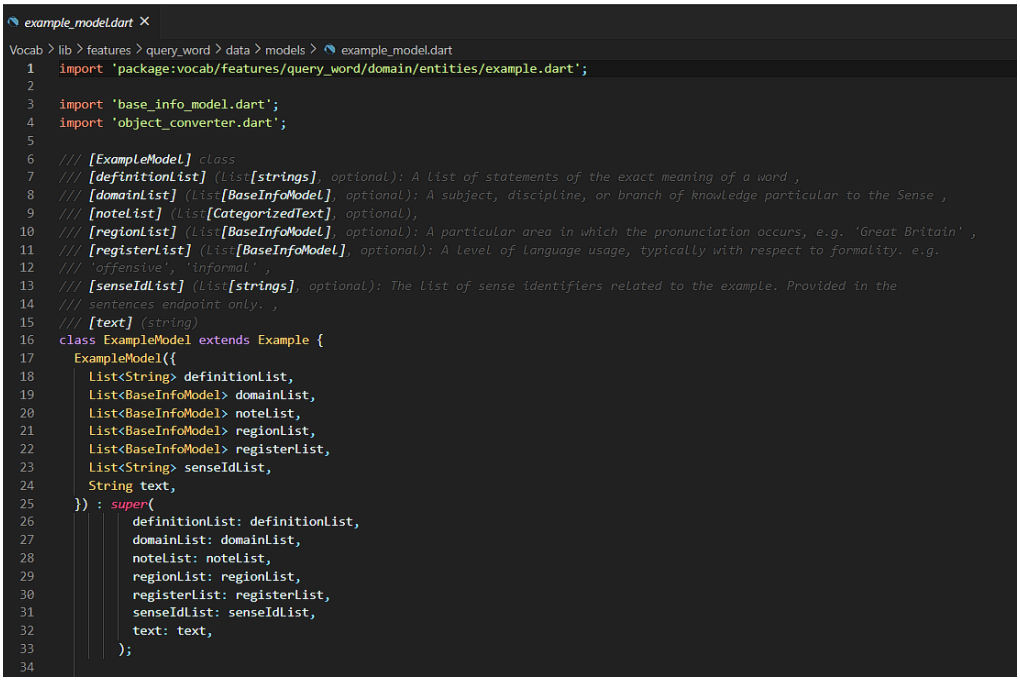


Figure 6.24: Example of Readable Code

One of good practices in software development is writing readable code. This is because coding is a collaborative endeavour as the source code isn't just carrying out a single mission in a vacuum. According to the norm, the developers usually make indentation, comments and spaces in the source code in order to achieve the criteria of readable code. In this project, certain comments were added to explain the reason and function that the line was coded. Moreover, the dart formatter tool was also implemented on the source code in order to make sure that all the source codes were indented and spaced in a consistent pattern. By having this good practice, the developer can:

1. Saves up a lot of time in modifying or changing the source code.
2. Maintains the source code in an efficient and effective way.

3. Helps other developers to understand the code when they are continuing or enhancing the developed source code.

6.3.3 Iterative Software Development

Classic software development cycle adopts the waterfall life cycle, through this approach, development continues linearly through requirement analysis to specification, implementation and unit testing, application testing and system testing. The main flaw with this methodology is that it drives forward danger through time such that it becomes costly to correct failures from earlier stages. An early proposal is likely to be defective with respect to the core specification, and, furthermore, the late detection of proposal deficiencies continues to result in expensive overruns or project cancellation. In this project, an alternative to the waterfall approach which is the iterative and incremental process was implemented. This approach is one of the constant experimentations, invention and execution with increasing iteration pushing the developer to consistently and repeatedly pull nearer the project's objective toward them. By using this approach, it can solve several root causes of the software development process:

1. In the early software development cycle, severe misunderstandings are made clear.
2. Continuous and iterative testing provides an accurate evaluation on the progress of the product.
3. Concrete evidence of the project status can be provided to the stakeholders throughout the software development life cycle.
4. Early detection of inconsistencies within the requirements, designs and implementation.

6.3.4 Software Quality Verification

Verifying the quality of a software is also considered as one of the best practices for the software development process. In this project, it involves designing test cases for each key scenario and each reflecting some factor of the expected functionality of the system. People can determine the reliability of a system by questioning which scenarios have crashed as well as which

scenarios and technology have not yet been implemented. On the other hand, the developer can evaluate the process of continuous and systematic evaluation at every iteration when the system is built iteratively. By performing this practice, it can also solve several root causes of the software development process:

1. Verification and testing are concentrated on the highest-risk region, thereby improving the quality and effectiveness of the software development process.
2. Defects are detected sooner, driving down the expenses of maintaining them dramatically.

CHAPTER 7

TESTING AND EVALUATION

7.1 Introduction

This chapter discusses the different kinds of tests that the trilingual mobile dictionary application has undergone. Specifically, four different tests such as unit testing, integration testing, usability testing and user acceptance testing were performed to ensure the built application is meeting the scopes and criteria that were agreed earlier.

7.2 Functional Testing

Functional testing is a form of system testing which complies that a software system has achieved its functional requirement or specification. Moreover, the functional tests are used to validate the output of a software application by giving informative inputs and comparing it to the functional specifications (Howden, 1980).

7.2.1 Unit Testing

As we know, unit testing is a kind of common functional testing that examines individual software units. The purpose of unit testing is to make sure that each unit of source code is able to work as intended. However, if the units of source code are not able to perform as expected, the developer can save time to catch the defects and bugs early in the development cycle by executing the unit testing. Furthermore, unit testing also aids the developer in comprehending the testing code base and allowing them to make swift changes. In terms of extra advantage, good unit tests can also serve as evidence for the project documentation. In this project, Visual Studio Code is used as a testing framework for both backend and frontend applications while the techniques such as Mockito were implemented to carry out test cases of unit testing. In fact, Mockito can be implemented by creating the dummy classes or interfaces. It enables the developer to define the result of certain function call as well as keep track on the communication between the system and tests.

network_info_test.dart

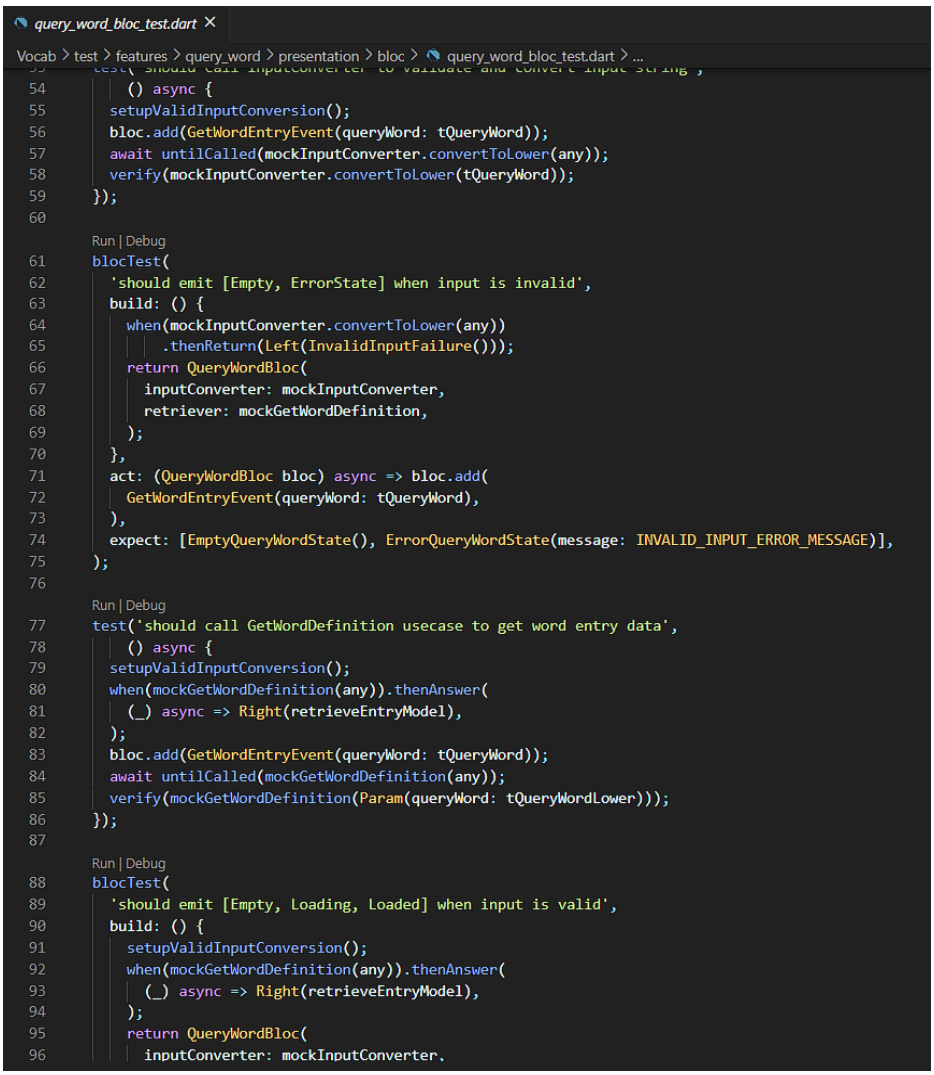
Vocab > test > core > network > network_info_test.dart > main > test("should forward isConnected call to DataConnectionChecker.hasConnection")

```dart
import 'package:data_connection_checker/data_connection_checker.dart';
import 'package:flutter_test/flutter_test.dart';
import 'package:mockito/mockito.dart';
import 'package:vocab/core/network/network_info.dart';

class MockDataConnectionChecker extends Mock implements DataConnectionChecker {}

Run | Debug
void main() {
  NetworkInfoImpl networkInfoImpl;
  MockDataConnectionChecker checker;

  setUp(() {
    checker = MockDataConnectionChecker();
    networkInfoImpl = NetworkInfoImpl(checker);
  });

  Run | Debug
  test(
    'should forward isConnected call to DataConnectionChecker.hasConnection',
    () async {
      final hasConnectionFuture = Future.value(true);
      when(checker.hasConnection).thenAnswer((_) async => hasConnectionFuture);
      final result = await networkInfoImpl.isConnected;
      verify(checker.hasConnection);
      expect(result, await hasConnectionFuture);
    },
  );
}
```

Figure 7.1 Implementation of Mockito in Unit Testing

7.2.2 Integration Testing

Integration testing is another functional testing in which software modules are logically integrated and evaluated together. A standard software project is made up from a number of software modules written by the developer. Thus, the goal of integration testing is to find out the existing bugs or errors when the software modules are communicating and integrating with each other. In this project, integration testing techniques such as stubs and driver and BLoC testing were implemented. In integration testing, stubs and drivers are dummy methods that help with software tests. These methods serve as a replacement for missing models in testing. Even though they do imitate data interaction with the calling module during tests, they do not execute the software module's entire programming logic (Luengruengroj and Suwannasart, 2019). On the other hand, the BLoC testing aims to address the over-requirement on the boilerplate by offering abstractions even for tests. In other words, the developer will no longer have to deal with the traditional stream.



```
query_word_bloc_test.dart
Vocab > test > features > query_word > presentation > bloc > query_word_bloc_test.dart > ...
    () async {
  setupValidInputConversion();
  bloc.add(GetWordEntryEvent(queryWord: tQueryWord));
  await untilCalled(mockInputConverter.convertToLower(any));
  verify(mockInputConverter.convertToLower(tQueryWord));
});

Run | Debug
blocTest(
  'should emit [Empty, ErrorState] when input is invalid',
  build: () {
    when(mockInputConverter.convertToLower(any))
        .thenReturn(Left(InvalidInputFailure()));
    return QueryWordBloc(
      inputConverter: mockInputConverter,
      retriever: mockGetWordDefinition,
    );
  },
  act: (QueryWordBloc bloc) async => bloc.add(
    GetWordEntryEvent(queryWord: tQueryWord),
  ),
  expect: [EmptyQueryWordState(), ErrorQueryWordState(message: INVALID_INPUT_ERROR_MESSAGE)],
);

Run | Debug
test('should call GetWordDefinition usecase to get word entry data',
    () async {
  setupValidInputConversion();
  when(mockGetWordDefinition(any)).thenAnswer(
    (_) async => Right(retrieveEntryModel),
  );
  bloc.add(GetWordEntryEvent(queryWord: tQueryWord));
  await untilCalled(mockGetWordDefinition(any));
  verify(mockGetWordDefinition(Param(queryWord: tQueryWordLower)));
});

Run | Debug
blocTest(
  'should emit [Empty, Loading, Loaded] when input is valid',
  build: () {
    setupValidInputConversion();
    when(mockGetWordDefinition(any)).thenAnswer(
      (_) async => Right(retrieveEntryModel),
    );
    return QueryWordBloc(
      inputConverter: mockInputConverter,
```

Figure 7.2: Implementation of BLoC Test in Unit Testing

7.2.3 Test Case and Results

7.2.3.1 Test Cases List

Table 7.1: Listing of Unit Testing and Integration Testing

No	Test Case ID	Test Case Title	Status
Unit Testing			
1	TC-CORE-01	Network Connection Test	Pass
2	TC-CORE-02	Input Converter Test	Pass
3	TC-U-FEATURE-01	Oxford Dictionary API Endpoint Test	Pass
4	TC-U-FEATURE-02	Oxford Dictionary API Repository Implementation Test	Pass
5	TC-U-FEATURE-03	Words API Endpoint Test	Pass

6	TC-U-FEATURE-04	Words API Repository Implementation Test	Pass
Integration Testing			
1	TC-I-FEATURE-01	Oxford Dictionary API Use Cases Test	Pass
2	TC-I-FEATURE-02	Oxford Dictionary API BLoC Test	Pass
3	TC-I-FEATURE-03	Words API Use Cases Test	Pass
4	TC-I-FEATURE-04	Words API BLoC Test	Pass

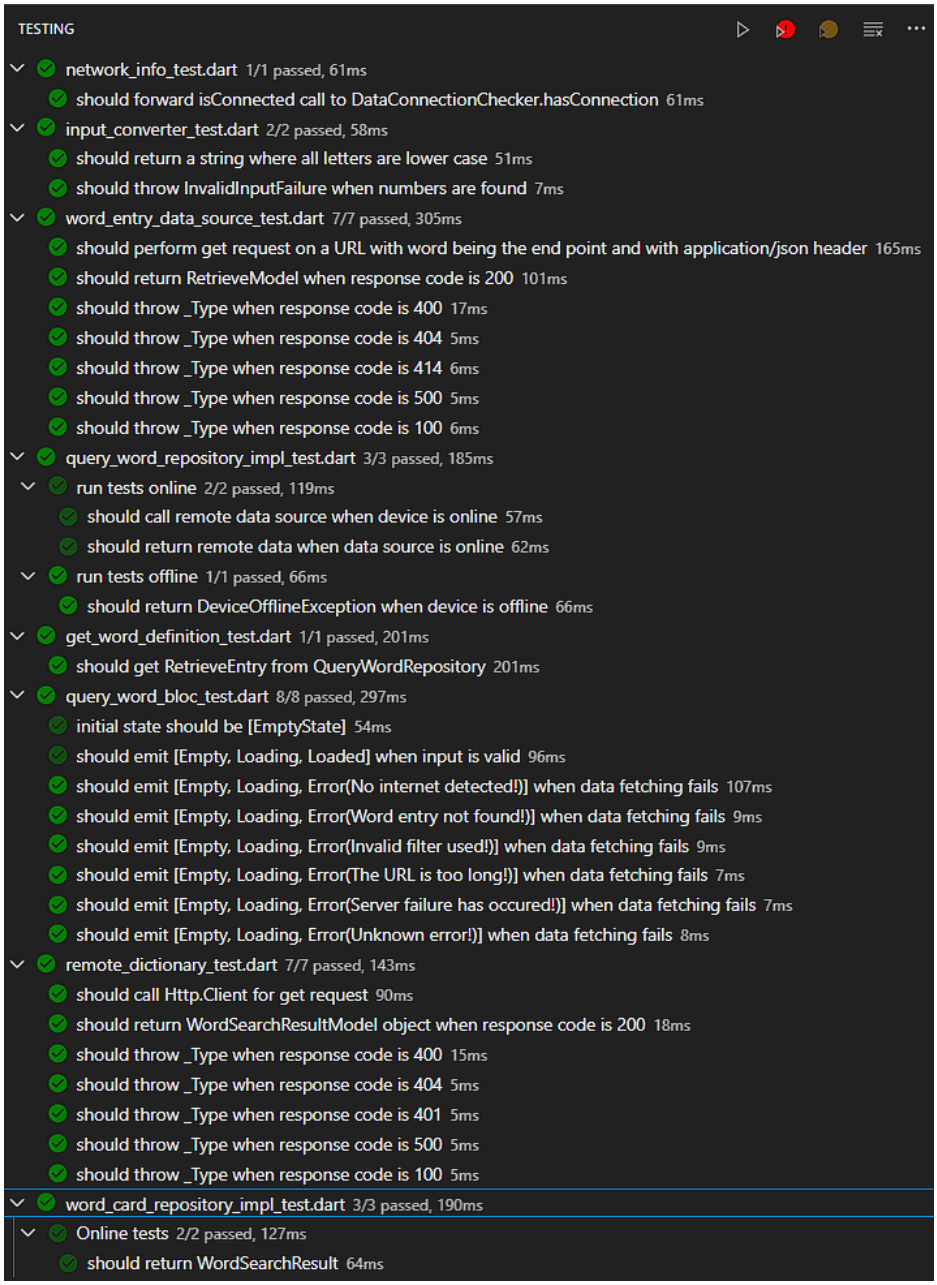


Figure 7.3: List of Passed Tests

7.2.3.2 Unit Test

Table 7.2: Test Case of Network Connection

Test Case ID	TC-CORE-01	**Status**	Pass
Test Case Title	Network Connection Test		
Pre-Condition	N/A		

Test Case Summary	**Test Steps**	**Test Data**	**Expected Result**	**Actual Result**
To examine whether the device is well connected to the network.	1) Establish a connection between the device and network.	N/A	It should forward isConnected call to DataConnectionChecker .hasConnection	It forwarded isConnected call to DataConnectionChecker .hasConnection

Table 7.3: Test Cases of Input Converter

Test Case ID	TC-CORE-02	**Status**	Pass
Test Case Title	Input Converter Test		
Pre-Condition	N/A		

Test Case Summary	**Test Steps**	**Test Data**	**Expected Result**	**Actual Result**
To examine whether the input is able to convert the mix of uppercase and lowercase to lowercase only.	1) The user navigates the route to the search page. 2) The user makes a mix cases of input. 3)The user presses “done” in order to execute the search.	“sTraTEgy”, “dIGest”, “HUmaN”	The search result should show the word “strategy”, “digest” and “human”.	The search result showed the word “strategy”, “digest” and “human”.
To examine	1) The user	“Cons7ru5”,	The search	The search page

whether an alphanumeric input is eligible.	navigates the route to the search page. 2) The user makes an alphanumeric input. 3)The user presses “done” in order to execute the search.	“J1ni4or”, “RaC9”	page should execute the message of invalid input failure which is “Invalid input detected”.	showed “Invalid input detected”.

Table 7.4: Test Cases of Oxford Dictionary API Endpoint

Test Case ID	TC-U-FEATURE-01	**Status**	Pass	
Test Case Title	Oxford Dictionary API Endpoint Test			
Pre-Condition	http.Client has been set up.			
Test Case Summary	**Test Steps**	**Test Data**	**Expected Result**	**Actual Result**
To examine whether the Oxford API is well established.	1) Developer defines a mock http.Client with response 200. 2) Verify the get request.	Fixture path, test query word, entryApiUrl, app_id, app_key, contentType	It should perform a get request on a URL with word being the end point and with application/json header.	It performed a get request on a URL with word being the end point and with application/json header.
To examine whether the Retrieve Model is returned when response code is 200.	1) Developer defines a mock http.Client with response 200. 2) Developer checks whether the Retrieve	Fixture path	It should return Retrieve Model when the response code is 200.	It returned the Retrieve Model when the response code is 200.

	Model is returned.			
To examine the exception code 400.	1) Developer defines the exception type and response code as the parameters. 2) Developer compares the expected result and actual result.	Exception type, code	It should throw an "Invalid Filter Exception" when the response code is 400.	It displayed "Invalid Filter Exception" when the response code is 400.
To examine the exception code 404.	1) Developer defines the exception type and response code as the parameters. 2) Developer compares the expected result and actual result.	Exception type, code	It should throw "Not Found Exception" when the response code is 404.	It displayed "Not Found Exception" when the response code is 404.
To examine the exception code 414.	1) Developer defines the exception type and response code as the parameters.	Exception type, code	It should throw "Too Long URL Exception" when the response code is 414.	It displayed "Too Long URL Exception" when the response code is 414.

	2) Developer compares the expected result and actual result.			
To examine the exception code 500.	1) Developer defines the exception type and response code as the parameters. 2) Developer compares the expected result and actual result.	Exception type, code	It should throw a "Server Exception" when the response code is 500.	It displayed a "Server Exception" when the response code is 500.
To examine the exception code 100.	1) Developer defines the exception type and response code as the parameters. 2) Developer compares the expected result and actual result.	Exception type, code	It should throw "Unknown Exception" when the response code is 100.	It displayed "Unknown Exception" when the response code is 100.

Table 7.5: Test Cases of Oxford Dictionary API Repository Implementation

Test Case ID	TC-U-FEATURE-02	**Status**	Pass
Test Case Title	Oxford Dictionary API Repository Implementation Test		

Pre-Condition	N/A			
Test Case Summary	**Test Steps**	**Test Data**	**Expected Result**	**Actual Result**
To examine whether the remote data source is called when the device is online.	1) The user establishes a connection to the network. 2) The user makes a valid input at the search bar.	Valid input	The user input should trigger a call on a remote data source.	The user input triggered a call on a remote data source.
To examine whether the remote data is returned when the data source is online.	1) The developer checks for the availability of Oxford Dictionary API. 2) The developer makes a valid input at the search bar.	Valid input	The search page should return the information of the vocabulary.	The search page showed the details of the vocabulary.
To examine whether the correct exception is thrown when the device is offline.	1) The user switches off the network connection of the device. 2) The user makes a valid input at the search bar.	Valid input	The search page should return a “Device Offline Exception” message.	The search page displayed the “Device Offline Exception” message.

Table 7.6: Test Cases of Words API Endpoint

Test Case ID	TC-U-FEATURE-03	**Status**	Pass
Test Case Title	Words API Endpoint Test		
Pre-Condition	http.Client has been set up.		

Test Case Summary	**Test Steps**	**Test Data**	**Expected Result**	**Actual Result**
To examine whether the Words API is well established.	1) Developer defines a mock http.Client with response 200. 2) Developer verifies the get request.	Fixture path, test query word, entryApiUrl, header host, header key, contentType	It should perform a get request on a URL with word being the end point and with application/json header.	It performed a get request on a URL with word being the end point and with application/json header.
To examine whether the Word Search Result Model is returned when response code is 200.	1) Developer defines a mock http.Client with response 200. 2) Developer checks whether the Word Search Result Model is returned.	Fixture path	It should return the Word Search Result Model when the response code is 200.	It returned the Word Search Result Model when the response code is 200.
To examine the exception code 400.	1) Developer defines the exception type and response code as the	Exception type, code	It should throw “Invalid Filter Exception” when the response code is 400.	It displayed “Invalid Filter Exception” when the response code is 400.

	parameters. 2) Developer compares the expected result and actual result.			
To examine the exception code 404.	1) Developer defines the exception type and response code as the parameters. 2) Developer compares the expected result and actual result.	Exception type, code	It should throw "Not Found Exception" when the response code is 404.	It displayed "Not Found Exception" when the response code is 404.
To examine the exception code 414.	1) Developer defines the exception type and response code as the parameters. 2) Developer compares the expected result and actual result.	Exception type, code	It should throw "Too Long URL Exception" when the response code is 414.	It displayed "Too Long URL Exception" when the response code is 414.
To examine the exception code 500.	1) Developer defines the exception	Exception type, code	It should throw "Server Exception"	It displayed "Server Exception" when the response code

	type and response code as the parameters. 2) Developer compares the expected result and actual result.		when the response code is 500.	is 500.
To examine the exception code 100.	1) Developer defines the exception type and response code as the parameters. 2) Developer compares the expected result and actual result.	Exception type, code	It should throw "Unknown Exception" when the response code is 100.	It displayed "Unknown Exception" when the response code is 100.

Table 7.7: Test Cases of Words API Repository Implementation

Test Case ID	TC-U-FEATURE-04	**Status**	Pass
Test Case Title	Words API Repository Implementation Test		
Pre-Condition	N/A		

Test Case Summary	**Test Steps**	**Test Data**	**Expected Result**	**Actual Result**
To examine whether the remote database is called when	1) The user establishes a connection to the network. 2) The user	Valid input	The user input should trigger a call on the remote database.	The user input triggered a call on the remote database.

the device is online.	makes a valid input at the search bar.			
To examine whether the Word Search Result is returned when the remote database is online.	1) The developer checks for the availability of Words API. 2) The developer makes a valid input at the search bar.	Valid input	The search page should return the information of the vocabulary.	The search page showed the details of the vocabulary.
To examine whether the correct exception is thrown when the device is offline.	1) The user switches off the network connection of the device. 2) The user makes a valid input at the search bar.	Valid input	The search page should return "Device Offline Exception" message.	The search page displayed the "Device Offline Exception" message.

7.2.3.3 Integration Testing

Table 7.8: Test Case of Oxford Dictionary API Use Cases

Test Case ID	TC-I-FEATURE-01	**Status**	Pass
Test Case Title	Oxford Dictionary API Use Cases Test		
Pre-Condition	N/A		

Test Case Summary	**Test Steps**	**Test Data**	**Expected Result**	**Actual Result**
To examine whether the "Retrieve	1) The developer defines the test	Test query word, fixture	The information of the retrieve entry such as	The information of the retrieve entry such as metadata,

Entry" has successfully passed from "Query Word Repository".	query word and fixture path.	path	metadata, id, word and resultList should get from the Query Word Repository.	id, word and resultList was retrieved from the Query Word Repository.

Table 7.9: Test Case of Oxford Dictionary API BLoC

Test Case ID	TC-I-FEATURE-02	**Status**	Pass	
Test Case Title	Oxford Dictionary API BLoC Test			
Pre-Condition	Device has connected to the network.			
Test Case Summary	**Test Steps**	**Test Data**	**Expected Result**	**Actual Result**
To examine whether the correct initial BLoC state has been displayed.	1) The user navigates the current page to the search page.	N/A	The search page should display "Start by searching a word!".	The search page showed "Start by searching a word!".
To examine whether the backend sequence of the search page has executed correctly.	1) The user navigates to the search page. 2) The user makes a valid input.	Valid input	The search page should first emit the empty query word state, followed by loading the query word state and then loaded query word state.	The search page has first emitted the empty query word state, followed by loading the query word state and then loaded query word state.
To examine whether the backend	1) The user disconnects his device from the	Valid input	Initially, the search page should emit the	Initially, the search page has emitted the empty query

sequence of the search page has executed correctly when the device is offline.	network. 2) The user navigates to the search page. 3) The user makes a valid input.		empty query word state, followed by loading query word state and then display the error message of "No internet detected!"	word state, followed by loading query word state and then display the error message of "No internet detected!"
To examine whether the backend sequence of the search page has executed correctly when the user input is invalid.	1) The user navigates to the search page. 2) The user makes an input at the search bar.	Invalid input/ Inexistence word	Firstly, the search page should emit the empty query word state, followed by loading query word state and then display the error message of "Word entry not found!"	Initially, the search page has emitted the empty query word state, followed by loading query word state and then display the error message of "Word entry not found!"
To examine whether the backend sequence of the search page has executed correctly when the invalid filter is implemented.	1) The developer pre-defines the invalid filter on the http request. 2) The developer navigates to the search page. 3) The developer makes a valid	Valid input, invalid filter	The search page should first emit the empty query word state, followed by loading query word state and then show the error message of "Invalid filter used!".	The search page has first emitted the empty query word state, followed by loading query word state and then show the error message of "Invalid filter used!".

	input at the search bar.			
To examine whether the backend sequence of the search page has executed correctly when the URL is too long.	1) The developer hard codes a long URL on the http request. 2) The developer navigates to the search page. 3) The developer makes a valid input at the search bar.	Valid input, long URL	First of all, the search page should first emit the empty query word state, followed by loading query word state and then show the error message of "The URL is too long!".	First of all, the search page has emitted the empty query word state, followed by loading query word state and then show the error message of "The URL is too long!".

Table 7.10: Test Case of Words API Use Cases

Test Case ID	TC-I-FEATURE-03	**Status**	Pass
Test Case Title	Words API Use Cases Test		
Pre-Condition	N/A		

Test Case Summary	**Test Steps**	**Test Data**	**Expected Result**	**Actual Result**
To examine whether the Word Card Repository has requested when getting word information.	1) The developer defines the test query word and fixture path.	Test query word, fixture path	The word information such as results, syllables, pronunciation and frequency should get from the Word Card Repository.	The word information such as results, syllables, pronunciation and frequency were retrieved from the Word Card Repository.

Table 7.11: Test Case of Words API BLoC

Test Case ID	TC-I-FEATURE-04	**Status**	Pass
Test Case Title	Words API BLoC Test		
Pre-Condition	Device has connected to the network.		

Test Case Summary	**Test Steps**	**Test Data**	**Expected Result**	**Actual Result**
To examine whether the correct initial BLoC state has been displayed.	1) The user navigates the current page to the search page.	N/A	The search page should display "Start by searching a word!".	The search page showed "Start by searching a word!".
To examine whether the backend sequence of the search page has executed correctly.	1) The user navigates to the search page. 2) The user makes a valid input.	Valid input	The search page should first emit the empty query word state, followed by loading query word state and then loaded query word state.	The search page has first emitted the empty query word state, followed by loading query word state and then loaded query word state.
To examine whether the backend sequence of the search page has executed correctly when the device is	1) The user disconnects his device from the network. 2) The user navigates to the search page. 3) The user makes a valid input.	Valid input	Initially, the search page should emit the empty query word state, followed by loading query word state and then display the error message	Initially, the search page has emitted the empty query word state, followed by loading query word state and then display the error message of "No internet

offline.			of “No internet detected!”	detected!”
To examine whether the backend sequence of the search page has executed correctly when the user input is invalid.	1) The user navigates to the search page. 2) The user makes an input at the search bar.	Invalid input/ Inexistence word	Firstly, the search page should emit the empty query word state, followed by loading query word state and then display the error message of “Word entry not found!”	Initially, the search page has emitted the empty query word state, followed by loading query word state and then display the error message of “Word entry not found!”
To examine whether the backend sequence of the search page has executed correctly when the invalid filter is implemented.	1) The developer pre-defines the invalid filter on the http request. 2) The developer navigates to the search page. 3) The developer makes a valid input at the search bar.	Valid input, invalid filter	The search page should first emit the empty query word state, followed by loading query word state and then show the error message of “Invalid filter used!”.	The search page has first emitted the empty query word state, followed by loading query word state and then show the error message of “Invalid filter used!”.
To examine whether the backend sequence of the search	1) The developer hard codes a long URL on the http request.	Valid input, long URL	First of all, the search page should first emit the empty query word	First of all, the search page has emitted the empty query word state, followed by

page has executed correctly when the URL is too long.	2) The developer navigates to the search page. 3) The developer makes a valid input at the search bar.		state, followed by loading query word state and then show the error message of "The URL is too long!".	loading query word state and then show the error message of "The URL is too long!".

7.3 Non-Functional Testing

Non-functional testing was carried out in order to examine a software application's non-functional attributes such as security, reliability, usability and flexibility and more. It's intended to assess a system's readiness by using non-functional criteria that are not covered by functional testing. In this project, usability is the main non-functional aspect that is being evaluated for this dictionary application.

7.3.1 Usability Testing

The System Usability Scale is a reliable "quick and dirty" tool for accessing the usability of an application. It has become a software industry standard, due to its simple scaling characteristic in small sample size. In this project, five users from different school levels were selected to participate in the software's usability test as the school students are the sole subject of this usability test. According to Table 7.12, five users were given eight scenarios in total and they were instructed to test them according to the defined scenarios. After completing the usability test, all users were asked to fill in the user satisfaction survey.

7.3.1.1 Standard Operating Procedure for Usability Testing

The usability testing was carried out by each participant under the supervision of the developer. If the participants have any concern or encounter any problems throughout the testing process, the developer will assist them so that

the testing will be completed successfully. On the other hand, the list below showed the entire procedure of the usability testing.

Steps of conducting usability testing:

1. The participants were initially asked to understand all the eight test scenarios that had been planned.
2. Next, the participants need to follow and perform the task as described in Table 7.12.
3. The developer had an eye on how the defined tasks were being executed and was ready to answer any inquiries from the participants.
4. Lastly, the participants must complete the user satisfaction form after finishing all the scenarios that stated in Table 7.12.

7.3.1.2 Usability Test Scenario List

Table 7.12: Listing of Usability Testing Scenarios

No	Test Scenario Title	Test Scenario Description
1	View daily new vocabulary	Imagine you are a great learner who wants to learn at least one new vocabulary in three languages on a daily basis. Therefore, you are required to load in the application in order to view some daily new vocabularies. Are you able to find the daily new vocabulary section?
2	Search a word	Now, you are intended to search a vocabulary so that you are able to understand the details of the searched word. How do you navigate to the search page and what actions do you take to search a vocabulary?
3	Lookup the details of a word in two different dictionaries	Once you have done searching a word, you are now able to check up the information from two different dictionaries which are Oxford dictionary and Cambridge dictionary.

		How would you view the details of two different dictionaries?
4	Favourite a vocabulary into the phrasebook	Let say you are interested in saving a word and its details to the phrasebook so that you may be able to refer back to what you have learnt so far. What is your action to favourite a vocabulary into the phrasebook or so-called favourite list?
5	Add a self-defined vocabulary and its' details to the phrasebook	Given that you have discovered a vocabulary which is not covered in both of the dictionaries. Thus, you wish to add a record of the word and its details to the phrasebook. Are you able to add a self-defined vocabulary and its details to the phrasebook?
6	Update or delete a vocabulary	Imagine you wish to update some information about the vocabulary followed by saving the changes. With your updated vocabulary details, are you able to see the results of changes? Next, you want to delete the particular vocabulary from the phrasebook. Are you able to remove or delete the vocabulary from the phrasebook?
7	Create a note to a vocabulary	Assume that the examples provided by the dictionary were too difficult to understand. Hence, you can overwrite the provided example

		by implementing your own idea in order to ease yourself in learning a new word. What is the action would you take to create a note for yourself?
8	Navigate between the pages	Imagine that you want to check up the favourited vocabulary in the phrasebook after you have done searching a word. In order to do so, what will be your action?

7.3.1.3 Result of System Usability Scale (SUS)

Table 7.13: Table of System Usability Scale Score

Participant	**Score of Statement**										**SUS Score**
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	
Participant #1	2	4	3	3	4	4	3	3	2	3	77.5
Participant #2	3	3	4	4	4	4	3	4	2	3	85
Participant #3	3	3	3	2	4	4	3	3	3	4	80
Participant #4	3	4	4	3	4	4	2	2	2	4	80
Participant #5	3	3	3	3	3	3	3	3	3	3	75
Average SUS Score											**79.5**

Table 7.14: Table of Analysing the Helpfulness of the Application

	Degree of Helpfulness of the Application				
	Not Helpful at All 1	**Not Helpful 2**	**Neutral 3**	**Helpful 4**	**Very Helpful 5**

Number of Participants			1	3	1

The results of the system usability scale were collected from Appendix C and then tabulated in the Table 7.13 after the usability testing was completed. Despite the participants found some unsatisfied parts of the application, the participants have rated an average of SUS score which is 79.5 to the overall of the mobile application. In addition, Table 7.14 has shown that three participants felt that the dictionary application is helpful to them followed by one felt that it is very helpful and another one found that it is neutral.

Table 7.15: Likes, Dislikes and Suggestions for Some Features of Mobile Application

Favourite Parts in the Application	**Disfavoured Parts in the Application**	**Suggestions on the Mobile Application**
• Informative details • Simple user interface to understand • Highly customizable at the phrasebook • Able to learn new vocabulary on a daily basis	• Font size of some wordings is too small • Too much information is included • Some unknown terminology is used such as API	• Should have some sort of cloud backup on the phrasebook • Should include Tamil as it is one of the major languages in Malaysia • Should have more information for the translated languages • Should include a history list for user to look back what they have searched • Should contain the vocal pronunciation of the word • Should include notification to remind user for learning daily new vocabulary

According to the Section B of user satisfaction survey from Appendix C, the participants' favourite features, least favourite features and comments were

aggregated into Table 7.15. Overall, some participants stated that they prefer certain features of the application because the features have achieved the aspect of user-friendly. Whereas, there are also some participants who felt that a few features are too difficult to understand or interpret when using the dictionary application. Furthermore, resolution steps were taken in order to resolve all the disfavoured features in the application after the user satisfaction surveys have been reviewed and analysed. In terms of suggestions, most of it is irrelevant as those recommendations are not within the project scope. However, it can become a reference for the feature enhancement in the future project development or upgrade.

In conclusion, the participants responded positively to the usability evaluation. In fact, the majority of the participants were comfortable with the user interface as well as the user experience throughout the execution of the test scenarios with the developed application.

7.3.2 User Acceptance Testing

User acceptance testing is the final stage of testing which was performed by the end user to verify and accept the software product before it is moved to the production area. In this project, user acceptance testing was conducted with five invited participants. Also, a list of predefined test cases was given to every participant and they were supposed to complete all of them in order to ensure that the application is user-friendly enough and has included all the features in project scope.

Based on the rules of user acceptance testing, the developer will not respond to any questions from the participants unless they have no other idea to continue on the subsequent test cases. Moreover, participants were also instructed to record the start end and end time for each test scenario before and after it is completed.

7.3.2.1 UAT Test Cases List

Table 7.16: Listing of UAT Test Description

No	Test Case ID	Tested Module	Test Description
1	UAT-01	View daily new	Able to check in the application

		vocabulary	successfully
			Able to allocate the daily new vocabulary section
2	UAT-02	Search vocabulary	Able to navigate from home page to search page
			Able to search multiple words and view the displayed information
			Able to lookup the details from two different dictionaries
3	UAT-03	Manage phrasebook/ favourite list	Able to save the word and its' details to the phrasebook
			Able to navigate from search page to phrasebook page
			Able to view the word and its' details that has just saved to the favourite list
			Able to edit the content of the word details according to personal preference
			Able to delete the saved vocabulary from the favourite list
			Able to add a self-defined word and its' details to the phrasebook
4	UAT-04	Manage note	Able to add the "example" of word details in phrasebook according to his/her own idea
			Able to amend the "example" of word details in phrasebook based on his/her own idea
			Able to delete or clear the "example" of word details in phrasebook
			Able to navigate from phrasebook page to home page

7.3.2.2 Result of User Acceptance Testing

Table 7.17: Test Result for UAT

Test Module	Test Case ID	Number of tests conducted	Number of tests passed	Feedback
View daily new vocabulary	UAT-01	5	5	-
Search vocabulary	UAT-02	5	5	-
Manage phrasebook/ favourite list	UAT-03	5	5	-
Manage note	UAT-04	5	5	-

Based on the results from Appendix E, all participants have successfully accomplished the UAT without too much assistance from the developer. During the user acceptance testing, the developer was able to get a deeper insight on the application's workflow and it also helped the developer to make sure that the application consists of the user interface in which it is able to offer a good user experience to the end users. Additionally, a small experiment was conducted with every participant in order to prove the achievement of the objective that stated earlier which is to reduce at least 10% of time usage in searching a particular vocabulary. As a result, all participants spent an average time of 7 seconds to search the details of the vocabulary by using the developed dictionary application. However, they consumed an average time of 3 minutes when they asked to search the same word by using a paper dictionary. In short, the trilingual mobile dictionary is a more efficient way to look up the details of the vocabulary.

7.4 Linkages Between Functional Requirements and Test Cases

Table 7.18: Achievement of Functional Requirements

Functional Requirement ID	Functional Requirement Statement	Use Case ID	Use Case Name	Test Case ID	Status

FR01	The system shall allow users to search the vocabulary in English and translate it to Mandarin and Malay.	UC01	Search and translate vocabulary	UAT-02, TC-CORE-02	Pass
FR02	The system shall allow users to view the detailed information of the vocabulary such as word properties, grammars, related words and sample sentences.	UC02	View information of vocabulary	UAT-02, TC-CORE-01	Pass
FR03	The system shall allow users to add more than one vocabulary into the phrasebook.	UC03	Add vocabulary into phrasebook	UAT-03	Pass
FR04	The system shall allow users to edit the vocabulary's content or delete the	UC03	Add vocabulary into phrasebook	UAT-03	Pass

	vocabulary from the phrasebook.				
FR05	The system shall allow users to attach notes or examples to vocabulary.	UC04	Attach note to vocabulary	UAT-04	Pass
FR06	The system shall allow users to edit or delete the note or example that is attached along with the vocabulary.	UC04	Attach note to vocabulary	UAT-04	Pass
FR07	The system shall allow users to view the daily new vocabulary on the main page of the application.	UC05	View daily new vocabulary	UAT-01	Pass
FR08	The system shall allow users to slide the daily new vocabulary on the main page of the application.	UC05	View daily new vocabulary	UAT-01	Pass

FR09	System administrator shall be able to update and maintain the vocabulary version.	UC06	Update and maintain vocabulary	TC-U-FEATURE-01, TC-U-FEATURE-02, TC-U-FEATURE-03, TC-U-FEATURE-04,	Pass

CHAPTER 8

CONCLUSION AND DISCUSSION

8.1 Introduction

This chapter has concluded how the trilingual mobile dictionary application tackles the challenges and reaches the project's objectives. Furthermore, the limitation of the application and some suggestions on the application's future enhancement were also listed in Section 8.3.

8.2 Conclusion

A trilingual mobile dictionary application was successfully deployed after going through 6 months of software development life cycle phases which is from the planning stage to testing stage. The developed application has achieved its initial planned objectives such as the list shown below.

1. To decrease at least 10% of time usage in searching a particular vocabulary.
2. To develop a functionable trilingual mobile dictionary application that consists of interactive tools such as searching, favourite list, related words, word properties and add note.
3. To compile and expand a comprehensive corpus of English, Mandarin and Malay vocabularies in the trilingual mobile dictionary application.

As a result, this application is now able to assist Malaysia school students by making it easy for them to search and understand the meaning of a vocabulary. Apart from this, the dictionary application also enables their end users to access the details or information from the two largest dictionary organizations in the world which are Oxford dictionary and Cambridge dictionary. The details can be viewed by the users have included etymology, sense, part of speech, example, definition, pronunciation, syllable and more. Moreover, users can fully customize their phrasebook by saving the interested vocabulary, editing or adding the saved word's content and removing the vocabulary from the phrasebook at any time. Last but not least, the trilingual mobile dictionary

application is also playing a role to help Malaysia school students to form a good habit of study as it provides the daily new vocabulary feature for the users to learn different words on a daily basis.

8.3 Limitations and Recommendations

Table 8.1: Limitations and Recommendations

No	Limitations	Recommendations
1	No notification for learning daily new vocabulary	In order to further enhance the trilingual mobile dictionary application, it should have a notification which can notify or remind the users to check out the new vocabulary on a daily basis. By having this feature, it can also help the users to cultivate the habit of learning.
2	No cloud backup for storing the details of the phrasebook	Cloud backup such as Google Drive should be implemented to store the data in phrasebook. This is because users might change their device from time to time and the saved information can only be transferred when there is a cloud backup. By having this feature, the dictionary application can achieve the aspect of reliability and it also improves the user's confidence when they are using the application.
3	No extra information in Mandarin and Malay	Due to the authority issue when accessing the full-featured Oxford Dictionary API, the current version of the trilingual mobile dictionary

		application is only consisting of the translation data. Instead, it should have information such as example sentences for Malay and Mandarin, grammar for Malay and Mandarin and etc. In the future enhancement, the dictionary application should have the access token to the full-featured Oxford Dictionary API so that the users are able to explore deeper into the details of Mandarin and Malay.
4	Absent of quizzes section	According to the research, quizzes are the fastest way to help the users to revise and retain the information that they have learnt earlier. On the other hand, quizzes can also assist the users in identifying the knowledge gap so that they can perform some corrective actions on their study (Narloch, Garbin and Turnage, 2006). By having this feature, it can analyse and keep track on the improvement of the user by weekly basis and then generate a set of quizzes that can succour the users to minimize the knowledge gap.
5	Absent of Tamil in the dictionary application	Tamil is one of the major languages in Malaysia. Hence, if the dictionary application is able to include the information of Tamil, it can attract more users to utilize it.

		On the other hand, the dictionary application can be a comprehensive source for the people who intend to learn Tamil by themselves. As an indirect effect, the application might be able to improve the Malaysians' linguistic knowledge as the users are able to learn a new language from the trilingual mobile dictionary application.
6	Absent of history list or recent searches in the application	Majority of the people can easily forget what they have done for a few days ago, therefore a history list or recent searches list must be included for them to recall the previous searches. Sometimes, users might want to favourite the vocabulary to the phrasebook but they just forgot to do so and could not remember the searched word anymore as it is considerably unfamiliar to the user. In order to overcome this issue, history list or recent searches list can be implemented.
7	No vocal pronunciation of the vocabulary	As a feature enhancement, vocal pronunciation for English, Malay and Mandarin need to be covered in the upgraded version of dictionary application as the current version of trilingual mobile dictionary application has only included the words' syllable in

		English. By having this feature, users can learn to pronounce the word accurately among the three languages. As a result, Malaysians can sharpen their knowledge and skills in word's pronunciation.
8	Unable to save the translated word to the phrasebook	Due to the limitation of time and Oxford Dictionary API, the translated word could not be saved in the phrasebook. In order to improvise the dictionary application, this feature has to be included as it plays a fairly vital role in saving the users' time. This is because users do not need to spend time on repeating the search process in the search page in order to retrieve the translated word.

Even though the trilingual mobile dictionary application has fulfilled all of the project objectives and project scopes, the dictionary application still has certain shortcomings that can be addressed in the future. Based on Table 8.1, it has illustrated some potential enhancements and suggestions that could extend more features to the end users and polish up the usability of the dictionary application at the same time.

REFERENCES

Anon 2021. *bloc | Dart Package*. [online] Available at: <https://pub.dev/packages/bloc> [Accessed 25 Feb. 2021].

Anon 2020. *Evolutionary Prototyping Methodology - Welcome to Shopper's Mart*. [electronic print] Available at: <https://shopeemart.weebly.com/sdlc.html> [Accessed 2 Sep. 2020].

Anon 2020. *Malaysia: mobile OS market share 2019 | Statista*. [online] Available at: <https://www.statista.com/statistics/929845/malaysia-mobile-os-share/> [Accessed 1 Sep. 2020].

Anon 2020. *Prototyping Methodology – The WritePass Journal : The WritePass Journal*. [electronic print] Available at: <https://writepass.com/journal/2012/12/evaluation-of-mendix-as-a-rapid-application-development-rad-tool/prototyping-methodology/> [Accessed 2 Sep. 2020].

Anon 2020. *Rapid Application Development (RAD) - Simplifier*. [electronic print] Available at: <https://www.simplifier.io/en/why-low-code/rapid-application-development/> [Accessed 2 Sep. 2020].

Anon 2020. *Ruby on Rails - Introduction - Tutorialspoint*. [online] Available at: <https://www.tutorialspoint.com/ruby-on-rails/rails-introduction.htm> [Accessed 30 Aug. 2020].

Anon 2020. *Ruby Programming Language*. [online] Available at: <https://www.ruby-lang.org/en/> [Accessed 30 Aug. 2020].

Anon 2020. *SDLC - Waterfall Model - Tutorialspoint*. [electronic print] Available at: <https://www.tutorialspoint.com/sdlc/sdlc_waterfall_model.htm> [Accessed 2 Sep. 2020].

Anon 2018. *Software Engineering | Prototyping Model - GeeksforGeeks*. [online] Available at: <https://www.geeksforgeeks.org/software-engineering-prototyping-model/> [Accessed 6 Jul. 2020].

Anon 2020. *Software Engineering | Prototyping Model - GeeksforGeeks*. [online] Available at: <https://www.geeksforgeeks.org/software-engineering-prototyping-model/> [Accessed 6 Jul. 2020].

Anon 2020. *Some Obstacles to Vocabulary Development | Adolescent Literacy Topics A-Z | AdLit.org*. [online] Available at: <http://www.adlit.org/article/19690/> [Accessed 4 Jul. 2020].

Anon 2020. *SQLite Home Page*. [online] Available at: <https://www.sqlite.org/index.html> [Accessed 30 Aug. 2020].

Anon 2017. Suruhanjaya Komunikasi dan Multimedia Malaysia Malaysian Communications and Multimedia Commission HAND PHONE USERS SURVEY 2017 STATISTICAL BRIEF NUMBER SEVENTEEN. [online] Available at: <http://www.mcmc.gov.my>. [Accessed 1 Sep. 2020].

Anon 2020. *Top 6 Best Dictionary APIs (for Developers in 2020) [38+ Reviewed]*. [online] Available at: <https://rapidapi.com/blog/dictionary-apis/> [Accessed 30 Aug. 2020].

Binani, S., Gutti, A. and Upadhyay, S., 2016. *ISSN: 2394-4714 Foundation of Computer Science FCS*. [online] *Communications on Applied Electronics (CAE)*, Available at: <www.caeaccess.org> [Accessed 21 Aug. 2020].

Carr, M. and Verner, J., n.d. *Prototyping and Software Development Approaches*.[online] Available at:< https://www.researchgate.net/profile/June_Verner/publication/238117215_Prototyping_and_Software_Development_Approaches/links/543fab1d0cf2fd72f99c8bc7.pdf> [Accessed 21 Aug. 2020]

Chan, A.Y.W., 2011. *The use of a monolingual dictionary for meaning determination by advanced Cantonese ESL learners in Hong Kong.* [online] Available at: < https://www.researchgate.net/publication/277482016_The_Use_of_a_Monolingual_Dictionary_for_Meaning_Determination_by_Advanced_Cantonese_ESL_Learners_in_Hong_Kong> [Accessed 7 Jul. 2020].

Charu, S., 2014. Comparison of SQL with HiveQL. *International Journal for Research in Technological Studies*, 1(1), pp.2348–1439.

Dutta, G., 2018. *WATERFALL MODEL*. [online]Available at: <https://www.academia.edu/38314403/WATERFALL_MODEL> [Accessed 8 Aug. 2020].

Eason, O.K., n.d. *Information Systems Development Methodologies Transitions: An Analysis of Waterfall to Agile Methodology Recommended Citation*. [online] Available at:< https://scholars.unh.edu/honors/286 >[Accessed 8 Aug. 2020].

Eisenberg, A. and Melton, J., 2000. SQL standardization: The next steps. *SIGMOD Record (ACM Special Interest Group on Management of Data)*, 29(1), pp.63–67.

Feoktistov, I., 2019. *Agile Software Development Lifecycle Phases Explained*. [online] Available at: <https://relevant.software/blog/agile-software-development-lifecycle-phases-explained/> [Accessed 2 Sep. 2020].

Feuerlicht, G., Pokorný, J. and Richta, K., 2009. Object-Relational Database Design: Can Your Application Benefit from SQL:2003? In: *Information Systems Development*. [online] Springer US.pp.975–987. Available at: <https://link.springer.com/chapter/10.1007/978-0-387-78578-3_30> [Accessed 21 Aug. 2020].

Fotache, M., 2013. NoSQL and SQL Databases for Mobile Applications. Case Study: MongoDB versus PostgreSQL. [online] Available at: <http://www.econ.ubb> [Accessed 21 Aug. 2020].

Hirschberg, M.A., 2016. *STN 2-1 Topic: Rapid Application Development (RAD)*. [online] Available at: <http://www.stc98.org> [Accessed 21 Aug. 2020].

Howden, W.E., 1980. Functional Program Testing. *IEEE Transactions on Software Engineering*, SE-6(2), pp.162–169.

Khalil, M.A. and Kotaiah, B., 2018. Implementation of agile methodology based on SCRUM tool. In: *2017 International Conference on Energy, Communication, Data Analytics and Soft Computing, ICECDS 2017*. [online] Institute of Electrical and Electronics Engineers Inc.pp.2351–2357. Available at: <https://ieeexplore.ieee.org/document/8389872/> [Accessed 8 Aug. 2020].

Kukulska-Hulme, A. and Shield, L., n.d. *An overview of mobile assisted language learning: From content delivery to supported collaboration and interaction Journal Item An Overview of Mobile Assisted Language Learning: from content delivery to supported collaboration and interaction.* [e-journal] 20(3), pp271-289. http://dx.doi.org/doi:10.1017/S0958344008000335

Luengruengroj, P. and Suwannasart, T., 2019. Stubs and drivers generator for class integration testing using sequence and class diagrams. In: *ACM International Conference Proceeding Series*. [online] New York, NY, USA: Association for Computing Machinery.pp.115–119. Available at: <http://dl.acm.org/doi/10.1145/3374549.3374572> [Accessed 24 Mar. 2021].

María del Pilar Ángeles, 2020. *Disadvantages of NoSQL technology - NOSQL Systems | Coursera*. [online] Available at: <https://www.coursera.org/lecture/nosql-databases/disadvantages-of-nosql-technology-Vepzo> [Accessed 1 Aug. 2020].

Narloch, R., Garbin, C.P. and Turnage, K.D., 2006. Benefits of Prelecture Quizzes. *Teaching of Psychology*, [online] 33(2), pp.109–112. Available at: <http://journals.sagepub.com/doi/10.1207/s15328023top3302_6> [Accessed 30 Mar. 2021].

Naz, R. and Khan, M.N.A., 2015. Rapid applications development techniques: A critical review. *International Journal of Software Engineering and its Applications*, 9(11), pp.163–176.

Parker, Z., Poe, S. and Vrbsky, S. V., 2013. Comparing NoSQL MongoDB to an SQL DB. In: *Proceedings of the Annual Southeast Conference*. [online] New York, New York, USA: ACM Press.p.1. Available at: <http://dl.acm.org/citation.cfm?doid=2498328.2500047> [Accessed 21 Aug. 2020].

Pecherzewska, A., & Knot, S. (2007). *Review of Existing EU Projects Dedicated to Dyslexia, Gaming in Education and M-Learning. WR08 Report to CallDysc Project. - References - Scientific Research Publishing.* [online] Available at: <https://www.scirp.org/(S(351jmbntvnsjt1aadkposzje))/reference/ReferencesPapers.aspx?ReferenceID=2206826> [Accessed 7 Jul. 2020].

Raut Professor, A.B., 2017. *NOSQL Database and Its Comparison with RDBMS*. [online] *International Journal of Computational Intelligence Research*, Available at: <http://www.ripublication.com>. [Accessed 21 Aug. 2020].

Schmitt, N., 2008. *Instructed second language vocabulary learning. Language Teaching Research.* [online] Available at: < https://scholar.google.com.my/scholar?q=Instructed+second+language+vocabulary+learning.+Language+Teaching+Research.&hl=en&as_sdt=0&as_vis=1&oi=scholart> [Accessed 7 Jul.2020].

Sharples, E., 2007. *Big Issues in Mobile Learning: Report of a workshop by the Kaleidoscope Network of Excellence.* [online] (January 2006), p.36. Available at: <http://hal.archivesouvertes.fr/docs/00/19/02/54/PDF/Sharples_Big_Issues.pdf%5Cnhttp://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.183.9628> [Accessed 7 Jul.2020].

Sunner, D., 2017. Agile: Adapting to need of the hour: Understanding Agile methodology and Agile techniques. In: *Proceedings of the 2016 2nd International Conference on Applied and Theoretical Computing and Communication Technology, iCATccT 2016*. [online] Institute of Electrical and Electronics Engineers Inc.pp.130–135. Available at: <http://ieeexplore.ieee.org/document/7911978/> [Accessed 8 Aug. 2020].

APPENDICES

APPENDIX A: Feedback Survey Form

Trilingual Mobile Dictionary Application Survey

Thank you for agreeing to take part in this survey. I am Wong Szee Yung, a final year software engineering student from Universiti Tunku Abdul Rahman. I am currently working on my final year project which is developing a trilingual mobile dictionary application. The purpose of this questionnaire is to collect some nominal, ordinal, internal and ratio data that can be used to demonstrate some of the basic statistical methods for analyzing quantitative data. The aim of this study is to determine whether the trilingual mobile dictionary application will bring any benefits to primary and secondary school students. The individual responses will be anonymous and the data collected will be used for only this final year project. Your participant is entirely voluntary. If there are questions you do not feel comfortable answering, please skip them. Thank you in advance for your cooperation.

1. 1. Gender

   Mark only one oval.

   - ◯ Male
   - ◯ Female
   - ◯ Prefer not to say

2. 2. Age

   Mark only one oval.

   - ◯ 0 to 6
   - ◯ 7 to 12
   - ◯ 13 to 18
   - ◯ 19 and above

3. 3. What are the problem(s) you faced when you are dealing with the unfamiliar vocabulary in Malay, English and Chinese?

Tick all that apply.

- [ ] Used up a lot of time in flipping and checking a particular vocabulary among 3 languages from paper dictionary
- [ ] Trilingual technology-based dictionary is not available in the market
- [ ] Mobile dictonary only provided superficial information of a word
- [ ] Google translate
- [ ] Other: _____

4. 4. What is your main purposes of consulting the dictionary?

Tick all that apply.

- [ ] To check the meaning of the word
- [ ] To check how the word can be applied in a sentence
- [ ] To understand the word properties
- [ ] To understand the actual pronunciation of the word
- [ ] Other: _____

5. 5. Do you look up more words when using paper dictionary or when consulting technology-based dictionary?

Mark only one oval.

- ◯ Paper dictionary
- ◯ Technology-based dictionary (Mobile dictionary, Web dictionary)

6. 6. How do you prefer to check the word that you are unfamiliar with?

Tick all that apply.

- [ ] Consult the paper dictionary
- [ ] Translate the vocabulary via Google Translate
- [ ] Search the vocabulary through web browser
- [ ] Search the vocabulary via mobile dictionary application
- [ ] Other: _____

7. 7. What is the mobile dictionary application that you are currently using?

Mark only one oval.

- ◯ Oxford Dictionary of English
- ◯ Oxford Chinese Dictionary
- ◯ Dictionary & Translator Free
- ◯ Malay-English Dictionary
- ◯ Malay Chinese Dictionary eKamus
- ◯ dict.cc
- ◯ None, more like web-based apps like google translate and some other random website
- ◯ Youdao
- ◯ Other: _____

8. 8. How often do you use mobile dictionary application per month?

Mark only one oval.

- ◯ 0 time
- ◯ 1 to 5 times
- ◯ 6 to 10 times
- ◯ 11 to 15 times
- ◯ 16 to 20 times

9. 9. Trilingual mobile dictionary application is an application which enable user to translate and look up the meaning among different languages such as Malay, Chinese and English. Apparently, Trilingual mobile dictionary application is more efficient and cost effective to search the meaning of the word compared to paper dictionary.

Mark only one oval.

	1	2	3	4	5	
Strongly disagree	◯	◯	◯	◯	◯	Strongly agree

10. 10. What is the mobile operating system that you are currently use?

Mark only one oval.

- ◯ iOS
- ◯ Andriod OS
- ◯ BlackBerry OS
- ◯ Other: _____

11. 11. What is the main criteria that you will be considered before downloading a mobile dictionary application?

Mark only one oval.

- ◯ User Interface
- ◯ Storage size of the application
- ◯ Dictionary sources
- ◯ Features that provided in the application
- ◯ user-friendliness
- ◯ Other: _____

12. 12. What are the features you would expect to see in a trilingual mobile dictionary application?

Tick all that apply.

- ☐ Search vocabularies
- ☐ Add vocabularies into the phrasebook
- ☐ Check search history
- ☐ Add note to a word
- ☐ Display daily new vocabulary
- ☐ Sentence structure
- ☐ Can be accessed using shortcut while using other applications and close it without leaving other applications

Other: ☐ _____

APPENDIX B: User Satisfaction Survey

User Satisfaction Survey

Participant No:	**Participant Name:**

Section A

Table 7.13 User Satisfaction Survey

Statement	**Strongly Disagree 1**	**Disagree 2**	**Neutral 3**	**Agree 4**	**Strongly Agree 5**
I think that I would like to use this application for checking the details of a vocabulary.					
I found that this is an unnecessarily complex application.					
I think the application is easy for me to use.					
To use this application, I feel I will need the assistance of a technical individual.					
I think that this application is simple to navigate without a lot of backtracking or data re-entry.					
I think there are too much inconsistencies in this application.					
I believe that most people will learn to use					

this application without consuming too much time.					
I found the application very awkward to use.					
I feel very confident in using the application.					
Before I could get started with this application, I had to learn a lot of things.					

Section B

Q1: How useful is this application to you in searching the details of a vocabulary? Please circle on the corresponding rate.

Not Helpful at All 1	**Not Helpful 2**	**Neutral 3**	**Helpful 4**	**Very Helpful 5**

Q2: Which feature in this application you prefer the most?

_____

Q3: Which feature in this application you dislike the most?

_____

Q4: What is your final comment or inquiry for this application?

_____

APPENDIX C: Results from Usability Testing

Participant No: 1	**Participant Name: Ding Shie Chin**

Section A

Table 7.13 User Satisfaction Survey

Statement	**Strongly Disagree 1**	**Disagree 2**	**Neutral 3**	**Agree 4**	**Strongly Agree 5**
I think that I would like to use this application for checking the details of a vocabulary.			✔		
I found that this is an unnecessarily complex application.	✔				
I think the application is easy for me to use.				✔	
To use this application, I feel I will need the assistance of a technical individual.		✔			
I think that this application is simple to navigate without a lot of backtracking or data re-entry.					✔
I think there are too much inconsistencies in this application.	✔				
I believe that most people will learn to use this application without consuming too much				✔	

time.					
I found the application very awkward to use.		✔			
I feel very confident in using the application.			✔		
Before I could get started with this application, I had to learn a lot of things.		✔			

Section B

Q1: How useful is this application to you in searching the details of a vocabulary? Please circle on the corresponding rate.

Not Helpful at All 1	**Not Helpful 2**	**Neutral 3**	**Helpful 4**	**Very Helpful 5**

Q2: Which feature in this application you prefer the most?

__user able to know a lot of information regarding the word____

Q3: Which feature in this application you dislike the most?

__too much information has included will make it looks messy at the same time__

Q4: What is your final comment or inquiry for this application?

__need to have more information for the translated languages_____

Participant No: 2	**Participant Name: Ong Jun Jie**

Section A

Table 7.13 User Satisfaction Survey

Statement	**Strongly Disagree 1**	**Disagree 2**	**Neutral 3**	**Agree 4**	**Strongly Agree 5**
I think that I would like to use this application for checking the details of a vocabulary.				✔	
I found that this is an unnecessarily complex application.		✔			
I think the application is easy for me to use.				✔	
To use this application, I feel I will need the assistance of a technical individual.	✔				
I think that this application is simple to navigate without a lot of backtracking or data re-entry.					✔
I think there are too much inconsistencies in this application.	✔				
I believe that most people will learn to use this application without consuming too much time.				✔	

I found the application very awkward to use.	✔				
I feel very confident in using the application.			✔		
Before I could get started with this application, I had to learn a lot of things.		✔			

Section B

Q1: How useful is this application to you in searching the details of a vocabulary? Please circle on the corresponding rate.

Not Helpful at All 1	**Not Helpful 2**	**Neutral 3**	**Helpful 4**	**Very Helpful 5**

Q2: Which feature in this application you prefer the most?

___daily new vocabulary_____

Q3: Which feature in this application you dislike the most?

___some terminology cannot be understood_____

Q4: What is your final comment or inquiry for this application?

___the application should have a vocal pronunciation function _____

Participant No: 3	**Participant Name: Christopher Ho**

Section A

Table 7.13 User Satisfaction Survey

Statement	**Strongly Disagree 1**	**Disagree 2**	**Neutral 3**	**Agree 4**	**Strongly Agree 5**
I think that I would like to use this application for checking the details of a vocabulary.				✔	
I found that this is an unnecessarily complex application.		✔			
I think the application is easy for me to use.				✔	
To use this application, I feel I will need the assistance of a technical individual.			✔		
I think that this application is simple to navigate without a lot of backtracking or data re-entry.					✔
I think there are too much inconsistencies in this application.	✔				
I believe that most people will learn to use this application without consuming too much time.				✔	

I found the application very awkward to use.		✔			
I feel very confident in using the application.				✔	
Before I could get started with this application, I had to learn a lot of things.	✔				

Section B

Q1: How useful is this application to you in searching the details of a vocabulary? Please circle on the corresponding rate.

Not Helpful at All 1	**Not Helpful 2**	**Neutral 3**	**Helpful 4**	**Very Helpful 5**

Q2: Which feature in this application you prefer the most?

__highly customizable in phrasebook_____

Q3: Which feature in this application you dislike the most?

__font size is too small_____

Q4: What is your final comment or inquiry for this application?

____-_____

Participant No: 4	**Participant Name: Lee Car Men**

Section A

Table 7.13 User Satisfaction Survey

Statement	**Strongly Disagree 1**	**Disagree 2**	**Neutral 3**	**Agree 4**	**Strongly Agree 5**
I think that I would like to use this application for checking the details of a vocabulary.				✔	
I found that this is an unnecessarily complex application.	✔				
I think the application is easy for me to use.					✔
To use this application, I feel I will need the assistance of a technical individual.		✔			
I think that this application is simple to navigate without a lot of backtracking or data re-entry.					✔
I think there are too much inconsistencies in this application.	✔				
I believe that most people will learn to use this application without consuming too much time.			✔		

I found the application very awkward to use.			✔		
I feel very confident in using the application.			✔		
Before I could get started with this application, I had to learn a lot of things.	✔				

Section B

Q1: How useful is this application to you in searching the details of a vocabulary? Please circle on the corresponding rate.

Not Helpful at All 1	**Not Helpful 2**	**Neutral 3**	**Helpful 4**	**Very Helpful 5**

Q2: Which feature in this application you prefer the most?

__simple and nice UI_____

Q3: Which feature in this application you dislike the most?

__maybe too much information has included_____

Q4: What is your final comment or inquiry for this application?

__include Tamil as it is one of the major languages in Malaysia, should have notification to remind user for reading daily new vocabulary, need to contain a history list to check back what are the previous searches_____

Participant No: 5	**Participant Name: Ti Hong Liang**

Section A

Table 7.13 User Satisfaction Survey

Statement	**Strongly Disagree 1**	**Disagree 2**	**Neutral 3**	**Agree 4**	**Strongly Agree 5**
I think that I would like to use this application for checking the details of a vocabulary.				✔	
I found that this is an unnecessarily complex application.		✔			
I think the application is easy for me to use.				✔	
To use this application, I feel I will need the assistance of a technical individual.		✔			
I think that this application is simple to navigate without a lot of backtracking or data re-entry.				✔	
I think there are too much inconsistencies in this application.		✔			
I believe that most people will learn to use this application without consuming too much time.				✔	

I found the application very awkward to use.		✔			
I feel very confident in using the application.				✔	
Before I could get started with this application, I had to learn a lot of things.		✔			

Section B

Q1: How useful is this application to you in searching the details of a vocabulary? Please circle on the corresponding rate.

Not Helpful at All 1	**Not Helpful 2**	**Neutral 3**	**Helpful 4**	**Very Helpful 5**

Q2: Which feature in this application you prefer the most?

_informative detail_____

Q3: Which feature in this application you dislike the most?

_font size seems a little too small _____

Q4: What is your final comment or inquiry for this application?

_maybe should have some sort of backup for the favourite list_____

APPENDIX D: User Acceptance Testing Template

User Acceptance Testing Template

Participant Number	
Participant Name	
Date of Testing	

Test Case ID	UAT-01	**Test Module**	View daily new vocabulary
Start Time		**End Time**	

Test Description	**Status**	**Comments**
Able to check in the application successfully		
Able to allocate the daily new vocabulary section		

Test Case ID	UAT-02	**Test Module**	Search vocabulary
Start Time		**End Time**	

Test Description	**Status**	**Comments**
Able to navigate from home page to search page		
Able to search multiple words and view the displayed information		
Able to lookup the details from two different dictionaries		

Test Case ID	UAT-03	**Test Module**	Manage phrasebook/ favourite list
Start Time		**End Time**	

Test Description	Status	Comments
Able to save the word and its' details to the phrasebook		
Able to navigate from search page to phrasebook page		
Able to view the word and its' details that has just saved to the favourite list		
Able to edit the content of the word details according to personal preference		
Able to delete the saved vocabulary from the favourite list		
Able to add a self-defined word and its' details to the phrasebook		

Test Case ID	UAT-04	**Test Module**	Manage note
Start Time		**End Time**	

Test Description	Status	Comments
Able to add the "example" of word details in phrasebook according to his/her own idea		
Able to amend the "example" of word		

details in phrasebook based on his/her own idea		
Able to delete or clear the “example” of word details in phrasebook		
Able to navigate from phrasebook page to home page		

APPENDIX E: Results from User Acceptance Testing

Participant Number	1
Participant Name	Gan Jes See
Date of Testing	19/3/2021

Test Case ID	UAT-01	**Test Module**	View daily new vocabulary
Start Time	11:30:00	**End Time**	11:30:10

Test Description	**Status**	**Comments**
Able to check in the application successfully	Pass	-
Able to allocate the daily new vocabulary section	Pass	-

Test Case ID	UAT-02	**Test Module**	Search vocabulary
Start Time	11:30:30	**End Time**	11:31:42

Test Description	**Status**	**Comments**
Able to navigate from home page to search page	Pass	-
Able to search multiple words and view the displayed information	Pass	-
Able to lookup the details from two different dictionaries	Pass	-

Test Case ID	UAT-03	**Test Module**	Manage phrasebook/ favourite list
Start Time	11:35:22	**End Time**	11:44:54

Test Description	**Status**	**Comments**

Able to save the word and its' details to the phrasebook	Pass	-
Able to navigate from search page to phrasebook page	Pass	-
Able to view the word and its' details that has just saved to the favourite list	Pass	-
Able to edit the content of the word details according to personal preference	Pass	-
Able to delete the saved vocabulary from the favourite list	Pass	-
Able to add a self-defined word and its' details to the phrasebook	Pass	-

Test Case ID	UAT-04	**Test Module**	Manage note
Start Time	11:48:07	**End Time**	11:51:33
Test Description		**Status**	**Comments**
Able to add the "example" of word details in phrasebook according to his/her own idea		Pass	-
Able to amend the "example" of word details in phrasebook		Pass	-

based on his/her own idea		
Able to delete or clear the “example” of word details in phrasebook	Pass	-
Able to navigate from phrasebook page to home page	Pass	-

Participant Number	2
Participant Name	Lean Chee Hong
Date of Testing	19/3/2021

Test Case ID	UAT-01	**Test Module**	View daily new vocabulary
Start Time	12:23:17	**End Time**	12:23:30

Test Description	**Status**	**Comments**
Able to check in the application successfully	Pass	-
Able to allocate the daily new vocabulary section	Pass	-

Test Case ID	UAT-02	**Test Module**	Search vocabulary
Start Time	12:25:26	**End Time**	12:26:09

Test Description	**Status**	**Comments**
Able to navigate from home page to search page	Pass	-
Able to search multiple words and view the displayed information	Pass	-
Able to lookup the details from two different dictionaries	Pass	-

Test Case ID	UAT-03	**Test Module**	Manage phrasebook/ favourite list
Start Time	12:34:46	**End Time**	12:50:55

Test Description	**Status**	**Comments**

Able to save the word and its' details to the phrasebook	Pass	-
Able to navigate from search page to phrasebook page	Pass	-
Able to view the word and its' details that has just saved to the favourite list	Pass	-
Able to edit the content of the word details according to personal preference	Pass	-
Able to delete the saved vocabulary from the favourite list	Pass	-
Able to add a self-defined word and its' details to the phrasebook	Pass	-

Test Case ID	UAT-04	**Test Module**	Manage note
Start Time	12:52:19	**End Time**	12:54:38
Test Description		**Status**	**Comments**
Able to add the "example" of word details in phrasebook according to his/her own idea		Pass	-
Able to amend the "example" of word details in phrasebook		Pass	-

based on his/her own idea		
Able to delete or clear the “example” of word details in phrasebook	Pass	-
Able to navigate from phrasebook page to home page	Pass	-

Participant Number	3
Participant Name	Ng Kin Han
Date of Testing	19/3/2021

Test Case ID	UAT-01	**Test Module**	View daily new vocabulary
Start Time	13:30:00	**End Time**	13:30:15

Test Description	**Status**	**Comments**
Able to check in the application successfully	Pass	-
Able to allocate the daily new vocabulary section	Pass	-

Test Case ID	UAT-02	**Test Module**	Search vocabulary
Start Time	13:30:50	**End Time**	13:33:47

Test Description	**Status**	**Comments**
Able to navigate from home page to search page	Pass	-
Able to search multiple words and view the displayed information	Pass	-
Able to lookup the details from two different dictionaries	Pass	-

Test Case ID	UAT-03	**Test Module**	Manage phrasebook/ favourite list
Start Time	13:38:44	**End Time**	13:47:03

Test Description	**Status**	**Comments**

Able to save the word and its' details to the phrasebook	Pass	-
Able to navigate from search page to phrasebook page	Pass	-
Able to view the word and its' details that has just saved to the favourite list	Pass	-
Able to edit the content of the word details according to personal preference	Pass	-
Able to delete the saved vocabulary from the favourite list	Pass	-
Able to add a self-defined word and its' details to the phrasebook	Pass	-

Test Case ID	UAT-04	**Test Module**	Manage note
Start Time	13:51:13	**End Time**	13:55:39
Test Description		**Status**	**Comments**
Able to add the "example" of word details in phrasebook according to his/her own idea		Pass	-
Able to amend the "example" of word details in phrasebook		Pass	-

based on his/her own idea		
Able to delete or clear the “example” of word details in phrasebook	Pass	-
Able to navigate from phrasebook page to home page	Pass	-

Participant Number	4
Participant Name	Yap Gay Chin
Date of Testing	19/3/2021

Test Case ID	UAT-01	**Test Module**	View daily new vocabulary
Start Time	15:00:15	**End Time**	15:00:50

Test Description	**Status**	**Comments**
Able to check in the application successfully	Pass	-
Able to allocate the daily new vocabulary section	Pass	-

Test Case ID	UAT-02	**Test Module**	Search vocabulary
Start Time	15:05:08	**End Time**	15:05:59

Test Description	**Status**	**Comments**
Able to navigate from home page to search page	Pass	-
Able to search multiple words and view the displayed information	Pass	-
Able to lookup the details from two different dictionaries	Pass	-

Test Case ID	UAT-03	**Test Module**	Manage phrasebook/ favourite list
Start Time	15:17:31	**End Time**	15:29:28

Test Description	**Status**	**Comments**

Able to save the word and its' details to the phrasebook	Pass	-
Able to navigate from search page to phrasebook page	Pass	-
Able to view the word and its' details that has just saved to the favourite list	Pass	-
Able to edit the content of the word details according to personal preference	Pass	-
Able to delete the saved vocabulary from the favourite list	Pass	-
Able to add a self-defined word and its' details to the phrasebook	Pass	-

Test Case ID	UAT-04	**Test Module**	Manage note
Start Time	15:31:06	**End Time**	15:33:02
Test Description		**Status**	**Comments**
Able to add the "example" of word details in phrasebook according to his/her own idea		Pass	-
Able to amend the "example" of word details in phrasebook		Pass	-

based on his/her own idea		
Able to delete or clear the “example” of word details in phrasebook	Pass	-
Able to navigate from phrasebook page to home page	Pass	-

Participant Number	5
Participant Name	Ng Pei Eer
Date of Testing	19/3/2021

Test Case ID	UAT-01	**Test Module**	View daily new vocabulary
Start Time	16:04:15	**End Time**	16:04:23

Test Description	**Status**	**Comments**
Able to check in the application successfully	Pass	-
Able to allocate the daily new vocabulary section	Pass	-

Test Case ID	UAT-02	**Test Module**	Search vocabulary
Start Time	16:06:47	**End Time**	16:09:00

Test Description	**Status**	**Comments**
Able to navigate from home page to search page	Pass	-
Able to search multiple words and view the displayed information	Pass	-
Able to lookup the details from two different dictionaries	Pass	-

Test Case ID	UAT-03	**Test Module**	Manage phrasebook/ favourite list
Start Time	16:12:21	**End Time**	16:30:27

Test Description	**Status**	**Comments**

Able to save the word and its' details to the phrasebook	Pass	-
Able to navigate from search page to phrasebook page	Pass	-
Able to view the word and its' details that has just saved to the favourite list	Pass	-
Able to edit the content of the word details according to personal preference	Pass	-
Able to delete the saved vocabulary from the favourite list	Pass	-
Able to add a self-defined word and its' details to the phrasebook	Pass	-

Test Case ID	UAT-04	**Test Module**	Manage note
Start Time	16:33:54	**End Time**	16:35:36
Test Description		**Status**	**Comments**
Able to add the "example" of word details in phrasebook according to his/her own idea		Pass	-
Able to amend the "example" of word details in phrasebook		Pass	-

based on his/her own idea		
Able to delete or clear the “example” of word details in phrasebook	Pass	-
Able to navigate from phrasebook page to home page	Pass	-