UTM
UNIVERSITI TEKNOLOGI MALAYSIA

SECJ2203: Software Engineering

System Documentation (SD)

Improvising Current DBP Dictionary

Version 2.0

24 JUNE 2022

School of Computing, Faculty of Engineering

Prepared by: Group 2

No.	Name	Matrix No.
1	AHMAD NAZRAN BIN YUSRI	A20EC0179
2	ANATASYA HUMAIRA	A20EC0261
3	MOHAND ALAA ABOUZEID MOHAMED	A20EC4042
4	NG YEN THONG	A20EC0107

Revision Page

a. Overview

The software design in this report is the first version of the Intelligent English Malay Dictionary. This report consists of the introduction, system-specific requirement, system architectural design, detailed description of components, data design, user interface design, requirement matrix, test cases, and test approach analysis.

b. Target Audience

This document is intended for the team members, developers, designers, testers and system administrators.

c. Project Team Members

Member Name	Role	Task	Status
Ahmad Nazran bin Yusri	Leader	● Purpose ● Activity Diagram ● Communication Interface ● Design Constraints ● Use case diagram ● UC001 ● UC002 ● UC003 ● P003 Subsystem ● Data Description ● Class Diagram ● Package Diagram	Completed
Mohand Alaa Abouzeid Mohamed	Software Developer	● Scope ● Hardware interface ● Domain model diagram ● Software system attributes ● UC004 ● UC005 ● UC006 ● P002 Subsystem ● Data Dictionary ● Class Diagram ● Package Diagram	Completed

Anatasya Humaira		● Use case diagram ● Definitions ● Reference ● User Interface ● UC007 ● UC008 ● UC009 ● UC010 ● UC011 ● P001 Subsystem	Not completed
Ng Yen Thong	Vice Leader	● Project Overview ● Software Interface ● Performance and other requirements ● Use case diagram ● UC012 ● UC013 ● UC014 ● P004 Subsystem ● Architecture style and rationale ● Class Diagram ● Package Diagram	Completed

d. Version Control History

Version	Primary Author(s)	Description of Version	Date Completed
1.0	Ng Yen Thong	Consists of Introduction and Specific Requirements.	10/6/2022
2.0	Mohand Alaa Abouzeid Mohamed	Consists of System Architectural Design, Detailed Description of Components and Data Design	24/6/2022

Table of Contents

1. Introduction

1.1 **Purpose**

The purpose of this system documentation is to report the specific requirements including external interface requirements, system features, performance and other requirements, design constraints and software system attributes of this project which is Improvising Current DBP Dictionary. Thus, providing clear details to all parties involved in this project to ensure better understanding. Besides that, this report also logs all the details on how our group conducts this project. The details mentioned are the overview, version control history, list of team members and the task distribution applied. This documentation will also play a crucial role in progressing this project as this documentation will be referred to a lot while completing the upcoming phases of the project. Therefore, this document is intended for the team members, developers, designers, testers and system administrators.

1.2 **Scope**

The software product is Intelligent English-Malay dictionary. This system is the new version of the Dewan Bahasa dan Pustaka dictionary. The system will be used by a variety of users, starting from normal users to professional users. Artificial Intelligence is the technology that the system is built on to introduce a satisfying software experience. The new system is developed to provide many useful features. starting with providing basic translation features to advanced translation features. The newly developed translation algorithm reduces the level of intricacy of the specific dictionary-driven rules. Also, it reduces the number of confusing Malay terms. The new website interface was built using modern UI/UX principles to produce a responsive, attractive, and convincing website. In addition, the new mobile application provides a handy and fast user experience. Furthermore, additional features like searching words by audio and image, sentence grammar API generation, Web browser library, and image-to-text will help a variety of users achieve their goals with the system. The main objective and goal of using this system is that users put less effort into the translation operation, so the translation will be faster and sufficient.

1.3 **Definitions, Acronyms and Abbreviation**

DBP : Dewan Bahasa dan Pustaka, the existing English-Malay dictionary that owned by Dewan Bahasa dan Pustaka

OS : An Operating System (OS) is the program that, after being initially loaded into the computer by a boot program, manages all of the other application programs in a computer.

RAM : a Random-Access Memory temporarily stores (remembers) everything that runs on the PC

LAN : Local Area Network

GUI : Graphical User Interface

ACLs : Access Control LIsts

BIOS : Basic Input/Output System

MT : Translation Machine

1.4 **References**

https://prpm.dbp.gov.my/cari1?keyword=kamus%20online

https://www.oxfordlearnersdictionaries.com/

https://dictionary.cambridge.org/

https://fusionauth.io/learn/expert-advice/authentication/login-authentication-workflows

https://acquire.io/blog/customer-service-workflow-examples/

https://lucid.app/lucidchart/62ccc974-db02-40a3-b326-c8a38c621dcf/edit?invitationId=inv_610eac9b-27d1-41df-9bd5-18a76fd78bda&page=0_0#

1.5 Overview

This System Documentation(SD) is divided into eight sections with some of them having various subsections. The sections of the System Documentation are:

1.0 Introduction: This section consists of five subsections that covers the system purpose, system scope, the definitions of terms, acronyms and abbreviation used in this documentation, as well as the references and document overview. Overall, it explains about

the contents of System Documentation (SD) and the system, its background, its functionalities, the audiences, and the references used to complete this system.

2.0 Specific Requirements: This section has five subsections that includes external interface requirements, system features, performance and other requirements, design constraints, as well as the software system attributes. Generally, in the first subsection it discusses user interfaces, hardware interfaces, software interfaces and communication interfaces of the system. Meanwhile, the second subsection describes the system features using use case description for each function that the system offers. In addition, the remaining subsections briefly explain about the design constraints, performance requirements and software system attributes.

3.0 System Architecture Design: In this section, it describes the architectural design. The system should be divided into subsections of architectural style and rationale, as well as the component model.

4.0 Detailed Description of Components: Here we describe the detailed description of components consisting of a complete package diagram to show the files and subfiles contained within the system. Also, the details then elaborated according to the content of the package diagram where every file has their class diagram, sequence diagram, entity, method, and the algorithm table.

5.0 Data Design: Data design contains data description that contains every entity name and their description. Beside that there is also a data dictionary which lists out the attribute name, type and description for each entity.

6.0 Interface Design: This section explains about the user interface design. In this section, the external, internal, physical, logical, H/W and also S/W interfaces are provided. Every screen for every transaction in the system is attached. The screens show a step by step process when the user starts to surf the website, register, login, main page as well as all the other main functions. Moreover, it also contains the screen for all the actors including which are only accessible for specific actors.

7.0 Requirements Matrix: In this section a cross-reference that traces components and data structures to the requirements in the SD is provided. A tabular format is used to show

which system components are satisfying each of the functional requirements represented as use cases.

8.0 Test Cases: This section consists of a few kinds of tests and each test will specify the test ID and name, additional description, input data, expected output data, actual output data as well as their results.

2. Specific Requirements

2.1 External Interface Requirements

2.1.1 User Interfaces

2.1.2 Hardware Interfaces

The system runs on a website and a mobile application. The website could run on both computers and smartphones. The application runs on the iOS and Android operating systems. The system server is running on an Amazon Web server. In addition system data is stored on the system local server. The server processes all the major functions and returns the output to the user.

2.1.3 Software Interfaces

1. Our system supports windows, macOS, iOS and android. For windows, the least version we can support is 7 or later and macOS is 10.14 or higher. Besides, we can support any version of iOS and android.
2. Visual Studio Code is used to develop our website. It helps to develop and create dynamic and interactive web pages. It is also widely used among web application developers because of its open-source technology. It is very easy and convenient because we can customize the website on our own.
3. We will implement the Amazon Web Server (AWS) because it offers flexibility, cost efficient and secure.
4. Maria DB technology is used in our project as it is an open-source relational database and is most often used in many projects.

No.	Name	Mnemonic	Specification number	Version number	Source
1	Visual Studio Code	VS Code	-	1.55.1	Microsoft
2	Amazon Web Server	AWS	-	-	GitHub
3	Maria DB	Maria DB	-	10.9	GitHub

2.1.4 **Communication Interfaces**

The system shall use HTTPS (Hypertext Transfer Protocol Secure) protocol for communication between the client and server over the internet. For offline mobile application, there's no communication needed as all of the data regarding the dictionary will be stored on end users' devices. For online mobile application, internet connection will be needed.

2.2 **System Features**

The system features include use case diagram (Figure 2.1), activity diagram (Figure 2.2) and domain model diagram (Figure 2.3).

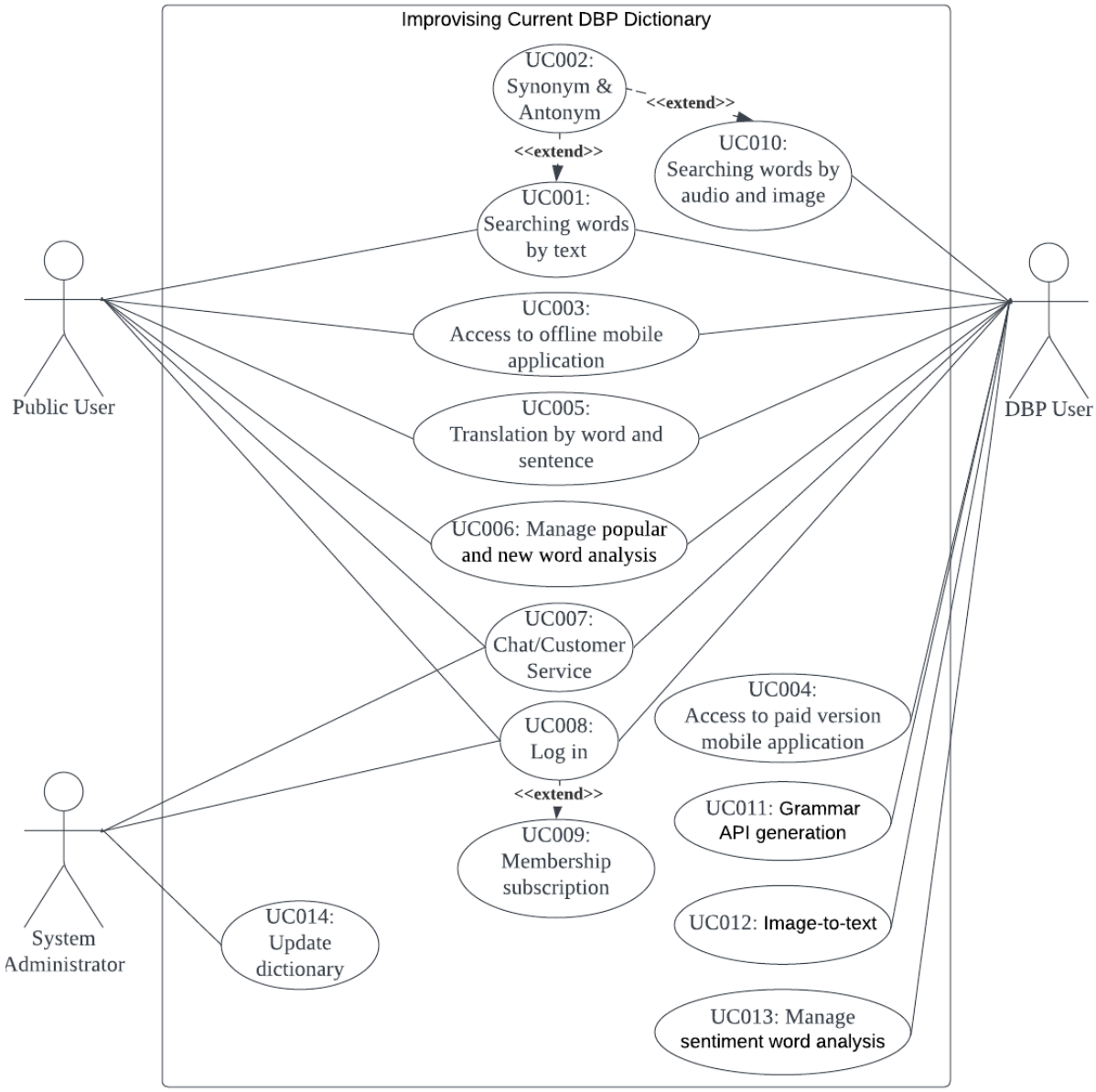

Figure 2.1: Use Case Diagram for Improvising Current DBP Dictionary

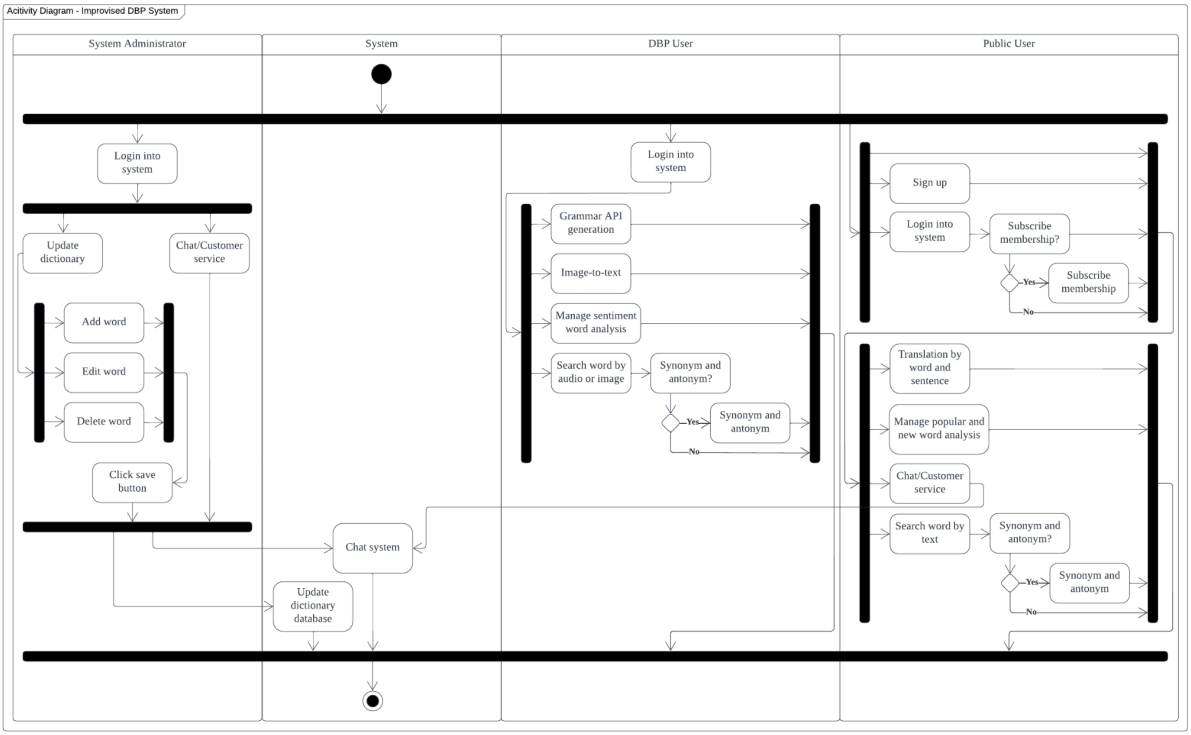

Figure 2.2: Activity Diagram for Improvising Current DBP Dictionary

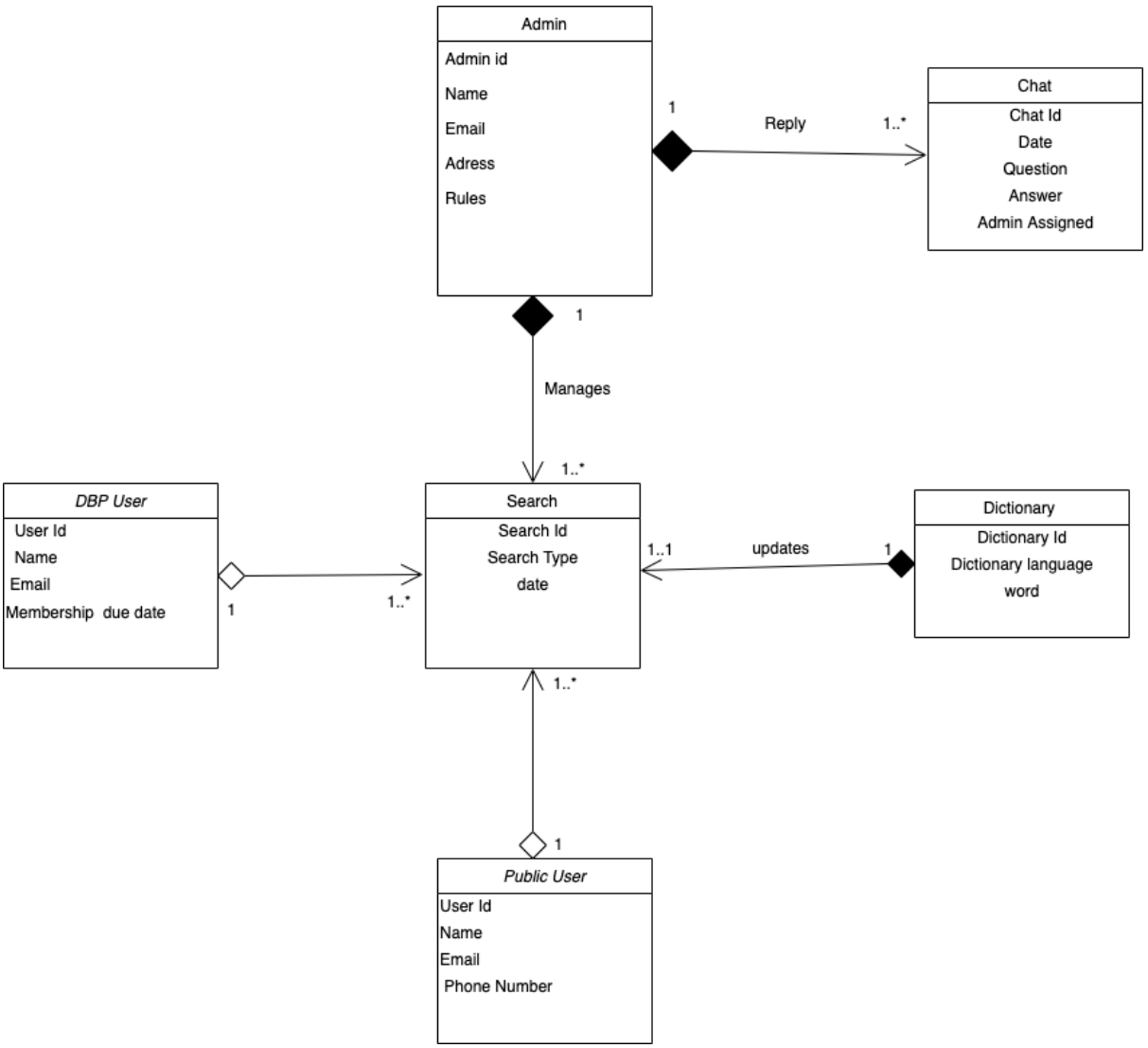

Figure 2.3: Domain Model for Improvising Current DBP Dictionary

2.2.1 **UC001: Use Case <Searching words by text>**

Table 2.1: Use Case Description for Searching words by text

Use case: Searching words by text
ID: UC001
Actors: Public User, DBP User
Description: This use case describes the process of searching definitions of words in the dictionary by text.
Extension points: <Synonym and Antonym>
Flow of events: 1. User searches for a word in the search bar using text. 2. The system will fetch the definition of word from the database. (EF1) 3. The system will display the definition of the corresponding text entered. 4. If the user chooses synonyms and antonyms options. 4.1. <Synonym and Antonym> 5. Use case ends.
Postconditions: The system successfully displays the definition of the text.
Exception flow: **1. No definition about the text.** 1.1. The system will display an error message. 1.2. Use case aborted

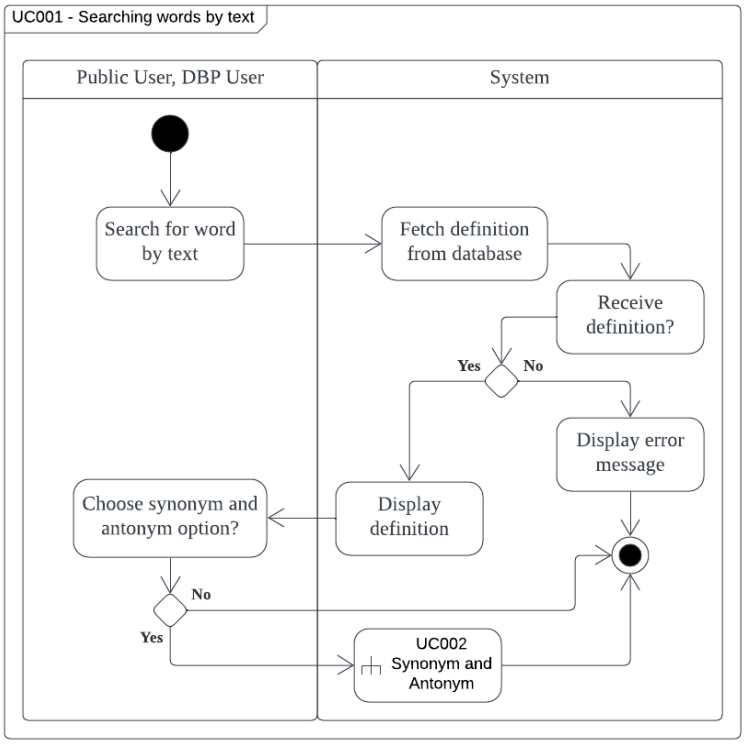

Figure 2.4: Activity Diagram for Searching words by text

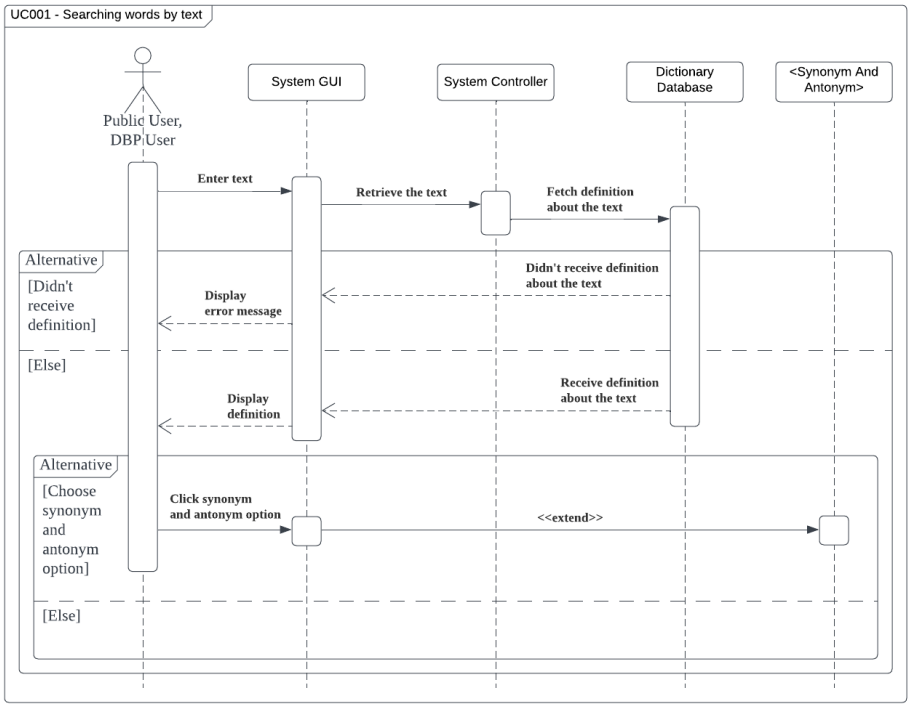

Figure 2.5: Sequence Diagram for Searching words by text

2.2.2 UC002: Use Case <Synonym and Antonym>

Table 2.2: Use Case Description for Synonym and Antonym

Use case: Synonym and Antonym
ID: UC002
Actors: Public User, DBP User
Description: This use case describes the process of getting the synonym and antonym of a word.
Precondition: User must insert a word by text, image or audio.
Flow of events: 1. The system will redirect users to a new page. 2. The system fetches the synonyms and antonyms for the corresponding word from the database. (EF1) 3. The system will display the synonyms and antonyms of the corresponding word entered. 4. Use case ends.
Postconditions: The system successfully displays the synonym and antonym of the text.
Exception flow: **1. No synonym and antonym about the word.** 1.1. The system will display an error message. 1.2. Use case aborted.

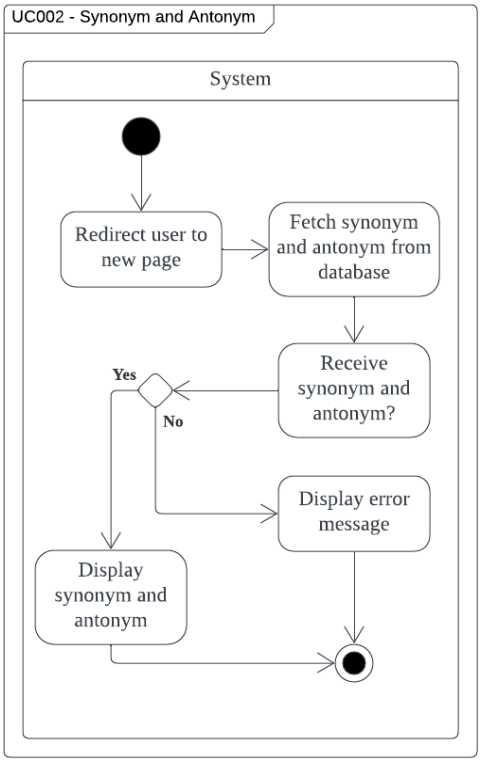

Figure 2.6: Activity Diagram for Synonym and Antonym

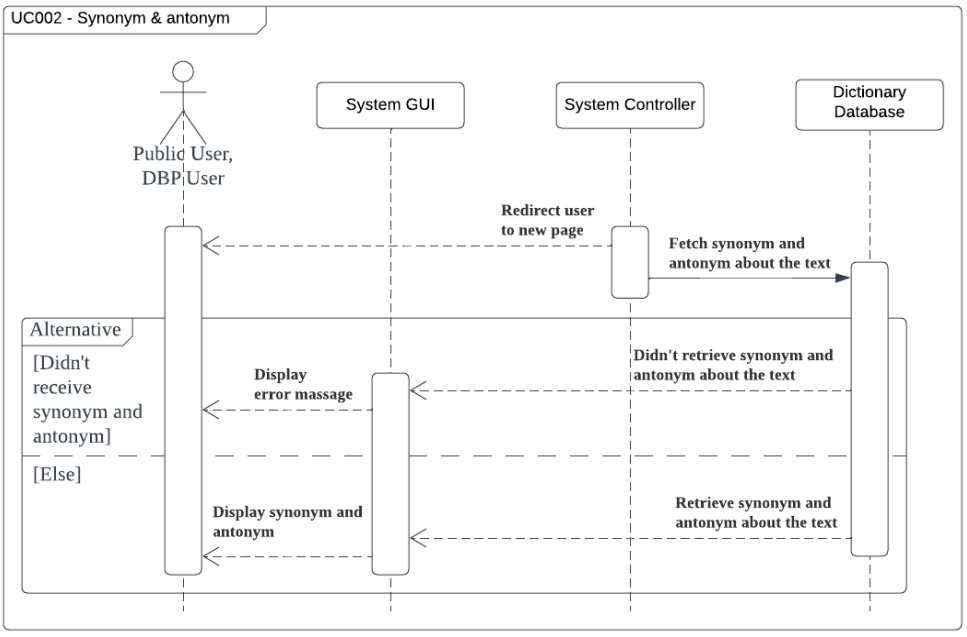

Figure 2.7: Sequence Diagram for Synonym and Antonym

2.2.3 **UC003: Use Case <Access to offline mobile application>**

Table 2.3: Use Case Description for Access to offline mobile application

Use case: Access to offline mobile application
ID: UC003
Actors: Public User
Description: This use case describes the process of accessing to offline mobile application
Preconditions: User must install the mobile application.
Flow of events: 1. Public user have access to 1.1. Searching for words by text. (AF1) 1.2. Synonym and antonym. (AF2) 1.3. Translation by word and sentence. (AF3) 2. Use case ends.
Alternative flow: **1. User chose to search for words.** 1.1. System invokes UC001. **2. User chose to find synonyms and antonyms.** 2.1. System invokes UC002. **3. User chose to translate words and sentences.** 3.1. System invokes UC005.

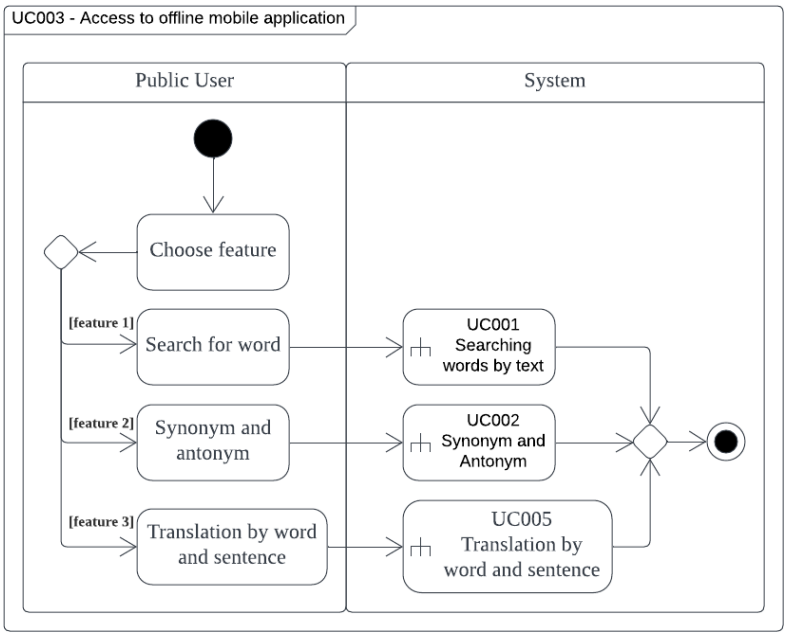

Figure 2.8: Activity Diagram for Access to offline mobile application

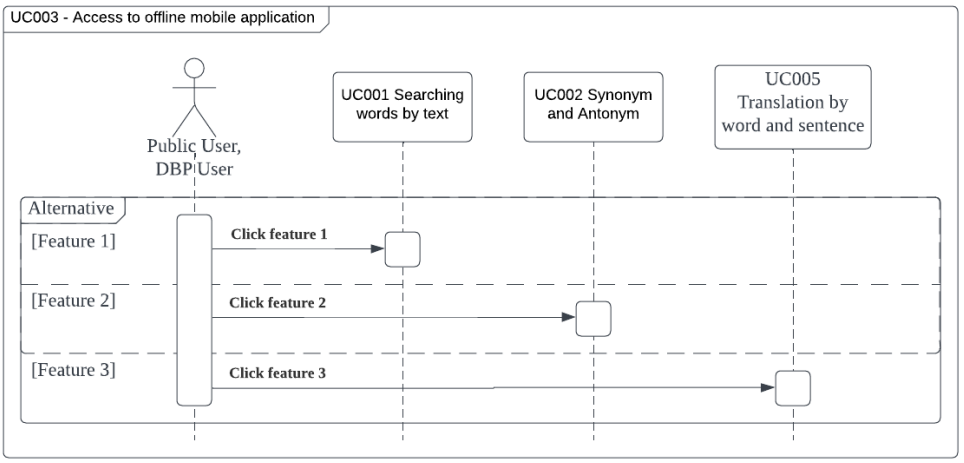

Figure 2.9: Sequence Diagram for Access to offline mobile application

2.2.4 UC004: Use Case <Access to paid mobile application>

Table 2.4: Use Case Description for Access to Paid mobile application

Use case: Access to Paid mobile application
ID: UC004
Actors: DBP User
Description: This use case describes The user will access the application's paid version.
Preconditions: User must install the mobile application.
Flow of events: 1. User enters username and password. 2. The system will check the accuracy of the user data from the database. (EF1) 3. The system will display that the login was successful. 4. If the user has paid membership. 4.1. The system will then open all the paid features in the application. 5.else 5.1. The system will not give permission to access paid features. 5.2 The system will display an option to make paid membership. 5. Use case ends..

Postconditions: The user successfully accesses the membership functions.
Exception flow: 1. Wrong username or password. 1.1. The system will display an error message. 1.2. Use case aborted.

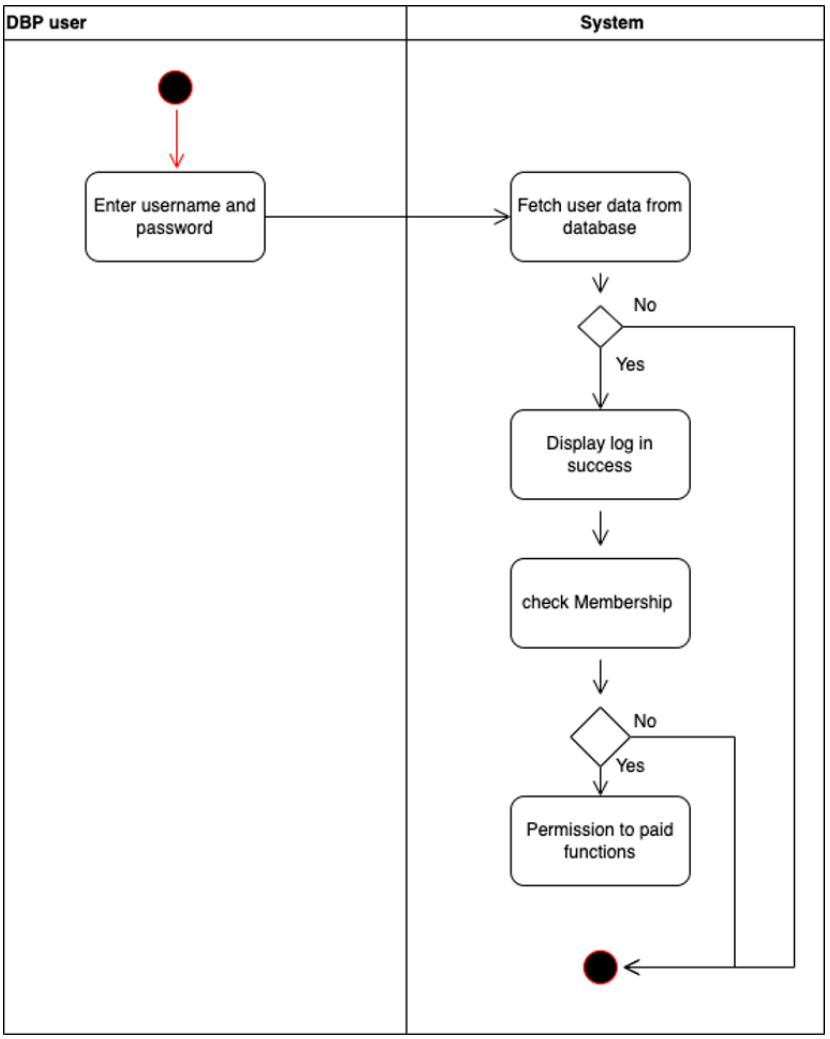

Figure 2.10: Activity Diagram for Access to Paid mobile application

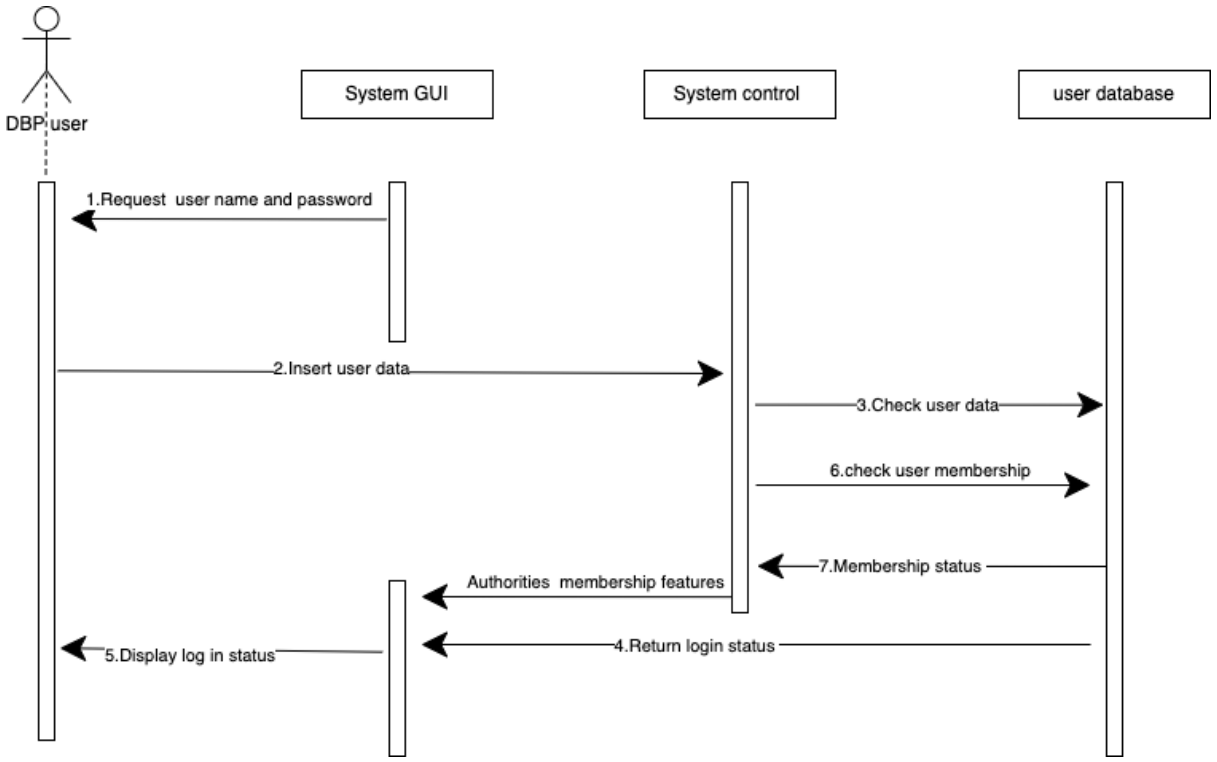

Figure 2.11: Sequence Diagram for Access to Paid Mobile Applications

2.2.5 UC005: Use Case <Translation by word and sentence>

Table 2.5: Use Case Description for Translation by word and sentence

Use case: Translation by word and sentence
ID: UC005
Actors: DBP User, Public user
Description: This use case describes the translation by word and sentence.
Preconditions: User must insert a word or sentence.
Flow of events: 1. The system will redirect users to a new page. 2. The user inserts a word or a sentence . 3. The system will fetch the definition of a word or the sentence from the database. (EF1). 4. The system will display the definition of the corresponding text entered. 5. If the user chooses to search by image options. 5.1< <Image-to-text>> 5.1 the system will fetch the definition of images text or the sentence from the database. (EF1).

6. Use case ends.
Postconditions: The system successfully identifies and displays the definition of the text.
Exception flow: 1. No definition of the text. 1.1. The system will display an error message. 1.2. Use case aborted.

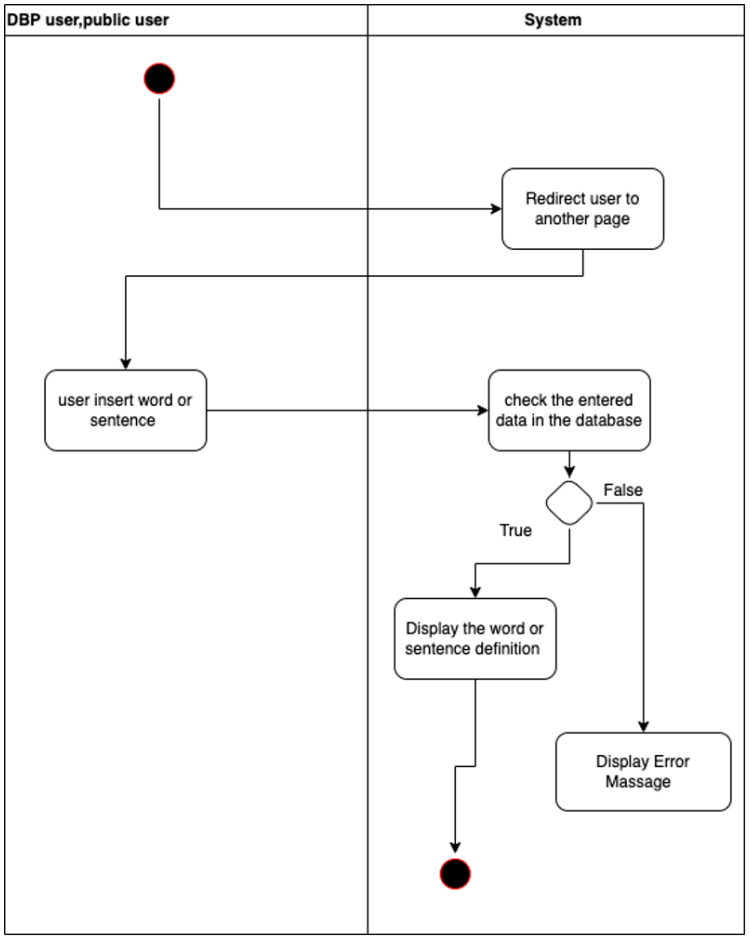

Figure 2.12: Activity Diagram for Translation by Word and Sentence

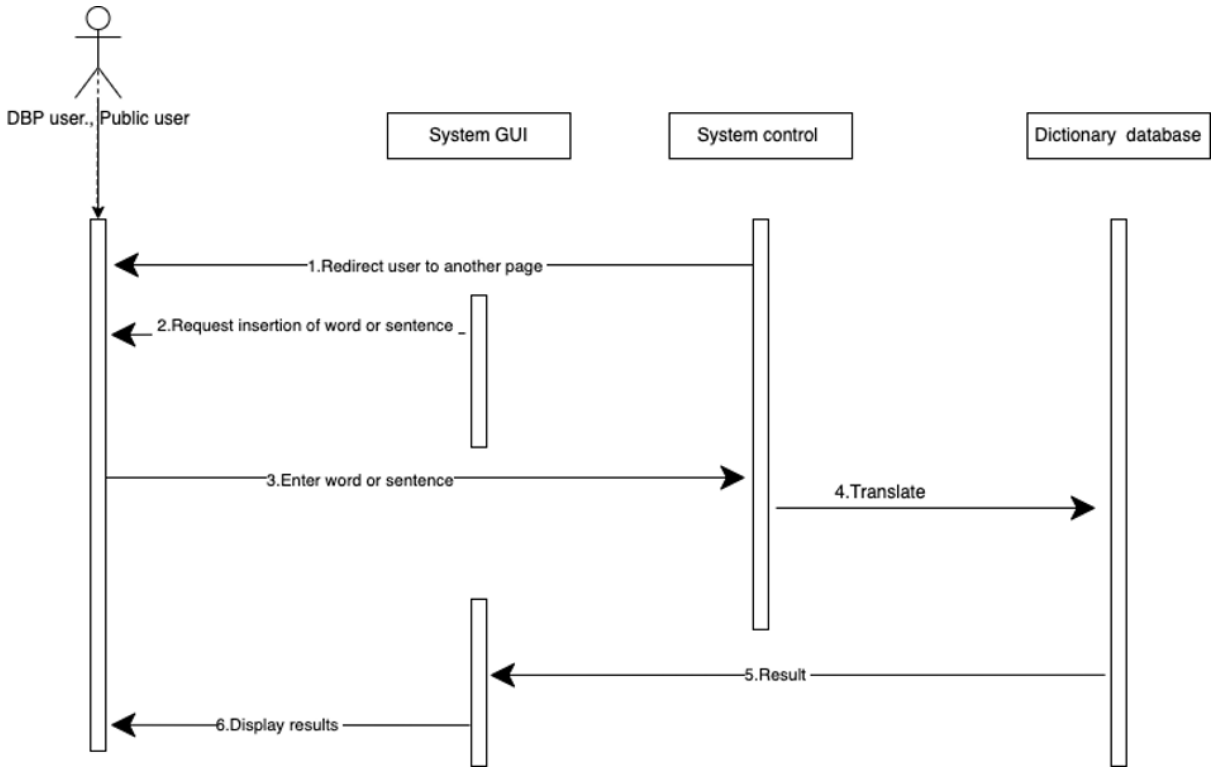

Figure 2.13: Sequence Diagram for Translation by Word and Sentence

2.2.6 UC006: Use Case <Manage popular word analysis>

Table 2.6: Use Case Description for Manage popular word analysis.

Use case: Manage popular word analysis
ID: UC006
Actors: DBP User,Public
Description: This use case describes the management of popular and new words analysis.
Preconditions: User must login in successfully.
Flow of events: 1. The system will redirect users to the "popular words" page. 2. The system will fetch the popular word from the database . (EF1). 3. The system will display all the words with their definitions. 4. If the user chooses new word analysis, 4.1. The system will ask the user to insert the words. 4.2. The system will fetch the definition of the text from the database. (EF1). 4.3.the system will display all the information of the word. 5. Use case ends..

Postconditions: The system successfully displays the definition of the text.
Exception flow: 1. No information was found about the text. 1.1. The system will display an error message. 1.2. Usecase aborted.

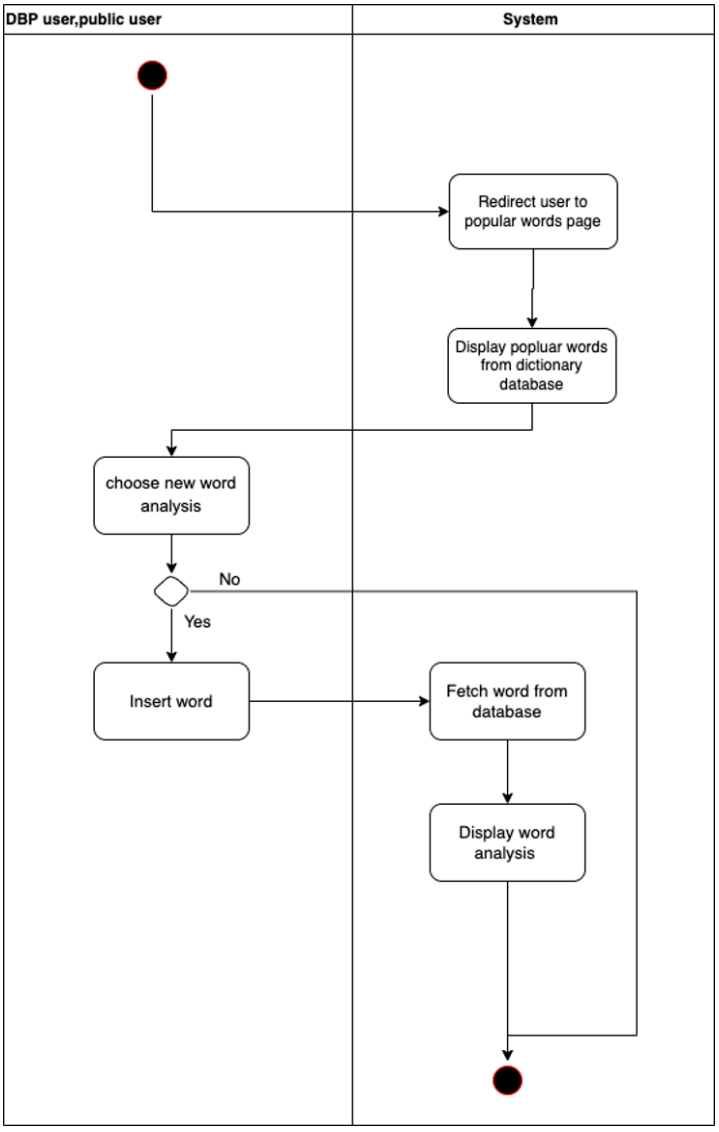

Figure 2.14: Activity Diagram for Managing Popular and New Word Analysis

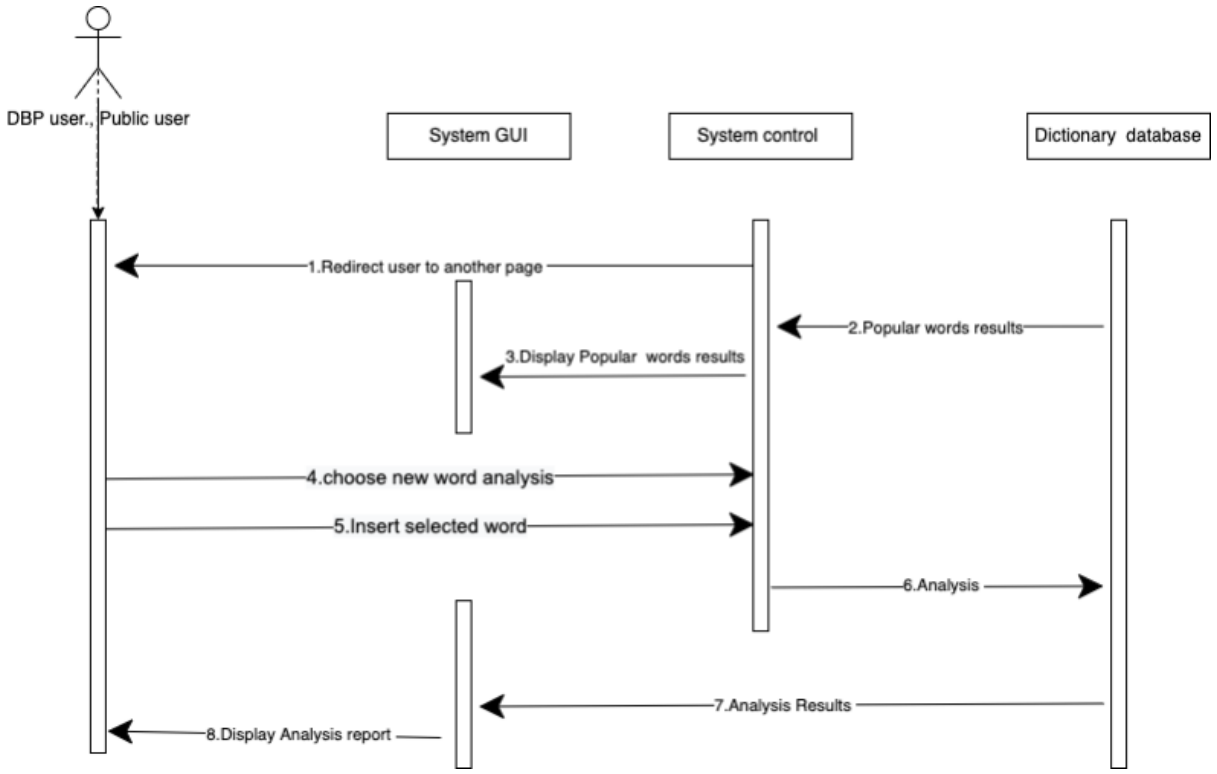

Figure 2.15: Sequence Diagram for Manage popular and new word analysis

2.2.7 **UC007: Use Case <Chat/Customer Service>**

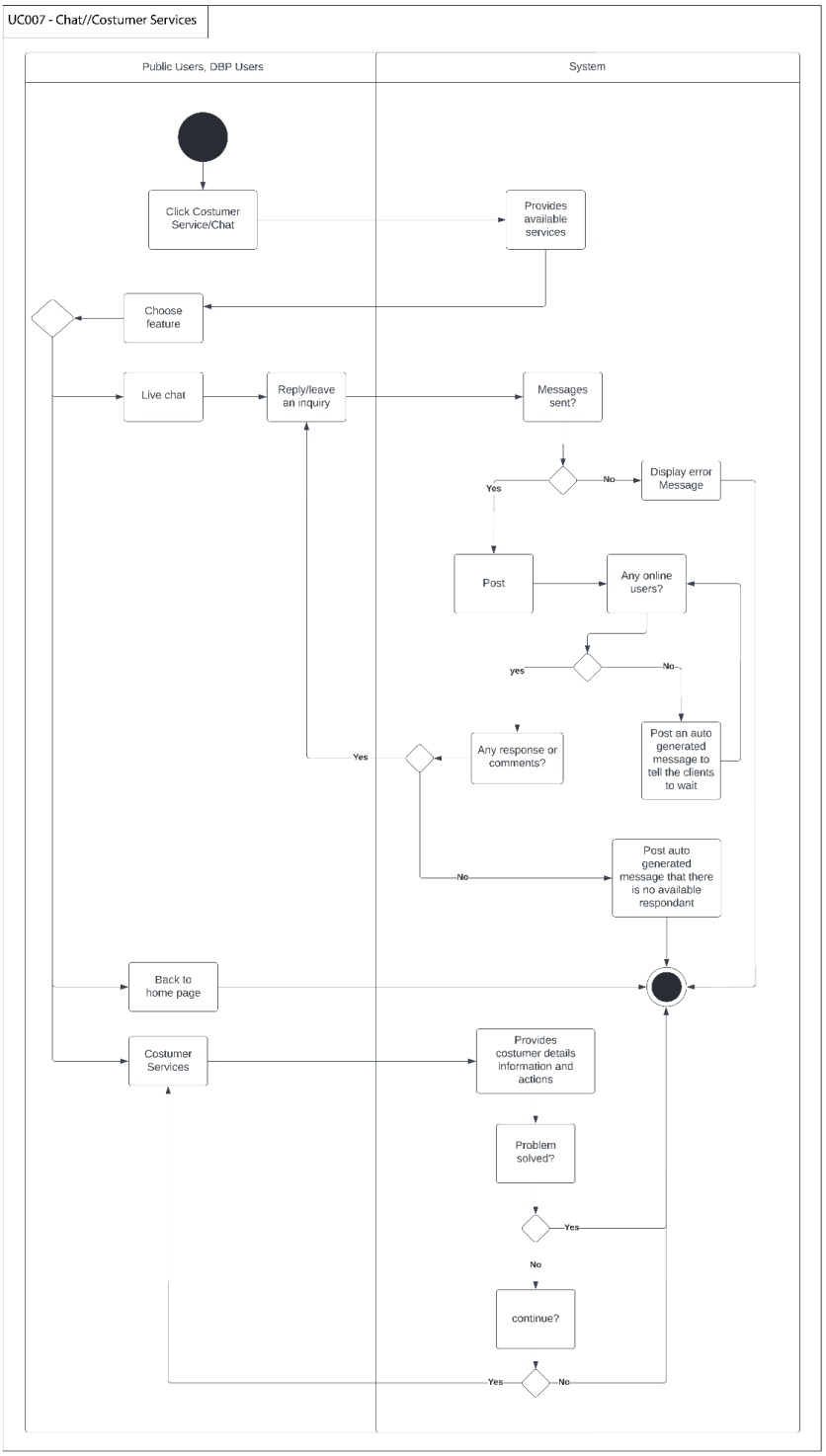

Figure 2.16: Activity Diagram for Chat and Customer Services

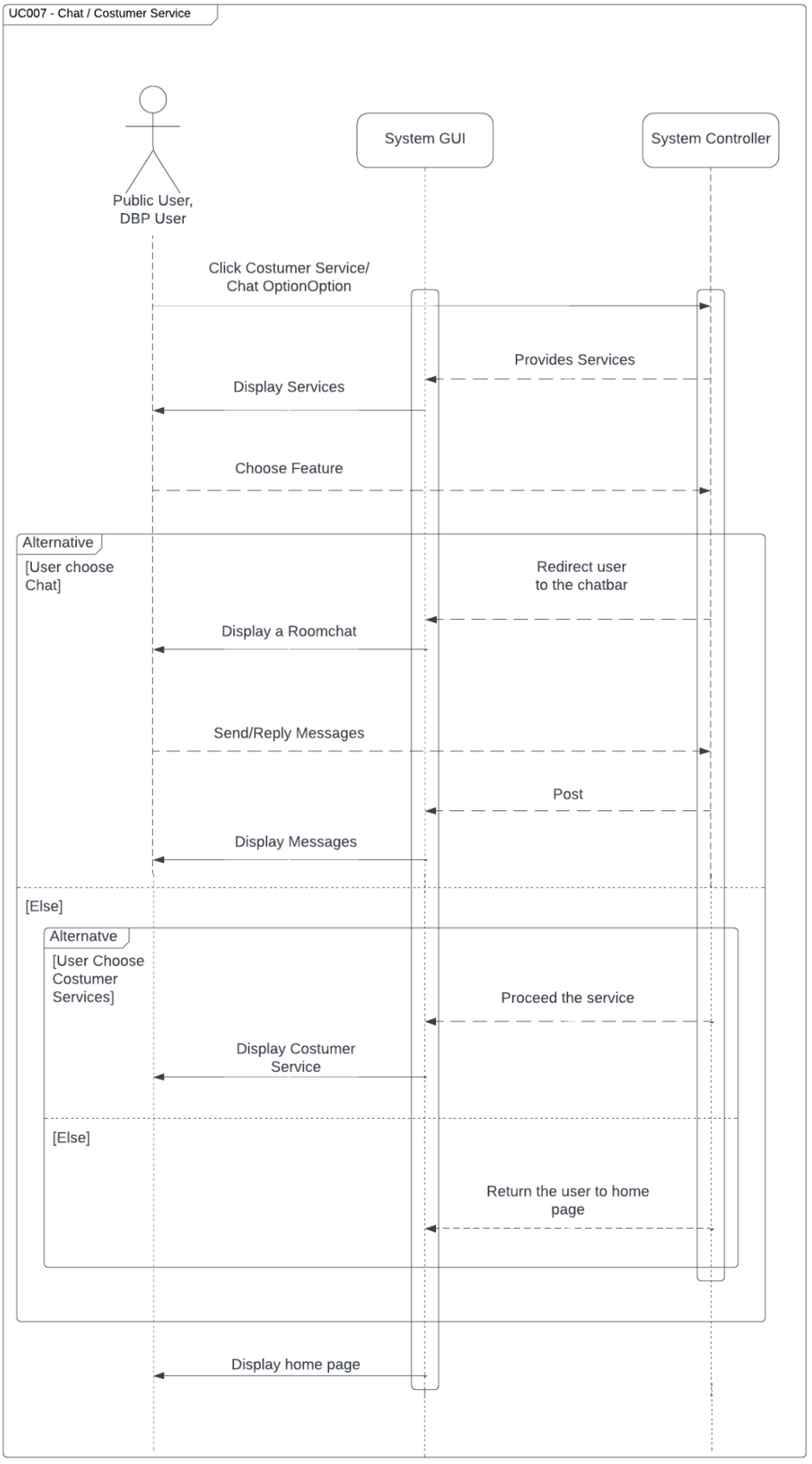

Figure 2.17: Sequence Diagram for Chat and Customer Services

2.2.8 **UC008: Use Case <Log In>**

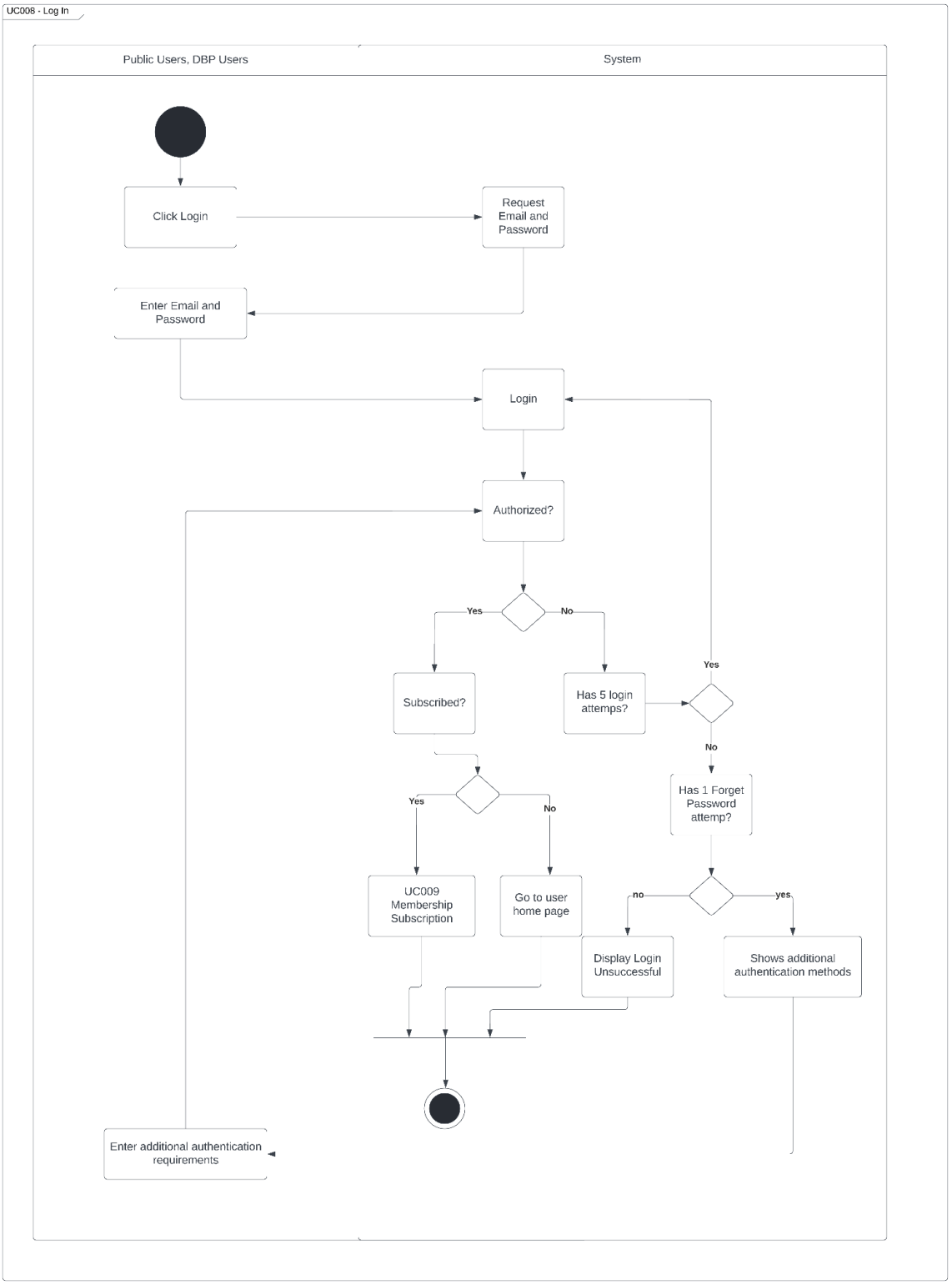

Figure 2.18: Activity Diagram for Logging In

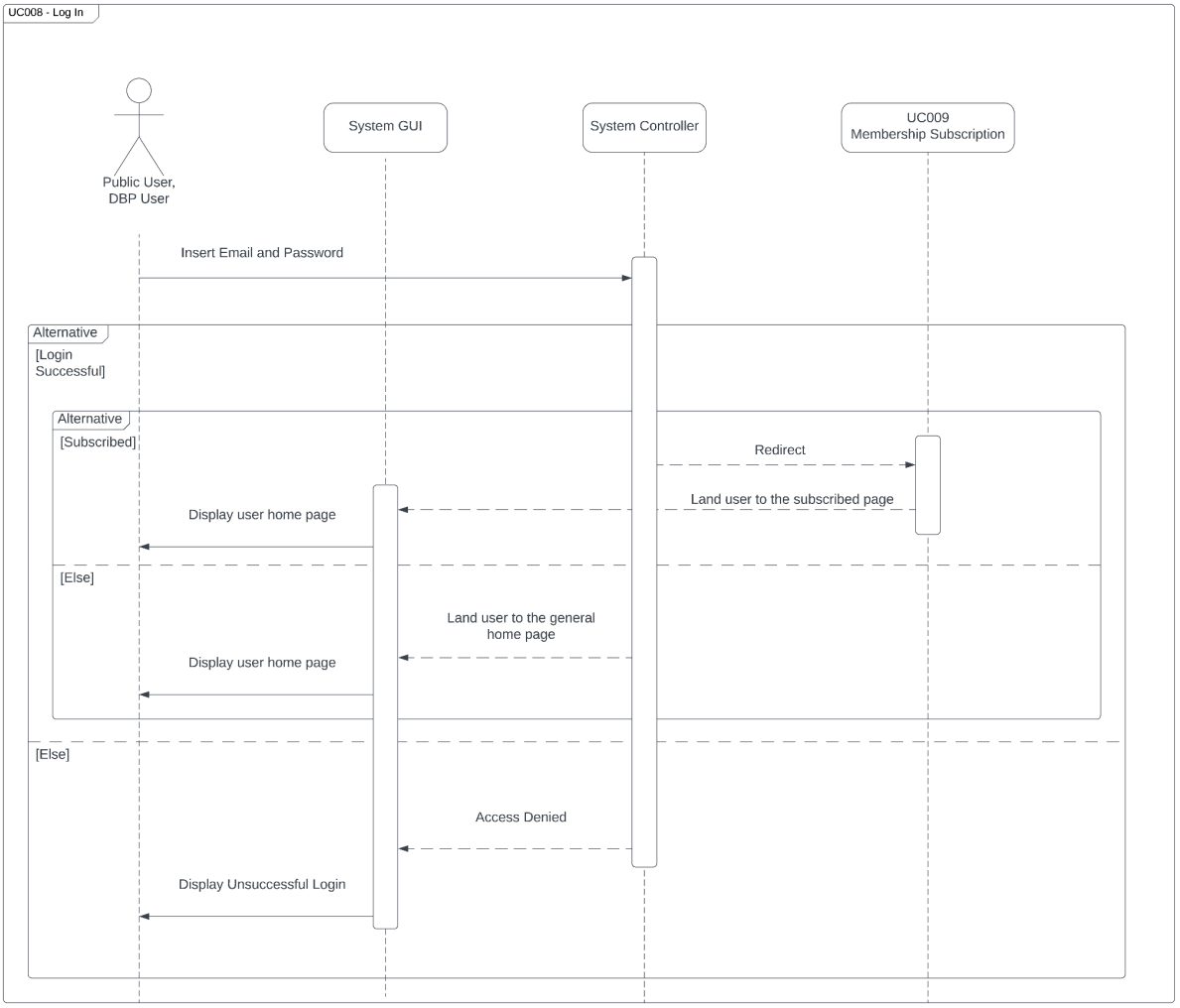

Figure 2.19: Sequence Diagram for Logging In

2.2.9 **UC009: Use Case <Membership Subscription>**

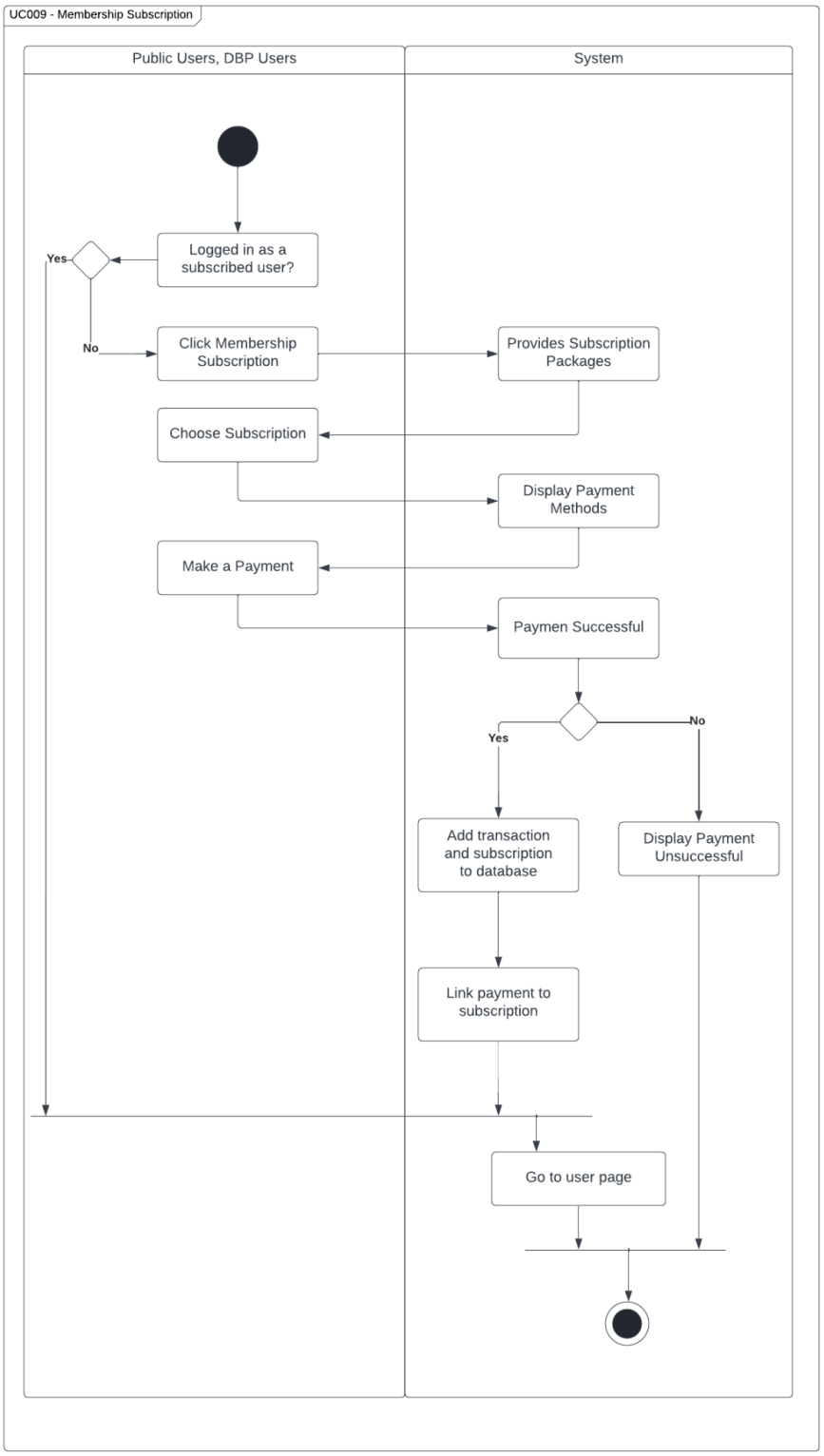

Figure 2.20: Activity Diagram for Membership Subscription

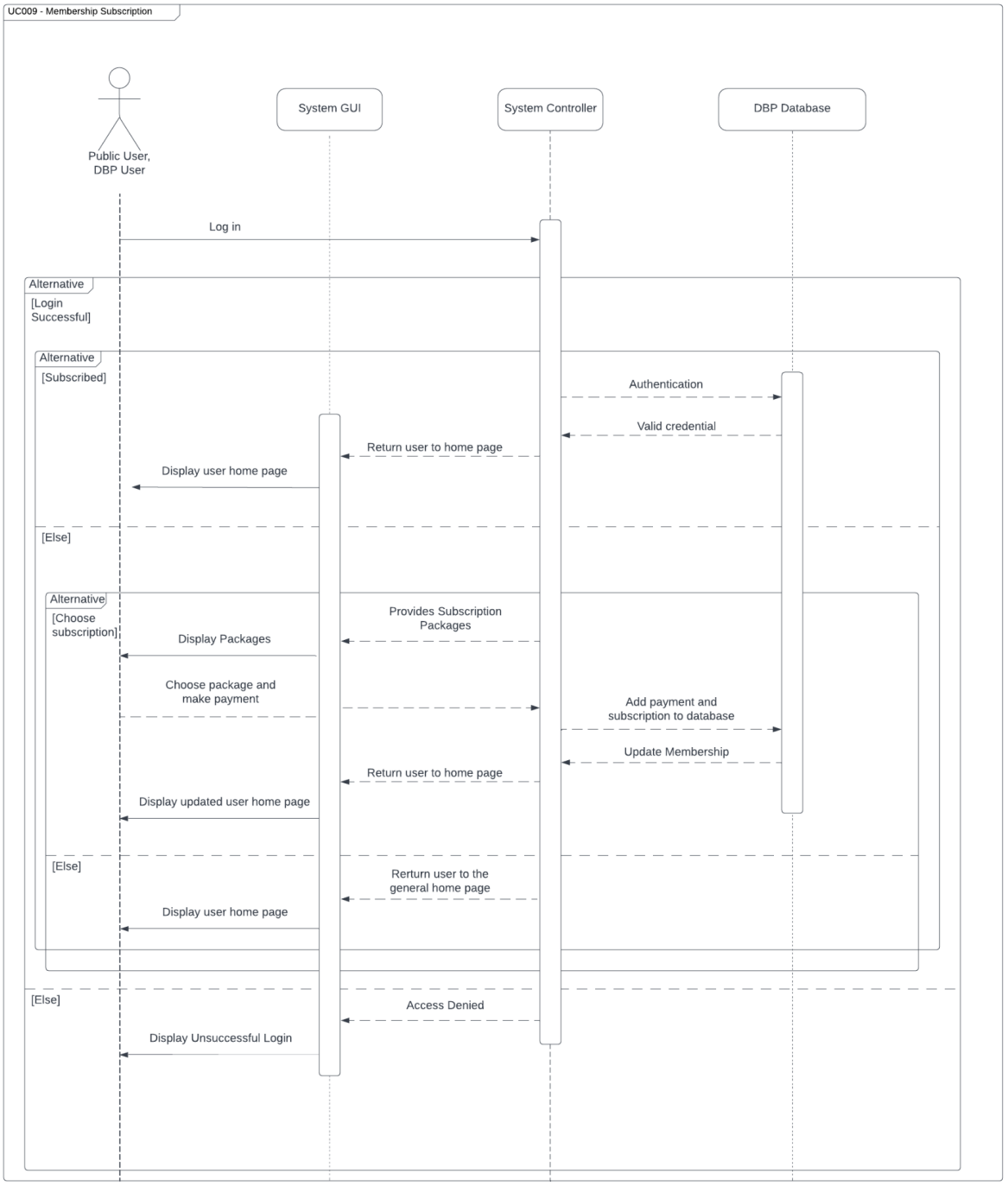

Figure 2.21: Sequence Diagram for Membership Subscription

2.2.10 **UC010: Use Case <Searching Words by Audio or Image>**

2.2.11 **UC011: Use Case <Grammar API Generation>**

2.2.12 **UC012: Use Case <Image-to-text>**

Table 2.7: Use Case Description for <Image-to-text>

Use case: <Image-to-text>
ID: UC012
Actors: DBP User
Brief Description: This use case describes the process of translating the image into text.
Preconditions: DBP User subscribes to the membership and successfully login to the system.
Flow of events: 1. The use case starts when the user selects the camera icon which refers to the “Image-to-text” function beside the search tab. 2. The system will request authorization for using the device’s camera. 3. If user allows the permission, 3.1. The camera will be turned on. 3.2. The user can capture the image of anything he/she wants to translate. 3.3. The system will ask the user to choose whether they want to translate from English to Malay or from Malay to English. 3.4. The system will translate any words that it captures in the image. 4. Else, 4.1. Exception 1 is followed. 5. The use case ends.
Postconditions: 1. Successful operation 1.1. User successfully translate the image. 1.2. User redirected to the home page. 2. Failure operation 2.1. System displays an error message.
Exception flow: 1. Unsuccessful authorization 1.1. The system will prompt the message that it did not get the permission from the user’s device for using the camera, and the user redirected to the home page.

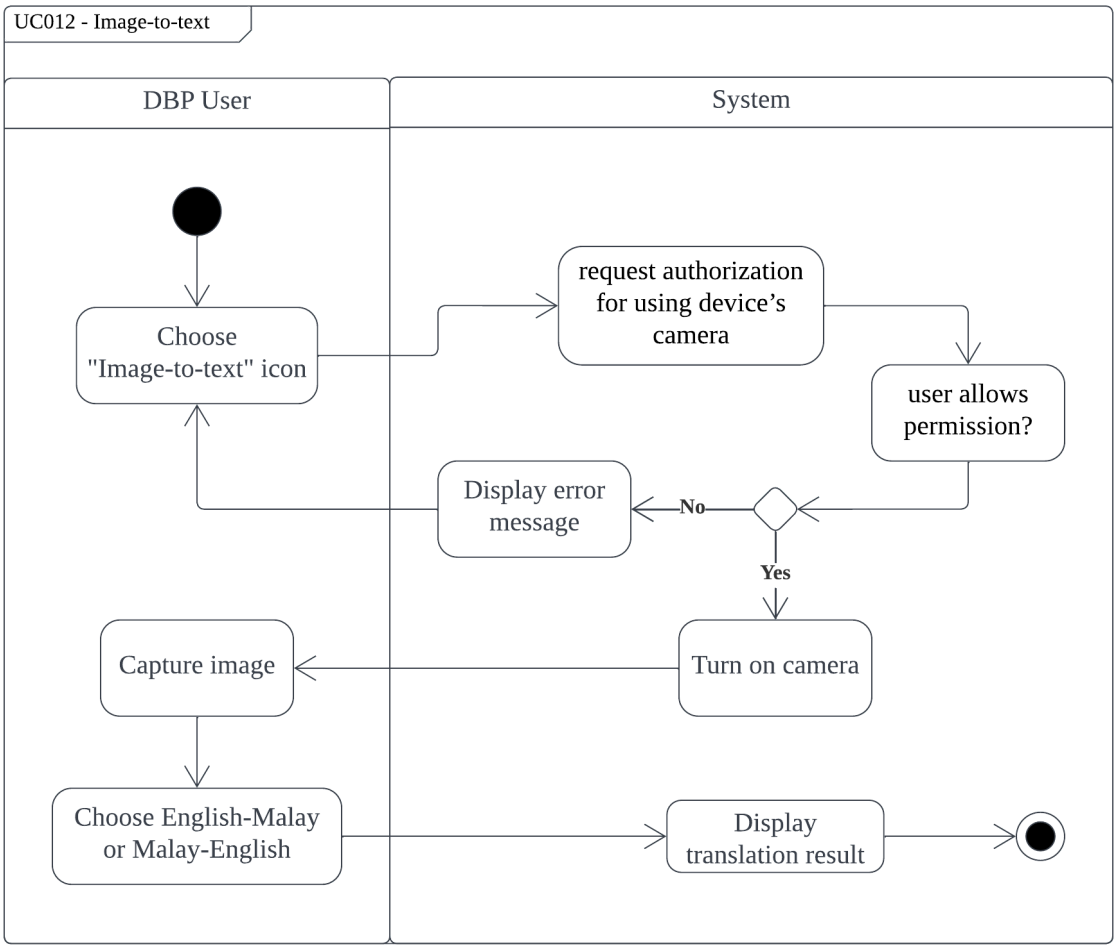

Figure 2.22: Activity Diagram for Translating Image-to-text

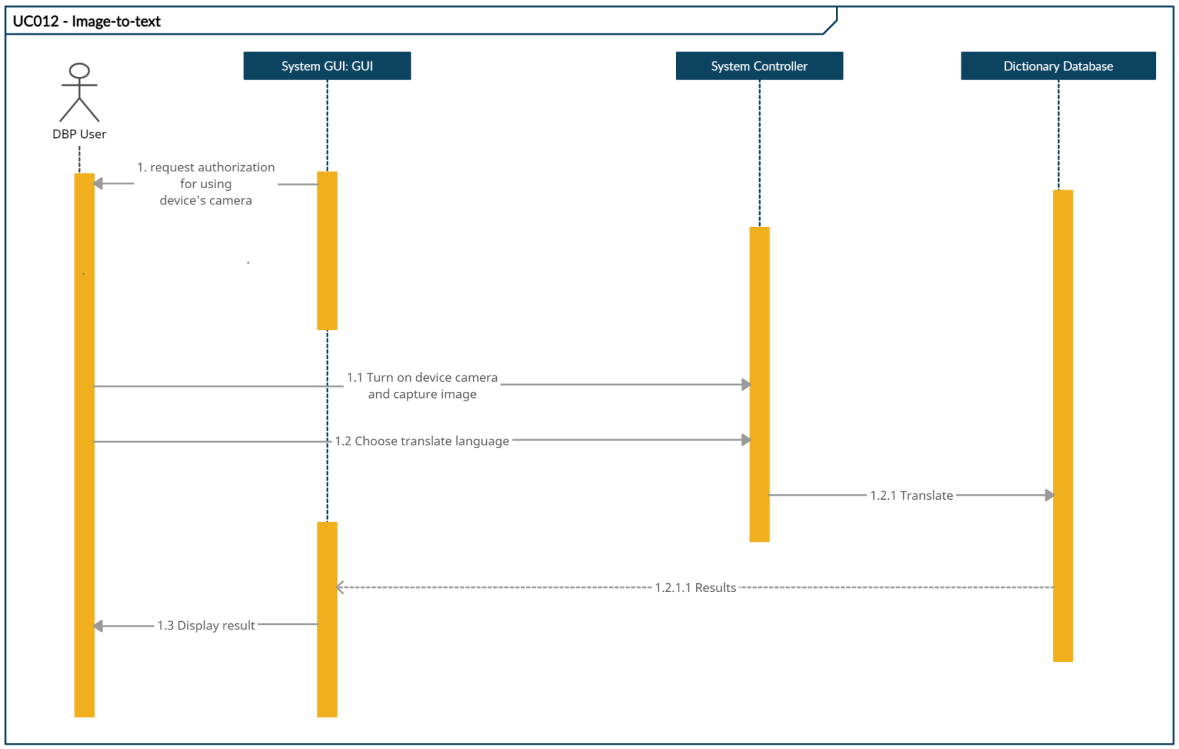

Figure 2.23: Sequence Diagram for Translating Image-to-text

2.2.13 **UC013: Use Case <Manage sentiment word analysis>**

Table 2.8: Use Case Description for <Manage sentiment word analysis>

Use case: <Manage sentiment word analysis>
ID: UC013
Actors: DBP User
Brief Description: This use case describes the process of managing the sentiment word analysis.
Preconditions: DBP User subscribes to the membership and successfully login to the system.
Flow of events: 1. DBP Users write and submit their feedback in the “Feedback” page. 2. System will analyze the feedback and get the keyword that represents the user’s words. 3. All of the feedback will be classified by the keywords. 4. If the user click on the “Sentiment Word Analysis Report”, 4.1. The system will display the result of all feedback in a sentiment analysis dashboard. 5. Else, 5.1. The user can click on the home icon to go back to the main page. 6. The use case ends.
Postconditions: 1. Successful operation 1.1. Users successfully submit their feedback.

1.2. Users successfully view the Sentiment Word Analysis Report. 1.3. Users are redirected to the home page. 2. Failure operation 2.1. System displays an error message. 2.2. Users are redirected to the home page.	
Exception flow (if any): 1. Failure operation 1.1. System displays an error message. 1.2. Users are redirected to the home page.	

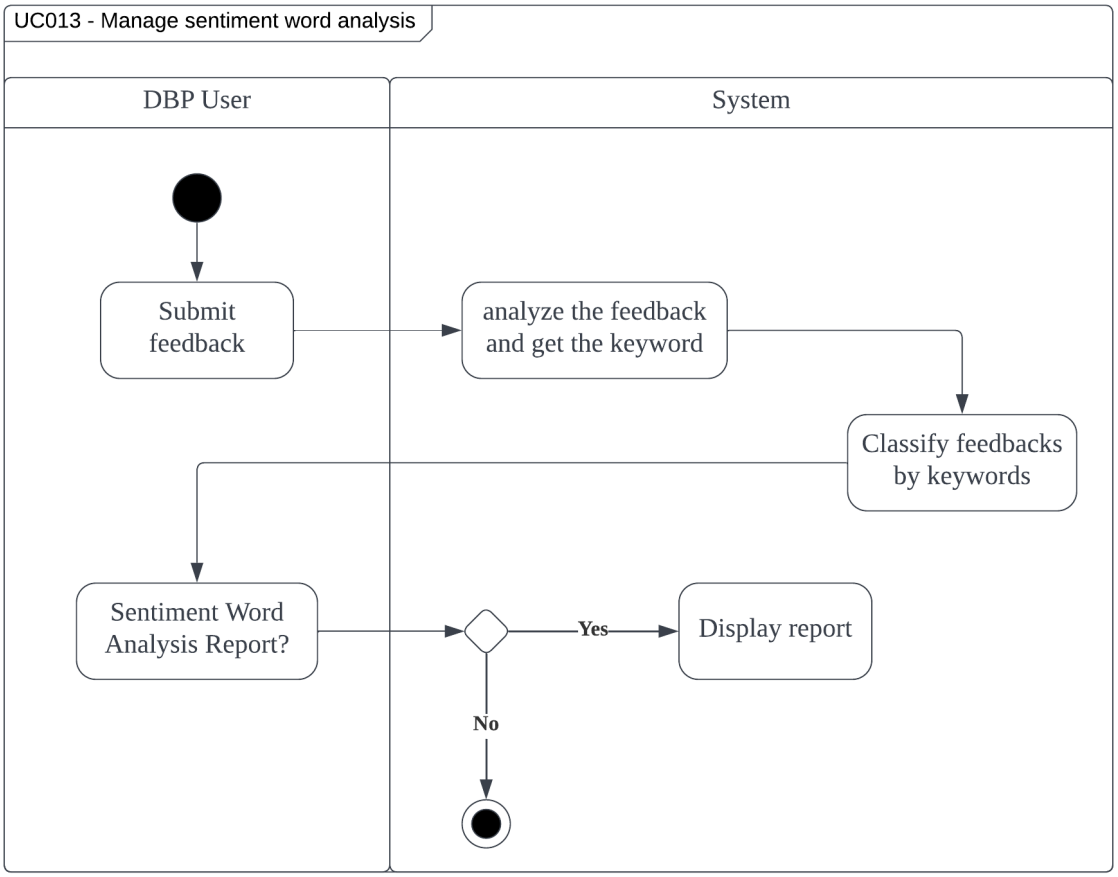

Figure 2.24: Activity Diagram for Manage sentiment word analysis

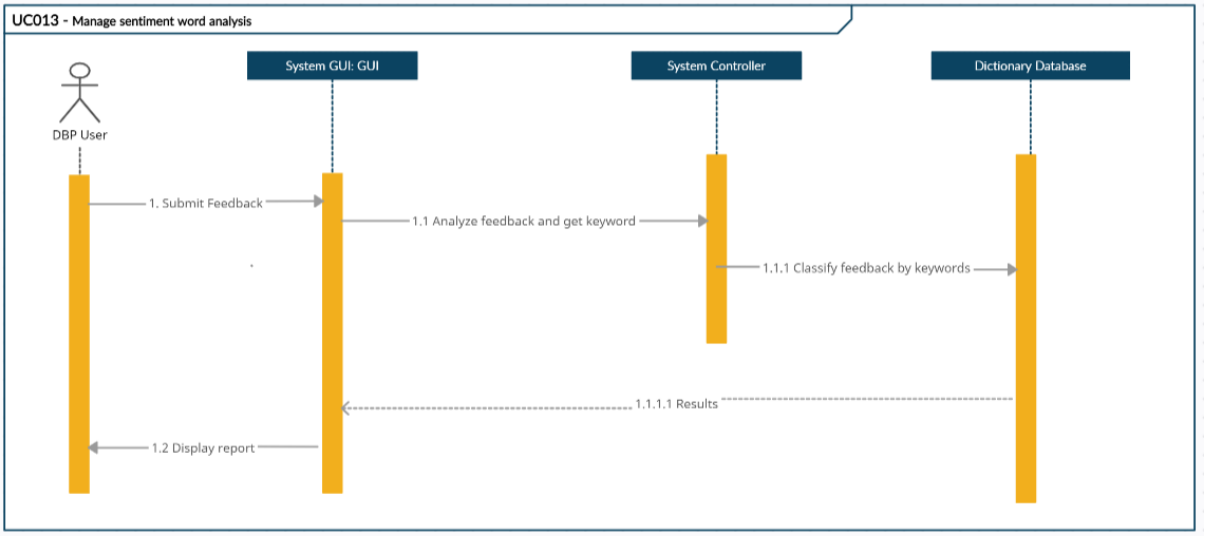

Figure 2.25: Sequence Diagram for Manage sentiment word analysis

2.2.14 **UC014: Use Case <Update dictionary>**

Table 2.9: Use Case Description for <Update Dictionary>

Use case: <Update dictionary>
ID: UC014
Actors: System Administrator
Brief Description: This use case describes the process of updating the dictionary.
Preconditions: System Administrator successfully login to the system.
Flow of events: 1. The use case starts when the system administrator selects the "Update Dictionary" button at the home page. 2. The system will list all the existing words in the dictionary ascendingly from the words starting with letter A. 3. If the system administrator choose the "Add New Words" button, 3.1. A "New Word" form will be given and the system administrator is required to fill in the form with the new word and its details. 4. At the right-hand side of every word, there will be an "Delete" button. 4.1. When the system administrator clicks on the delete button, the system will prompt a message to confirm the action. 5. At the right-hand side of every word, there will be an "Edit" button. 5.1. When the system administrator clicks on the edit button, the system will go to the edit form of the specified word. 5.2. In the edit form, the system administrator can do any changes to the words and its description. 5.3. The system administrator can click on the "Save" button to save their changes to the words. 6. For the add, delete and edit words function, the system will require the system administrator to type their password. 6.1. If the password is correct, the operation is successful and the dictionary will be updated.

6.2. If the password is incorrect, Exception 1 is followed.

7. When the system administrator clicks on the arrow button at the end of the page,
   - 7.1. The system will go to the next page, previous page, first page or last page according to the system administrator's choice of arrow button.
   - 7.2. If the system fails to redirect to the pages, Exception 1 is followed.
8. When the system administrator clicks on the search tab,
   - 8.1. The system administrator are required to choose the type of searching.
   - 8.2. If "Search by page number" is chosen,
     - 8.2.1. The system administrator can enter the number of pages in the search tab.
   - 8.3. If "Search by word" is chosen,
     - 8.3.1. The system administrator will enter the word they want to find.
   - 8.4. If "Search by starting letter" is chosen,
     - 8.4.1. The system administrator can enter the starting letter that they want to search.
   - 8.5. The system will show the searching result.
   - 8.6. Else,
     - 8.6.1. Exception 1 is followed.
9. When the system administrator clicks on the "Cancel" button at the end of the page,
   - 9.1. The system administrator redirected to the home page.
10. The use case ends.

Postconditions:

1. Successful Operation
   - 1.1. System administrator successfully updated the dictionary.
   - 1.2. System administrator redirected to the home page.
2. Failure Operation
   - 2.1. System displays an error message.

Exception flow:

1. Failure operation
   - 1.1. System displays an error message.
   - 1.2. System redirect to the home page.

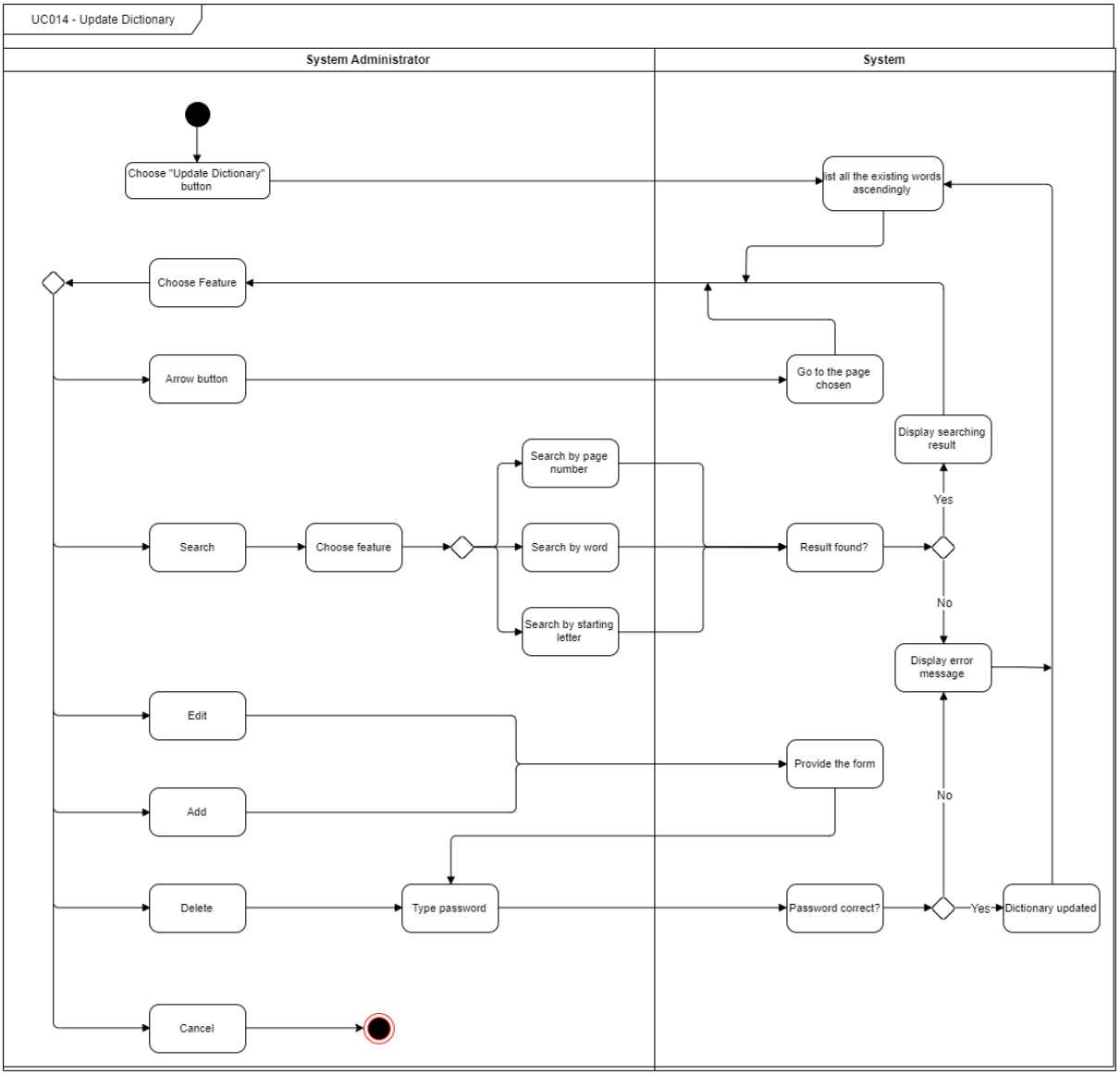

Figure 2.26: Activity Diagram for Updating Dictionary

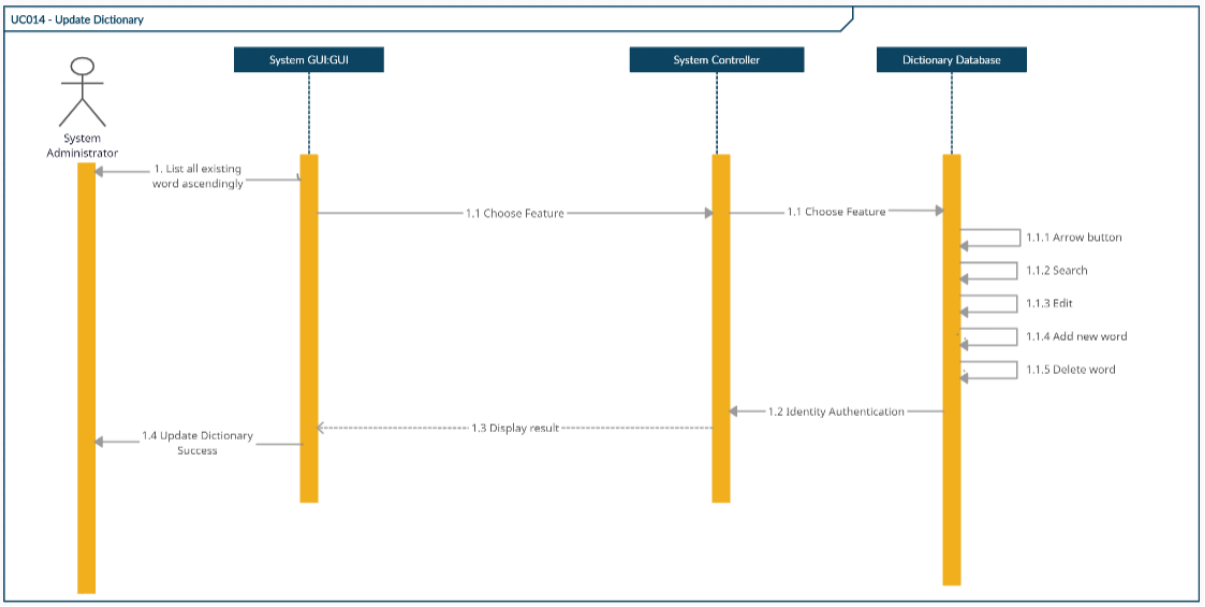

Figure 2.27: Sequence Diagram for Update Dictionary

2.3 **Performance and Other Requirements**

The system should be optimized in terms of utilization and performance. Therefore, the requirements are as below:

Table 2.10: Performance and Other Requirements

Requirement	Description
Security	• User authentication: Any user who wants to do the translation needs to have access to the system. The system should recognize the user through their unique id and password. • Database: Any insertion, deletion or update on the database can be made by the system administrator only.
Portability	• The website should be both mobile-friendly and desktop-friendly. It is compatible with various browsers (E.g: Google Chrome, Safari, Mozilla Firefox etc).
Usability	• All functions in the system work accurately and efficiently. The system will produce correct output for the user.
Maintainability	• The system offers the efficiency of data backup.

2.4 **Design Constraints**

There are some design constraints that are needed to be applied in completing this project. Those constraints are as follow:

Table 2.11: Design constraints

Constraint	Description
Brand and Style Guidelines	● Any design created shall adhere to DBP's brand identity in terms of colors, font and professionnalisme. ● Every page shall have a header with DBP's logo.
Device Specifics	● The system shall be compatible on both website and mobile application. ● The interface should be minimalistic but compact with features for every device used.
Usability	● The interface should be easy to use for new users with none to little learning curve.
Compliance	● Any pages displayed shall not contain any copyrighted items that can violate the copyright laws.
User Feedback	● The system shall provide a way to gather user's feedback in order to improve the product.

2.5 **Software System Attributes**

1. Speed: The system functions must be fast and accurate for all kinds of tasks.
2. Scalability: The system must be able to handle a large number of all at once.
3. Integrity or security: The system's sensitive functions can only be accessed by authorized users.
4. Usability: The system's user interface must be appealing, simple to understand, and simple to use for all sorts of users.
5. Availability: The system must be able to operate 24 hours a day, seven days a week.
6. Reliability: Any changes to the dictionary will be quickly updated to the system, with a maximum downtime of five seconds.

3. System Architectural Design

3.1 Architecture Style and Rationale

For our DBP Dictionary, we have decided to choose the layered architecture style. As mentioned, this style is suitable to be used when building new facilities on top of existing systems. For our system, we are building this proposed system based on the existing dictionary system which exists on the market now. Moreover, our proposed system also requires multi-level security since the data of our users and system administrator will be stored and used in our system. Therefore, we have determined that the layered architecture style is the most suitable style for our proposed system.

3.2 Class Diagram

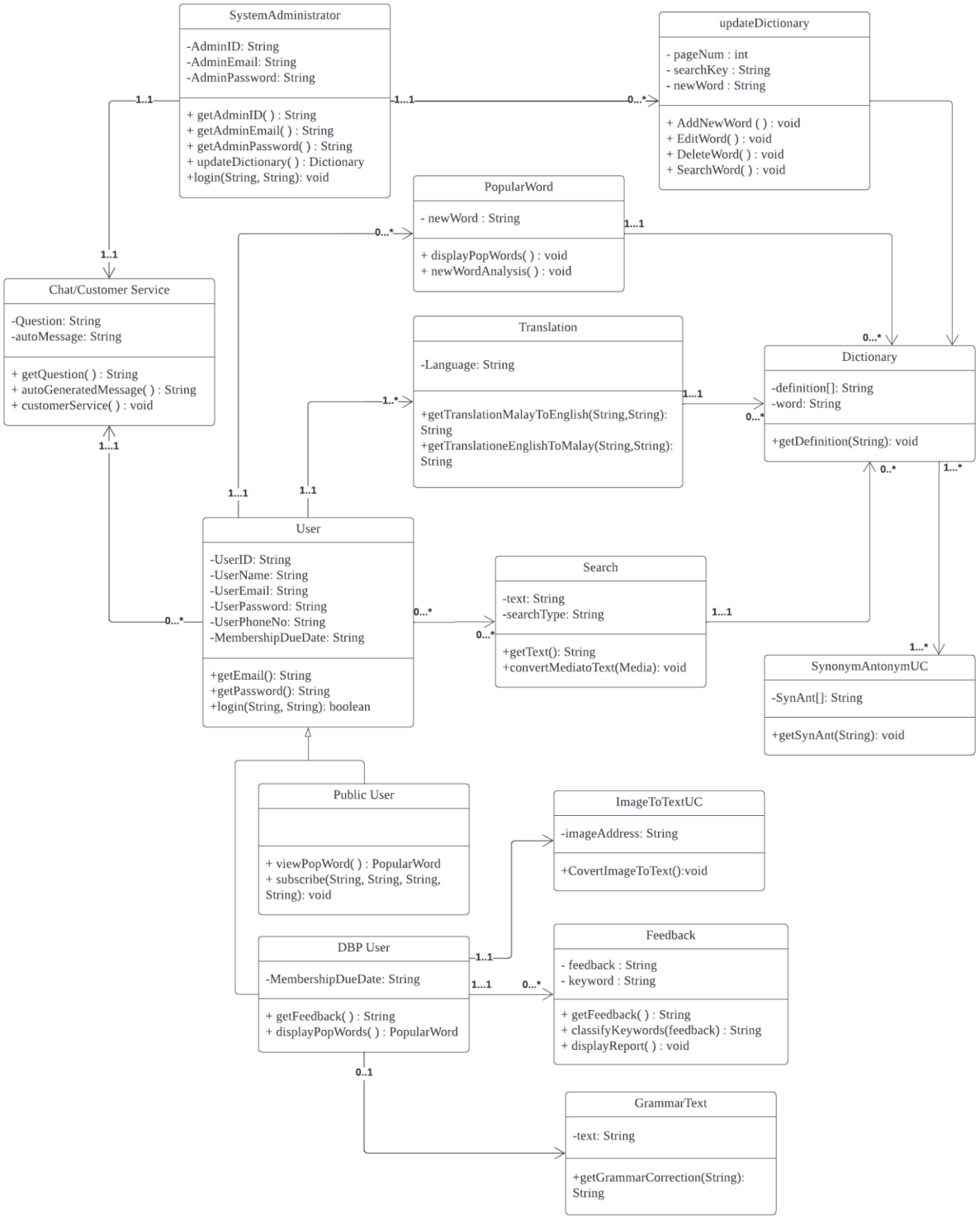

Figure 3.1: Class Diagram of Improvised DBP Dictionary System

Table 3.2.1: Method in Entity Translation

Entity Name	Translation
Method Name	getTranslationFromMalayToEnglish

Input	Language, Text(Word or Sentence)
Output	Translated text
Algorithm	1. Start. 2. Provide a new translation form. 3. Require the user to fill in Translation details. 4. Read Entered Text. 5. Display the translation. 6. End.

Table 3.2.2: Method in Entity Translation

Entity Name	Translation
Method Name	getTranslationFromEnglishToMalay
Input	Language,Text(Word or Sentence)
Output	Translated text
Algorithm	1. Start. 2. Provide a new translation form. 3. Require the user to fill in Translation details. 4. Read Selected Language. 5. Read Entered Text. 6. Display the translation. 7. End.

Table 3.2.3: Method in Entity GrammarText

Entity Name	GrammarText
Method Name	getGrammarCorrection
Input	Text
Output	Corrected text
Algorithm	1. Start 2. Require the user to log into the system to check if he/she is a DBP user. 3. Require the user to enter targeted text. 4. Display Edited Text 5. End

Table 3.2.4: Method in Entity SystemAdministrator

Entity Name	SystemAdministrator
Method Name	Login
Input	username , password
Output	-
Algorithm	1. Start 2. Create a new login form 3. Require admin to insert username and password 4. check if username and password correct 5. display the status of the system. 6. End

Table 3.2.5: Method in Entity Public User

Entity Name	Public User
Method Name	viewPopWord
Input	-
Output	popular word data
Algorithm	1.Start 2. The user must log in successfully. 3.display poplar word 4.End

Table 3.2.6: Method in Entity Public User

Entity Name	Public User
Method Name	subscribe
Input	all user-standard data(name,email,password, phone number).
Output	-
Algorithm	1. Start. 2. Requires the user to insert all standard data. 3. Display the status of the operation. 4. End.

Table 3.2.7: Method in Entity DBP User

Entity Name	DBP user
Method Name	displayPopWords
Input	-
Output	Popular word
Algorithm	1. Start 2. Check if the user has membership. 3. Display popular word 4. End

Table 3.2.8: Method in Entity ImageToTextUC

Entity Name	ImageToTextUC
Method Name	CovertImageToText
Input	media
Output	-
Algorithm	1. Start 2. require user to insert image 3. process image 4. extract data from the image 5. display extracted data to the user. 6. End

Table 3.2.9: Method in Entity SynonymAntonym

Entity Name	SynonymAntonym
Method Name	getSynAnt
Input	word
Output	list of synonym and antonym for word
Algorithm	1. Start 2. The method will take a variable word with data type String from Main as its parameter. 3. The database will be searched to find the synonyms and antonyms of the word. 4. The synonyms and antonyms will later be stored in an array variable named SynAnt[] with datatype String. 5. The variable SynAnt[] will then be printed out. 6. End

Table 3.2.10: Method in Entity Dictionary

Entity Name	Dictionary
Method Name	getDefinition
Input	word
Output	definition for word
Algorithm	1. Start 2. The method will take a variable word with data type String from Main as its parameter. 3. The database will be searched to find the definition of the word. 4. The definition will later be stored in an array variable named definition[] with datatype String. 5. The variable definition[] will then be printed out. 6. End

Table 3.2.11: Method in Entity Search

Entity Name	Search
Method Name	convertMediatoText
Input	Media
Output	Media convert to text as a string
Algorithm	1. Start 2. The method will take the Media from Main as its parameter. 3. The Media then will be converted to text form. 4. Variable text in this entity then will be set to the converted text. 5. End

Table 3.2.12: Method in Entity Search

Entity Name	Search
Method Name	getText
Input	-
Output	text
Algorithm	1. Start 2. The method will return the text with data type String. 3. End

Table 3.2.13: Method in Entity User

Entity Name	User
Method Name	getEmail
Input	-
Output	UserEmail
Algorithm	1. Start 2. The method will return UserEmail with data type String. 3. End

Table 3.2.14: Method in Entity User

Entity Name	User
Method Name	getPassword
Input	-
Output	UserPassword
Algorithm	1. Start 2. The method will return the UserPassword with data type String. 3. End

Table 3.2.15: Method in Entity Dictionary

Entity Name	Dictionary
Method Name	AddNewWord
Input	-
Output	The added word
Algorithm	1. Start. 2. Provide a new word form. 3. Require the System Administrator to fill in all details. 4. Require the System Administrator to enter their password. 5. If the password is correct then a new word is added successfully. 6. Display the new word. 7. End.

Table 3.2.16: Method in Entity Dictionary

Entity Name	Dictionary
Method Name	EditWord
Input	-
Output	The updated word
Algorithm	1. Start. 2. Provide Edit Word Form. 3. System Administrator is allowed to make changes for the word. 4. If the “Save” button is clicked, require System Administrator to enter their password. 5. If the password is correct then the word is updated successfully. 6. Display the updated word. 7. End.

Table 3.2.17: Method in Entity PopularWord

Entity Name	PopularWord
Method Name	displayPopWords
Input	-
Output	The popular words
Algorithm	1. Start. 2. The system will fetch the popular words from the database. 3. Display all the words with their definitions. 4. End.

Table 3.2.18: Method in Entity PopularWord

Entity Name	PopularWord
Method Name	newWordAnalysis
Input	new word
Output	Information of the word
Algorithm	1. Start. 2. The user inserts new words. 3. The system will fetch the words from the database. 4. Display the words with their definitions. 5. End.

Table 3.2.19: Method in Entity Chat/Customer Service

Entity Name	Chat/Customer Service
Method Name	getQuestion
Input	Question/User's inquiry
Output	-
Algorithm	1. Start. 2. User insert questions. 3. The question is returned to the system as a string. 4. End.

Table 3.2.20: Method in Entity Chat/Customer Service

Entity Name	Chat/Customer Service
Method Name	customerService
Input	Customer's inquiry
Output	Answer by system or admin
Algorithm	1. Start. 2. User choose live chat or customer service. 3. User insert their inquiry. 4. If live chat is chosen, 4.1. System displays auto generated messages. 5. If customer service is chosen, 5.1. System provides information needed by customers. 6. End.

Table 3.2.21: Method in Entity Feedback

Entity Name	Feedback
Method Name	getFeedback
Input	User's Feedback
Output	-
Algorithm	1. Start. 2. User insert feedback. 3. Return the feedback as a string to classifyKeywords(feedback) method. 4. End.

Table 3.2.22: Method in Entity Feedback

Entity Name	Feedback
Method Name	displayReport
Input	-
Output	Sentiment Word Analysis Report
Algorithm	1. Start. 2. System displays the analysis report according to the users' feedback. 3. End.

4. Detailed Description of Components

4.1 Complete Package Diagram

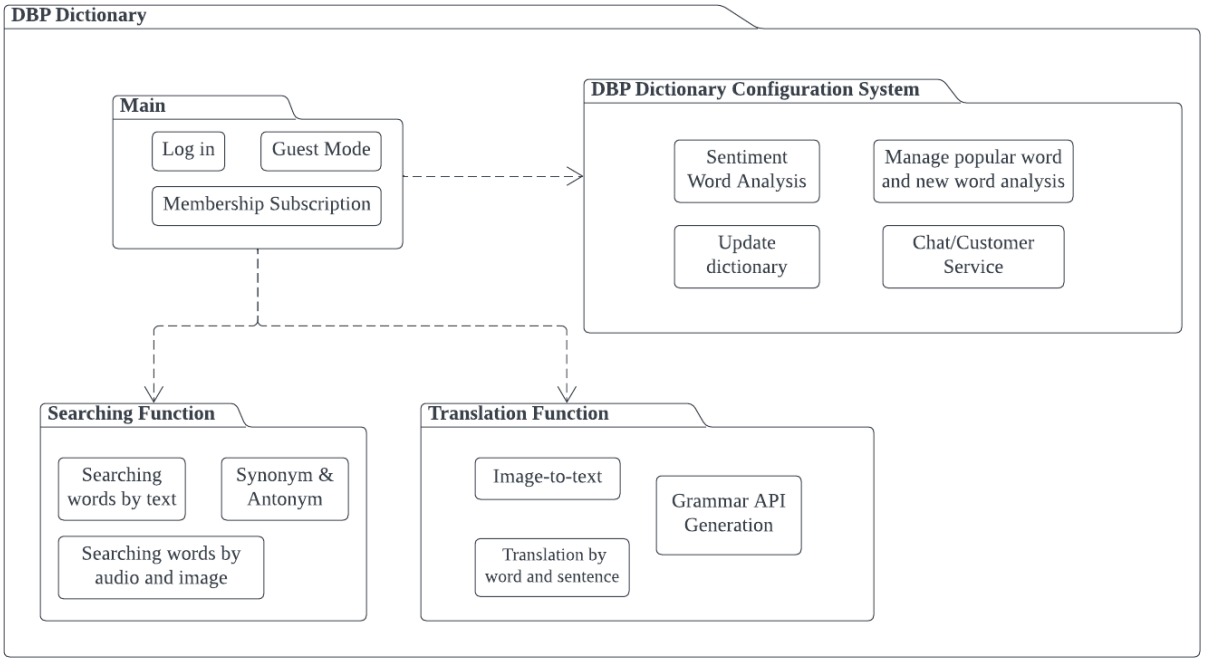

Figure 4.1: Package Diagram for DBP Dictionary

4.2 Detailed Description

4.2.1 P001: Main Subsystem

4.2.1.1 Sequence Diagram

4.2.2 **P002: Translation Function Subsystem (Mohand)**

The translation function subsystem is some of the essential futures in the system. It consists of translation by word and sentence functions, grammar correction, and image-to-text functions. In the beginning, the user will have access to the view layer, which consists of the translation window, ImageToText window, grammar correction window, DBP user window, and public user window. Each window directs the user to its function. For example, the translation window directs the user to the translation by word and sentence functions.In addition, the DBP window and public user window direct the user to their main profile page based on their membership. Each domain layer function has its own set of process techniques. The translation process requires the user to enter the selected language and enter the target language in order for the system's algorithm to do the task and produce output sufficiently. In addition, grammar correction uses API software, which consists of an advanced AI system that has powerful functionality. Also, the image to text function uses advanced software to extract the text in any image. All the functions have access to all the system data. The data access layer consists of all the data related to the software and the user, like users' databases, dictionary databases, and image databases.

4.2.2.1 **Sequence Diagram**

a) SD001: Sequence diagram for Translation function

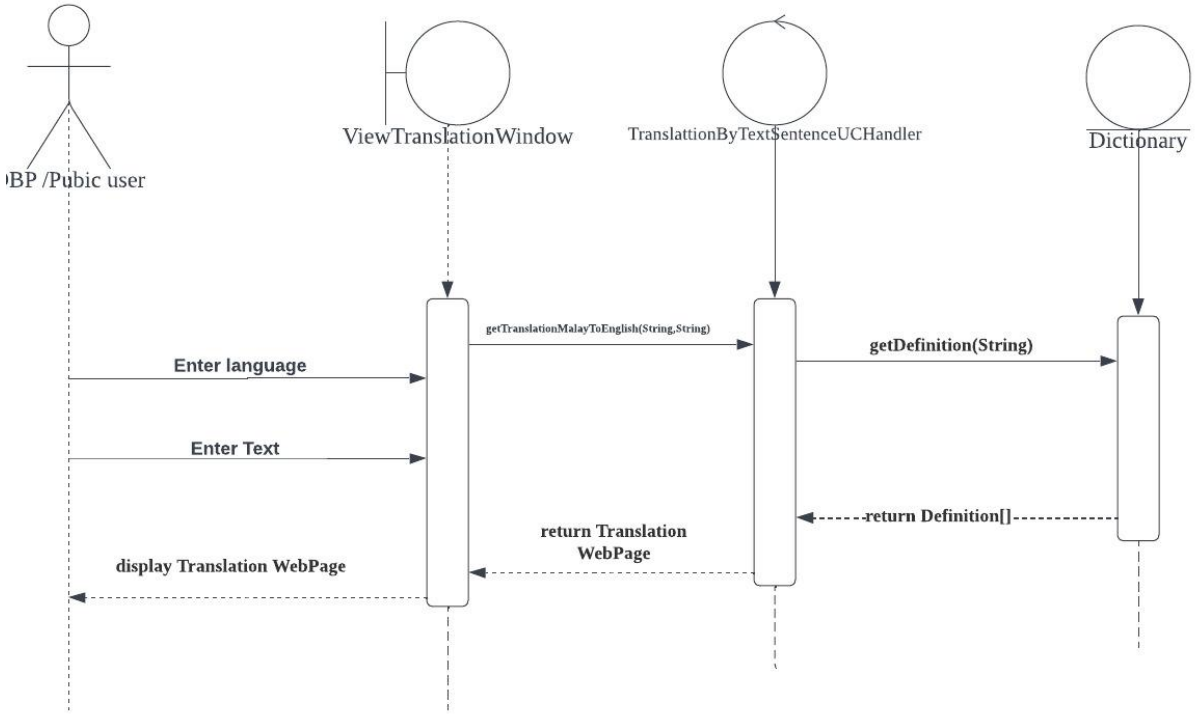

Figure 4.2: Sequence diagram for Translation function

a) SD002: Sequence diagram for **Grammar Correction**

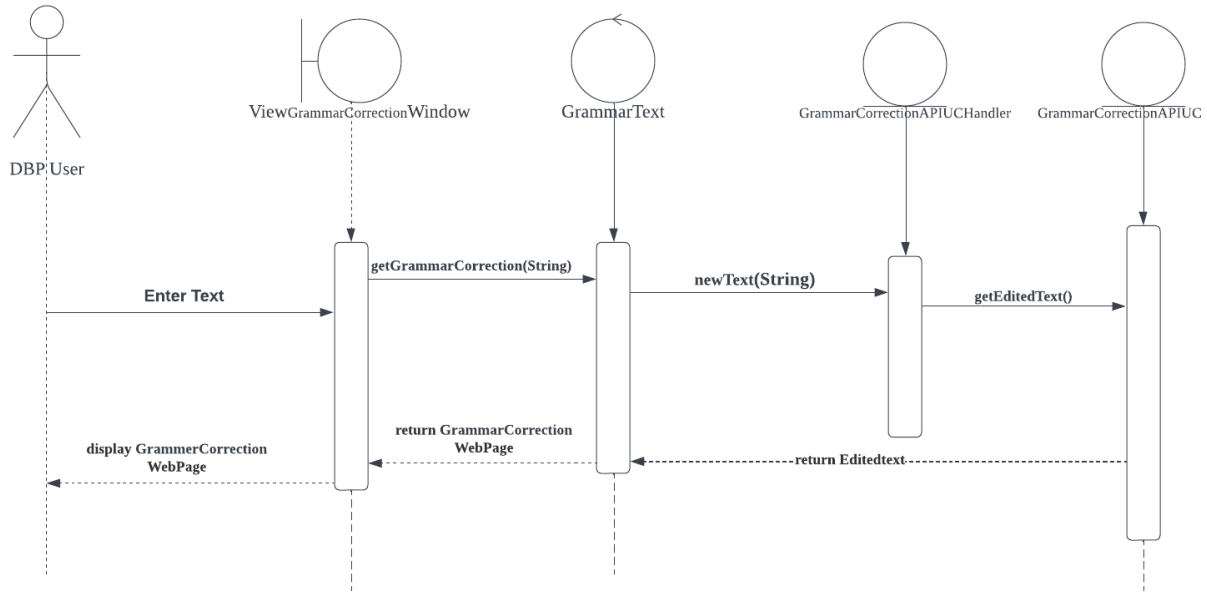

Figure 4.3: Sequence diagram for grammar correction

4.2.3 **P003: Searching Function Subsystem**

This subsystem consists of use cases in the category of searching functions such as searching words by text, searching words by audio and image and getting synonyms and antonyms. These are the options that the users have while searching for the meaning of a word or getting the synonyms and antonyms of the word. First of all, in the view layer of this subsystem, it consists of ViewPublicUser, ViewDBPUser, ViewDefinition and ViewSynonymAntonym. For ViewPublicUser, it is the main page for Public User. Next, for ViewDBPUser, it is the main page for DBP User. Meanwhile for ViewDefinition, it is the page that will display the word and its definition. In the ViewDefinition page, users have an option whether to get the synonyms and antonyms of the word. If the user clicks on the button, it will bring the user to the ViewSynonymAntonym page where it will display the synonyms and antonyms. Secondly, in the domain layer, it consists of SearchWordbyTextUCHandler, SearchWordbyMediaUCHandler and SynonymAntonymUCHandler. The handler classes will act as the mediator between boundary and entity. Lastly, the data access layer consists of DictionaryDA and SynonymAntonymDa. For DictionaryDa, it is the database for the dictionary while SynonymAntonymDa is the database for Synonym and Antonym.

4.2.3.1 **Sequence Diagram**

a) SD001: Sequence diagram for Search by Text

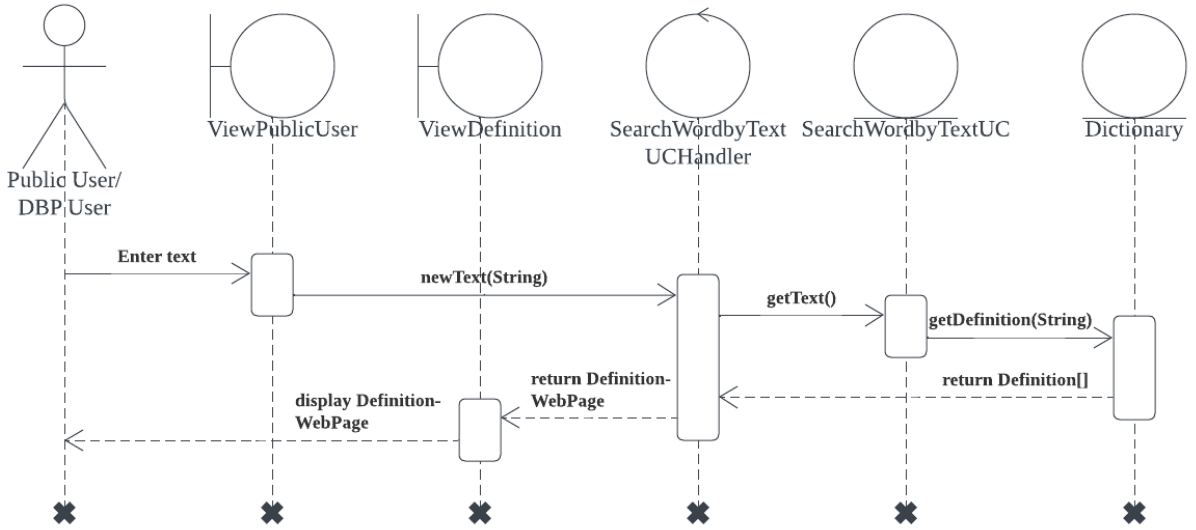

Figure 4.4: Sequence diagram for Search by Text

b) SD002: Sequence diagram for Search by Audio or Image

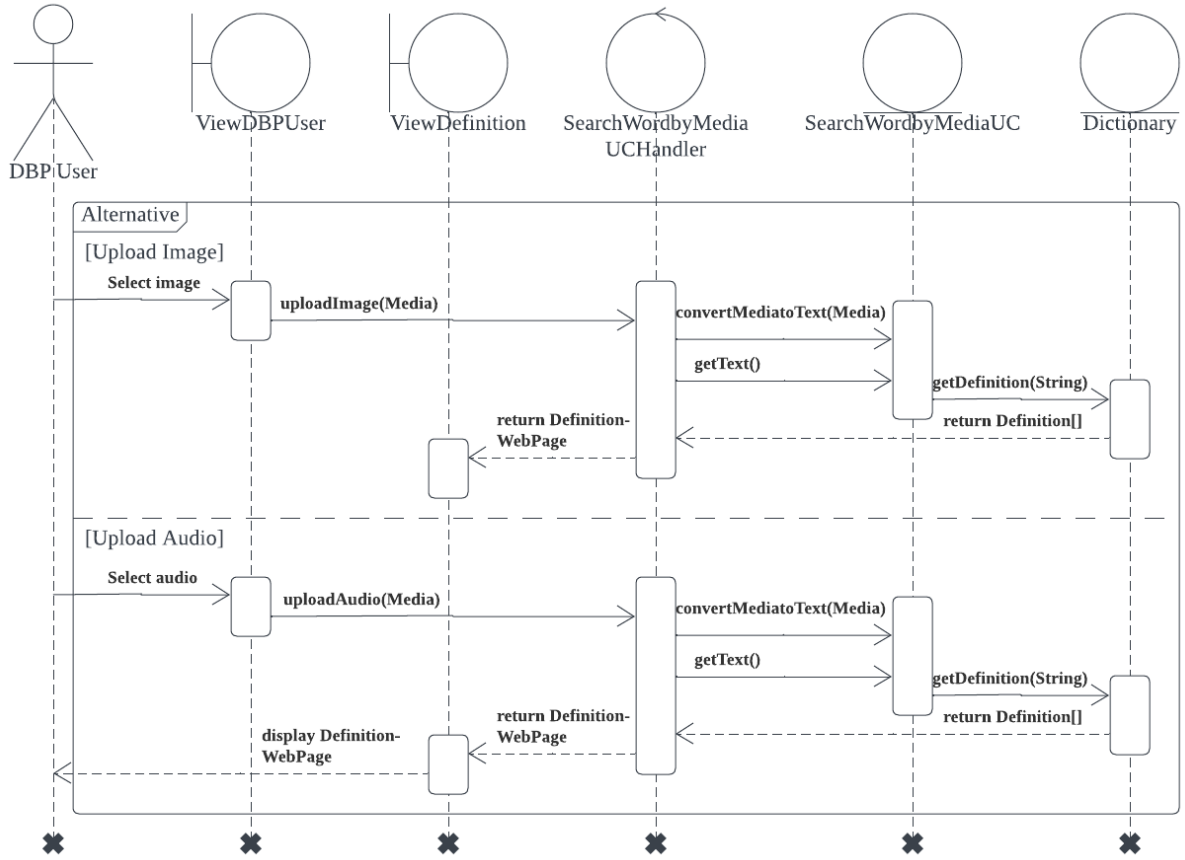

Figure 4.5: Sequence diagram for Search by Audio or Image

c) SD003: Sequence diagram for Synonym & Antonym

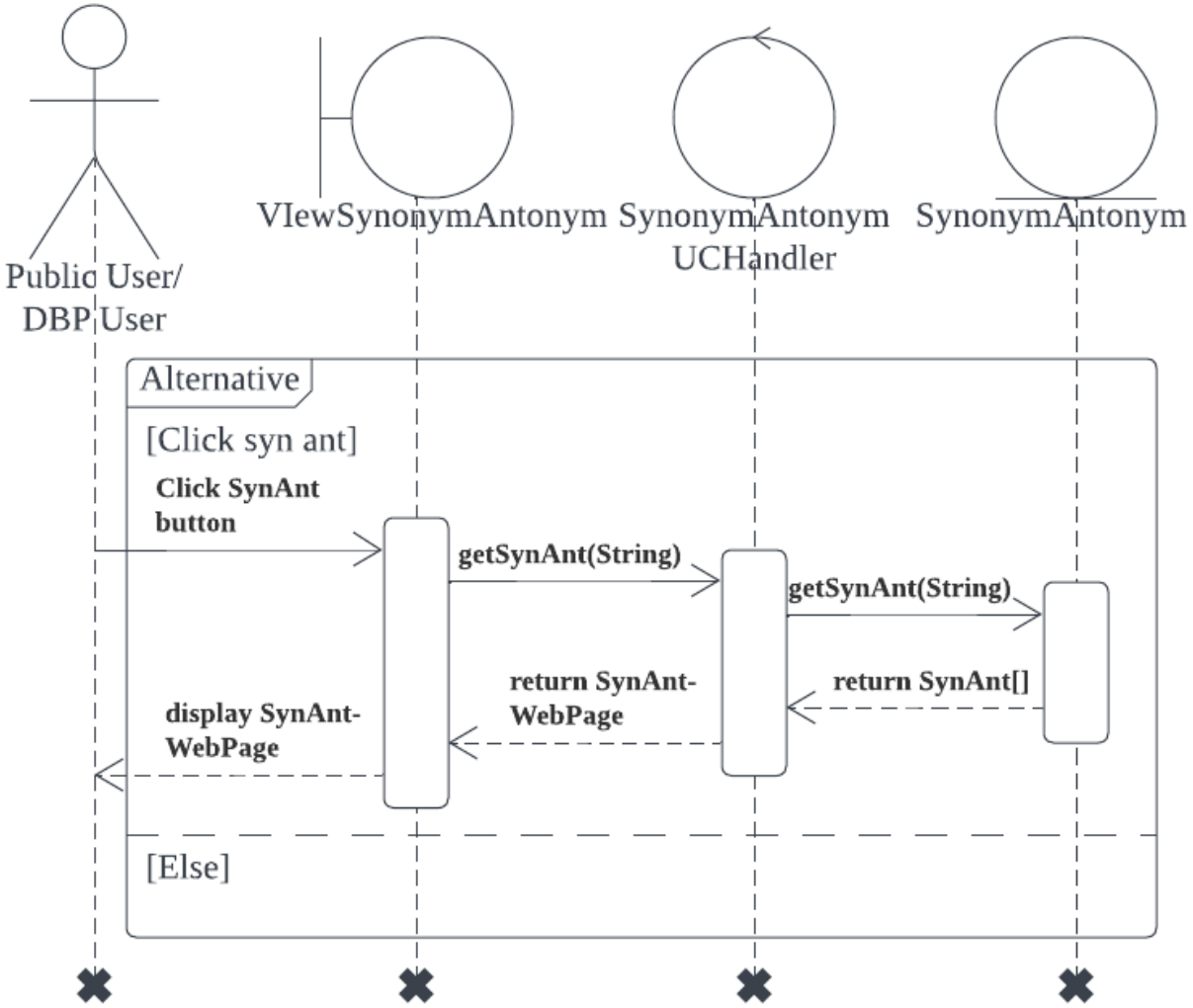

Figure 4.6: Sequence diagram for Synonym & Antonym

4.2.4 **P004: DBP Dictionary Configuration System**

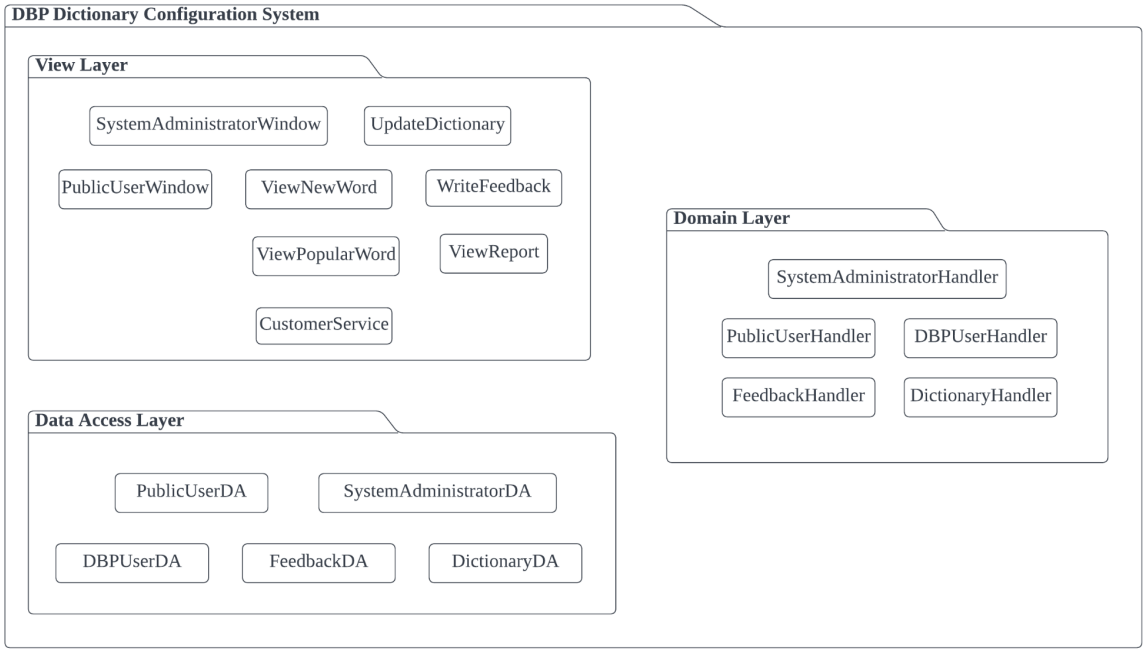

Figure 4.7: Package Diagram for Dental Clinic Maintenance Subsystem

This subsystem is named DBP Dictionary Configuration System which includes all the use cases that take part in the configuration job of the DBP Dictionary. In the View Layer, it includes the SystemAdministratorWindow, PublicUserWindow and DBPUserWindow which refers to the main page of system administrator, public user and DBP user respectively. In SystemAdministratorWindow, there is an UpdateDictionary window and a CustomerService window. Besides, the ViewNewWord and ViewPopularWord window which relate to the 2 analysis functions are available for both public user and DBP user. DBP users have a unique window that is WriteFeedback and ViewReport as this function sentiment word analysis is only available for DBP users. Beside that, we have 5 handlers in the domain layer that are SystemAdministratorHandler, DBPUserHandler, PublicUserHandler, FeedbackHandler and DictionaryHandler. These handler classes act as the mediator between boundary and entity. Next we have the data access layer. In this layer, the PublicUserDA, DBPUserDA and SystemAdministratorDA are the databases which store the information of public user, DBP user and system administrator respectively. Other than that, the FeedbackDA is used to store the

feedback given by DBP users while the DictionaryDA is the database for all vocabs and their information.

4.2.4.1 Sequence Diagram

a) SD002: Sequence diagram for Manage sentiment word analysis

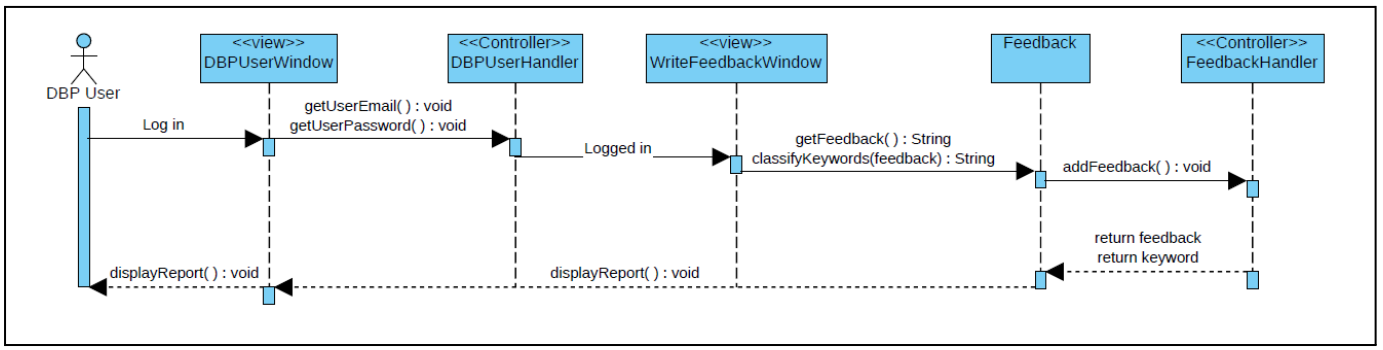

Figure 4.8: Sequence diagram for Manage sentiment word analysis

b) SD003: Sequence diagram for Update Dictionary

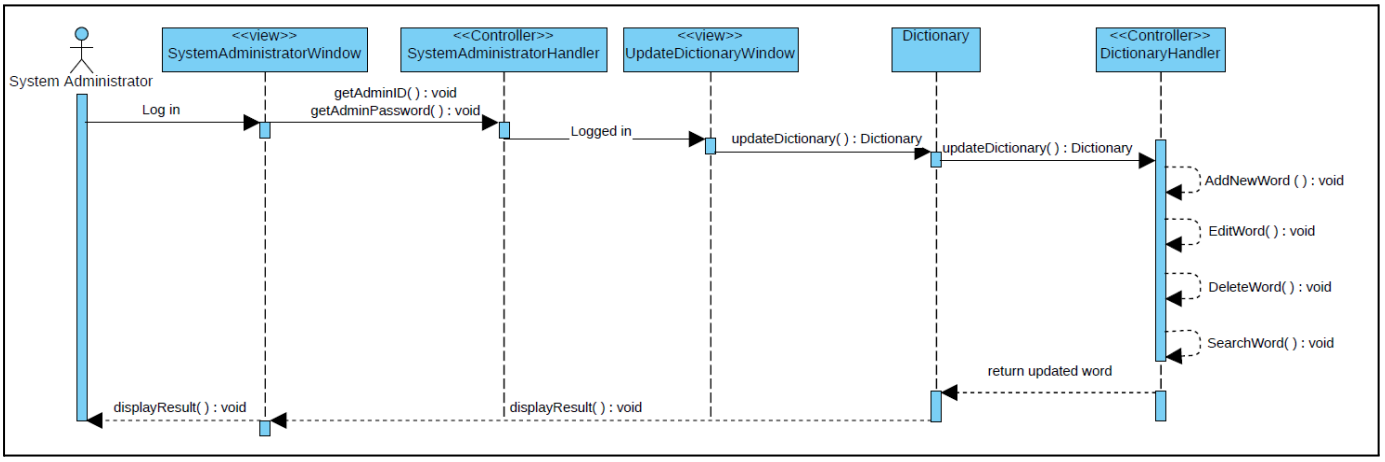

Figure 4.9: Sequence diagram for Update Dictionary

5. Data Design

5.1 Data Description

The major data or systems entities are stored into a relational database named DBP Dictionary database, processed and organized into 12 entities as listed in Table 5.1.

Table 5.1: Description of Entities in the Database

No.	Entity Name	Description
1	SystemAdministrator	Stored the data and attributes of system administrator, therapists and staff.
2	updateDictionary	Stored the attributes of the dictionary that will be used to update the dictionary.
3	Chat/Customer Service	Stored the data and chats between user and administrator.
4	DBPUser	Stored the data and attributes of paid user which is DBP User
5	PublicUser	Stored the data and attributes of free user which is Public User
6	PopularWord	Stored the data of new popular words.
7	ImageToTextUC	Stored the image uploaded by the user to be converted to text.
8	Feedback	Stored the feedback received by DBP User to be used in sentiment word analysis.
9	Translation	Stored the translated word entered by the user.
10	Search	Stored the data of search functions and search type.
11	Dictionary	Stored the data and attributes of the dictionary.
12	SynonymAntonymUC	Stored the arrays of synonyms and antonyms of entered words.

5.2 **Data Dictionary**

5.2.1 **Entity: <**SystemAdministrator**>**

Attribute Name	**Type**	**Description**
AdminID	String	admin unique id
AdminEmail	String	admin email
AdminPassword	String	admin unique password

5.2.2 **Entity: <**updateDictionary**>**

Attribute Name	**Type**	**Description**
pageNum	int	Dictionary Page number
searchKey	String	Search main identifier
newWord	String	The new inserted word

5.2.3 **Entity: <**Chat/Customer Service**>**

Attribute Name	**Type**	**Description**
Question	String	user new question
autoMessage	String	automated message

5.2.4 **Entity: <**DBPUser**>**

Attribute Name	**Type**	**Description**
UserID	String	user unique identifier
UserName	String	The name of DBP user
UserEmail	String	The email of a DBP user
UserPassword	String	DBP user's unique password
UserPhoneNo	String	DBP user’s phone number
MembershipDueDate	String	DBP user membership due date

5.2.5 **Entity: <**PublicUser**>**

Attribute Name	**Type**	**Description**
UserID	String	public user unique identifier
UserName	String	The name of public user
UserEmail	String	The email of a public user
UserPassword	String	public user's unique password
UserPhoneNo	String	public user phone number

5.2.6 **Entity: <**PopularWord**>**

Attribute Name	**Type**	**Description**
newWord	String	The newly added word

5.2.7 **Entity: <**ImageToTextUC**>**

Attribute Name	**Type**	**Description**
imageAddress	String	The location of the new image in the system

5.2.8 **Entity: <**Feedback**>**

Attribute Name	**Type**	**Description**
feedback	String	feedback description
keyword	String	feedback unique keyword

5.2.9 **Entity: <**Translation**>**

Attribute Name	**Type**	**Description**
Language	String	the translated to language

5.2.10 **Entity: <**Search**>**

Attribute Name	**Type**	**Description**
text	String	Search keyword
searchType	String	Search type

5.2.11 **Entity: <**Dictionary**>**

Attribute Name	**Type**	**Description**
definition	String	The definition in the dictionary
word	String	The keyword of definition

5.2.12 **Entity: <**SynonymAntonymUC**>**

Attribute Name	**Type**	**Description**
SynAnt	String	SynonymAntonym keyword

Product Owner	Product ID	Product Name	Status
AHMAD NAZRAN BIN YUSRI	UC001	Searching words by text	Completed
	UC010	Searching words by audio and image	Completed
	UC002	Synonym and Antonym	Completed
ANATASYA HUMAIRA	UC008	Login	Not completed
	UC009	Membership subscription	Not completed
MOHAND ALAA ABOUZEID MOHAMED	UC012	image to text	Completed
	UC005	Translation by Word and Sentence	Completed
	UC011	GrammarAPI generation	Completed
NG YEN THONG	UC012	Image-to-text	Completed
	UC013	Manage sentiment word analysis	Completed
	UC014	Update dictionary	Completed