
CHAT ROOM SERVER DESIGN

Lab 1

Sarah Alzahrani

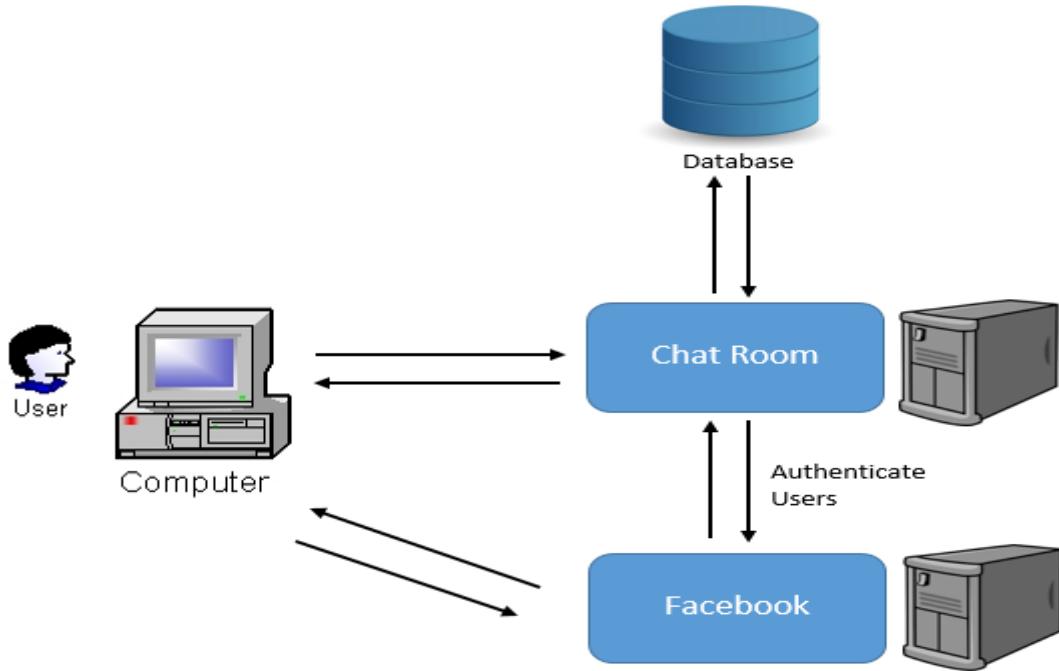
14331356

2014-2015

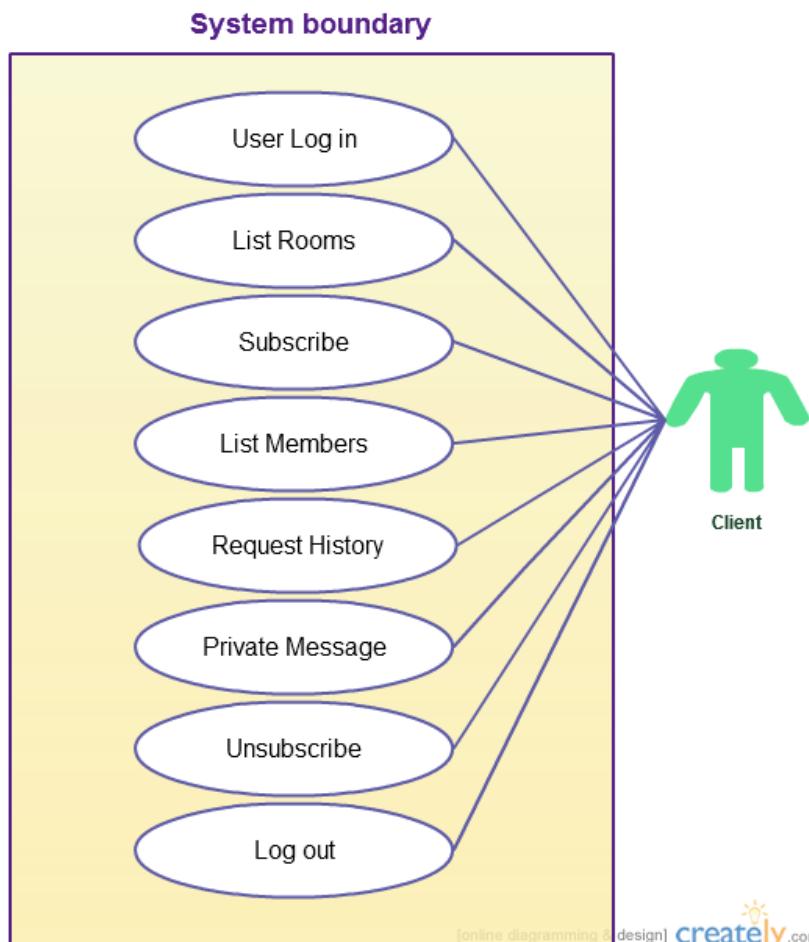
NETWORK APPLICATION II

Trinity College Dublin

Architecture:

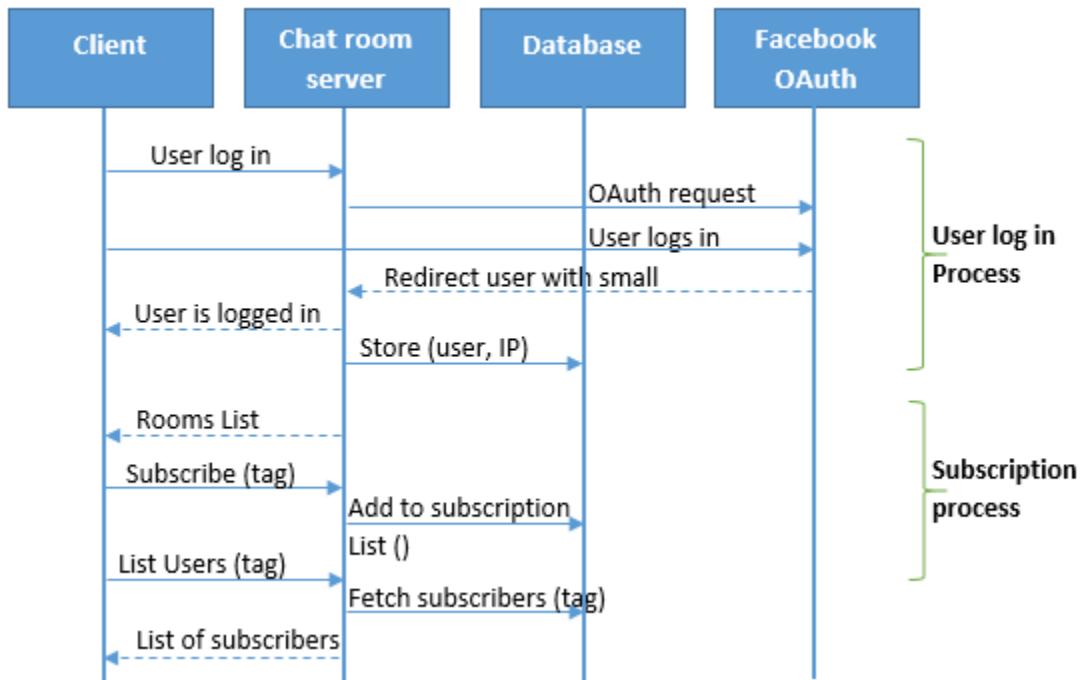


System boundary:



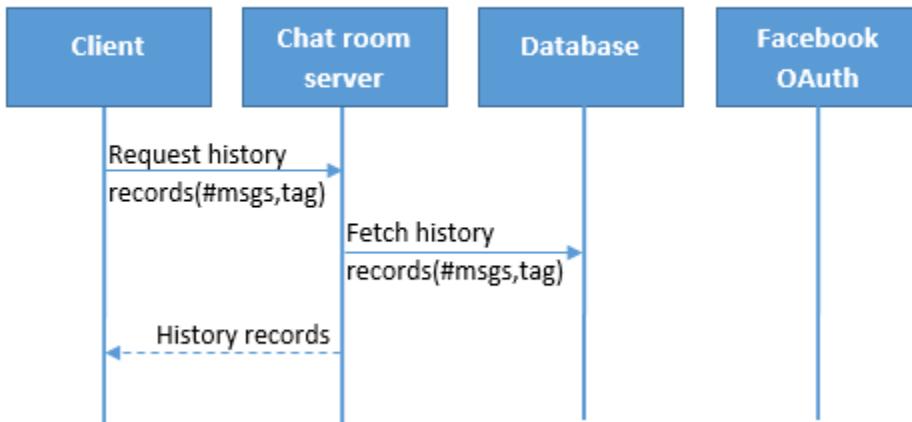
Group Messaging

Chatrooms, Client state:



- After user log in, the server should list all distinct rooms alternately users would use List Rooms command to list them.
- It has been agreed that the server would use “Tag model” to distinguish client messages destined for different rooms.
- In this case user would be able to join specific room by using subscribe command along with room’s tag.
- After the subscription process users would be able to list all members in that room by using command List users.
- Server should has a subscriptions lists for all chat rooms that contain members’ identities.
- If user stop sending messages for specific time (specifies by server) Server should send a message to the user to check if he or she still there or not. In this case server should maintain last seen activity that has been done by the user.

Latecomers:



- It has been decided that the server should maintain history tables for all chat rooms (for example: for the last 24 hour).
- In this case user would be able to request history log over specific period of time by using this command “History {number of messages} {chatroom tag}”.

Private Messaging

- In order to support private messages It has been agreed that these messages will be supported by using only one tag (user identity) for example : if message is being sent to Sarah the. Sara is the tag name in this case. (@Sarah: hello Sarah)
- *assumption: users identities are unique.

Identity

- It has been agreed that the server would use Facebook oAuth as a third party authentication service provider.
- The Server also has a table or file which contains all registered users' identities along with their current IP addresses, if the IP address field is null this means user is offline.