

Introduction

In this project you will be developing an Android app that acts as a client to a simple social network called Chit Chat. Chit Chat is a chronologically sorted stream of text messages from its users. Each message has a count of likes and dislikes.

Your Chit Chat client should be capable of displaying recent messages in chronological order, including their timestamps, content, and number of likes & dislikes. I have developed a simple Chit Chat server. You can connect to it to retrieve messages, post a message, and like or dislike a message. The server will respond to you in JSON. I have described the endpoints, but not the return values in this document — you'll need to do some sleuthing.

Each message on Chit Chat has a unique `_id` property associated with it. This is what you will use for distinguishing it from other posts and for incrementing the number of likes or dislikes. Your client should identify itself when posting messages to Chit Chat by a client name.

Your client should never display the same post twice in the feed (unless someone posted it twice). Chit Chat posts are only one level deep. There is no concept of “comments.” Your client should include a “refresh” button or support pull-to-refresh. It should allow the user to post new messages, as well as like & dislike posts. A user should not be able to like or dislike a post multiple times, and the number of likes or dislikes should increment after a user presses the appropriate button.

Your client must make all network connections to the server asynchronously. The feed should refresh after the user posts a message.

Connecting to Chit Chat

The first thing you will need to do is register for an API Key with Chit Chat. To do this you can either connect to the endpoint <https://www.stepoutnyc.com/chitchat/register> with a POST request and a parameter of “email” with your email. Or, you can visit the convenient website <https://www.stepoutnyc.com/static/register.html>

After you send your POST request or fill out the form on that website with your champlain.edu email address, you will receive an email with your credentials (aka Key). Note that the key is unique to your email address and you should not share it with your classmates. If you don't get the email with your Key, be sure to check your spam folder. If you still don't get it, please email me and include the email address you tried to sign up with. Note that Chit Chat will only register Champlain email addresses.

Chit Chat API

Retrieve Messages

Endpoint

GET request to <https://www.stepoutnyc.com/chitchat>

Required Parameters

key: <your key>

client: <your email address>

Optional Parameters

skip: <number of messages to skip> (default 0)

limit: <maximum number of messages> (default 20)

Send Message

Endpoint

POST request to <https://www.stepoutnyc.com/chitchat> with URL encoded parameters

Required Parameters

key: <your key>

client: <your email address>

message: <user's message>

Optional Parameters

lat: <number of degrees latitude> (default 0)

lon: <number of degrees longitude> (default 0)

Like Message

Endpoint

GET request to <https://www.stepoutnyc.com/chitchat/like/<message id>>

Required Parameters

key: <your key>

client: <your email address>

Dislike Message

Endpoint

GET request to <https://www.stepoutnyc.com/chitchat/dislike/<message id>>

Required Parameters

key: <your key>

client: <your email address>

Resources

You may find third party libraries like `okhttp` useful for making HTTP requests. You may find the Android class `AsyncTask` helpful for working asynchronously. Android has built-in JSON support via the `org.json` package, but you may find third party libraries like `GSON` and `Jackson` superior alternatives. If you have no prior experience with JSON, take a few minutes to read up on it first. Checkout chapter 25 of your book for help with all of these topics.

Reminders

- Remember to cite *any* sources that you use code from using comments. Even if it's just a few lines.
- You have the option of working with a partner of your choice or working solo on this project. If you work with a partner, as per the course's group work policy, you will both receive the same grade.
- This is a social network that approximately 75 people will be accessing and your email address is associated with all of the messages you post. Do not abuse the privilege of doing a project on a social network by posting offensive messages. I will not be pleased.

Grading

Your base project will be graded on four criteria:

- Does your project work correctly without bugs? If I cannot run your project or it constantly crashes, you will not do well. Please do not send me a project that does not build.
- Design - Is your project well structured? Does it use a smart object-oriented design?
- Conventions - Does your project follow standard Android conventions as laid out by *Android Programming: The Big Nerd Ranch Guide, 3rd Edition*?
- Cleanliness of implementation - intelligent variable names, well commented, proper indentation... all the little things.

Extra Credit

The following four options are the only extra credit that will be available on this assignment.

- Use the geolocation APIs to attach lat/lons to your messages (the server supports this) and determine the distance of other messages from the user based on latitude/longitude. (1 point)
- Permanently only allow a user to like or dislike a post once from the same device. (1 point)
- Offer the user the choice of sorting posts by likes or dislikes in addition to time. (1 point)
- Add a load more button to the bottom of the feed that retrieves older messages. (1 point)