Princess Kim

CS300 Xie Fei

CS300 MobileNet Project Write Up

In the end, MobileNet became only 80% of what I wanted it to be. Many accommodations needed to be made because of the lack of knowledge of Android Studio I held, and the limitations of Android Studio and Quickblox. The biggest disappointment was not being able to differentiate online and offline users. The biggest fault of mine was not being prepared for this (AKA not looking up the limitations of certain sites), rather I looked through different web servers to host MobileNet’s database on, and picked the one with the most useful functions and services instead of comparing the downfalls of each. I had assumed a database server meant for user registration would be able to provide a way for users to have an online and offline status, and it wasn’t until I was 70% done did I run into this problem. This is my first time programming with so many libraries that I had to look up and learn about, before using. As for Android Studio, the available UI service it provided was very appealing, and after watching a few tutorials, I decided to start MobileNet. Unfortunately, The UI started crashing and breaking instantly. Things were not aligning correctly and the process that once seemed so easy, seemed so unintuitive all of a sudden. I overcame this by learning the xml code and hand coded each button, image, and text onto each page of the app.

As for the functionalities of the project, there were two that I could not make work. Getting users online/offline status, and creating an “all chat”. However, as a way to work around not being able to create an all chat, I decided to just make a group chat. Users can click on the user list (all existing users, rather than online users) and can choose from 1 to many users. Choosing no users and clicking on the “create new chat” button results in a creation of an “all chat” but since it is not implemented, sending or receiving messages is impossible. As for the others, in both private and group chats, sending and receiving works just fine. Messages are sent in real time, and if a new chat is created with yourself in it, a box will pop up with the users inside that chat (excluding your name). The other more important feature that I was able to get to work was the signing in and registering an account. Accounts are stored within Quickblox, where I had to make an account and become an admin of my own “app”. Information is stored onto its database, and the code within Android Studio accesses the database to ensure that users login correctly, and even register an account with valid information. I was however unable to develop a way to add a Battlenet ID into the app.

Extra features I wanted to add into MobileNet that thought were essential to Instant messaging programs were: Edit message, delete message, delete chat, and updating user account. Updating the user account was an idea when I had created multiple accounts with long passwords. I wanted a way to change the password without having to delete the account on Quickblox and re-registering with the same name. I figured it would be a standard function for most apps, so once developed, I added a pseudo Log-Out button, which resulted in me finding out that grabbing a user’s online or offline status was impossible with Quickblox. Apparently, the closest you can get is to grab the last time a user logged in (not log out) and compare it to the current time. This is not the same as “online” or “offline” and so instead of trying to make a fake status, I decided to scratch that out and add other features to accommodate the lack of status.