|  |
| --- |
| Software Engineering, Fall, 2014  Department of CSIE in National Taiwan University |
| Software Requirement Specification |
| TransOn |
|  |
|  |
| Advisor: 李允中教授 |
| Members: 許睿陵、秦睿謙、程至賢、林奕安  彭彥肇、林宣丞、王凱昇、吳佳芷 |

|  |
| --- |
|  |

# Outline

[1 Outline 2](#_Toc402867435)

[Revision History 3](#_Toc402867436)

[1. System Architecture 4](#_Toc402867437)

[1.1 Introduction 4](#_Toc402867438)

[1.2 Architecture Description 4](#_Toc402867439)

[2. Functional Requirements 5](#_Toc402867440)

[2.1 Client 5](#_Toc402867441)

[2.2 Server 7](#_Toc402867442)

[3. Interface Requirements 9](#_Toc402867443)

[3.1 External Interface Requirements 9](#_Toc402867444)

[3.1.1 Client 9](#_Toc402867445)

[3.1.2 Server 9](#_Toc402867446)

[3.2 Internal Interface Requirements 9](#_Toc402867447)

[3.2.1 Client to Server 9](#_Toc402867448)

[3.2.2 Client 9](#_Toc402867449)

[3.2.3 Server 11](#_Toc402867450)

[4. Nonfunctional Requirements 12](#_Toc402867451)

[4.1 Client 12](#_Toc402867452)

[4.2 Server 12](#_Toc402867453)

[5. Use Case Diagram 13](#_Toc402867454)

[5.1 Use Case Diagram 13](#_Toc402867455)

[5.2 Actor Description and Actor Use Case Matrix 14](#_Toc402867456)

[5.3 Use case specification 15](#_Toc402867457)

[6. Traceability Matrix 37](#_Toc402867458)

[6.1 Traceability Matrix of Requirements V.S Requirements 37](#_Toc402867459)

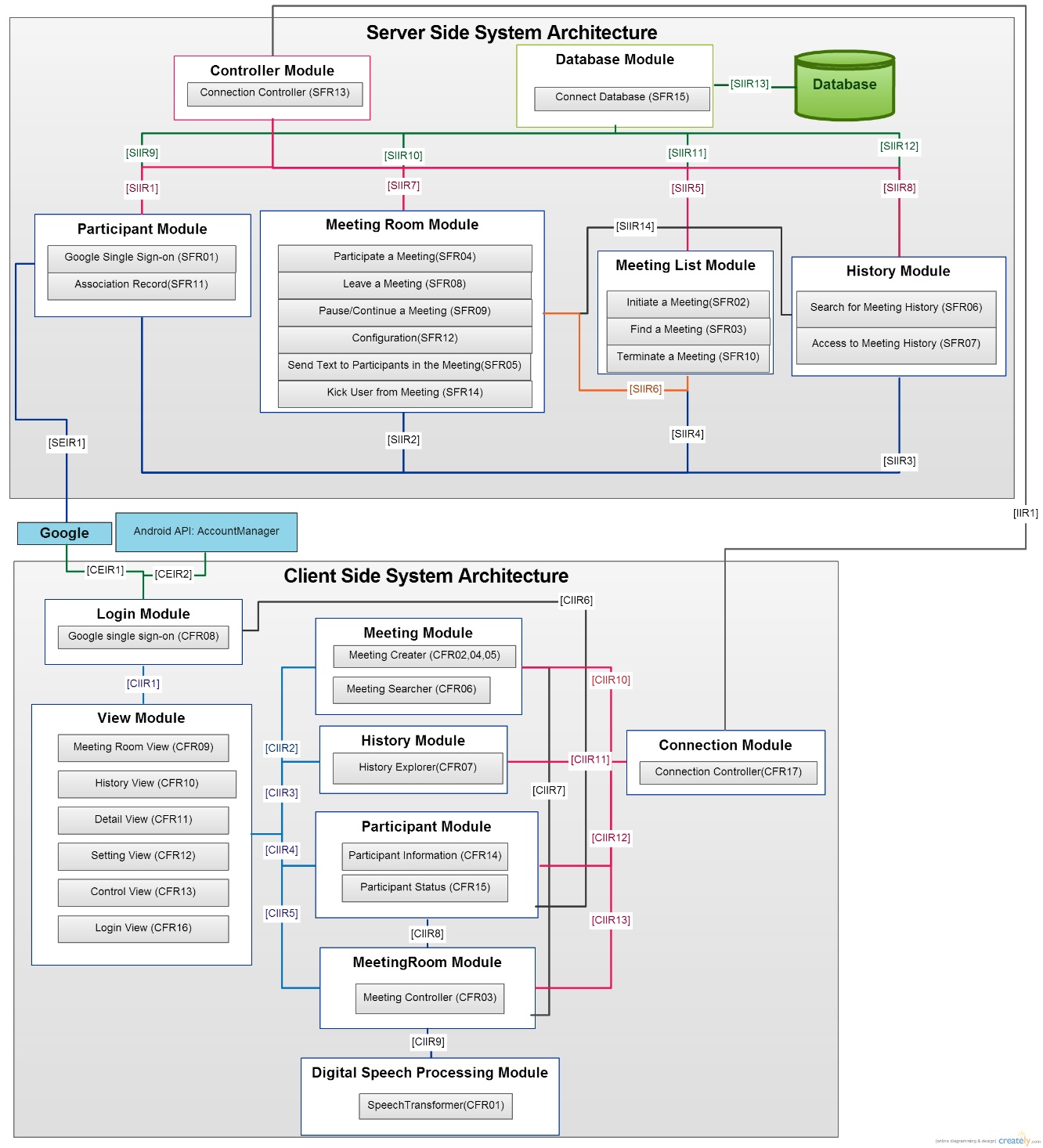
[6.2 Traceability Matrix of Requirements V.S Use Case 37](#_Toc402867460)

[6.3 Traceability Matrix of Requirements V.S Subsystem Function 37](#_Toc402867461)

Revision History

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Revision | Responsibility | Date | Change Message | Reviewer | Review Date |
| 0.1 | 王凱昇 | 11/05/2014 | Initiated | All members | 12/03/2013 |

1. System Architecture
   1. Introduction
   2. Architecture Description



1. Functional Requirements
2. Client

|  |  |
| --- | --- |
|  |  |
| CFR01 | **Speech transformer** |
|  | Speech transformer need to transform speech to text. |
| CFR02 | **Meeting Creator** |
|  | The meeting creator can create meeting and set up meeting information.  Server will return meeting ID back. |
| CFR03 | **Meeting controller** |
|  | 1. Controller provides kicking other participants (by adding them into black list) functionality for administrator. 2. Controller provides mute/unmute participants (which can't be reverted by themselves) functionality for administrator. 3. Controller provides pausing/continuing meeting (by muting all participants) functionality for administrator. 4. Controller provides terminating meeting (freeze the record) functionality for administrator. 5. Controller provides accepting participants to meeting functionality for administrator. 6. Controller provides setting/canceling a participant as an administrator functionality for administrator. 7. Controller can remove participants themselves from meeting. 8. Controller can mute/unmute participants themselves (which can revert by themselves.) |
| CFR04 | **Meeting information** |
|  | Meeting information contains subject, location, latitude, longitude, time, description, initiator, permission policy, duration, list of participants. |
| CFR05 | **Permission policy** |
|  | Permission include transparency (whether show in nearby list) and authority request (whether ask for attendance) |
| CFR06 | **Ongoing meeting searcher** |
|  | Ongoing meeting searcher will list all public meeting nearby. |
| CFR07 | **History explorer** |
|  | History explorer can search the meeting which the user have participated. It can get the history record of specified meeting by server. |
| CFR08 | **User Login with Google Single Sign-on** |
|  | User can use his/her Google account to login the system. |
| CFR09 | **Meeting room view** |
|  | Meeting room will show every messages come from all participants with scroll. Messages are sorted by timestamp. In addition, there is a mute button to mute himself, and a return (leave) button to leave the meeting. |
| CFR10 | **History view** |
|  | The history view first shows options of searching history, and shows results.  After user chooses a record, history view will show conversations in text. |
| CFR11 | **Detail view** |
|  | Detail view displays meeting information. Detail view can be selected in Meeting room and History view. |
| CFR12 | **Setting view** |
|  | The setting view can set the meeting information when user change the setting of meeting. The setting view can be selected in "detail" view. |
| CFR13 | **Control view** |
|  | Control view can be accessed in Meeting room view by administrator. Administrator can pause and terminate the meeting in this view. Also, administrator can kick participants or accept participants in this view. |
| CFR14 | **Participant Information** |
|  | Some participant information is grabbed from google by signing in google account. We will grab name, google account. |
| CFR15 | **Participant status** |
|  | Participant status should associate the status (mute and admin/general) of a participant with the corresponding meeting. |
| CFR16 | **Login View** |
|  | Login View can let user to enter Google Single Sign-on to login, or let user select the stored Google account via Android Account Manager. |
| CFR17 | **Connection Controller** |
|  | Connection controller can connect to server and manage connections for corresponding functions. |
| CFR18 | **Lobby View** |
|  | Lobby View is the default page user would see after user logins. Lobby View has the entrance of History Search View, Meeting Search View. |
| CFR19 | **Meeting Search View** |
|  | Meeting Search View can exploit location, meeting id to get into a meeting. |
| CFR20 | **History Search View** |
|  | The history view shows results. |
| CFR21 | **View after leave** |
|  | Leaving meeting room in any ways will redirect to Meeting Search View |

1. Server

|  |  |
| --- | --- |
|  |  |
| SFR01 | **Google Single Sign-on** |
|  | User can use his/her Google account to login the system. |
| SFR02 | **Initiate a Meeting** |
|  | User can initiate a meeting in the system, when a meeting is initiated, server will generate a meeting ID and send the meeting ID to the user and add the meeting to the ongoing meeting list. |
| SFR03 | **Find a Meeting** |
|  | User can send meeting ID to server, server will show the corresponding meeting to user. User can also ask server to search nearby meetings, server will show the nearby meetings in the ongoing meeting list to user.  Server will store an ongoing meeting list which include all the non-terminated meetings. |
| SFR04 | **Participate a Meeting** |
|  | When user see a meeting, there are 4 situations:   1. The meeting is private, 2. The meeting is secret, 3. The user is in the blacklist of the meeting, or 4. Otherwise.   Server will allow the user to participate the meeting in situation 4, and will ask administrator for permission to participate the meeting in situation 1.  If user successfully participate the meeting, the server will add the user to the participant list of the meeting. |
| SFR05 | **Send Text to Participants in the Meeting** |
|  | When user send text to server, server will send the text to other users in the participant list of the meeting and store the text in the meeting history. |
| SFR06 | **Search for Meeting History** |
|  | User can send the meeting ID to server, server will show the corresponding meeting history. |
| SFR07 | **Access to Meeting History** |
|  | 1. The user is in the participant list of the meeting,   Server will allow the user to access the meeting history when user is in the readable user list of meeting. |
| SFR08 | **Leave a Meeting** |
|  | User can leave the meeting, server will remove the user from participant list of the meeting. |
| SFR09 | **Pause and Continue a Meeting** |
|  | Administrator can pause the meeting, server will mute all the users in the participant list. Administrator can continue the paused meeting, server will unmute all the users in the participant list. |
| SFR10 | **Terminate a Meeting** |
|  | Administrator can terminate the meeting, server will remove the meeting from the ongoing meeting list. |
| SFR11 | **Association Record** |
|  | Equal to combination of CFR04, CFR14 and CFR15 |
| SFR12 | **Configuration** |
|  | Administrator can change setting of meeting which include editing permission, change admin list, change blacklist, and mute setting. |
| SFR13 | **Connection Controller** |
|  | Controller read the user command from connection and delegate to the appropriate modules. |
| SFR14 | **Kick User from Meeting** |
|  | Administrator can kick a user from meeting, server will remove the user from participant list of the meeting. |
| SFR15 | **Connect database** |
|  | Database module establishes the connection to database and is in charge of all the queries. |

1. Interface Requirements
   1. External Interface Requirements
      1. Client

|  |  |
| --- | --- |
|  |  |
| CEIR1 | **Login Module to Google** |
|  | Participant Module send Single Sign-on request to Google and receive general information from Google. |
| CEIR2 | **Login Module to Android API: Account Manager** |
|  | Login Module ask Account Manager which google account to use. |

* + 1. Server

|  |  |
| --- | --- |
|  |  |
| SEIR1 | **Participant Module to Google** |
|  | Participant Module send Single Sign-on request to Google and receive the result from Google. |

* 1. Internal Interface Requirements
     1. Client to Server

|  |  |
| --- | --- |
|  |  |
| IIR1 | **Client side Connection Module to Server side Connection Module** |
|  | The Connection Module of client side can send/receive packet to/from Connection Module of server side. |

* + 1. Client

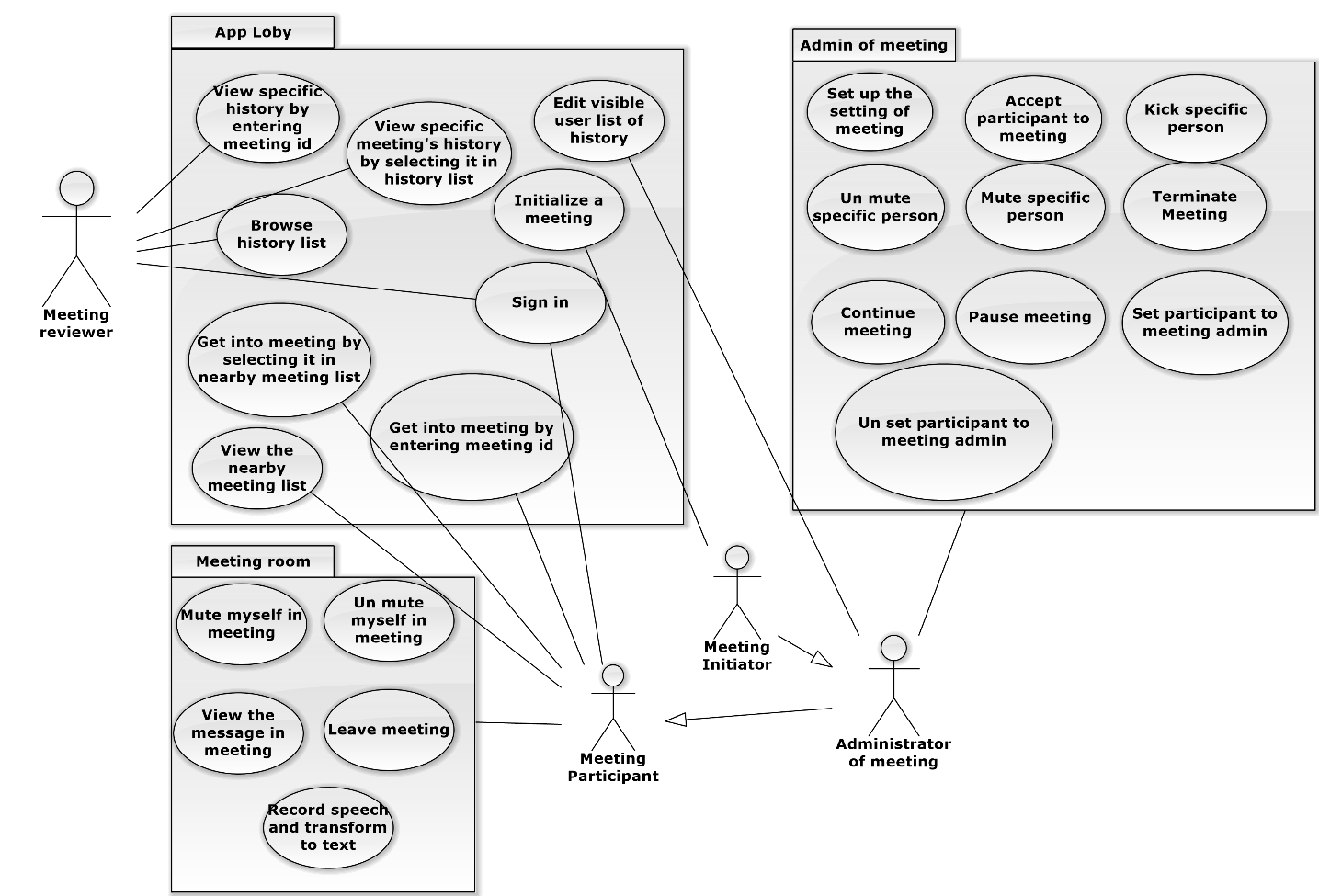
|  |  |
| --- | --- |
|  |  |
| CIIR1 | **View Module to Login Module** |
|  | View Module can send the login information entered by user from Login View to Login Module. |
| CIIR2 | **View Module to Meeting Module** |
|  | View Module can send/get the setting information of meeting to/from Meeting Module, and receive the nearby public meeting from Meeting Module. |
| CIIR3 | **View Module to History Module** |
|  | View Module can get the meeting list which the user have participated or is permitted by its administrators and history record of specified meeting. |
| CIIR4 | **View Module to Participant Module** |
|  | View Module can get/set the participant status from/to Participant Module. |
| CIIR5 | **View Module to Meeting Room Module** |
|  | View Module can set all control setting (pause, kick, exit...) of meeting and get all message sent from participants by Meeting Room Module. |
| CIIR6 | **Participant Module to Login Module** |
|  | Participant Module can use Login Module to get Google Account Information. |
| CIIR7 | **Meeting Room Module to Meeting Module** |
|  | Meeting Controller can use Meeting Module to create meeting and search a Meeting. |
| CIIR8 | **Meeting Room Module to Participant Module** |
|  | Meeting Room module can use Participant Module to mute the client. |
| CIIR9 | **Meeting Room Module to Digital Speech Processing Module** |
|  | Meeting Room module can use Digital Speech Processing Module to transform a speech into a script. |
| CIIR10 | **Meeting Module to Connection Module** |
|  | Meeting Module can use Connection Module to send request of getting ongoing meeting list or creating a new meeting to server. |
| CIIR11 | **History Module to Connection Module** |
|  | History Module can use Connection Module to send exploration request to server side. |
| CIIR12 | **Participant Module to Connection Module** |
|  | Participant Module can use Connection Module to send google access token to server |
| CIIR13 | **Meeting Room Module to Connection Module** |
|  | Meeting Room Module can use Connection Module to send message or control command to server. |

* + 1. Server

|  |  |
| --- | --- |
|  |  |
| SIIR1 | **Controller Module to Participant Module** |
|  | Controller Module can send the login information entered by user from Login View to Login Module. |
| SIIR2 | **Meeting Room Module to Participant Module** |
|  | Meeting Room Module can get/set participant status from/to Participant Module. |
| SIIR3 | **History Module to Participant Module** |
|  | History Module can get the participant information (SFR07 1.~4.) from Participant Module |
| SIIR4 | **Meeting List Module to Participant Module** |
|  | Meeting List Module can get/set participant status from/to Participant Module. |
| SIIR5 | **Controller Module to Meeting List Module** |
|  | Controller Module can control the setting (initiate, find, terminate) to set a meeting room by Meeting List Module. |
| SIIR6 | **Meeting List Module to Meeting Room Module** |
|  | Meeting List Module can initiate/terminate a meeting room from Meeting Room Module. |
| SIIR7 | **Controller Module to Meeting Room Module** |
|  | Controller Module can control the setting (participate, leave, pause, kick...) to set a meeting room by Meeting Room Module. |
| SIIR8 | **Controller Module to History Module** |
|  | Controller Module can search/access the meeting history by History Module. |
| SIIR9 | **Participant module to database module** |
|  | Participant module queries database through database module. |
| SIIR10 | **Meeting room module to database module** |
|  | Meeting room module queries database through database module. |
| SIIR11 | **Meeting list module to database module** |
|  | Meeting list module queries database through database module. |
| SIIR12 | **History module to database module** |
|  | History module queries database through database module. |
| SIIR13 | **Database module to database** |
|  | Database module need to make connection with database. (refer SFR15) |
| SIIR14 | **History module to meeting room module** |
|  | History Module can use configuration functionality of Meeting Room Module to reset history viewing permission. |

1. Nonfunctional Requirements
   1. Client
   2. Server
2. Use Case Diagram

## Use Case Diagram



## Actor Description and Actor Use Case Matrix

|  |  |
| --- | --- |
| Actor Name | Actor Description |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

Table 2: Actor Description Table

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Use Case\Actor |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

Table 3: Actor Use Case Matrix

## Use case specification

|  |  |  |
| --- | --- | --- |
| Use Case ID | TOUC1 | |
| Use Case Name | Sign in | |
| Goal |  | |
| Requirements | [CFR08: Google Single Sign-on], [CFR16: Login View], [CIIR1: View Module to Login Module], [CIIR6], [CIIR12], [CFR17: Connection Controller], [CEIR1], [CEIR2], [IIR1], [SFR01: Google Single Sign-on], [SFR11: Association Record], [SFR13: Connection Controller], [SIIR1: Controller Module to Participant Module], [SEIR1] | |
| Description |  | |
| Actor | Meeting Reviewer, Meeting Participant | |
| Assumptions |  | |
| Constraints |  | |
| Priority | High | |
| Pre-Conditions | None | |
| Post-Conditions | User login to system. | |
| Basic Flow | Actor | System |
| 1. User enter Google username and password in Login View. | 1. Client send login request to server. 2. Server connect to Google to check username and password. 3. Reply the result to user. |
| Alternative Flows | If user failed to login (ex: username or password is wrong), user need to login again. | |
| Exceptional Flows |  | |
| Inclusion Use Case |  | |
| Extension Use Case |  | |
| Business Rules |  | |
| Artifacts |  | |
| Use Case Glossary |  | |
| Notes |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID | TOUC2 | |
| Use Case Name | View specific history by entering meeting id | |
| Goal |  | |
| Requirements | [CFR10: History View], [CFR07: History Module], [CFR17: Connection Controller], [CIIR3: View Module to History Module], [CIIR11], [IIR1], [SFR06: Search for Meeting History], [SFR07: Access to Meeting History], [SFR13: Connection Controller], [SIIR8] | |
| Description |  | |
| Actor | Meeting Reviewer | |
| Assumptions |  | |
| Constraints |  | |
| Priority | High | |
| Pre-Conditions | Sign in | |
| Post-Conditions | None | |
| Basic Flow | Actor | System |
| 1. User enter meeting id in History View. | 1. Client send view history request to server. 2. Server send specific history corresponding to meeting id to Client. 3. Client display the specific history. |
| Alternative Flows | If user enter wrong meeting id, client will display error to user. | |
| Exceptional Flows |  | |
| Inclusion Use Case |  | |
| Extension Use Case |  | |
| Business Rules |  | |
| Artifacts |  | |
| Use Case Glossary |  | |
| Notes |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID | TOUC3 | |
| Use Case Name | View specific meeting's history by selecting it in history list | |
| Goal |  | |
| Requirements | [CFR10: History View], [CFR07: History Module], [CFR17: Connection Controller], [CIIR3: View Module to History Module], [CIIR11], [IIR1], [SFR07: Access to Meeting History], [SFR11: Association Record], [SFR13: Connection Controller], [SIIR3], [SIIR8] | |
| Description |  | |
| Actor | Meeting Reviewer | |
| Assumptions |  | |
| Constraints |  | |
| Priority | High | |
| Pre-Conditions | Sign in | |
| Post-Conditions | None | |
| Basic Flow | Actor | System |
| 1. User select a meeting history in history list. | 1. Client ask server for specific meeting history in history list. 2. Server send specific meeting history to client. 3. Client display the history. |
| Alternative Flows | None | |
| Exceptional Flows |  | |
| Inclusion Use Case |  | |
| Extension Use Case |  | |
| Business Rules |  | |
| Artifacts |  | |
| Use Case Glossary |  | |
| Notes |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID | TOUC4 | |
| Use Case Name | Browse history list | |
| Goal |  | |
| Requirements | IIR1,[CFR07: History explorer],[SFR07: Access to Meeting History],CIIR3,CIIR11, SIIR8 | |
| Description |  | |
| Actor | Meeting Reviewer | |
| Assumptions |  | |
| Constraints |  | |
| Priority | High | |
| Pre-Conditions | Sign in | |
| Post-Conditions | None | |
| Basic Flow | Actor | System |
| 1. Client enter History View to view a visible meeting history. 2. The meetings are showed in the view. | 1. Client ask server for meeting that one is able to see. 2. Server send the list of meeting that client able to see. |
| Alternative Flows |  | |
| Exceptional Flows |  | |
| Inclusion Use Case |  | |
| Extension Use Case |  | |
| Business Rules |  | |
| Artifacts |  | |
| Use Case Glossary |  | |
| Notes |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID | TOUC5 | |
| Use Case Name | Get into meeting by selecting it in nearby meeting list | |
| Goal |  | |
| Requirements |  | |
| Description |  | |
| Actor | Meeting Participant | |
| Assumptions |  | |
| Constraints |  | |
| Priority |  | |
| Pre-Conditions | TOUC6: View the nearby meeting list | |
| Post-Conditions | None | |
| Basic Flow | Actor | System |
| 1. Client chooses one of the meeting in the nearby list to enter. | 1. Client asks server for entering a specific meeting. 2. Server adds the client to the meeting. . 3. Server sends a success message to the client. |
| Alternative Flows | Actor | System |
| 1. Client chooses one of the meeting in the nearby list to enter. 2. Client enters the Meeting Room View. | 1. Client asks server for entering a specific meeting. 2. The meeting requires the permission to enter. 3. The server sends a message to all administrator. 4. One Permission would make the server to add the client to the meeting. 5. The Server send a success message to the client. |
| Exceptional Flows |  | |
| Inclusion Use Case |  | |
| Extension Use Case |  | |
| Business Rules |  | |
| Artifacts |  | |
| Use Case Glossary |  | |
| Notes |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID | TOUC6 | |
| Use Case Name | View the nearby meeting list | |
| Goal |  | |
| Requirements |  | |
| Description |  | |
| Actor | Meeting Participant | |
| Assumptions |  | |
| Constraints |  | |
| Priority |  | |
| Pre-Conditions | Sign in | |
| Post-Conditions | None | |
| Basic Flow | Actor | System |
| 1. After setting a distance, User can select the meeting nearby to enter.. 2. The View displays the meeting found. | 1. Server search the meeting within the set distance from the client and sends to the client. |
| Alternative Flows | None | |
| Exceptional Flows |  | |
| Inclusion Use Case |  | |
| Extension Use Case |  | |
| Business Rules |  | |
| Artifacts |  | |
| Use Case Glossary |  | |
| Notes |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID | TOUC7 | |
| Use Case Name | Get into meeting by entering meeting id | |
| Goal |  | |
| Requirements | [CFR16: Login View], [CFR06: Meeting Searcher], [CFR17: Connection Controller], [CIIR2], [CIIR10], [IIR1], [SFR03: Find a Meeting], [SFR04: Participate a Meeting], [SFR01: Google Single Sign-on], [SFR13: Connection Controller], [SIIR2], [SIIR4], [SIIR5], [SIIR7] | |
| Description |  | |
| Actor | Meeting Participant | |
| Assumptions |  | |
| Constraints |  | |
| Priority | High | |
| Pre-Conditions | Sign in | |
| Post-Conditions | None | |
| Basic Flow | Actor | System |
| 1. A login user select to search a meeting by entering a meeting id. 2. User get into the meeting. | 1. Client query specific meeting by id in server’s ongoing meeting list. 2. Server check the authority of user. 3. Server send specific meeting to client. |
| Alternative Flows | If the meeting is a private meeting, Server should send message to administrator for privilege. | |
| Exceptional Flows |  | |
| Inclusion Use Case |  | |
| Extension Use Case |  | |
| Business Rules |  | |
| Artifacts |  | |
| Use Case Glossary |  | |
| Notes |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID | TOUC8 | |
| Use Case Name | Initialize a meeting | |
| Goal |  | |
| Requirements | [CFR16: Login View], [CFR12: Setting View], [CFR02: Meeting Creator], [CFR09: Meeting Room View], [CFR03: Meeting Controller], [CFR17: Connection Controller], [CIIR2], [CIIR10], [CIIR05], [CIIR7], [IIR1], [SFR02: Initiate a Meeting], [SFR04: Participate a Meeting], [SFR13: Connection Controller], [SFR01: Google Single Sign-on], [SFR11: Association Record], [SIIR5], [SIIR6], [SIIR4], [SEIR1] | |
| Description |  | |
| Actor | Meeting Initiator | |
| Assumptions |  | |
| Constraints |  | |
| Priority | High | |
| Pre-Conditions | Sign in | |
| Post-Conditions | None | |
| Basic Flow | Actor | System |
| 1. User choose to create a new meeting by clicking create a new meeting button. 2. User get into the newly created meeting as an administrator. | 1. Client ask server to create a new meeting. 2. Server return the newly created meeting to client. |
| Alternative Flows | None | |
| Exceptional Flows |  | |
| Inclusion Use Case |  | |
| Extension Use Case |  | |
| Business Rules |  | |
| Artifacts |  | |
| Use Case Glossary |  | |
| Notes |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID | TOUC9 | |
| Use Case Name | Edit visible user list of history | |
| Goal |  | |
| Requirements | [CFR10: History View], [CFR07: History Explorer], [CFR17: Connection Controller], [CIIR3], [CIIR11], [IIR1], [SFR07: Access to Meeting History], [SFR12: Configuration], [SFR13: Connection Controller], [SFR15: Connect Database], [SIIR8], [SIIR14], [SIIR12], [SIIR13] | |
| Description |  | |
| Actor | Meeting Reviewer | |
| Assumptions |  | |
| Constraints |  | |
| Priority | High | |
| Pre-Conditions | [View specific history by entering meeting id] or [View specific meeting’s history by selecting it in history list] | |
| Post-Conditions | None | |
| Basic Flow | Actor | System |
| 1. User select setting in history view. 2. User get success message. | 1. Client ask server to reset history viewing permission. 2. Server check the user account for authority. 3. Server reset the permission. 4. Server sent ack to client. |
| Alternative Flows | If Server authority check failed, the server sent failed message back. User get failed message | |
| Exceptional Flows |  | |
| Inclusion Use Case |  | |
| Extension Use Case |  | |
| Business Rules |  | |
| Artifacts |  | |
| Use Case Glossary |  | |
| Notes |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID | TOUC10 | |
| Use Case Name | Mute myself in meeting | |
| Goal |  | |
| Requirements | [CRF03: Meeting Controller], [CFR09: Meeting Room View], [CIIR5] | |
| Description | Close the recorder of the phone and prevent myself from sending message | |
| Actor | Meeting Participant | |
| Assumptions |  | |
| Constraints | The current status is not mute | |
| Priority | Medium | |
| Pre-Conditions | The user is in a meeting and at the main view of a meeting room | |
| Post-Conditions | The recorder of the phone is closed and user cannot send any message | |
| Basic Flow | Actor | System |
| 1. User press the mute button | 1. Meeting controller change the status to mute |
| Alternative Flows |  | |
| Exceptional Flows |  | |
| Inclusion Use Case |  | |
| Extension Use Case |  | |
| Business Rules |  | |
| Artifacts |  | |
| Use Case Glossary |  | |
| Notes |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID | TOUC11 | |
| Use Case Name | Unmute myself in meeting | |
| Goal |  | |
| Requirements | [CRF03: Meeting Controller], [CFR09: Meeting Room View], [CIIR5] | |
| Description | Open the recorder of the phone, start recognizing voice and send the recognized result | |
| Actor | Meeting Participant | |
| Assumptions |  | |
| Constraints | The current status is mute and is not mute by the admin | |
| Priority | Medium | |
| Pre-Conditions | The user is in a meeting and at the main view of a meeting room | |
| Post-Conditions | The recorder of the phone is opened and user can send message | |
| Basic Flow | Actor | System |
| 1. User press the unmute button | 1. Meeting controller change the status to unmute |
| Alternative Flows |  | |
| Exceptional Flows |  | |
| Inclusion Use Case |  | |
| Extension Use Case |  | |
| Business Rules |  | |
| Artifacts |  | |
| Use Case Glossary |  | |
| Notes |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID | TOUC12 | |
| Use Case Name | View the messages in meeting | |
| Goal |  | |
| Requirements | [CFR09: Meeting Room View] | |
| Description | Know what everyone has talked in current meeting | |
| Actor | Meeting Participant | |
| Assumptions |  | |
| Constraints |  | |
| Priority | High | |
| Pre-Conditions | The user is in a meeting and at the main view of a meeting room | |
| Post-Conditions |  | |
| Basic Flow | Actor | System |
| 1. User can browse the messages by reading, scrolling up, and scrolling down. | 1. Handle the scrolling up or down event, and show the messages |
| Alternative Flows |  | |
| Exceptional Flows |  | |
| Inclusion Use Case |  | |
| Extension Use Case |  | |
| Business Rules |  | |
| Artifacts |  | |
| Use Case Glossary |  | |
| Notes |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID | TOUC13 | |
| Use Case Name | Leave meeting | |
| Goal |  | |
| Requirements | [CRF03: Meeting Controller], [CFR09: Meeting Room View], [CIIR5], [SFR08: Leave a Meeting], [SFR13: Connection Controller], [SIIR7] | |
| Description | Leave the meeting room, won't send or receive any messages in the future | |
| Actor | Meeting Participant | |
| Assumptions |  | |
| Constraints |  | |
| Priority | High | |
| Pre-Conditions | The user is in a meeting and at the main view of a meeting room | |
| Post-Conditions | None | |
| Basic Flow | Actor | System |
| 1. User press the return (leave) button 2. User can press "yes" to leave the meeting, or press "no" to remain in the meeting | 1. An alert window shows up 2. If the "no" button is pressed, then goto 8. 3. Leave the meeting, and return to history view 4. Send message to server 5. Server modify the status of the meeting and the participant, end 6. Close the alert window |
| Alternative Flows |  | |
| Exceptional Flows |  | |
| Inclusion Use Case |  | |
| Extension Use Case |  | |
| Business Rules |  | |
| Artifacts |  | |
| Use Case Glossary |  | |
| Notes |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID | TOUC14 |  |
| Use Case Name | Set up the setting of meeting | |
| Goal |  | |
| Requirements | [CFR12: Setting view],[CFR02:Meeting  Creator],[CFR04:Meeting information],[CFR05:Permission policy] | |
| Description |  | |
| Actor | Administrator of Meeting | |
| Assumptions |  | |
| Constraints |  | |
| Priority | High | |
| Pre-Conditions | The meeting is in the user history or ongoing | |
| Post-Conditions |  | |
| Basic Flow | Actor | System |
|  | 1. Administrator sets the meeting setting. | 1. Meeting Module send the setting to server. 2. Server sends the updated setting to all participants. 3. Setting view sends the setting to Meeting Module. |
| Alternative Flows | If administrator failed to set meeting setting, he/she may be deprived permission when he/she was setting. | |
| Exceptional Flows |  | |
| Inclusion Use Case |  | |
| Extension Use Case |  | |
| Business Rules |  | |
| Artifacts |  | |
| Use Case Glossary |  | |
| Notes |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID | TOUC15 | |
| Use Case Name | Accept participant to meeting | |
| Goal |  | |
| Requirements | [CFR13],[CIIR04],[CIIR05],[CFR03],[CIIR13],[CFR17],[CIIR12] | |
| Description |  | |
| Actor | Administrator of Meeting | |
| Assumptions |  | |
| Constraints |  | |
| Priority | High | |
| Pre-Conditions | Another user send a participant request to the meeting which is administrated by user. | |
| Post-Conditions | Another user will be in the participant list | |
| Basic Flow | Actor | System |
| 1. User accept the participant’s request. | 1. System send the Controlling setting to server 2. Server update the Association Record 3. Server send the updated meeting information to all participants. |
| Alternative Flows | None | |
| Exceptional Flows |  | |
| Inclusion Use Case |  | |
| Extension Use Case |  | |
| Business Rules |  | |
| Artifacts |  | |
| Use Case Glossary |  | |
| Notes |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID | TOUC16 | |
| Use Case Name | Kick specific person | |
| Goal |  | |
| Requirements | [CFR13],[CIIR05],[CFR03],[CIIR04],[CFR14],[CFR15],[CIIR13],[CIIR12] | |
| Description |  | |
| Actor | Administrator of Meeting | |
| Assumptions |  | |
| Constraints |  | |
| Priority | High | |
| Pre-Conditions |  | |
| Post-Conditions | The specified participant is out of participant list | |
| Basic Flow | Actor | System |
| 1. Meeting Administrator select a specific participant and kick it. | 1. System sends a controlling setting to server. 2. Server updates the Association Record 3. Server send the updated meeting information to all participants. |
| Alternative Flows | If administrator failed to kick others, he/she may be deprived permission when he/she was kicking. | |
| Exceptional Flows |  | |
| Inclusion Use Case |  | |
| Extension Use Case |  | |
| Business Rules |  | |
| Artifacts |  | |
| Use Case Glossary |  | |
| Notes |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID | TOUC17 | |
| Use Case Name | Unmute specific person | |
| Goal |  | |
| Requirements | [CFR13],[CIIR05],[CFR03],[CIIR04],[CFR14],[CFR15],[CIIR13],[CIIR12] | |
| Description |  | |
| Actor | Administrator of meeting | |
| Assumptions |  | |
| Constraints |  | |
| Priority | High | |
| Pre-Conditions | The specific person was muted before. | |
| Post-Conditions | The specific person is not muted | |
| Basic Flow | Actor | System |
| 1. Meeting Administrator select a specific participant and unmute him/her. | 1. System sends a controlling setting to server. 2. Server updates the Association Record 3. Server send the updated meeting information to all participants. |
| Alternative Flows | If administrator failed to unmute others, he/she may be deprived permission when he/she was unmuting. | |
| Exceptional Flows |  | |
| Inclusion Use Case |  | |
| Extension Use Case |  | |
| Business Rules |  | |
| Artifacts |  | |
| Use Case Glossary |  | |
| Notes |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID | TOUC18 | |
| Use Case Name | Mute specific person | |
| Goal |  | |
| Requirements | [CFR13],[CIIR05],[CFR03],[CIIR04],[CFR14],[CFR15],[CIIR13],[CIIR12] | |
| Description |  | |
| Actor | Administrator of meeting | |
| Assumptions |  | |
| Constraints |  | |
| Priority | High | |
| Pre-Conditions | The specific person was unmuted before. | |
| Post-Conditions | The specific person is muted | |
| Basic Flow | Actor | System |
| 1. Meeting Administrator select a specific participant and mute him/her. | 1. System sends a controlling setting to server. 2. Server updates the Association Record 3. Server send the updated meeting information to all participants. |
| Alternative Flows | If administrator failed to mute others, he/she may be deprived permission when he/she was muting. | |
| Exceptional Flows |  | |
| Inclusion Use Case |  | |
| Extension Use Case |  | |
| Business Rules |  | |
| Artifacts |  | |
| Use Case Glossary |  | |
| Notes |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID | TOUC19 | |
| Use Case Name | Terminate Meeting | |
| Goal |  | |
| Requirements | [CFR03: Meeting controller], [CFR13: Control view], [SFR09: Pause and Continue a Meeting], [CIIR5] , [CIIR13] , [SIIR7] | |
| Description |  | |
| Actor | Administrator of meeting | |
| Assumptions |  | |
| Constraints |  | |
| Priority | Low | |
| Pre-Conditions | The meeting which user wants to terminate must be suspended. | |
| Post-Conditions | The meeting has been terminate. | |
| Basic Flow | Actor | System |
| 1. User presses the terminate button. | 1. User asks system to terminate the meeting. 2. System unmutes all users in the meeting. |
| Alternative Flows | None | |
| Exceptional Flows |  | |
| Inclusion Use Case |  | |
| Extension Use Case |  | |
| Business Rules |  | |
| Artifacts |  | |
| Use Case Glossary |  | |
| Notes |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID | TOUC20 | |
| Use Case Name | Continue meeting | |
| Goal |  | |
| Requirements | [CFR03: Meeting controller], [CFR13: Control view], [SFR09: Pause and Continue a Meeting], [CIIR5] , [CIIR13] , [SIIR7] | |
| Description |  | |
| Actor | Administrator of meeting | |
| Assumptions |  | |
| Constraints |  | |
| Priority | Low | |
| Pre-Conditions | The meeting which user wants to continue must be suspended | |
| Post-Conditions | The meeting has been continued | |
| Basic Flow | Actor | System |
| 1. User presses the continue button | 1. User asks system to continue the meeting 2. System un mutes all users in the meeting |
| Alternative Flows | None | |
| Exceptional Flows |  | |
| Inclusion Use Case |  | |
| Extension Use Case |  | |
| Business Rules |  | |
| Artifacts |  | |
| Use Case Glossary |  | |
| Notes |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID | TOUC21 | |
| Use Case Name | Pause meeting | |
| Goal |  | |
| Requirements | [CFR03: Meeting controller], [CFR13: Control view], [SFR09: Pause and Continue a Meeting], [CIIR5] , [CIIR13] , [SIIR7] | |
| Description |  | |
| Actor | Administrator of meeting | |
| Assumptions |  | |
| Constraints |  | |
| Priority | Low | |
| Pre-Conditions | The meeting which user wants to continue must be ongoing | |
| Post-Conditions | The meeting has been paused | |
| Basic Flow | Actor | System |
| 1. User presses the pause meeting button | 1. User asks system to pause a meeting 2. System mutes all users in the meeting |
| Alternative Flows | None | |
| Exceptional Flows |  | |
| Inclusion Use Case |  | |
| Extension Use Case |  | |
| Business Rules |  | |
| Artifacts |  | |
| Use Case Glossary |  | |
| Notes |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID | TOUC22 | |
| Use Case Name | Set participant to meeting admin | |
| Goal |  | |
| Requirements | [CFR03: Meeting controller], [CFR13: Control view], [SFR12: Configuration], [CIIR5] , [CIIR13] , [SIIR7] | |
| Description |  | |
| Actor | Administrator of meeting | |
| Assumptions |  | |
| Constraints |  | |
| Priority | Medium | |
| Pre-Conditions | None | |
| Post-Conditions | The participant has been set as meeting admin | |
| Basic Flow | Actor | System |
| 1. User sets somebody as a meeting admin | 1. User asks system to let somebody become meeting admin 2. The specific person has been set to a meeting admin |
| Alternative Flows | None | |
| Exceptional Flows |  | |
| Inclusion Use Case |  | |
| Extension Use Case |  | |
| Business Rules |  | |
| Artifacts |  | |
| Use Case Glossary |  | |
| Notes |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID | TOUC23 | |
| Use Case Name | Unset participant to meeting admin | |
| Goal |  | |
| Requirements | [CFR03: Meeting controller], [CFR13: Control view], [SFR12: Configuration], [CIIR5] , [CIIR13] , [SIIR7] | |
| Description |  | |
| Actor | Administrator of meeting | |
| Assumptions |  | |
| Constraints |  | |
| Priority | Medium | |
| Pre-Conditions | None | |
| Post-Conditions | The participant has been set as meeting participant | |
| Basic Flow | Actor | System |
| 1. User unsets somebody to meeting admin | 1. User asks system to let somebody become meeting participant from meeting admin 2. The specific person has been un set to a meeting admin |
| Alternative Flows | None | |
| Exceptional Flows |  | |
| Inclusion Use Case |  | |
| Extension Use Case |  | |
| Business Rules |  | |
| Artifacts |  | |
| Use Case Glossary |  | |
| Notes |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID | TOUC24 | |
| Use Case Name | Record speech and transform to text | |
| Goal |  | |
| Requirements | [CFR01: Speech transformer], [CFR09: Meeting room view], [CIIR9] | |
| Description |  | |
| Actor | Meeting participant | |
| Assumptions |  | |
| Constraints |  | |
| Priority | High | |
| Pre-Conditions | Must be in a meeting with the status of un-mute | |
| Post-Conditions | System records the new messages and transform to text and shows the new message to every users in the meeting | |
| Basic Flow | Actor | System |
| 1. User speaks some sentences | 1. Records those sentences 2. Transform those sentences from speech to text 3. Show the new messages to every users in the meeting |
| Alternative Flows | None | |
| Exceptional Flows |  | |
| Inclusion Use Case |  | |
| Extension Use Case |  | |
| Business Rules |  | |
| Artifacts |  | |
| Use Case Glossary |  | |
| Notes |  | |

1. Traceability Matrix

## Traceability Matrix of Requirements V.S Requirements

## Traceability Matrix of Requirements V.S Use Case

## Traceability Matrix of Requirements V.S Subsystem Function