# Further Development

### Server:

* Use of Streaming State Transfer protocol in JGroups/spooling bytes from file when transferring between servers to avoid OutOfMemoryExceptions in Java when handling extremely large files.
* Persistent storage for messages, usernames and associated GCM IDs, etc.
* Handle GCM IDs on device being renewed. Associated information is in the Google Cloud Messaging documentation. On the server, you would need to reassign the username to the new GCM ID.
* Further testing with a larger number of servers to test for overhead of message ordering.
* Usage of the TCP protocol to simulate multicasting instead of UDP to allow server cluster to be formed outside of a LAN.
* Change ChatServerProtocol to be either Singleton or static; no need for more than one instance.
* Change PushNotification to send to particular list of connected devices rather than every single device stored that is currently connected.

### Client-Server Communications

* Security - Both a login system and secure sockets (SSL).
* Consider a more established format for serialization for objects passed between server and client e.g. JSON. Useful if client platforms were expanded beyond Android.
* Be able to request that a message only be sent to a particular subset of online clients rather than be sent to all clients on the cluster.
* Needs a bug fix for when all servers are taken offline but the phone client is not; any messages will remain on the phone and the message retrieval mechanism will not work until the number of messages on the server exceeds the number of messages on the phone.

### Client

* UI/Layout Improvements
* Persistent data storage of messages, recorded audio files (Recorded audio files are just deleted at the moment in the onDestroy() event, so this could be easily changed).
* Expansion beyond audio, perhaps video files.
* More usage delivery/message ordering guarantee aspect.
* DownloadMediaPlayer and StreamMediaPlayer are extremely similar and could be refactored to subclass a common class, which then subclasses PlayAudioBase.