|  |  |
| --- | --- |
| **Create an Account - Full Use Case Description** | |
| **Brief Description:** | |
| When a user  Wants to start using the chat system  They create an account So that they can continue into the application and use it. | |
| **Trigger:** | |
| A user wants to create an account. | |
| **Actors:** | |
| 1. **User**: A user on the client side application, who wants to create an account. 2. **System**: The client side application. | |
| **Pre-Conditions:** | |
| The user has downloaded the client side application. The user has internet connection. | |
| **Post-Conditions:** | |
| The user will have an account, whose information will be stored on the server. The user will be able to log into their account using the information they provided. | |
| **Normal Flow** | |
| User | System |
| 1. User selects “Create New Account”. | 1. System presents “Create Account” screen. |
| 1. User enters their full name, email address, and username. | 1. System validates format of inputs, and that email address and user handle/id is new. |
| 1. User selects “Create Account”. | 1. System creates the account, and notifies user of success. |
| **Alternate Flows:** | |
| **Invalid Input** If at step 4, any of the input is invalid, then: | |
| User | System |
|  | 4.1 System notifies user of incorrect input. |
| The flow resumes from step 3 of the normal flow. | |
| **Exceptional Flows:** | |
| **User quits process** If at any point during the process, the user quits the process, then their account is not created. | |
| **Key Scenarios:** | |
| 1. **Regular successful account creation** The user enters all the correct data and their account is created. 2. **Invalid input** The user enters some incorrect data, but eventually they fix it and their account is successfully created. 3. **User quits process** The user quits the process at some point before the account has been created, therefore the account is not created. | |

|  |  |
| --- | --- |
| **Add User To Contacts [Request Contact Add] - Full Use Case Description** | |
| **Brief Description:** | |
| When a user Wants to add another user as a contact  They search for that user and request to add them as a contact  So that they can easily find and interact with them in the system. | |
| **Trigger** | |
| A user wants to add another user as a contact. | |
| **Actors** | |
| **Requester**: The user requesting the contact-add. **Receiver**: The user receiving the contact-add request. **System**: The Requester’s client side application. | |
| **Pre-Conditions:** | |
| The Requester is connected to the internet. Both Requester and Receiver have accounts. | |
| **Post-Conditions:** | |
| A contact-add request will have been sent from the Requester to the Receiver. | |
| **Normal Flow** | |
| Requester | System |
| 1. User selects “Add Contact”. | 1. System requests search criteria to find a user. |
| 1. User enters the user handle/id or email address of the user they wish to add. | 1. System displays the user that matched this search criteria. |
| 1. User selects “add user” on the presented user. | 1. System notifies user that a contact-add request has been sent to the specified user. |
| **Alternate Flows:** | |
| **No Match For Search Criteria** If at step 4 there is no match for the search criteria or the search criteria is invalid, then: | |
| Requester | System |
|  | 4.1. System notifies user of problem (no match or invalid search criteria) |
| The flow resumes from step 3 of the normal flow. | |
| **Exceptional Flows:** | |
| NONE | |
| **Key Scenarios:** | |
| 1. **Request Contact Add** Requester successfully requests a contact-add with Receiver. 2. **No Match For Search Criteria** There is no match for the search criteria or the search criteria is invalid. The Requester is alerted and tries again. | |

|  |  |
| --- | --- |
| **Add User To Contacts [Accept Contact Add] - Full Use Case Description** | |
| **Brief Description:** | |
| When a user receives a contact-add request, and they Want to become contacts with the requester They select the contact-add request, and accept it So that they can easily find and interact with the requester in the system. | |
| **Trigger:** | |
| Another user requests to add this user as a contact, generating a contact-add request on this users client application. | |
| **Actors:** | |
| **Requester**: The user requesting the contact-add. **Receiver**: The user receiving the contact-add request. **System**: The Receiver’s client side application. | |
| **Pre-Conditions:** | |
| The Receiver is connected to the internet. Both Requester and Receiver have accounts. | |
| **Post-Conditions:** | |
| The Receiver and Requester will now be contacts. | |
| **Normal Flow:** | |
| Receiver | System |
|  | 1. System notifies user that they have a contact-add request. |
| 1. User selects this contact-add request. | 1. System displays the contact-add request. |
| 1. User selects “Accept” on this contact-add request. | 1. System notifies user that they have are now contacts with this user. |
| **Alternate Flows:** | |
| **Decline Contact Add** When at step 4. the Receiver selects “Decline” rather than “Accept”, the contact request is cancelled, and the Receiver is notified that they successfully declined the contact-add request. | |
| **Exceptional Flows:** | |
| NONE | |
| **Key Scenarios:** | |
| 1. **Accept Contact Add** Receiver accepts the contact-add request, and the two users successfully become contacts. 2. **Decline Contact Add** Receiver declines the contact-add request, and the request is cancelled. | |

|  |  |
| --- | --- |
| **View Contacts - Full Use Case Description** | |
| **Brief Description:** | |
| When a user  Wants to view their contacts They select “View Contacts” So that they can view their contacts and maybe start a chat with one of them. | |
| **Trigger:** | |
| A user wants to view their contacts. | |
| **Actors:** | |
| 1. **User**: A user on the client side application, who wants to view their contacts. 2. **System**: The client side application. | |
| **Pre-Conditions:** | |
| The user has downloaded the client side application. The user has internet connection. The user has an account. | |
| **Post-Conditions:** | |
| The user is viewing/viewed their contacts. | |
| **Normal Flow** | |
| User | System |
| 1. User selects “Contacts”. | 1. System displays all of the users contacts. |
| **Alternate Flows:** | |
| **User Has No Contacts** If at step 1, the user has no contacts, then the system notifies the user that they do not have any contacts to view, and the use case is terminated. | |
| **Exceptional Flows:** | |
| NONE | |
| **Key Scenarios:** | |
| 1. **View Contacts** The user successfully views their contacts. 2. **User Has No Contacts** The user has no contacts to view, so they are alerted, and the use case is terminated. | |

|  |  |
| --- | --- |
| **View Chats [Via Chats] - Full Use Case Description** | |
| **Brief Description:** | |
| When a user  Wants to view their chats They select “Chats” to view all their chats, and then select a specific chat to view it So that they can see and participate in their chats. | |
| **Trigger:** | |
| A user wants to see and/or participate in their chats.. | |
| **Actors:** | |
| 1. **User**: A user on the client side application, who wants to view their chats.. 2. **System**: The client side application. | |
| **Pre-Conditions:** | |
| The user has downloaded the client side application. The user has internet connection. The user has an account. | |
| **Post-Conditions:** | |
| The user will see/have seen their chats. | |
| **Normal Flow** | |
| User | System |
| 1. User selects “Chats” | 1. System displays users chats. |
| 1. User selects a specific chat to see. | 1. System displays the specific chat selected. |
| **Alternate Flows:** | |
| **User Doesn't Want To See A Specific Chat** If at step 3, the user does not want to see a specific chat, then they do something else with the system, and the use case ends. | |
| **User Has No Chats** If at step 2, the user does not have any chats, then the system alerts them that they have no chats to display, and the use case is terminated. | |
| **Exceptional Flows:** | |
| NONE | |
| **Key Scenarios:** | |
| 1. **View Chats** The user successfully views their chats. 2. **User Doesn’t Want To See A Specific Chat** The user exits the use case after viewing all their chats. 3. **User Has No Chats** The user has no chats to display, so they are alerted, and the use case is terminated. | |

|  |  |
| --- | --- |
| **View Chats [Via Contacts] - Full Use Case Description** | |
| **Brief Description:** | |
| When a user  Wants to view/interact with a chat from a specific contact They open this existing chat So that they can view/interact with the chat | |
| **Trigger:** | |
| A user wants to view/interact with a pre existing chat with a contact | |
| **Actors:** | |
| 1. **User**: A user on the client side application, who wants to view/interact with a chat. 2. **Contact**: Another account, that is associated with the chat that the user wants to view. 3. **Client**:: The client side application. | |
| **Pre-Conditions:** | |
| The user has downloaded the client side application. The user has internet connection. The user has an account. The user has the user they want to chat with as a contact. The user has a pre existing chat with the contact. | |
| **Post-Conditions:** | |
| The user will be viewing/interacting with the chat. | |
| **Normal Flow** | |
| User | Client |
| 1. User completes “View Contacts” use case, and selects “Chat” on a specific contact. | 1. System displays the chat with that contact, so that the user can view and interact with it.. |
| **Alternate Flows:** | |
| **There Is No Pre-Existing Chat Between The User And The Contact** If at step 2, the client has trouble connecting to the server, then the client alerts the user and tries to connect again.  The flow resumes from step 2 of the normal flow. | |
|
| **Exceptional Flows:** | |
| **No Response From Server** If at step 4, the client does not receive a response from the server for an extended period of time, the client alerts the user, and the use case is terminated. | |
| **Key Scenarios:** | |
| 1. **Create New Chat** The user successfully creates a new chat with the contact. 2. **Issue Connecting To Server** The client has an issue connecting to the server, and continues to try until the user cancels. 3. **No Response From Server** The client does not get a response from the server, so it terminates the use case. | |

**SUB USE CASES**

|  |  |  |
| --- | --- | --- |
| **Send Message - Full Sub-Use Case Description** | | |
| **Brief Description:** | | |
| When a user  Wants to send a message to one of their contacts They open an existing chat or create a new chat with the contact, type a message, and press send So that they can send a message to a specific contact | | |
| **Trigger:** | | |
| A user wants to send a message to one of their contacts | | |
| **Actors:** | | |
| 1. **User**: A user on the client side application, who wants to send a message to one of their contacts. 2. **Contact**: The contact that the message is being sent to. 3. **Client**: The client side application. 4. **Server**: The server that the client side application is talking to. | | |
| **Pre-Conditions:** | | |
| The user has downloaded the client side application. The user has internet connection. The user has an account. The user has logged in. The user has completed either of the “View Chats” use case, and is currently viewing a specific chat. | | |
| **Post-Conditions:** | | |
| User will have sent a message to Contact. | | |
| **Normal Flow** | | |
| User | Client | Server |
| 1. Type a message and press send. | 1. Display the message with the status ‘sending’, and send the message to the server. | 1. Add the message to the chat, and notify the client that the message was successfully received. |
|  | 1. Change the status of the message to ‘sent’. |  |
| **Alternate Flows:** | | |
| **No Connection To The Server When Sending Message** If at step 2, the client cannot connect to the server, then the client will alert the user and continue to trying to connect to the server.  When a connection is established, the flow resumes from step 2 of the normal flow. | | |
|
| **No Alert Received From Server** If at step 4, the client is not notified from the server that the message was successfully received, then the message status stays as ‘sending’ until the client does hear back from the server. | | |
| **Exceptional Flows:** | | |
| NONE | | |
| **Key Scenarios:** | | |
| 1. **Send Message** The user successfully sends a message to the Contact. 2. **No Connection To The Server When Sending Message** The client failed to connect to the server when sending the message, and must continue to try to reconnect before continuing. 3. **No Alert Received From Server** The client has not heard back from the server, so the message status remains as ‘sending’ until the client does hear back from the server. | | |

|  |  |
| --- | --- |
| **Receive Message - Full Sub-Use Case Description** | |
| **Brief Description:** | |
| When a user  Receives a message They receive a notification So that they will know and can read the message | |
| **Trigger:** | |
| The server sends a message to the client. | |
| **Actors:** | |
| 1. **User**: A user on the client side application, who is receiving a message 2. **Client**: The client side application. 3. **Server**: The server that the client talks to, and who sent the message. | |
| **Pre-Conditions:** | |
| The user has downloaded the client side application. The user has internet connection. The user has an account. The user has logged in. | |
| **Post-Conditions:** | |
| The user will have received a notification that they have a new message, and they will be able to view this message. | |
| **Normal Flow** | |
| User | Client |
|  | 1. The client adds the new message to the corresponding chat, and sets a notification. |
| 1. The user sees the notification, and can view the message. |  |
| **Alternate Flows:** | |
| **User Has The Relevant Chat Open** If at step 1, the user already has the relevant chat open, then the new message will be appended, but the notification will be skipped. And the flow will resume from step 4 of the normal flow. | |
| **Exceptional Flows:** | |
| NONE | |
| **Key Scenarios:** | |
| 1. **Receive Message** The user successfully received the new message, and a notification of this. 2. **User Has The Relevant Chat Open** The user successfully received the new message, but did not receive a notification as they were already in the relevant chat. | |

|  |  |
| --- | --- |
| **[UNCOMPLETE] Notification - Full Sub-Use Case Description** | |
| **Brief Description:** | |
| When a user  Wants to  They  So that | |
| **Trigger:** | |
|  | |
| **Actors:** | |
| 1. **User**: A user on the client side application, who wants to x.. 2. **System**: The client side application. | |
| **Pre-Conditions:** | |
| The user has downloaded the client side application. The user has internet connection. | |
| **Post-Conditions:** | |
|  | |
| **Normal Flow** | |
| User | System |
|  |  |
|  |  |
|  |  |
| **Alternate Flows:** | |
| **One** If at step x, xxxxxx, then: | |
| User | System |
|  | x.1 System |
| The flow resumes from step x of the normal flow. | |
| **Exceptional Flows:** | |
| **One** If x, then x. | |
| **Key Scenarios:** | |
| 1. **One** one 2. **Two** two. | |

|  |  |  |
| --- | --- | --- |
| **Create Chat - Full Sub-Use Case Description** | | |
| **Brief Description:** | | |
| When a user  Wants to chat with one of their contacts that they don't already have a chat with They find the contact they want to chat with and select “Chat” So that they can start a chat with the contact | | |
| **Trigger:** | | |
| A user wants to start a chat with one of their contacts. | | |
| **Actors:** | | |
| 1. **User**: A user on the client side application, who wants to start a chat with one of their contacts. 2. **Contact**: Another account, that the user wants to start a chat with. 3. **Client**:: The client side application. 4. **Server**: The server that the client talks to. | | |
| **Pre-Conditions:** | | |
| The user has downloaded the client side application. The user has internet connection. The user has an account. The user has the user they want to chat with as a contact. | | |
| **Post-Conditions:** | | |
| A chat will have been created between the user and the contact. | | |
| **Normal Flow** | | |
| User | Client | Server |
| 1. User completes “View Contacts” use case, and selects “Chat” on a specific contact. | 1. System sends a message to the server to create a chat with the user and contact. | 1. .Server creates a chat with the two given accounts. Sends the chat data to both clients either now or when they come online next. |
|  | 1. Displays the chat to the user to interact with. |  |
| **Alternate Flows:** | | |
| **Issue Connecting To Server** If at step 2, the client has trouble connecting to the server, then the client alerts the user and tries to connect again.  The flow resumes from step 2 of the normal flow. | | |
|
| **Exceptional Flows:** | | |
| **No Response From Server** If at step 4, the client does not receive a response from the server for an extended period of time, the client alerts the user, and the use case is terminated. | | |
| **Key Scenarios:** | | |
| 1. **Create New Chat** The user successfully creates a new chat with the contact. 2. **Issue Connecting To Server** The client has an issue connecting to the server, and continues to try until the user cancels. 3. **No Response From Server** The client does not get a response from the server, so it terminates the use case. | | |

|  |  |  |
| --- | --- | --- |
| **Log In - Full Sub-Use Case Description** | | |
| **Brief Description:** | | |
| When a user  Wants to log in They enter their details and press log in So that they can use the system | | |
| **Trigger:** | | |
| A user wants to log in. | | |
| **Actors:** | | |
| 1. **User**: A user on the client side application, who wants to log in. 2. **Client**: The client side application. 3. **Server**: The server that the client talks to. | | |
| **Pre-Conditions:** | | |
| The user has downloaded the client side application. The user has internet connection.  The user has an account. | | |
| **Post-Conditions:** | | |
| The user will be logged in. | | |
| **Normal Flow** | | |
| User | Client | Server |
| 1. User enters email and password, then presses “Log In”. | 1. Client validates that input is formatted correctly, then sends it off to the server. | 1. Server validates that the input is correct, logs the user in, and alerts the client. |
|  | 1. Client notifies the user that they have successfully logged in. |  |
| **Alternate Flows:** | | |
| **Invalid Input Format** If at step 2, the client recognises that the format of the input is incorrect, then it notifies the user.  The flow resumes from step 1 of the normal flow. | | |
| **No Matching Details** If at step 3, the server fails to find an account that matches the input, then it notifies the client, which notifies the user.  The flow resumes from step 1 of the normal flow. | | |
| **No Connection To Server** If at step 2, the client cannot connect to the server, then the client notifies the user, and continues to try to connect. When a connection is established, the flow resumes from step 2 of the normal flow. | | |
| **No Response From Server** If at step 4, the client has not heard back from the server, then it notifies the user, and continues waiting. When a response is received from the server, the flow resumes from step 4 of the normal flow. | | |
| **Exceptional Flows:** | | |
| NONE | | |
| **Key Scenarios:** | | |
| 1. **Log In** The user successfully logs in. 2. **Invalid Input Format** The user entered an invalid format for some of the input, and must try again. 3. **No Matching Details** There is no account with the given details, the user must try again. 4. **No Connection To Server** The client could not connect to the server to send it the log in data, so it notifies the user and continues trying to connect. 5. **No Response From Server** The client has not heard back from the server for an extended period of time, so it notifies the user and continues waiting. | | |

|  |  |  |
| --- | --- | --- |
| **Log Out - Full Sub-Use Case Description** | | |
| **Brief Description:** | | |
| When a user  Wants to log out They select log out So that they will no longer be logged into the system | | |
| **Trigger:** | | |
| A user wants to log out. | | |
| **Actors:** | | |
| 1. **User**: A user on the client side application, who wants to log out. 2. **Client**: The client side application. 3. **Server**: The server that the client is talking to. | | |
| **Pre-Conditions:** | | |
| The user has downloaded the client side application. The user has internet connection. The user has an account. The user is logged into their account. | | |
| **Post-Conditions:** | | |
| The user will not be logged into the system on the device used. | | |
| **Normal Flow** | | |
| User | Client | Server |
| 1. Selects “Log Out”. | 1. Ends the connection to the server. And notifies the user that they have successfully logged out. |  |
| **Alternate Flows:** | | |
| NONE | | |
| **Exceptional Flows:** | | |
| NONE | | |
| **Key Scenarios:** | | |
| 1. **Log Out** The user successfully logged out. | | |