**Boku Wa Shinitai**

**MyMajeksServer  
  
Use Case Report**

**Revision History**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Authors** | **Description of Change** | **Sections** | **Rev** | **Date** |
|  |  |  |  |  |
|  |  |  |  |  |

**Table of Contents**

[1 Team Description 4](#_Toc903772)

[2 Project Description 5](#_Toc903773)

[2.1 Use Case Diagram 7](#_Toc903774)

[2.2 Use Case Lists 8](#_Toc903775)

[2.2.1 <Create New Account> 9](#_Toc903776)

[2.2.2 <Add friend/Remove> 9](#_Toc903777)

[2.2.3 <Call/Message/Video Chat Friend> 10](#_Toc903778)

[2.2.4 <Add Additional User to a Personal Group Chat> 12](#_Toc903779)

[2.2.5 <Change Status> 12](#_Toc903780)

[2.2.6 <Create Channel > 13](#_Toc903781)

[2.2.7 <Block/Remove Friend> 14](#_Toc903782)

# Team Description

|  |  |
| --- | --- |
| **Team Member Name** | **Email Address** |
| Sami Halwani | 4sakenmage1@csu.fullerton.edu |
| Aarib Abdullah |  |
| Aatib Abdullah |  |
| Andrew Shaker |  |
|  |  |
|  |  |

# Project Description

**Statement of Purpose:**

The purpose of MyMajeks is to create an easy to use communication platform for casual users, gamers, and business environments which allows the user to connect with peers, create lobbies for voice, text, and video chat, and give live updates of activity of other users they are connected to.

TLDR: Communication platform that allows users to communicate with peers.

**Detailed Description:**

The application will consist of 2 main parts:

* Users
* Channels

**The User:**

The User will have to create an account using an email or username, and a password to log in. The user will have the ability to customize their account by changing their DisplayName. The user will also have the option to host private lobbies. Lobbies will default to close on login (Busy), unless the user changes their user/lobby status. The user/lobby statuses will include:

* + Available - Open for all users (Can customize to allow certain users to wait for an acceptation to the lobby)
  + Away – User is idle, but still available to host lobbies
  + Busy – User is in another Lobby or has closed their lobby
  + Offline - User is offline and cannot host lobbies.

Users will also have access to a friend’s list, where they can find their peer’s usernames to add them. By adding friends, you will be able to freely communicate with them using a direct text chat, Lobby hosted voice, video or audio chat. The friend’s activity will automatically show up in the Primary Users Feed unless otherwise specified. These private lobbies will also allow for multiple users (2+).

**Channel Lobby:**

Users will be able to connect with friends in custom group lobbies, with a minimum of 3 people. Each Lobby will include a new text Channels, voice Channels distinct from their personal conversations. Within the Channel, and all users will receive notification of any text updates. Lobbies will only run so long as at least one of the users are online.

Channel Properties

* + Channels will include voice Channels and text Channels
  + Text Sub-Channel:
    - Allow strings for up to 2,000 characters (Constant character count) to be posted within text chat
      * Admins will be allowed to set chat limits (including spam detection and dictionary filters)
  + Voice Sub-Channel:
    - Users will be able to connect to channels where voice chat is enabled so long as there is someone within in the sub-channel
    - Admins will be able to kick/ban/mute/move users between channels
    - Voice sub-channels can be privileged by admins to allow channels accessed by certain ranks or certain users

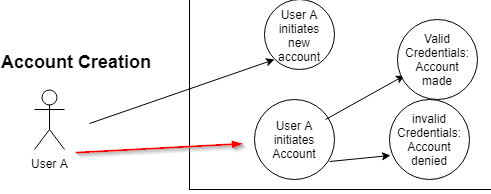
Users Within Channel

* + Channels will have different ranking systems created by the admins of the channel (The original 3 creators of the channel)
  + Admins will be allowed to create voice channels and text channels with specific privileges
  + Admins will be able to modify ranking privileges
  + Users will be able to chat/message in channels they have access to
  + Users will be able to private message other users within the channel regardless of friendship (except if there is a block)

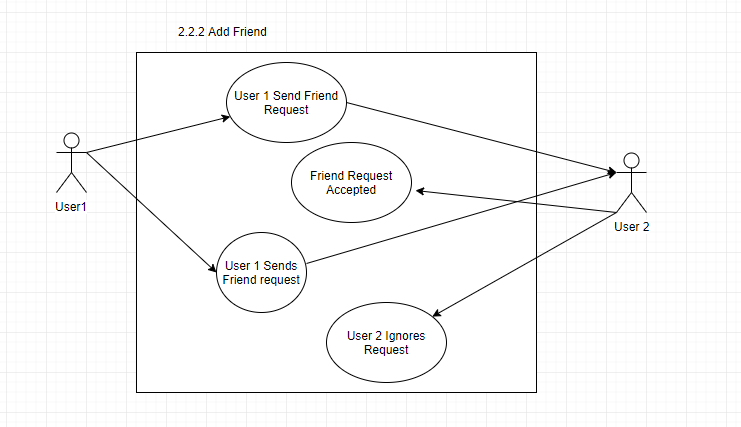
## Use Case Diagram



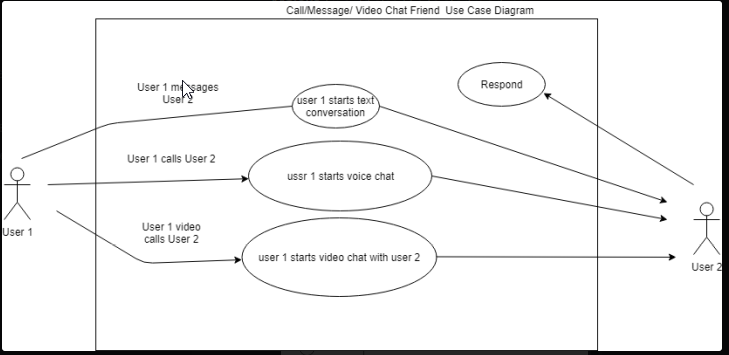
2.2.1 Create Account



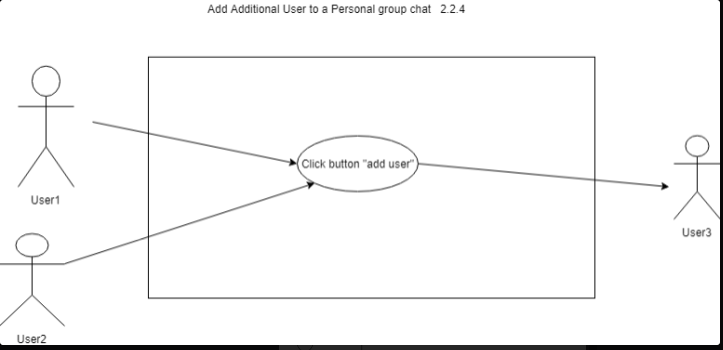
2.2.2 Add Friend



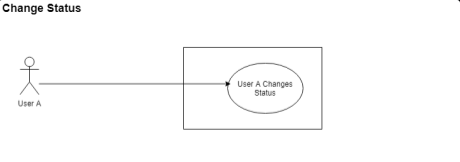
2.2.3 Message, Call, Video Chat

****

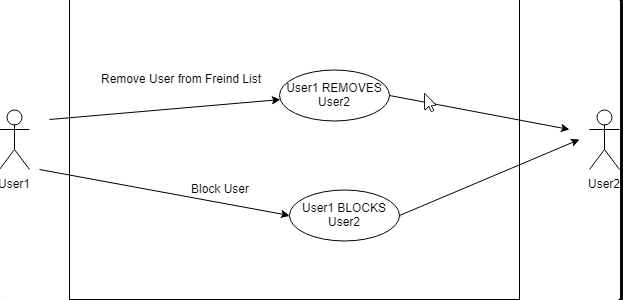
2.2.4



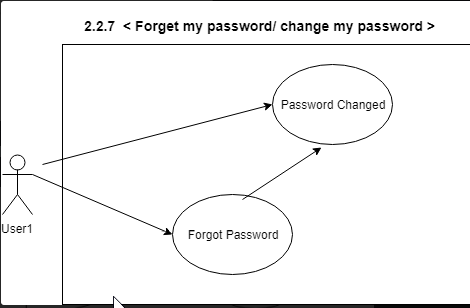
2.5 Change status



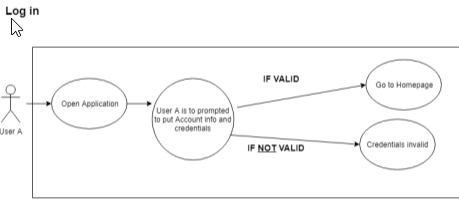
2.2.6 Block Friend/ Remove Friend

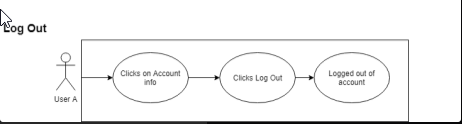


2.2.7 Password Change / Forgot Password



2.2.8 Log-in/Log out





## Use Case Lists

|  |  |  |
| --- | --- | --- |
| **Use Case** | | |
| **Sequence Number** | **Actor** | **Goal** |
| 1 | User | Call/message/video Chat Another User (Friend | Lobby Acquaintance) X |
| 2 | User | Add Another user to Call/message group/ video chat X |
| 3 | User | Forgot or change password X |
| 4 | New User | Register Account X |
| 5 | User | Add Friend X |
| 6 | User | Change Status X |
| 7 | User | Block User/Remove Friend X |
| 8 | User | Login/logout X |

### <Create New Account>

Primary Actor: New User

Secondary Actors(s):

Goal in Context: Create a new account login for MyMajekServer

Preconditions: User account information must be unique and not in the database

Additional Description:

#### <**Create new account using New User Button** >

* User Opens client and clicks on ‘Create New Account’
* UI asks user for:
  + Email[string]
  + Username[string]
  + Password[string]
  + FirstName[string]
  + Lastname[String]
* Server checks username and email to confirm that it doesn’t match with anyone on file (If inputs are unique account is created)

<Scenario 1 . Credentials are valid>

User is Logged in

<Scenario 2. User Information taken>

User is Prompted Username is Taken.

* User is returned to login screen after account is created.

### <Add friend>

Primary Actor: User1

Secondary Actors(s): User2

Goal in Context: Connect two users by adding them to friends list

Preconditions: Both users must have valid account and must not currently be friends

Additional Description:

#### <User1 Sends Friend Request to User 2 >

* Part 1: send Request
  + While on home screen, User click on friends list
  + Clicks button to ‘Add friend’
  + User1 is required to input a username to add user2(User2 Username)
  + If correct username is entered, friend request is sent to user2
  + If incorrect username is entered, message” Invalid username entered” will be displayed”
  + User1 must wait on user2 reply to continue with process
* Part 2: User2 Reply
  + While on home screen, user2 will receive notification on friends list.
  + User2 click on friends list and see’s request from user1
  + If user2 accepts request, user1 is added to friends list
  + If user2 denies request, user1 is added to ignored list
    - Ignored list still allows for users to accept friend request in the future, but will not display a notification
  + Once user2 accepts or denies request, he is returned to the friends list.

### <Call/Message/Video Chat Friend>

Primary Actor: User1

Secondary Actors(s): User2

Goal in Context: Initiate or continue Call/Message/Video Chat

Preconditions: User 1 and User 2 must be friends

Additional Description:

#### <User1 messages User2 >

* User1 Clicks on User2 Profile from the friends list
* User1 Inputs a message into the dialogue box
* User1 Sends Message to conversation
  + If conversation doesn’t exist between both users, create a conversation

#### <User1 Calls User2 >

* User1 Clicks on User2 Profile from the friends list
* User1 Clicks on ‘Voice Call’ Button
  + Text-style message is sent along with dialogue to accept or deny call
* User2 can either accept or deny call Notification
  + If call is accepted, Text-Style message is sent stating call is accepted and voice chat links together
  + If call is denied or not answer, text-style message is sent stating user2 is busy
  + If user2 status is busy, call will automatically be ignored, and text-style message will be sent
  + If conversation doesn’t exist between both users, create a conversation

#### <User1 Video calls User2 >

* User1 Clicks on User2 Profile from the friends list
* User1 Clicks on ‘Video Call’ Button
  + Text-style message is sent along with dialogue to accept or deny call
* User2 can either accept or deny call Notification
  + If call is accepted, Text-Style message is sent stating call is accepted and Video and Voice chat links together
  + If call is denied or not answer, text-style message is sent stating user2 is busy
  + If user2 status is busy, call will automatically be ignored, and text-style message will be sent
  + If conversation doesn’t exist between both users, create a conversation

### <Add Additional User to a Personal Group Chat>

Primary Actor: User1

Secondary Actors(s): User2 and User 3

Goal in Context: Connect 3 or more users by adding them

Preconditions: User1 Must have User 2 and User 3 Added to Friends list, Existing Chat between User 1 and User2

Additional Description:

#### <Adding User 3 To chat >

* Once a chat has been started, User1 Will Click on the “Add User” Button to existing conversation
* User1 will Select from friends list to add to Text chat
* New Conversation is generated between User1, User2, and User3 and acts like a regular conversation
  + Conversation information is unique and does not contain logs from previous conversation.

### <Change Status>

Primary Actor: User1

Secondary Actors(s):

Goal in Context: Change Status to reflect Availability

Preconditions: User 1 must have a valid account

Additional Description:

#### <Change User Status >

* User Clicks on the status bubble on his profile Icon
* Will be given options to pick from:
  + Available - Open for all users (Can customize to allow certain users to wait for an acceptation to the lobby)
  + Away – User is idle, but still available to host lobbies
  + Busy – User is in another Lobby or has closed their lobby
  + Offline - User is offline and cannot host lobbies.

### <Block/Remove Friend>

Primary Actor: User1

Secondary Actors(s): User2

Goal in Context: Remove or Block Users

Preconditions: User2 must be a valid user

Additional Description:

#### <Remove User from Friends list >

* User 1 open Friends list
* User 1 Click on User2 Profile
* User1 Click “Remove Friend”, Button
  + User2 will not be notified of friend removal
* User1 is return to friends list

#### <Block a User>

* User 1 Opens friends list
* User click on Block user button
  + User1 inputs User2 Username
* User1 is prompted to confirm block
* If they accept, User2 is added to block list and User1 is returned to Friends list

### < Forget my password/ change my password >

Primary Actor: User

Secondary Actors(s): None

Goal in Context: Change User login information

Preconditions: User Must be Valid Account, User must have access to email.

Additional Description:

#### < Forgot Password>

* On Home screen, User Click on Forgot Password
  + User is Prompted to Enter the Account Username and Email
* If Email is Valid, Sent Temporary Password to User
  + Password will be randomly generated, and user is responsible for changing it Later

#### < Change Password >

* On Home Screen, User click on his profile
* User will then click on the Change password Option
  + User is Prompted to Enter Old Password
  + User is prompted to Enter new Password Twice (Confirmation Step)
* Email is sent to user email informing proper password change.

### < Log out /Log in >

Primary Actor: User

Secondary Actors(s): None

Goal in Context: User Log in/Log out of MyMajeks Server

Preconditions: User account must be valid

Additional Description:

#### < Log in >

* User Opens Application
* User is prompted to put in Account information
* If Valid, Users is directed to Homepage and Status is Changed to Online.
  + User is prompted that Login Credentials are Invalid

#### <Log-out>

* User Click on Account Profile
* User Clicks “Log-Out” Button
* User is Logged out His profile and Status is Changed to Offline