|  |  |  |
| --- | --- | --- |
| **Use case name:** | User requests random chat | |
| **Scenario:** | A user initiates a request for a random chat | |
| **Triggering event:** | User requests a random chat | |
| **Brief description:** | A registered user requests to participate in a random anonymous chat. | |
| **Actors:** | User | |
| **Related use cases:** | User Logs In | |
| **Stakeholders:** | User | |
| **Preconditions:** | User must be registered with an account  Another user must also be available for a chat | |
| **Postconditions:** | Available spots in class is reduced by one  Member account schedule is updated | |
| **Flow of activities:** | **Actor**   1. User selects to be entered into a random one on one chat 2. User is entered into chat | **System**   1. System takes the user ID and area and randomly selects another user   1.2 System joins the 2 users |
| **Exception conditions:** | 1.1 No match is found for user or area | |

The following diagram is the use case diagram for the user request a random chat user case. The actor in this diagram is the user and the use case “Request Chat” is associated with the user. This use case also includes the “User logs in” use case as a user would need to be logged in to start a chat.



The next diagram is the activity diagram for the request random chat use case. This begins with the user requesting to be entered into a random chat. The system would then need to find a user that is also requesting a chat and within the appropriate area. If a user is not found then the user would be entered into a queue to wait for a chat. The user would be asked if they want to wait. If they choose no then the activity is completed. If they choose yes then the user’s state is maintained until a chat becomes available. If a User is found then the two people are connected and a chat screen is displayed.



The next diagram is the system sequence diagram for requesting a random chat. The first input to the system is a request for a random chat with the user ID passed to the system. A Boolean of whether a user was found or not is then passed back to the user. The user is then entered into the chat and the user’s username is passed to the system. The message ID and the username is then passed back from the system.

