Hall for Graduation Exhibition

Software Requirement Specification

2021.10.31.

Team 9

Jihee Kim

Bomin Namkoong

Jaehyeong Park

Jongin Park

Jaekwang Shin

Seokjun Chung

Gyumin Han

CONTENTS

LIST OF TABLES

LIST OF FIGURES

1. Introduction

1.1. Purpose

This document is a Software Requirements Specification (SRS) for providing Metaverse hall for graduation exhibition service. This service is designed and implemented by Team 9 of the Introduction to Software Engineering on Sungkyunkwan University. The requirements for this service are summarized, analyzed, and the system is designed and implemented based on the contents described after.

In this document, Team 9 is the main reader, and Team 9 designs and implements the functions of the metaverse graduation exhibition hall service according to this specification. Additionally, professors, TAs, and team members in the Introduction to Software Engineering class can be the main readers.

The purpose of this document is to outline and publish the Requirement Specification for a new metaverse hall service especially for graduation exhibition. Unlike other metaverse rooms on VRChat, the service provides special space for graduation exhibition, and many visitors can view exhibits even they are on another space, 24 hours. Guests can appreciate various works, share opinions, and ask questions.

1.2. Scope

This metaverse hall for graduation exhibition service is meant to ease the come and go to graduation exhibition regardless of time and space. Users can come to look graduation exhibits more easily. The system is based on VRChat platform, developed by Unity 2019.4.30f1 version. Because VRChat doesn't support external database sync, every data will be managed in VRChat platform. Above all, our most important goal is to make comfortable user experience along with the trusted operation of graduation exhibition. Even if it is an online graduation exhibition, guests will feel that they actually communicate with people in the space and feel comfortable with it.

1.3. Definitions, Acronyms, and Abbreviation

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

[Table 1] Table of acronyms and abbreviations

Acronym& Abbreviation	Explanation
GNU	GNU's Not Unix
GUI	Graphic User Interface
OS	Operating System
PC	Personal Computer
RAM	Random-Access Memory
SDK	Software Development Kit
URL	Uniform Resource Locator
VR	Virtual Reality

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

[Table 2] Table of terms and definitions

Term	Definition
User	Someone who uses a system
System administrator	Someone who manage the world
Server	A computer or computer program which manages access to a centralized resource or service in a network
Software	The programs and other operating information used by a computer
Open-source	A code that is designed to be publicly accessible
VRChat	An online virtual world platform
Widget	Little applications which can be placed on a widget host

1.4. References

- IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications, In IEEEXplore Digital Library
  http://ieeexplore.ieee.org/Xplore/guesthome.jsp

- Team 1. "Software Requirement Specification". SKKU, Last Modified: May. 14, 2020.
  https://github.com/skkuse/2020spring_41class_team1/blob/master/docs/SRS_TEAM1.pdf

1.5. Overview

The remainder of this document is composed of three chapters and appendixes. On second chapter, overall description of service on product perspective, product functions, user characteristics and constraints with important assumptions and dependencies of service. On third chapter, the specific requirements are described in detail. It handles external interface, functional, performance requirements, design constraints and standards compliance. The last chapter provides supporting information of this document. It describes how this document is written and history of the document. All members in team 9 equally contributed the production of this project.

2. Overall Description

2.1. Product Perspective

This product is designed for three kinds of people. First is people who want to register one's graduation work for exhibition, second is the people who want to watch graduation exhibition and leave comment on exhibits, the last is professors who want to promote their labs. This service will make people more comfortable to participate on graduation exhibition regardless of time and space. This service will run on VRChat platform, so that participants on exhibition can interact each other, watch exhibit together, or comment on it.

2.1.1. System Interfaces

The number of 'Like's, count of views, user's exhibit information and comment histories are stored on VRChat platform.

2.1.2. User Interfaces

An interface is provided through basic GUI of VRChat platform, and information can be input through user's keyboard typing. Depending on the information entered by user, they can upload their graduation exhibit, or leave comment.

2.1.3. Hardware Interfaces

The system is intended for PC that can run VRChat. The PC must have at least 4GB RAM and NVIDIA GeForce® GTX 970/AMD Radeon™ R9 290 Graphic card. Both processor and OS should support 64-bit.

2.1.4. Software Interfaces

The system is intended for Windows OS version at least Windows 7 and targeting Windows 10.

2.1.5. Memory Constraints

The system should run on PC with least 4GB RAM for primary memory.

2.2. Product Functions

2.2.1. Input Processing Using Keyboard and Mouse

Users can transmit their instructions through keyboard & mouse.

2.2.2. Poster Exhibition

Users can see the poster of the participant's graduation work and magnify the poster by clicking the poster.

2.2.3. Playing Demo Video

Users can watch the demo video of the work by clicking the play button displayed on the poster while the poster is magnified.

2.2.4. Shifting Between Rooms

Users can move to another space through the portal. This portal allows the user more convenient way to move to the other team's space and use services provided by them. After accessing the portal, the user can know the list of spaces that they can move from there through the portal click and can select the space the user wants to move.

2.2.5. Voting a Poster

It is a process in which users can vote for the work they want. Users can vote for their favorite posters and each ach user can vote only once. Accordingly, it is possible to know posters that are popular with users.

2.2.6. Searching Poster

It is a process in which users can search for the exhibition location of the work they want. Users can find the location of posters containing the information they want using a search bar in the main hall. Also, users can find the locations of posters corresponding to a specific category

through the search bar and enjoy posters searched by users directly from the search bar.

2.3. User Characteristics

2.3.1. System Administrator

System administrator is limited to those who those who has sufficient knowledge of the system and who has a general understanding of the system. It is assumed that system administrator has sufficient capabilities to deal with system problems. Also, it is assumed that system administrator has majored in computer science or similar studies, has completed training to become a network administrator or a system administrator, or has equivalent qualifications. In addition, it is assumed that the system administrator has ability to reflect the update of Unity and VRChat. Also, by analyzing the reviews, system administrator should be able to revise the system more user-friendly and comfortably.

2.3.2. User

Users generally referred to in this document are divided into authors and audience.

The author is the person who wrote the poster displayed at the exhibition, displays the poster and uploads a demo video. They should be free to use the functions provided by the system.

Visitors watch the displayed posters, play the uploaded video, and move around the space to enjoy the exhibition. After watching the exhibition, they are going to be vote for their favorite works. Since they do various activities within the system, an overall understanding of the system is needed.

2.4. Constraints

The system will be designed and developed based on the contents mentioned in this document. Other details are designed and developed by selecting the direction preferred by the developer, but the following items are observed.

- Use the technology that has already been widely proven.

- Users should be able to come and go hall anywhere VRChat is available, 24 hours.

- Use open source software whenever possible.
- Avoid using technology or software that requires a separate license or pays for royalty. (Exclude this provision if this is the only technology or software that the system must

require.)

- Decide in the direction of seeking improvement of overall system performance.
- Decide in a more user-friendly and convenient direction.
- Consider integration to other VRChat projects.
- Consider future scalability and availability of the system.
- Optimize the source code to prevent waste of system resources.
- Consider future maintenance and add sufficient comments when writing the source code.
- Develop on Windows 10 environment and Unity 2019.4.30f1 that is compatible with VRChat.

2.5. Assumptions and Dependencies

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

3. Specific Requirements

3.1. External Interface Requirements

3.1.1. User Interfaces

[Table 3] User Interface of input processing using keyboard & mouse

Name	Basic User Interaction Using Keyboard & Mouse
Purpose/Description	Users can transmit their instructions through keyboard & mouse.
Input source/ Output destination	User / VRChat Client
Range / Accuracy / Margin of error	Range according to the number of inputs by keyboard & mouse / Accuracy according to the accuracy of inputs from users / Margin of error of inputs by keyboard & mouse
Unit	A click or a type

Time/ Velocity	Asynchronous user input / Instant execution of a user instruction
Relationship with other input/outputs	Depending on the input, the client processes it or requests a command to the server.
Format and configuration of screen	N/A
Format and configuration of window	N/A
Data type	Int type value of a button code
Instruction type	Command mapping according to the button of Widget and input value.
Exit message	N/A

[Table 4] User Interface of poster exhibition

Name	Poster Exhibition
Purpose/Description	Users can see the poster of the participant's graduation work. Users can magnify the poster by clicking the poster.
Input source/ Output destination	Host server - Client
Range / Accuracy / Margin of error	Depends on the number of participants / N/A / N/A
Unit	A Poster
Time/ Velocity	After users click a poster / Communication time between the server and the user device
Relationship with	N/A

other input/outputs	
Format and configuration of screen	1. The posters of the participants are listed in a relatively small size in the room. 2. When users click on the poster, the poster is magnified and displayed.
Format and configuration of window	Poster is stored in advance in PDF form
Data type	Image, Text, Widget, PDF
Instruction type	Command mapping according to the button of Widget and input value.
Exit message	"Done with the Poster"
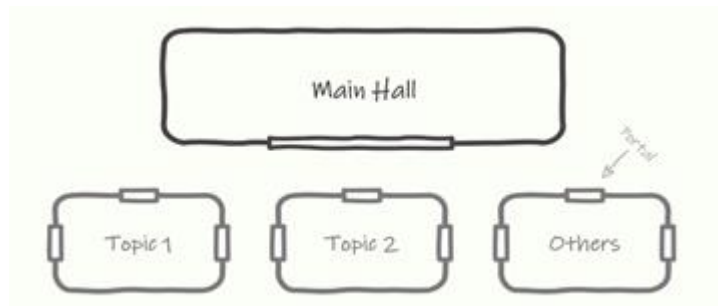
[Table 5] User Interface of playing demo video

Name	Playing demo video
Purpose/Description	Users can watch the demo video of the work by clicking the play button displayed on the poster while the poster is magnified.
Input source/ Output destination	Server - Client
Range / Accuracy / Margin of error	Depending on the existence of the demo video of participants / N/A / N/A
Unit	A demo video of a participant uploaded on YouTube.
Time/ Velocity	The time when the user clicks the play button displayed on the poster / Depends on the processing time of the user device
Relationship with	It shows a demo video of the graduation poster clicked by the user.

other input/outputs	
Format and configuration of screen	Play button to watch a video. Video player who plays demo videos.
Format and configuration of window	Image, Widget, Text
Data type	Video, Image, Widget
Instruction type	Command mapping according to the button of Widget and input value.
Exit message	"End video"

[Table 6] User Interface of shifting between rooms

Name	Shifting between rooms
Purpose/Description	Users can move to another space through the portal. This portal allows the user more convenient way to move to the other team's space and use services provided by them. After accessing the portal, the user can know the list of spaces that they can move from there through the portal click, and can select the space the user wants to move.
Input source/ Output destination	Server - Client
Range / Accuracy / Margin of error	Other spaces registered by the administrator./ N/A / N/A
Unit	Portal, Other space registered by the administrator.
Time/ Velocity	The time when the user clicks on the portal / Depends on the processing time of the user device

Relationship with other input/outputs	N/A
Format and configuration of screen	There is a portal in the room where you can move to another space. After accessing the portal, clicking on the portal appears a list of other movable spaces. You can choose another space you want to move.
Format and configuration of window	Image, Widget, Text
Data type	Image, Widget
Instruction type	Command mapping according to the button of Widget and input value.
Exit message	"Exit"

[Table 7] User Interface of voting a poster

Name	Voting a poster
Purpose/Description	Users can vote for their favorite posters. Each user can vote only once. Accordingly, it is possible to know posters that are popular with users.
Input source/ Output	Server - Client

destination	
Range / Accuracy / Margin of error	N/A./ N/A / N/A
Unit	A voting action by a user
Time/ Velocity	The time when the user votes on a poster / Communication time between the server and the user device
Relationship with other input/outputs	N/A
Format and configuration of screen	The user receives one unique ball when entering. Users can vote their favorite poster by putting the ball into ballot box in front of the poster. The ballot box shows how many users put the ball in the ballot box.
Format and configuration of window	Image, Widget, Text
Data type	Image, Widget
Instruction type	Command mapping according to the button of Widget and input value.
Exit message	N/A

[Table 8] User Interface of searching poster

Name	Searching posters
Purpose/Description	Users can find the location of posters containing the information they want using a search bar in the main hall. The user can find the locations of posters corresponding to a specific category through the search bar.

	Users can enjoy posters searched by users directly from the search bar.
Input source/ Output destination	Server - Client
Range / Accuracy / Margin of error	Posters registered by participants / Depends on searching algorithm / Margin of error set by searching algorithm
Unit	A search action by a user
Time/ Velocity	The time when the user searches a poster / Communication time between the server and the user device
Relationship with other input/outputs	N/A
Format and configuration of screen	A search bar where users can search what they want is located in the main hall. Users can search the posters that contains the information what they want by moving in front of the search bar and clicking it. The list and location of posters searched through the search bar are displayed by searching. Users can look at the poster by moving to the location indicated by search bar or they can also look at the poster directly by clicking the name of the poster in the list indicated by search bar.
Format and configuration of window	Image, Widget, Text
Data type	Image, Widget
Instruction type	Command mapping according to the button of Widget and input value.
Exit message	"Finish searching"

3.1.2. Hardware Interfaces

[Table 9] Hardware interface of applicable device for the system

Name	interface
Purpose/Description	Enable users to take advantage of the service provided by the system / PC that can run VRChat (at least 4GB RAM and NVDIA GeForce® GTX 970/AMD Radeon™ R9 290 Graphic card)

3.2. Functional Requirements

3.2.1. Use Case

[Table 10] Use case of poster exhibition

Use case name	Register
Actor	User
Description	Audience can see the poster of the participant's graduation work. Each item can be magnified by clicking the poster.
Normal Course	1. The user clicks a poster hang up on the wall 2. The poster is magnified so that the user can see well
Precondition	Posters are put up in advance on the wall of the world
Post Condition	N/A
Assumptions	All users turned in their graduate work posters in advance

[Table 11] Use case of shifting between rooms

Use case name	Register
Actor	User
Description	Users can move to other room by taking the portal.
Normal Course	1. The user takes the portal in the world 2. The user moves to other room which is already set by admin 3. By repeating 1~2, the user can get around the exhibition rooms
Precondition	The portal is created in advance and is directed to specific VRChat locations

Use case name	Register
Post Condition	N/A
Assumptions	The user decided to go to other room

[Table 12] Use case of playing demo video

Use case name	Register
Actor	User
Description	Users can watch the explanatory video about displayed works while the poster is magnified.
Normal Course	1. The user clicks the play button on the video 2. Video is displayed and the user listens to an explanatory video about the work
Precondition	Each URL is pasted into VRCUrlInputField, which exists as a child object of Prefab Demo video is uploaded on whitelisted hosts
Post Condition	N/A
Assumptions	All users' demo video is uploaded on supported video hosts

[Table 13] Use case of voting a poster

Use case name	Register
Actor	User
Description	It is a process in which users can vote for the work they want.
Normal Course	1. All users must be in the exhibition hall. 2. All users can vote only once for the work they want.
Precondition	N/A
Post Condition	N/A
Assumptions	N/A

[Table 14] Use case of searching poster

Use case name	Register
Actor	User
Description	It is a process in which users can search for the exhibition location of the work they want.

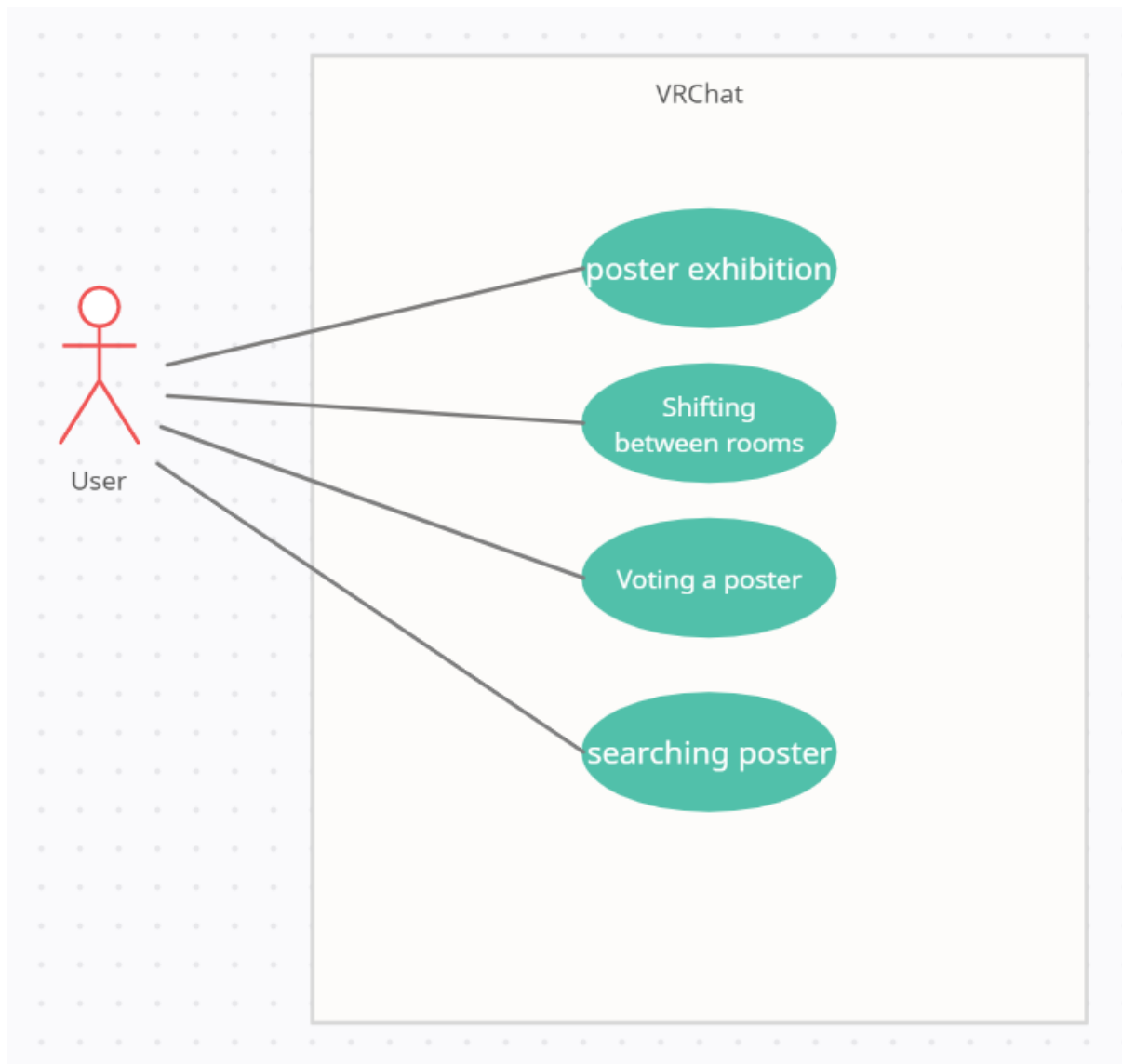| Use case name	Register
Normal Course	1. All users must be in the exhibition hall. 2. User can search for the information you want at the search bar in the main hall. 3. If the information exists, the desired search result will come out, and if not, it will not be found.
Precondition	N/A
Post Condition	N/A
Assumptions	N/A

3.2.2. Use Case Diagram

Figure 1 Usecase Diagram

3.3. Performance Requirements

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

3.3.1. Static numerical requirement

- The system management tool supports only one administrator.

- The VRCHAT supports simultaneous user for each PC. user can switch accounts and
  access.

- The system can run smoothly on PC with least 4GB RAM, 3.3 GHz 4 processors and NVDIA GeForce® GTX 970/AMD Radeon™ R9 290 Graphic card. Both processor and OS should support 64bit.

3.3.2. Dynamic numerical requirement

- Functions depend on the server status of VRChat.

3.4. Design Constraints

To avoid the strictness of GNU general public licenses, any components used in creating the world must be under MIT license. The world is designed with a specific SDK version (VRCSDK3), and migration between different SDK versions is not supported. The world runs on VRChat and can be accessed via VRChat platform, while the administrator must be able to manage and upload it through the online virtual platform.

3.5. Standards Compliance

All scripts in the world belong to whitelisted world components mentioned in VRChat documentation. Others are programmed by following the Unity standards and the VRChat SDK3. The world should be made public so that all users can access it by Community Labs. In addition, the system is expected to be optimized by complying with performance guidelines.

3.6. Software System Characteristics

This section describes several non-functional requirements of our system including system properties and constrains. The entire will be defined by dividing it into three requirements: product requirements, organic requirements, and external requirements.

3.6.1. Product Requirements

This section specifies or constrains the runtime behavior of delivered product.

3.6.1.1. Usability Requirements

The system should be easy to start even for first-time users and should be comfortable for users to use. In addition, it should be well-designed so that the user does not cause many errors. A manual of how to use the system should be provided in the first main hall and each room. Objects in the system must follow a simple, commonly used format so that users can easily recognize them.

3.6.1.2. Performance Requirements

See section 3.3. Performance Requirements.

3.6.1.3. Dependability Requirements

When one user causes an error, it should not affect the computer or server. When one user causes an error, it should not cause another user to not use the system. Also, errors should not cause damage to the computer itself. Only users with error are forced to shut down the system or shut it down on their own.

3.6.1.4. Security Requirements

The system should not allow other users to view user information. While using the system, the personal information of the person who provided the graduation work and visit records of users should not be visible without consent. Also, the users should not be able to obtain the original graduation work.

3.6.2. Organizational Requirements

This section describes several requirements about organizational policies and procedures. It includes both customer's and developer's organization.

3.6.2.1. Operational Requirements

The system provides the function of exhibiting graduation works. Users can view graduation posters and work description videos on the system. The system should always be available during the exhibition period.

3.6.2.2. Development Requirements

Use VRChat as a development platform. Develop a system using VRChat SDK in Unity version 2019.4.30f1. Testing of the system is also done through VRChat.

3.6.3. External Requirements

This section describes several requirements arise from factors which are external to the system and its development process.

3.6.3.1. Regulatory Requirements

The system must be developed in compliance with laws and regulations. When developing

each object in the system development process, several open source may be used. At this time, the copyright law must be observed for all open source used. The opensource code must be changed or copyrighted in accordance with the licenses of each open source. In addition, copyright law must be followed for works displayed. The user's consent must be obtained before displaying. After displaying the work, personal information about the work must be deleted in accordance with the copyright law.

3.6.3.2. Safety/security requirements

The system should not expose the user's information to the outside or provide it to an external institution without consent. Personal information about the owner of the displayed work and the record of the user's visit using the system should not be delivered to the outside.

3.7. Organizing the Specific Requirements

In this section, we describe the system model using graphical notation based on Unified Modeling Language (UML). System model describes the relationship among the system, sub-systems, components, and surrounding environments, showing more specific requirements.
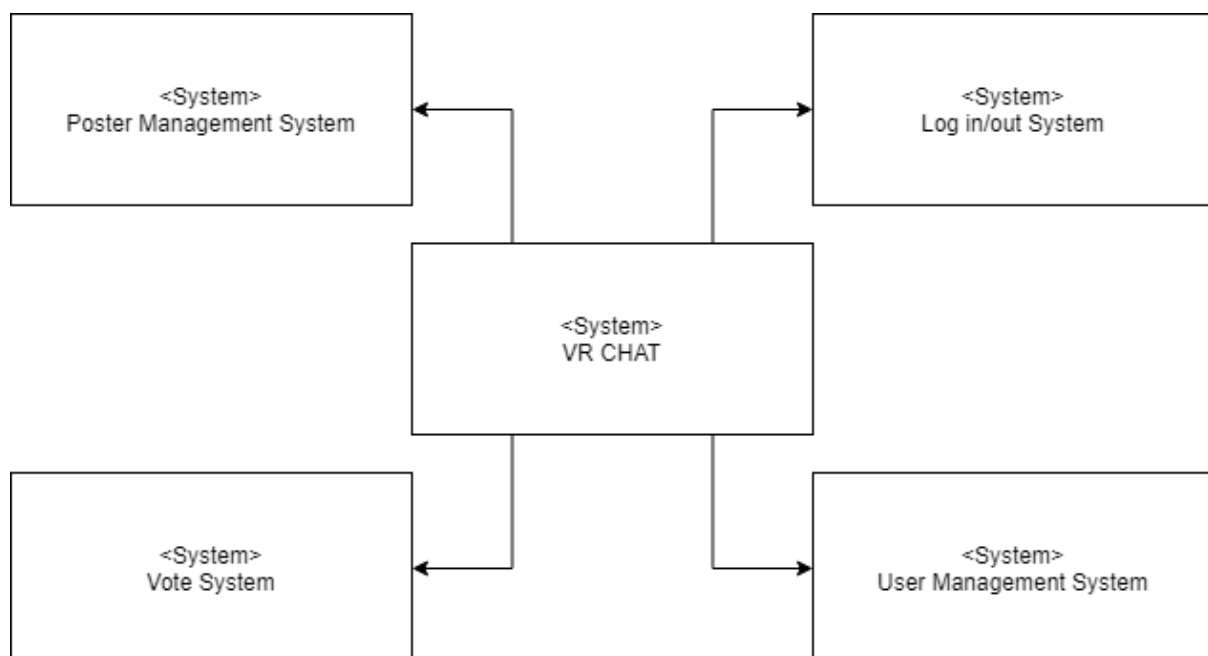
3.7.1. Context Model
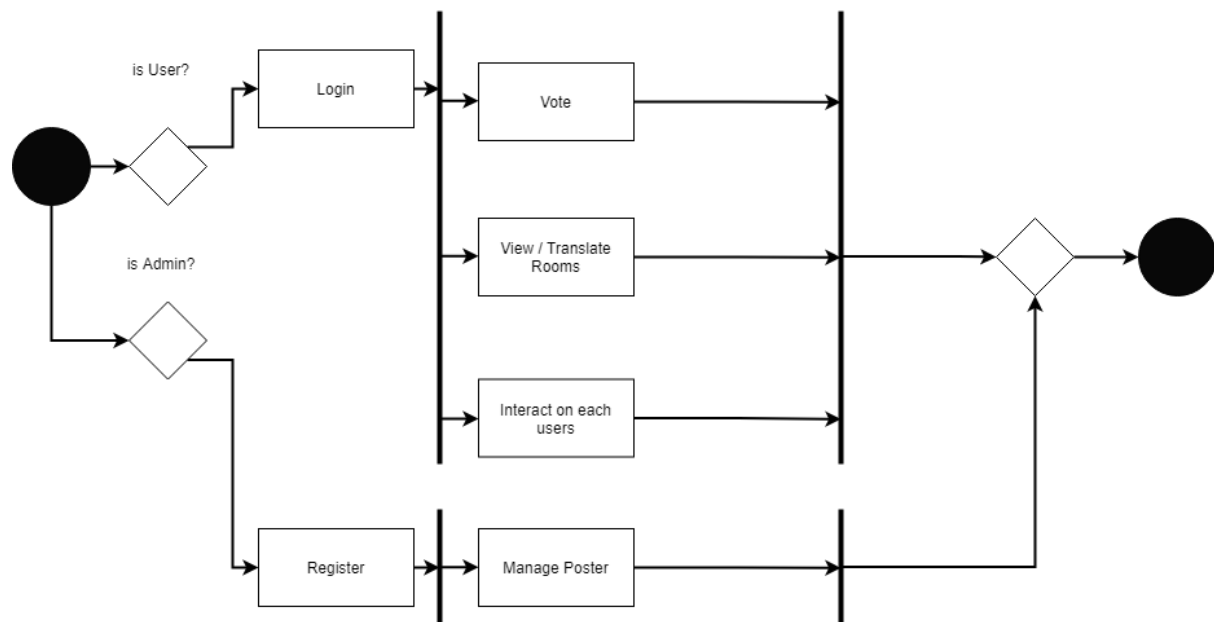

Figure 2 Context model

3.7.2.   Process Model

Figure 3 Process model

3.7.3.   Interaction Model

See 3.2.2. Use Case Diagram

3.7.4.   Behavior Model

3.7.4.1. **Sequence Diagram**

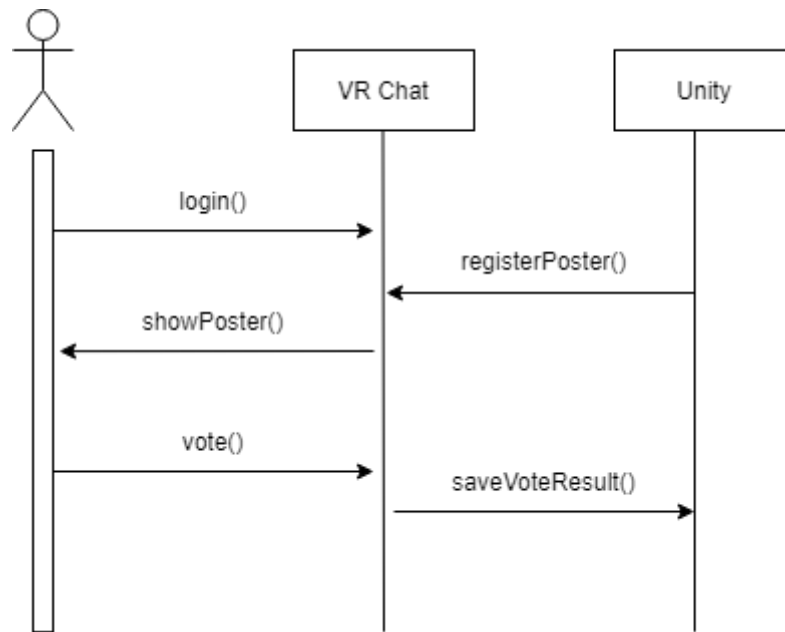It describes mainly the sequence of processing which is our main system.

Figure 4 Sequence diagram

3.8. System Architecture

This Section presents a high-level overview of the anticipated system architecture, showing the distribution of functions across system modules. We use MVC pattern as graphical models of the architecture.
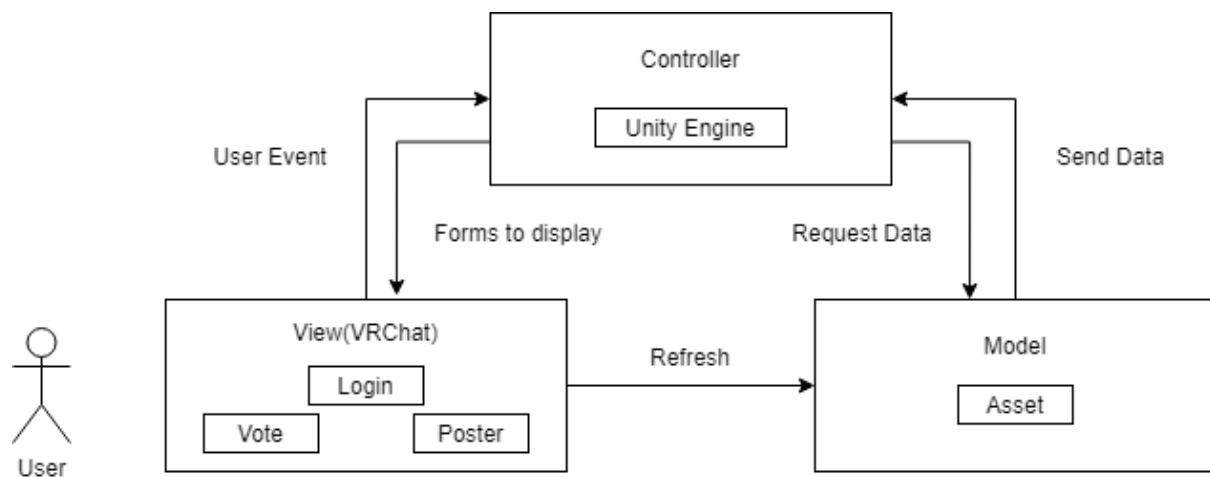

Figure 5 System architecture

3.9. System Evolution

This section describes the fundamental assumptions about the system base and the evolution of future changes in customer needs. This section may help prevent system designers from designing the system in a direction that does not accept change. It is also useful to determine which change request is impossible by the assumption of the system when it comes in.

3.9.1. Assumption and limitation

Our system is developed using VRChat SDK on the unity. In this development environment, it is impossible to link with external databases such as firebase or mysql, and since there is no function to enter text in VRChat the user's writing is also limited. Therefore, it is impossible to implement functions such as information registration that require a high level of database.

The system is developed by focusing on the function of displaying works such as displaying posters and playing YouTube videos explaining the works rather than functions such as user feedback on work exhibitions.

3.9.2. Analysis of change of user requirements.

Our system is developed with a focus on exhibitions of works in the Department of Software. However, there may be demands not only for the software department but also for exhibitions of works by art colleges. The works displayed should be able to be expanded to works such as paintings and 3D models, not just posters or papers.

With the development of VR technology in the future, users can demand not only exhibitions of works, but also real-time communication with the owner of the works, and the ability to directly demonstrate the works. To apply these functions with minimal changes in their original functions, we must leave enough space to apply the functions without moving other objects within the map. In addition, a document describing the dependence on each function should be created so that there is no difficulty in adding the function.

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

[Table 15] Document History

Date	Version	Description	Writer
2021/10/27	0.1	contents and overview	Jihee Kim
2021/10/27	1.0	Addition of 1.1, 1.2	Jaekwang Shin
2021/10/27	1.1	Addition of 1.3, 1.4	Bomin Namkoong
2021/10/27	1.2	Addition of 3.1.1, 3.3	Gyumin Han
2021/10/28	1.3	Addition of 2.1, 2.4, 2.5	Jaekwang Shin
2021/10/28	1.4	Addition of 2.2, 2.3	Bomin Namkoong
2021/10/28	1.5	Addition of 3.4, 3.5	Seokjun Chung
2021/10/29	1.6	Addition of 3.6, 3.9	Jihee Kim
2021/10/30	1.7	Addition of 3.3.2, Revision of 3.3.1	Gyumin Han
2021/10/30	1.8	Addition of 3.5, Revision of 3.3.1	Seokjun Chung
2021/10/30	1.9	Revision of 1.2, 2.1	Bomin Namkoong
2021/10/30	2.0	Revision of 2.2, 2.3	Jaekwang Shin
2021/10/30	2.1	Addition of 3.1	Jongin Park
2021/10/31	2.2	Revision of 3.1	Jongin Park
2021/10/31	2.3	Addition of 3.7, 3.8	Jaehyeong Park