IMPORTANT DATES

- December 1
  - Project 2 submission.
PROJECT OBJECTIVE

- Implement a chat client-server application.
BASIC FEATURES

- Each user will have a username and a password.
- Users can create and join a chat-room in the server.
- A user should be able to send messages to all users or a selected set of one or more users in the same chat-room.
- There can be more than one virtual chat-room within the server. However, a user can be in a single chat room at a time.
- Plain text messages spanning one or more lines can be sent.
BASIC FEATURES

- The server should provide a list of current users and current chat-rooms.

- The system should be reasonably scalable (as the number of users grows) and its response time should be low; this can be shown statistically, as well as with an argument based on the techniques you use.

- All messages should appear in the same order for all the clients.
IMPLEMENTATION

- **Server**
  - Any Technology

- **Client**
  - Android App

- **Communication Protocol (application layer)**
  - Client management.
  - Message management & delivering.

- **Friendly User Interface**
  - Help Information (By typing “/help” in the group message window).
  - Clean, understandable
MESSAGE ORDERING

- absolute ordering
- consistent ordering
- causal ordering
- first-in-first-out ordering
DEMOS

- We will use two laptops
- In machine 1
  - The server
  - Two Android emulators
- In machine 2
  - Two Android emulators
DEMOS

- Example scenarios:
  - Connect to the server and register
  - Join an existing channel
  - Send a massage to all
  - Send a message to a single client
  - Send a message to a group of clients
  - Quit from channel and create a channel
  - Delete channel
Select Server

- KeyboardFail
- QuakeNet
- Undernet
- EFNet
- Rizon
- Ustream
- IRC-Hispano
- DALnet
- ChatZona
- OFTC
- GameSurge
- IRCHighway
- SwiftIRC
- Freenode
Edit server

Changes will only take effect next time you connect to the server.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Server Name</td>
<td>server</td>
<td>This value can be anything you want, it's just a symbolic value to identify the server.</td>
</tr>
<tr>
<td>Host Name</td>
<td>192.168.2.1</td>
<td>This is the hostname or IP address of the server, normally this will be in the format irc.xxx.net.</td>
</tr>
<tr>
<td>Port Number</td>
<td>6667</td>
<td>Default value is 6667 unless server operator has chosen another value.</td>
</tr>
<tr>
<td>Username</td>
<td>queensu</td>
<td>This is the username you will be logged in with when connecting to this server.</td>
</tr>
<tr>
<td>Password</td>
<td></td>
<td>This is the password you will be identified with when connecting to the server. Only registered users need to use this.</td>
</tr>
<tr>
<td>User Hostname</td>
<td></td>
<td>This allows you to set the hostname that will be shown when people query your user.</td>
</tr>
<tr>
<td>User Realname</td>
<td></td>
<td>This allows you to set your real name (as opposed to your username) that will be shown when people query your user.</td>
</tr>
</tbody>
</table>

by Underdose at timestamp 1350687970

> Dark-Knight Breaking Bad Spoiler Lol at Beneke running foot getting caught in the rug falls and hits head in table, dead? if so, most lol death ever

> @GM is really a hard job

> General Motors?

> @game master

> the current one piece rp im doing has 9 players

Code: -b Dweagle!*@

> rp

> :p

> we're having fun so stfu
PROJECT REPORTS

▪ Introduction
  ▪ Objectives, functionalities implemented, and brief introduction to your implementation.
  ▪ Briefly discuss any APIs and libraries used in the project.

▪ Methodology
  ▪ Brief outline of protocol used to send/receive messages.
  ▪ How multiple clients are managed by a server?

▪ Analysis of solutions
  ▪ Advantages, disadvantages, and tradeoff.

▪ Conclusions
  ▪ Your learning points.
GROUPING & SUBMISSION

- Grouping
  - Keep your current group formation

- Submission
  - Same as project 1
  - Submit via email your compressed project files
SEVERAL WEBSITES MIGHT HELP YOU!

- **Books**
  - Elliotte Rusty Harold, *Java Network Programming*, O'Reilly;
  - Cay S. Horstmann, Gary Cornell, *Core Java2*, Prentice Hall.

- **Online**
  - [http://ta.cnci.org/basicirc/228-basic-java-socket-programming-creating-a-simple-irc](http://ta.cnci.org/basicirc/228-basic-java-socket-programming-creating-a-simple-irc)
Reading the Android Dev Guide

The Android Developer’s Guide

Android Basics

What is Android?
ANDROID

- **Application Fundamentals**

- **Framework Topics**
  - Services: [http://developer.android.com/guide/topics/fundamentals/services.html](http://developer.android.com/guide/topics/fundamentals/services.html)
A critical part of any development is testing and debugging. Especially for this assignment, you need to test your app with two emulator instances. The following two articles describe this well:

The Android Developer’s Guide also has great tutorials you can follow. These tutorials will teach you the basics. Please do the following two tutorials at least: