

Requirement Specification

METaverse LIBRARY SYSTEM

Introduction to Software Engineering 41 - Group 11

| | |
|------------|---------------------------|
| 2016312818 | Changmin Park |
| 2018311925 | Dajung Choi |
| 2018312257 | Youngsuh Chin |
| 2019313857 | Amirah Binti Ahmad Nadzri |
| 2016314576 | Jaehyuk Choi |
| 2016314494 | Wonjae Lee |
| 2019313897 | Angel Fitri Sari |

CONTENTS

| | |
|--|-----------|
| 1. Introduction..... | 5 |
| 1.1. Purpose | 5 |
| 1.2. Scope | 5 |
| 1.3. Definitions, Acronyms, and Abbreviation | 6 |
| 1.4. References | 7 |
| 1.5. Overview | 7 |
| 2. Overall Description | 8 |
| 2.1. Product Perspective | 8 |
| 2.1.1. System Interfaces..... | 8 |
| 2.1.2. User Interfaces | 8 |
| 2.1.3. Hardware Interfaces | 8 |
| 2.1.4. Software Interfaces | 8 |
| 2.1.5. Communications Interfaces..... | 8 |
| 2.1.6. Memory Constraints | 8 |
| 2.1.7. Operations | 9 |
| 2.1.7.1. System administrator..... | 9 |
| 2.1.7.2. User..... | 9 |
| 2.2. Product Functions..... | 10 |
| 2.2.1. Register | 10 |
| 2.2.2. Search books | 10 |
| 2.2.3. Provide foreign book translation | 11 |
| 2.2.4. Take notes on books | 11 |
| 2.2.5. Share informations with friends | 11 |
| 2.2.6. Profile | 11 |
| 2.2.7. Review Analysis | 11 |
| 2.3. User Characteristics | 12 |
| 2.3.1. System Administrator | 12 |
| 2.3.2. User | 12 |
| 2.3.3. Vendor | 12 |
| 2.4. Constraints | 13 |
| 2.5. Assumptions and Dependencies..... | 13 |
| 3. Specific Requirements | 14 |
| 3.1. Functional Requirements | 14 |

| | |
|---|----|
| 3.1.1. Use Case..... | 14 |
| 3.1.2. Use Case Diagram | 20 |
| 3.1.3. Data Dictionary..... | 20 |
| 3.1.4. Data Flow Diagram | 23 |
| 3.2. Performance Requirements | 24 |
| 3.2.1. Static numerical requirement..... | 24 |
| 3.2.2. Dynamic numerical requirement | 24 |
| 3.3. Logical Database Requirements..... | 24 |
| 3.4. Design Constraints | 24 |
| 3.5. Standards compliance..... | 25 |
| 3.6. Software System Characteristics | 25 |
| 3.6.1. Product Requirements | 25 |
| 3.6.2. Organizational Requirements | 26 |
| 3.6.3. External Requirements | 27 |
| 3.7. Organizing the Specific Requirements | 27 |
| 3.7.1. Context Model..... | 28 |
| 3.7.2. Process Model | 29 |
| 3.7.3. Behavior Model..... | 29 |
| 3.8. System Architecture | 31 |
| 3.9. System Evolution..... | 31 |
| 3.9.1. Limitation and Assumption | 31 |
| 3.9.2. Evolutions of Hardware and Change of User Requirements | 32 |
| 4. Supporting Information | 32 |
| 4.1. Software Requirement Specification..... | 32 |
| 4.2. Document History | 32 |

LIST OF FIGURES

| | |
|------------------------------------|----|
| FIGURE 1 USE CASE DIAGRAM | 20 |
| FIGURE 2 DATA FLOW DIAGRAM | 23 |
| FIGURE 3 CONTEXT MODEL | 28 |
| FIGURE 4 PROCESS MODEL | 29 |
| FIGURE 5 SEQUENCE DIAGRAM | 30 |
| FIGURE 6 SYSTEM ARCHITECTURE | 31 |

LIST OF TABLES

| | |
|---|----|
| TABLE 1 ACRONYMS & ABBREVIATIONS..... | 6 |
| TABLE 2 TECHNICAL TERMS..... | 6 |
| TABLE 3 USE CASE - REGISTER..... | 14 |
| TABLE 4 USE CASE - LOG-IN/OUT | 15 |
| TABLE 5 USE CASE - SEARCH BOOKS..... | 16 |
| TABLE 6 USE CASE - BOOK TRANSLATION..... | 17 |
| TABLE 7 USE CASE - TAKE NOTES | 18 |
| TABLE 8 USE CASE - SHARE INFORMATION / BOOK MEETING | 19 |
| TABLE 9 PROFILE DATA | 20 |
| TABLE 10 BOOK DATA..... | 21 |
| TABLE 11 GLOBAL NOTE DATA..... | 21 |
| TABLE 12 LOCAL NOTE DATA | 22 |
| TABLE 13 DOCUMENT HISTORY..... | 33 |

1. Introduction

1.1. Purpose

This document is a Software Requirements Specification for providing METAVERSE LIBRARY SYSTEM services. This service is designed and implemented by Group 11 of the Introduction to Software Engineering at Sungkyunkwan University. The requirements for this are summarized, analyzed, and the system is designed and implemented based on the contents described.

1.2. Scope

The METAVERSE LIBRARY SYSTEM is meant to provide a method for students to virtually experience and use the facilities that libraries provide. The system provides great accessibility to a library as there are limitations that might not be able to be handled, such as geographical distance and compulsory social distancing requirements due to COVID-19 pandemic in real life. We can save architectural cost while serving the same purpose as a real-life library with unlimited extension of space.

1.3. Definitions, Acronyms, and Abbreviation

The following table explains the acronyms and abbreviations used in this document.

| Acronyms & Abbreviations | Explanation |
|--------------------------|-----------------------------------|
| API | Application Programming Interface |
| DB | Data Base |
| RAM | Random Access Memory |
| SSN | Social Security Number |
| UI | User Interface |
| HTTP | Hypertext Transfer Protocol |
| GPU | Graphics Processing Unit |

Table 1 Acronyms & Abbreviations

The following table defines certain technical terms used in this document.

| Terms | Definitions |
|----------------------|---|
| User | Someone who uses a system |
| Client (user device) | A user device/user that connected to server |
| Software | The programs and other operating information used by a computer |

Table 2 technical terms

1.4. References

- Team 1. "Software Requirement Specification". SKKU, Last Modified: Jun. 15, 2020. Accessed: Oct 24 2021, https://github.com/skkuse/2020spring_41class_team1/blob/master/docs/SRS_TEAM1.pdf
- VRCPrefabs, "VRCPrefabs", 2021, Accessed: Oct 27 2021, <https://vrcprefabs.com/>
- iVRy, Steam Community, "How to use VRChat with a smartphone and Google Cardboard viewer", Last Modified: Jan 14 2018, Accessed: Oct 27 2021, <https://steamcommunity.com/app/438100/discussions/0/2595630410184136626/>

1.5. Overview

The rest of this software requirement statement contains three chapters and appendices. The second chapter (Overall Description) provides an overall description of the product perspective, including multiple interfaces, system functions, and system interactions with other systems. This chapter also introduces interactions with various types of stakeholders and systems. This chapter also describes item details, system constraints, and product assumptions and dependencies. Chapter 3 (Specific Requirements) provides detailed requirements, including descriptions of various system interfaces and software system characteristics. Various specification techniques are used to more accurately specify the requirements for various users. Chapter 4 (Supporting Information) mainly deals with the priorities of requirements. This includes a support document for the METAVERSE LIBRARY SYSTEM and a timeline for this Software Requirement Specification document. All members contributed equally to the production of this project.

2. Overall Description

2.1. Product Perspective

This product is designed for those who do not understand technical words. This system will allow the library to be used remotely and even communicate with people. The system uses a database of users and libraries.

2.1.1. System Interfaces

The user's personal information is stored in a database through SQLite. Information and contents of the book are received through API calls.

2.1.2. User Interfaces

An interface is provided through the screen of the computer or laptop, and information can be input through a simple click of the computer and a keyboard.

The administrator can access the Laptop database.

2.1.3. Hardware Interfaces

Minimum hardware requirements are an Intel i5-4950 processor with 4GB of memory, NVIDIA GeForce GTX 970 graphics card and at least 1GB of storage space.

2.1.4. Software Interfaces

This system is supported from at least Windows 8.1, and Windows 10 is recommended. And DirectX ver.11 is also required.

2.1.5. Communications Interfaces

Users communicate with HTTP protocol.

2.1.6. Memory Constraints

This system requires at least 4GB of RAM and 1GB of free storage space to run.

2.1.7. Operations

2.1.7.1. System administrator

- Login
 - System administrator can login by his/her own access point with an administrator account.
- Manage System
 - Monitoring server robustness and occurrence of process failure.
- Manage DB
 - System administrators can manage user data such as account key, profile, character in secure.
 - System administrator can manage notes data of each book.

2.1.7.2. User

- Login
 - Users can login using google accounts, skku accounts, etc.
- Register
 - Users can register using google accounts, skku accounts, etc.
 - When the user registers into the system, the user enters some information like ID, name, age, gender, etc.
- Search books
 - Users can search and get the book that they want.
- Provide foreign book translation
 - Users can get a translation of the book in real-time.
- Take notes on books
 - Users can write some notes on the e-book and save it as global or local save.
- Share information with friends
 - Users can communicate with other users using this function in the metaverse space.

- They can share their opinion about the books.
- They can toggle the share screen which has other user's opinions or notes
- Profile
 - Users can edit their personal information or characters.

2.2. Product Functions

2.2.1. Register

Users must log in to the system to use the function to modify and save books. If the user has already registered, the user can log in by entering ID and password. Otherwise, the user can register to the library system after entering the essential information.

Users can also click the "forgot ID" or "forgot password" button in cases where they already registered for the system but forgot their login details.

2.2.2. Search books

Users can search for books in the library space and find the books they want. They can also find the books you want while moving within the metaverse library space. Users can search for the desired book through the title, author, publisher, etc. of the book.

2.2.3. Provide foreign book translation

Users can log in to Sungkyunkwan University library to view files that can be accessed for free. Translate the application through the translation button included in the nav bar of the viewed file.

There is an interchange button for the translation, where the user can click and the translated file will be changed again to its original language.

2.2.4. Take notes on books

Users are able to pick their selections of books and are provided with the functionality of taking notes on the book itself and saving their progress for future applications.

The notes taken can be added, edited or deleted later on after the progress is saved. This is to ensure users can always update their progress in their notes and refer to the notes back anytime at their convenience.

2.2.5. Share information with friends

Various information, for instance, the progress of notes taken by the user on the selected books, can also be shared by other multiple users that are linked/related to the user's profile.

2.2.6. Profile

Users will be able to update their personal information and provide a bio description on their profile. Users can also choose details that will be visible towards other users of the system.

2.2.7. Review Analysis

Users will be able to review what the activities that they have done and any adjustments that have been made. This will also allow them to go back to the changes that they made in case they cannot trace back where they have been to in the system by themselves.

2.3. User Characteristics

2.3.1. System Administrator

System Administrator generally requires a sufficient understanding of the system. The system referred to here means the overall Metaverse Library System. They can access the system, correct errors that occur, and are responsible for maintenance and improvement. It requires server management experience to manage user databases and proficiency in VRchat and Unity for 3D modeling, and should be aware of the actual situation of Sungkyunkwan University's library.

2.3.2. User

User generally refers to a member of Sungkyunkwan University who can log in to Sungkyunkwan University Library (url "<https://lib.skku.edu/#/>"). Users may want to read, rent, and share contents of books in the library. Users include Korean students who have difficulty reading major books written in English, and international students who are poor at Korean yet who want to read books written in Korean.

2.3.3. Vendor

It is assumed that Vendor is an academic information team and librarians of Sungkyunkwan library who supports smart space planning and facility operation, information device operation planning, library data use support, electronic data/continuous publication selection and use.

2.4. Constraints

In developing the Metaverse library system, the following matters should be considered. Constraints are considered in a number of ways, including policy aspects, equipment aspects, and market research aspects.

- Cost and reliability must be considered in using the Open source API
- Copyright aspects should be considered in the use of e-book APIs
- The cost of system maintenance must be considered
- The service should be provided only to users who have access to Sungkyunkwan University's library service
- Natural 3D avatar modeling that is less heterogeneous is needed
- It should be compatible with the Windows 10 operating system, the basic operating system of library computers
- The maximum number of concurrent users of the metaverse system should be considered
- Detailed specification work is required to minimize the cost of system improvement
- It should be easy for users to use and be able to correct their own wrong choices

2.5. Assumptions and Dependencies

All systems in this document are written on the assumption that they are designed and implemented based on Vrchat. Therefore, all contents are written based on the Vrchat with Unity 2019.4.30f1 and may not be applied to other operating systems or versions.

3. Specific Requirements

3.1. Functional Requirements

3.1.1. Use Case

| Use case name | Register |
|----------------|--|
| Actor | Unregistered user |
| Description | It is the process in which the user registers information in the system for the first time to use the library's service. |
| Normal Course | <ol style="list-style-type: none">1. A user registers to the system for using the library system.2. The user enters some information according to the registration form. The required information includes:<ol style="list-style-type: none">a. E-mail addressb. Namec. Student IDd. Gendere. ... etc3. Authenticate whether the information is correct through the email address entered by the user.4. After authentication is completed, user information may be registered in the system by pressing the registration button. |
| Precondition | The user is not registered to the system. There is no the same email address in the system DB. |
| Post Condition | Personal information should be safely stored in the database. |
| Assumptions | N/A |

Table 3 Use Case - Register

| Use case name | Log-in/out |
|----------------|--|
| Actor | Registered user |
| Description | <p><Log-in></p> <p>This is a process in which the registered user tries to get into the system to use the library services.</p> <p><Log-out></p> <p>This is a process when the logged-in user tries to get out of the system.</p> |
| Normal Course | <p><Log-in></p> <ol style="list-style-type: none">1. A user is already registered in the system.2. The user enters an ID and password.3. If the information is correct, the system allows the user to get into the system and the user is now able to take advantage of all the services provided by the system <p><Log-out></p> <ol style="list-style-type: none">1. If the user wants to get out of the system, the user can click the 'Log-out' button.2. The system can arbitrarily close the session for the user who closed the metaverse library system without log-out. |
| Precondition | <p><Log-in></p> <p>The user should be registered in the system as a member and not logged-in status.</p> <p><Log-out></p> <p>The user should be logged in to the system.</p> |
| Post Condition | N/A |
| Assumptions | N/A |

Table 4 Use Case - Log-in/out

| Use case name | Search Books |
|----------------|--|
| Actor | Registered user |
| Description | When a user enters the library metaverse space, there is a search bar where he can find the book he wants and a space where books can be freely selected. Here, you can read the book you want, use the translation function, and take notes. |
| Normal Course | <ol style="list-style-type: none">1. After logging-in, the user is in the metaverse library.2. <ol style="list-style-type: none">a) If there is a book that the user wants, search for the book at the search bar to obtain location information.b) If you want to look around and choose books for no fixed purpose, look around in the zone where the book is plugged in and choose the book.3. After obtaining the location of the book, go to the shelf at the location and select the book.4. The user can click on the book and read it. |
| Precondition | The user should be in a logged-in status and connected to network |
| Post Condition | N/A |
| Assumptions | N/A |

Table 5 Use Case - Search books

| Use case name | Book Translation |
|----------------|--|
| Actor | Registered user |
| Description | Translate the contents of the original book into the desired language. |
| Normal Course | <ol style="list-style-type: none">1. A user has some difficulty understanding the original book or wants to read the book in another language.2. Press the translation button in the upper right corner of the book.3. Select the part of the book you want to translate and the language you want, and press the Translate button.4. The user can read a book in two versions, the original and the translation. |
| Precondition | The user should be in a logged-in status and connected to the network. The user is reading a book. |
| Post Condition | N/A |
| Assumptions | N/A |

Table 6 Use Case - Book translation

| Use case name | Take notes on books |
|----------------|--|
| Actor | Registered user |
| Description | It is a function that allows you to take notes on books in the library. The original book must not be damaged because the book is read with others. Therefore, when saving after writing, the original book remains and a new written book is created. |
| Normal Course | <ol style="list-style-type: none">1. The user picks up the desired book and reads it.2. The user has found a phrase they like and wants to underline it and highlight it.3. Click the Edit button in the upper right corner.4. The edit button is expanded and various edit buttons appear.5. Click the button to underline and use the mouse to underline the desired phrase.6. Click the Save button.7. A selection button appears to select whether to save as local or global. |
| Precondition | Initially, there was only one original book. |
| Post Condition | If the user chooses to save locally, the newly saved book is only visible to the user. If you save it globally, the newly created book will be visible to everyone. |
| Assumptions | N/A |

Table 7 Use Case - Take notes

| Use case name | Share information with friend / Book Meeting |
|----------------|--|
| Actor | Registered users |
| Description | It is a service that allows users to share information about books and have conversations with each other. |
| Normal Course | <ol style="list-style-type: none">1. Users gather in a meeting room about a certain book.2. Gathered users have a conversation about book-related content through a microphone.3. User A wants to share the notes in the book.4. Change A's handwriting book local to global so that other people can see it.5. Read what other users have written and continue the conversation about it. |
| Precondition | N/A |
| Post Condition | N/A |
| Assumptions | Users must have a microphone. |

Table 8 Use Case - Share information / Book Meeting

3.1.2. Use Case Diagram

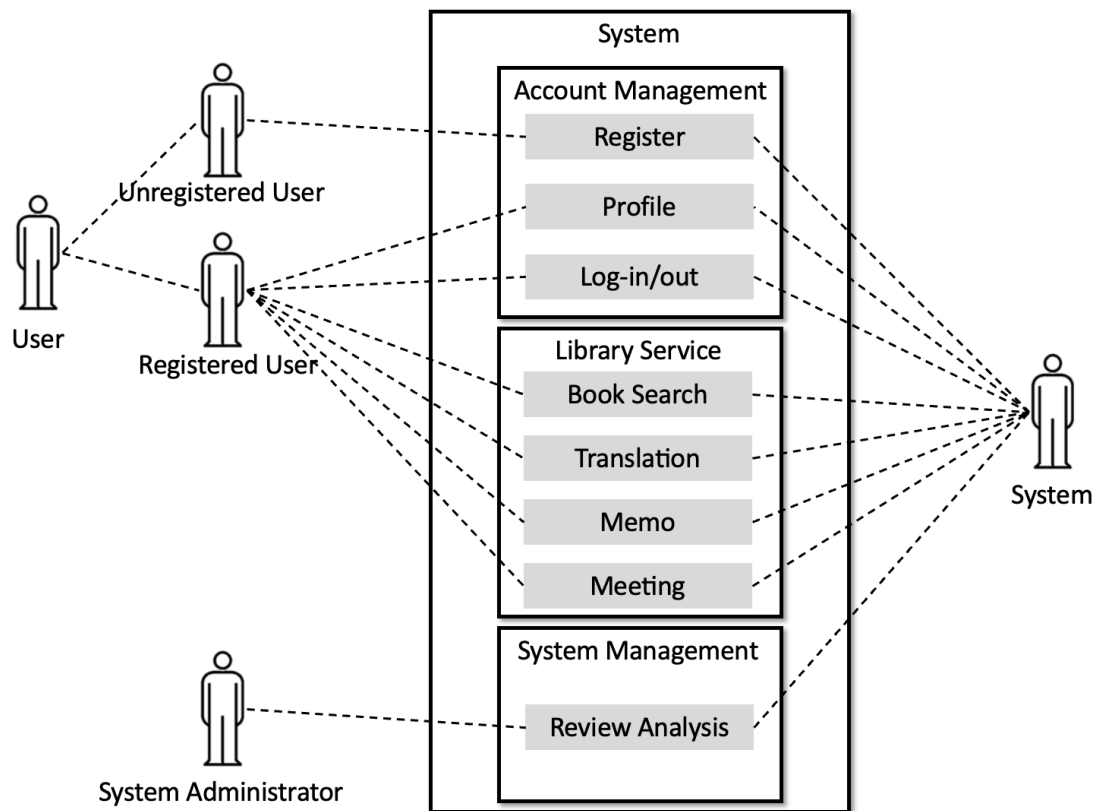


Figure 1 Use Case Diagram

3.1.3. Data Dictionary

| Field | Key | Constraint | Description |
|--------|-----|------------|----------------|
| ID | PK | Not Null | User id(email) |
| Name | | Not Null | User name |
| Age | | | User's age |
| Gender | | | User's gender |

Table 9 Profile Data

| Field | Key | Constraint | Description |
|---------|-----|------------|---|
| Index | PK | Not Null | Book id(hash number) |
| Title | | Not Null | Book information: Title |
| Author | | Not Null | Book information: Author |
| Tag | | | Book information: Category, keyword or tag name |
| Content | | | List of Book's content divided into pages |

Table 10 Book Data

| Field | Key | Constraint | Description |
|--------------|-----|------------|--|
| ID | PK | Not Null | Note id(hash number, not Local Note Data's ID) |
| User profile | | | Ref. Profile Data |
| Note content | | Not Null | Ref. Local Note Data |

Table 11 Global Note Data

| Field | Key | Constraint | Description |
|----------|-----|------------|--------------------------------------|
| ID | PK | Not Null | Local note id |
| Book | | Not Null | Ref. Book Data |
| Page | | Not Null | Page number of selected books |
| Position | | Not Null | Position in page where to write note |
| Content | | | Content of note |

Table 12 Local Note Data

3.1.4. Data Flow Diagram

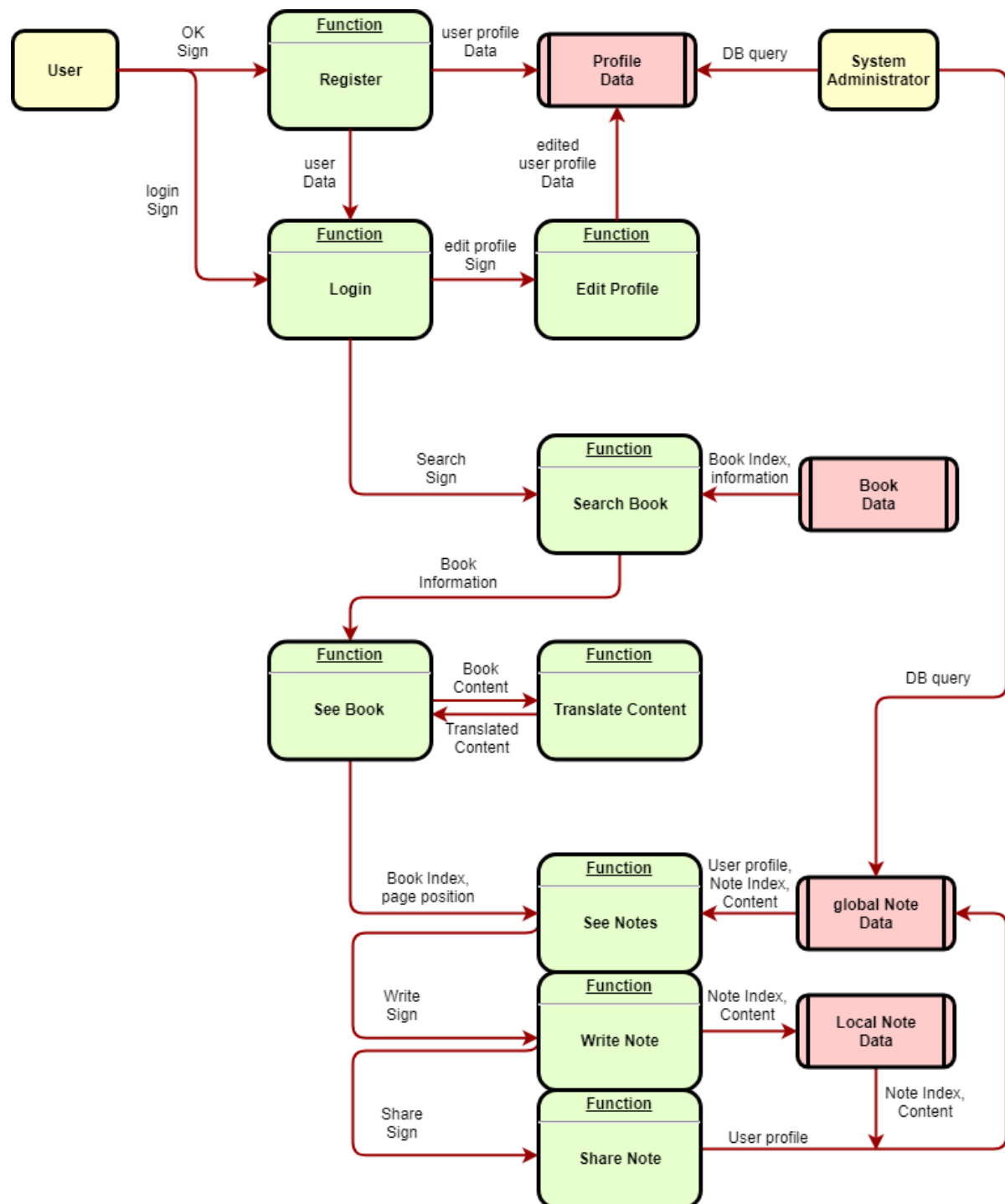


Figure 2 Data flow Diagram

3.2. Performance Requirements

The following requirements are based on estimates and may be changed in the final application.

3.2.1. Static numerical requirement

In this system, only one user can be registered per account. However, this does not prevent a single user from using multiple accounts.

This system allows multiple people to read one book. However, viewing multiple books at once is not supported by the system.

The system requires a minimum of 4 GB of memory and 1 GB of free storage space for storage and execution.

3.2.2. Dynamic numerical requirement

The system can be up and running in 10 seconds.

Searching for a book will result in 3 seconds.

The login process will take 5 seconds.

If more than 100 users are gathered at once, the system may become overloaded.

A computer is recommended to implement the above dynamic numerical requirements.

3.3. Logical Database Requirements

The system uses a community called VCRPrefabs that runs a database that collects and organizes prefabs, tools, and assets for the platform that will be used, VRChat. It runs on both 2.0 and 3.0 SDKs and Udon, and it also contains hundreds of assets.

3.4. Design Constraints

The system uses VRChat, an online virtual world platform that allows users to interact with each other in the user-created 3D world. The system should be able to access from various m

obile devices with Android operating system and Apple operating system, and the administrator must be able to access and manage it through a web browser.

3.5. Standards compliance

This system used Unity 2019.4.30f1 and VRChat 2021.3.4p2 as standard. Unity programming is done in C++, and the names of functions and variables have camel notation and underscore notation applied.

3.6. Software System Characteristics

Product requirements specify or constrain the runtime behavior of the software. Our system should satisfy the following requirements.

3.6.1. Product Requirements

3.6.1.1. Usability Requirements

Students of SKKU are able to access the system without any difficulty and are provided with just the necessary functions to select the books and utilize the provided features of the library system.

3.6.1.2. Performance Requirements

The Metaverse Library system should always respond well towards the user's actions. This means that it is able to react quickly and ensure that large resources are able to be processed without hindering the system. It is also essential to keep the system stable, preventing vulnerabilities such as system crashes.

3.6.1.3. Reliability Requirements

The system is capable of performing its intended function under numerous circumstances without error. Information provided by the system should always be accurate and consistent.

3.6.1.4. Security Requirements

Security requirements will follow three principles; confidentiality, integrity and availability. Personal information of the user will have to be thoroughly protected, unauthorized changes should be monitored to prevent inaccurate information and the user has reliable access whe

re information can be accessed anywhere at any time of the day, provided that they have rights to access them.

3.6.2. Organizational Requirements

These requirements are broad system requirements derived from policies and procedures in the customer's and developer's organizations

3.6.2.1. Environmental Requirements

Book data in this system uses those whose copyrights have expired or are open online. So, how efficiently you collect the book data is key to this system.

3.6.2.2. Operational Requirement

Users are identified by ID or email. This system is more friendly to computers than VR devices which is still uncomfortable. This will provide a search function that will quickly show you if the book you are looking for is available. The system will present the book to the user in an ebook-like format.

3.6.3. External Requirements

It deals with external factors of the system and all requirements derived from the development process.

3.6.3.1. Safety / Security Requirement

The system should guarantee that any personal information is not accessed by an external system. The system should be safe enough to prevent user's data from being damaged from external natural disasters.

3.6.3.2. Regulatory Requirement

'Google Translation API' is not completely free. User's personal privacy should not be violated according to law. The system should be developed in accordance with a national privacy standard.

3.7. Organizing the Specific Requirements

In this section, we describe the system model using graphical notation. System model describes the relationship among the system, components, and surrounding environments, showing more specific requirements

3.7.1. Context Model

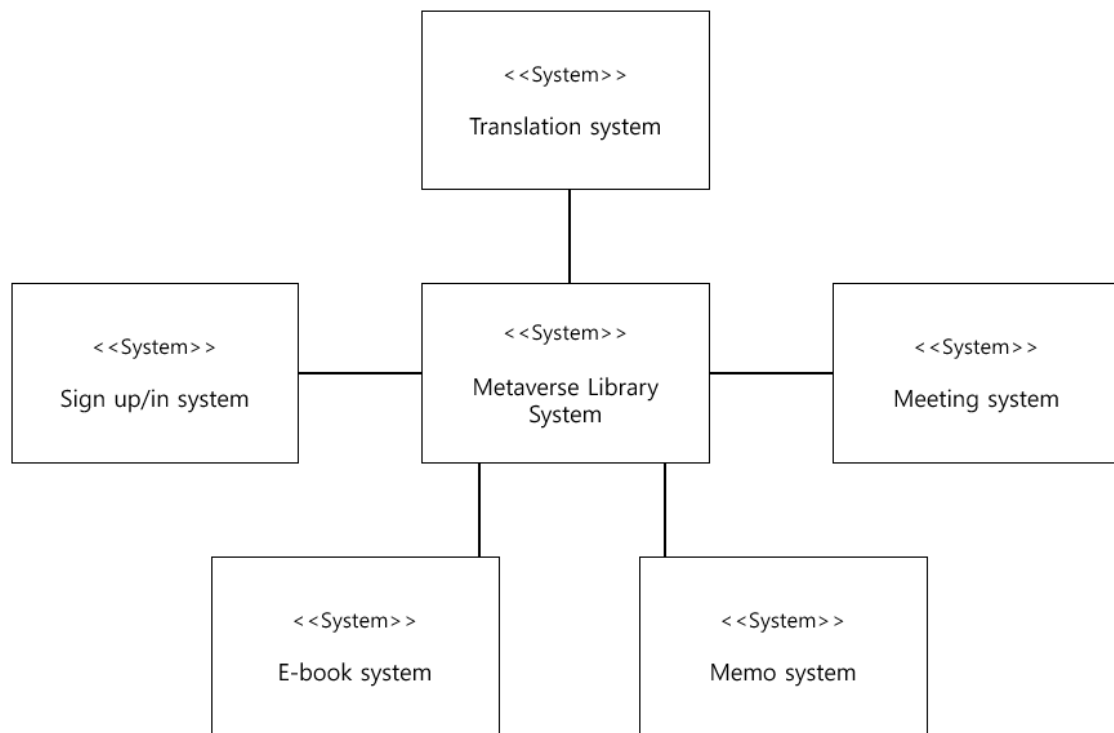


Figure 3 Context Model

3.7.2. Process Model

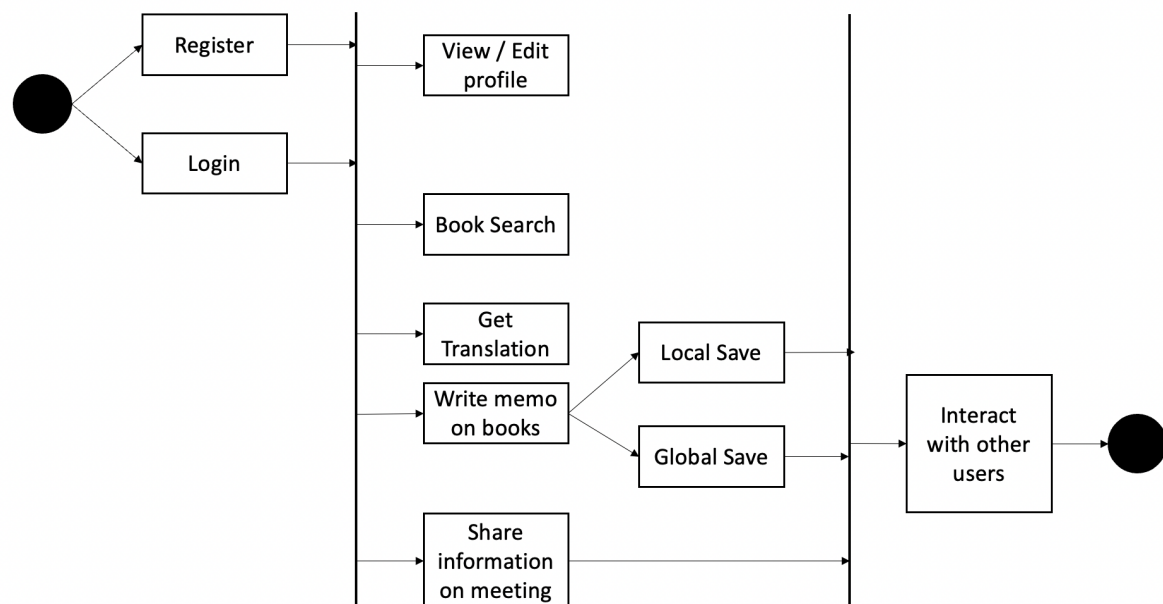


Figure 4 Process Model

3.7.3. Behavior Model

3.7.3.1. Data Flow Diagram

See 3.1.4. Data Flow Diagram

3.7.3.2. Sequence Diagram

It describes 5 main sequences of processes in our system; which are searching books, translating content of selected books, showing all notes written by other users, and also writing notes and then sharing notes to other users.

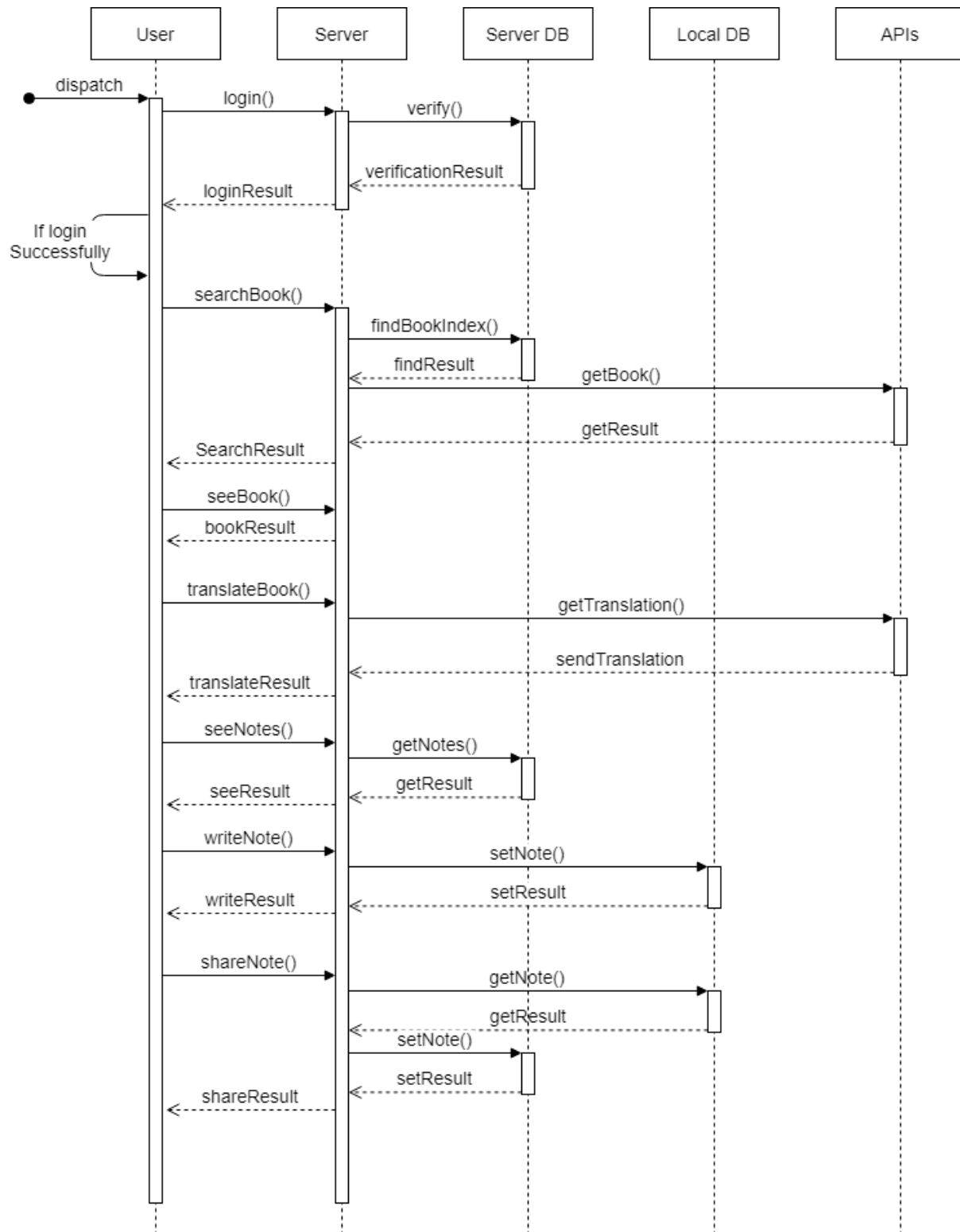


Figure 5 Sequence Diagram

3.8. System Architecture

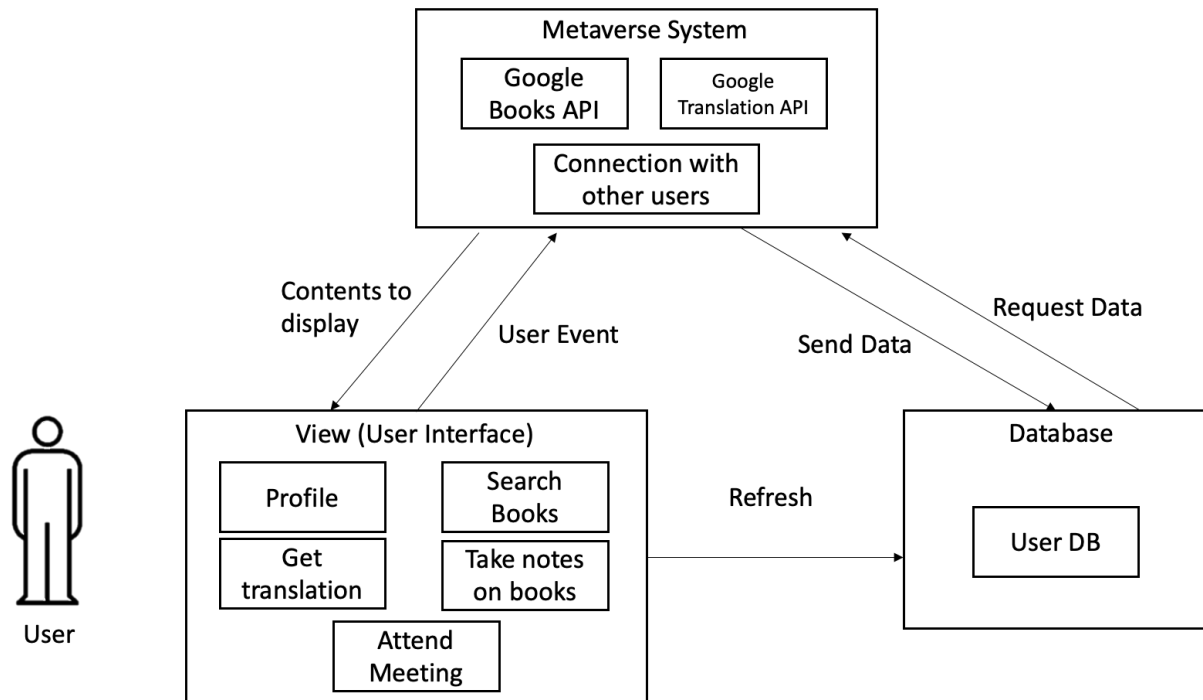


Figure 6 System Architecture

3.9. System Evolution

This part describes the basic assumptions on which the system is based and the changes expected due to hardware evolution and changing user needs. This part will help system designers to avoid system-limited changes in the future.

3.9.1. Limitation and Assumption

Currently, due to the inconvenience of VR devices, it is expected that the system will be run more in a PC environment than in a VR environment. So this system is intended to be built on the assumption that it will work on a computer. In fact, vrchat can run on various vr devices including Oculus. There is a difference between the environment running on a computer and the environment running on a VR device such as Oculus. Since this system was created wi

th the expectation that users will use it on their computer, unexpected errors may occur when using the system on a VR device.

3.9.2. Evolutions of Hardware and Change of User Requirements

The system expansion from computers to VR devices, which is the above-mentioned constraint, is the first expected development. Second, the user makes additional system requests. What is typically expected is to update the type or number of supported books. This can be solved through periodic PDF file updates. In addition, functional requirements may be added, which should be added with care not to conflict with the existing system.

4. Supporting Information

4.1. Software Requirement Specification

This software requirements specification was written in accordance with the IEEE Recommendation (IEEE Recommended Practice for Software Requirements Specifications, IEEE-Std-830).

4.2. Document History

| Date | Version | Description | Writer |
|------------|---------|--|------------------|
| 2021/10/29 | 0.1 | Style and overview | Youngsuh Chin |
| 2021/10/30 | 1.0 | Addition of 1.1, 1.2, 1.3, 1.5, 2.1, 2.1.2 | Youngsuh Chin |
| 2021/10/30 | 1.1 | Addition of 2.2.3, 2.3.1, 2.3.2, 2.3.3 | Jaehyuk Choi |
| 2021/10/30 | 1.2 | Addition of 2.2.4, 2.2.5 | Angel Fitri Sari |

| | | | |
|------------|-----|---|------------------|
| 2021/10/30 | 1.3 | Addition of 2.2.1, 2.2.2 | Dajung Choi |
| 2021/10/30 | 1.4 | Modify Contents, Addition of 3.1.1, 3.1.2, 3.6.3 | Youngsuh Chin |
| 2021/10/30 | 1.5 | Addition of 2.3.4 | Jaehyuk Choi |
| 2021/10/30 | 1.6 | Addition of 3.6.1, 2.2.6 | Angel Fitri Sari |
| 2021/10/30 | 1.7 | Addition of 3.1.1, 3.7.1, 1.3, 3.1.3 | Dajung Choi |
| 2021/10/30 | 1.8 | Addition 2.1.3. 2.1.4 2.1.5 3.9 3.9.1 | Wonjae Lee |
| 2021/10/31 | 1.9 | Addition 2.2.1, 2.2.3, 2.2.4, 2.2.7 | Amirah |
| 2021/10/31 | 2.0 | Addition of 1.4, 2.1.1, 2.1.7.2, 2.5, 3.7, 3.7.2 | Youngsuh Chin |
| 2021/10/31 | 2.1 | Addition 3.9.2 3.1.1 3.6.2 3.2 3.5 | Wonjae Lee |
| 2021/10/31 | 2.2 | Addition of 2.1.7.1, 2.1.7.2, 3.1.4, 3.7.3.1 Revision of 3.1.2, 3.1.3. | Changmin Park |
| 2021/10/31 | 2.3 | Addition 3.4 | Amirah |
| 2021/10/31 | 2.4 | Addition 3.7.3.2 Revision of 1.4 | Changmin Park |
| 2021/10/31 | 2.5 | Addition of 3.8 | Youngsuh Chin |
| 2021/10/31 | 2.6 | Addition 3.3 | Amirah |

Table 13 Document History