Tinder Clone: Client Meeting Notes

7/21/21

- System may run on a LAN – no external internet access/interaction is required.
- Plan already in place for remaining two group members to absorb missing member's work
- Will finish implementation within the next two days and begin testing using the current version of the test plan specification

7/07/21

- Include class diagrams for our design
- Schedule document should clearly show what each team member is working on
- Design should be finalized before our presentation

6/28/21

- Use multi-thread sockets for server to be able to authenticate multiple user requests
- Message Passing, how will a user request get passed to the server and back to the user
- What is client based and what is server based
- For the login class an authentication message will be displayed for the user, should have a maximum 3 authentication messages in our program
- Save all client requests to server in a queue

6/23/21

- More class association and how all of these various classes are able to interact with each other (diagrams)
- For the next design check in, design has to be ready to implement

6/21/21

- Contain more notes in client meetings
- All revisions of requirements document should be saved in GitHub
- Is using the MatchDB feasible for tracking user choices. Yes as long as the DB is not in sequel or some other program (will be in a text file).
- Have 3 modules Login Module, Match Module, DataStore Module. Login module will handle the username, password, and userID, along with the profile info. The match module will keep track of user swipes, and match if both users have swiped right. The

datastore module will contain the data entries and keep a list of all the databases, (UserDB, MatchDB, and ProfileDB)

- Do we need to delete account/attributes/ profile pictures? Not needed can get complicated
- Keep track of all Revisions to Requirements Document. We have multiple versions 1.1, is our first then 2.1

6/7/21 and 6/14/21

- Not web based
- Client- server based
- Server application with many clients application
- Client must be a separate executable, clients connect to server
- Simulate (fake GPS) GPS location, randomly generated, use coordinates to determine a location, show distance between cities.
- Track user choices (A, B, C, D, E) see if users are going to have higher probabilities in certain groups. Or wait for all groups to swipe and match when they both swipe right (Will match users when they both swipe right)
- Every time a user logins location will be randomized different location every login
- Will support multiple clients in a typical server and client configuration
- Focus mostly on User profile
- Emulate functionality of Tinder