Assignment No. 6

Chat App with Services and Cloud

Objective: Create chatting app with basic Services and Cloud.

Description:

In this assignment, we will be implementing a simple cloud-based chat app, where clients exchange chat messages through a Web service. We are given a simple chat server that apps will communicate with via HTTP.

The main user interface for the app presents a screen with a text box for entering a message to be sent, and a “send” button. The rest of the screen is a list view that for now will just show the messages posted by this app. A settings screen where the server URI and this chat instance’s client name can be specified, to be saved in shared preferences.

This service stays alive as long as it is connected to the activity. As soon as activity is destroyed service stops. Service that is used to receive message is started service. This is started using startService method. It calls onStartCommand method in return. This service runs for indefinite amount of time unless stopped manually by stopself() and stopService(). Execution takes place in onHandleelement() method.

The message input and message list interface just likes the previous design. The content provider lists the clients registered with the chat service. When synchronize the server downloads the new message to the client.

I have created couple of classes to handle the request in the background thread like request process and request and register classes along with rest method which is given below.

REST Method:

```
URL url = new URL(request.getRequestUri().toString());
// Send POST data request
Log.d("RestMethod", url.toString());
URLConnection connection = url.openConnection();
HttpURLConnection conn = (HttpURLConnection) connection;
conn.setReadTimeout(10000);
conn.setConnectTimeout(15000);
conn.setRequestProperty("X-latitude", "40.744906");
conn.setRequestProperty("X-longitude", "40.744906");
conn.setRequestProperty("Content-Type", "application/json");
conn.setRequestProperty("Accept", "application/json");
conn.setRequestMethod("POST");
response.status = conn.getResponseCode();
Log.d("Response Code", String.valueOf(response.status));
```

Conclusion: In this way I have successfully created chat app and using services and saved messages using http protocols to save that messages on the cloud.