

## Hollo Scenario II

Group 7 - Mateusz Wszolek, Darius Skiedra, Shreyas Gaonkar, Heeba Mohammed

The Hollo application will eventually become a cross platform application where users can communicate with each other via several means. Users will be able to send text based messages to each other and the text will be stored for future or until the user manually deletes. Users will also be able to make calls to one another in real time, in addition to video chat with each other, and use holograms interactions; these features are planned for future iterations beyond this scenario.

Currently we have semi functioning application where the user is able to create a new account using a custom, temporary command line program. The user is also able to log in to their account and receive a confirmation whether it was successful or not. We also have a graphical user interface ready for the PC/Mac OS X/Linux. We have a basic mobile client with a graphical user interface (GUI) capable of sending basic C strings to a remote server connection.

### \* Adding/Deleting Friends Scenario:

The user will be able to add a friend if he or she knows the friend's user name. The application will inform a user when the request has been accepted by the other user via on screen notification. New friends will show up in the friends list and can be selected to send messages instantaneously. Receiving user also has an option to refuse a pending friend request as well as delete current friends individually. To delete a friend from a friend list, a user selects a friend in the friends list and right clicks or selects options on it and can then choose to delete. The deleted user will also have the other user removed from their friends list.

### \* Communication with Friends Scenario:

When a user wants to send a message to another user who already is on their contact list ("friend"), they need to select them and choose send message option. A new window interface will appear where a user can type out their message to be sent to their friend. If a user has contacted this user previously, last couple messages will be displayed in the interface. When the user is sending the message to another user, they must be available and online, the other client will not receive the message. When the other user does login, he will then receive the message.

\* These two were already being done in Demo-1. Though there were errors. What's new?

→ wasn't it just a command line screen?

otherwise

Previously, it was mentioned that user can change their status. Ho

A special type of communication is sending a location to another user. A user on a mobile device can choose to send ~~their~~ <sup>his/her</sup> location to another mobile user. Users can send their Location on request via integration with Google Maps Application Program Interface (API).

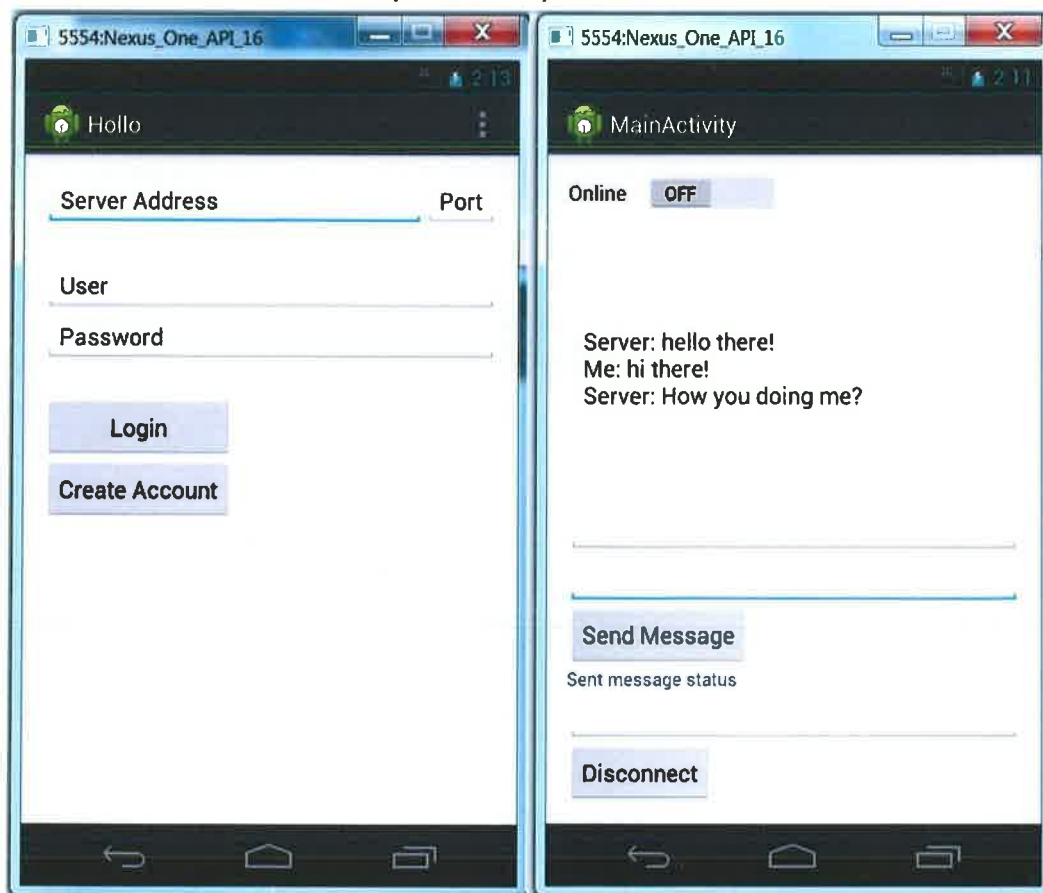
### Usability Scenario:

Users will be able to use the application on multiple platforms. Starting supported platforms include: Android OS, various Linux based operating systems, Mac OS X, and Windows. The user experience will be uniform across all platforms.

### Changing Status Scenario:

A user will have an option to select their status. Statuses to choose from are: Available, Away, Busy or Offline. When a user changes their status, their friends will receive the status update.

### Graphical User interface (test build):



Note Adding/Deleting & communication & status changing was mentioned in Scenario-1, Although not implemented properly in demo. I fail to see a lot of new features in this scenario. So marks are low.