**Project-2 Text Views and Internet Requests**

**Project Description**

The main purpose of the project is to communicate with an external API via secure method using Android platform and using a request queue singleton pattern to communicate with the network calls and requests.

**Goals of the Project**

1. Create a layout with full screen scrollable text View

2. Install Volley library for making network calls asynchronously.

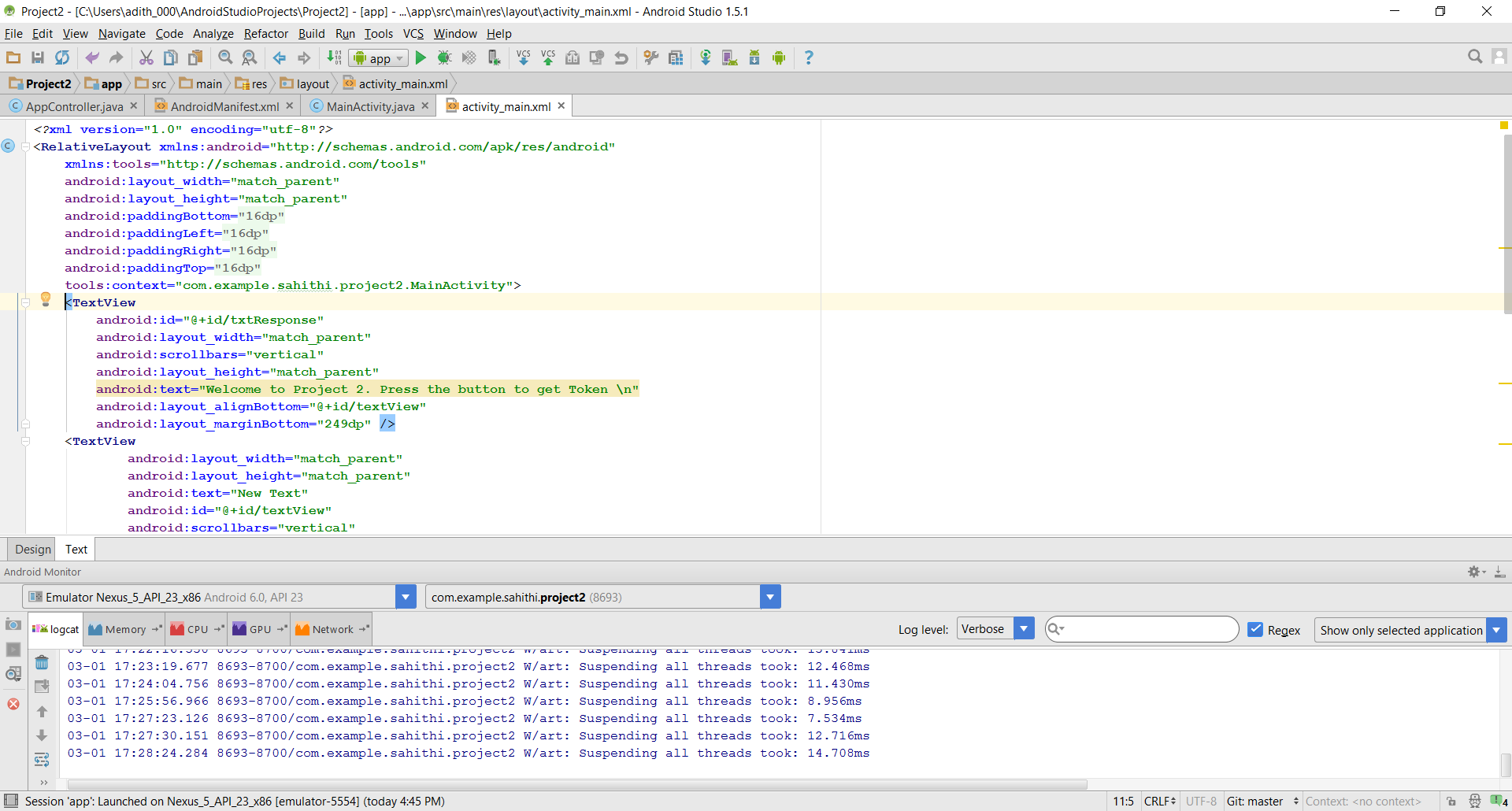
3. Setup volley for basic communications with the API.

4. Create a request queue singleton to communicate with the API.

5. Use JSONObject to retrieve JSON responses from the API.

**Layout with full screen scrollable text View**

For Scrollable text views I created two text views with vertical scroll bars. Also I created two buttons one to generate token, and one to generate message on the activity main XML file. I used classroom demonstration of scrollable text views and layouts. A snippet of my Layout File.



**Figure 1- XML Layout**

**Installing Volley**

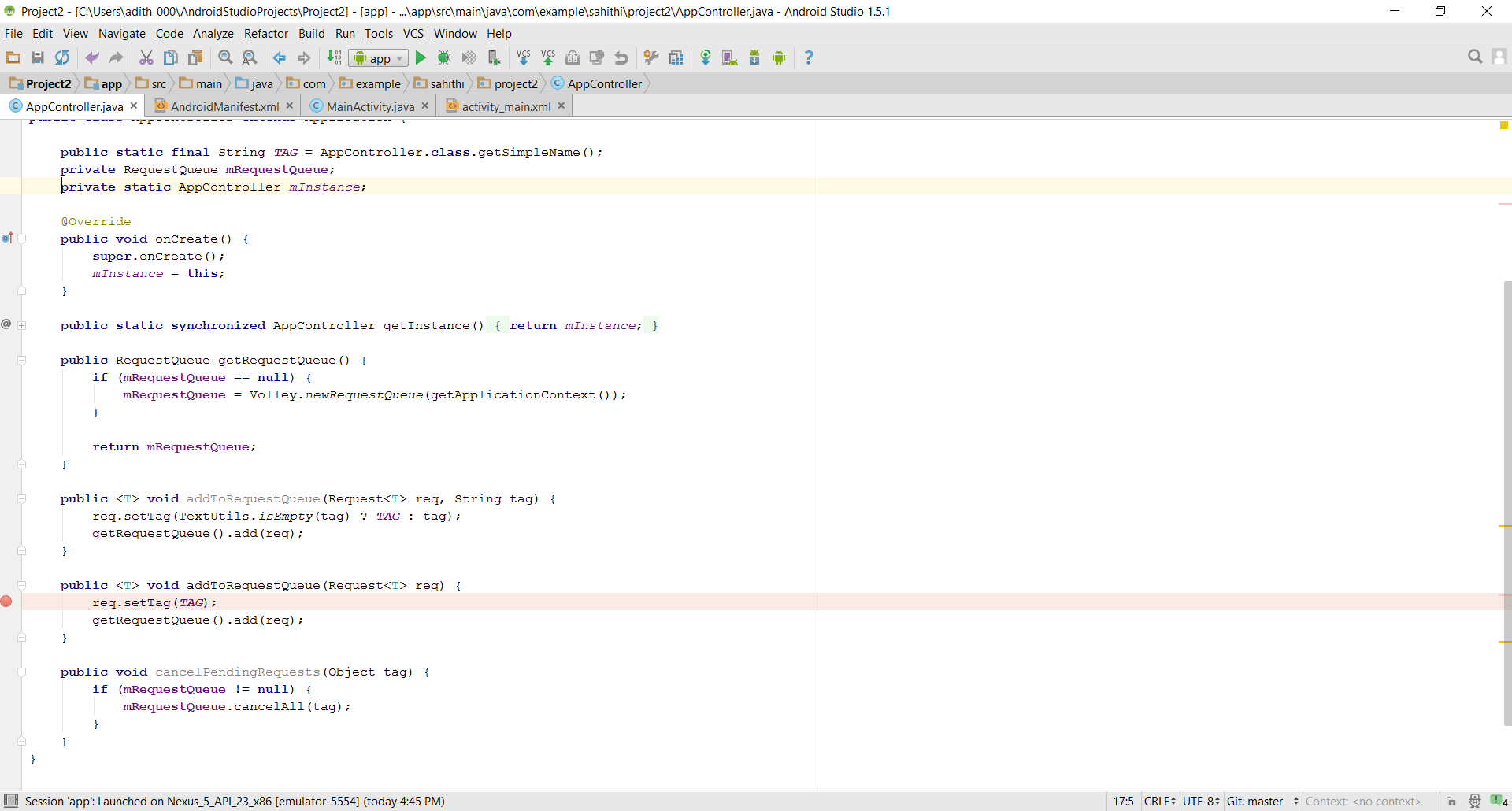
As described in the project description for installing Volley I downloaded GIT for windows and cloned volley into one of the directories. On Android studio I inserted volley module into my project and the volley library was included in my project. There were few issues after I installed volley but Android studio told me what the errors was and I installed Android 22 package and Volley was ready to use.

**Request Queue Singleton Class using Volley Library**

**Volley Library**

Android Volley was introduced to make networking calls much easier. Volley networks calls works asynchronously. Volley comes with a lot of features such as request queuing and prioritization, effective request cache and memory management, extensibility and customization of the library to our needs, and cancelling the requests.

For communication with the API we use Volley Library .To do that I created an AppController Java file where I created a singleton class to initialize all the volley core objects such as RequestQueue, addToRequestQueue, and canceling requests. In this way we can communicate with JSON objects and responses using Volley. We also have to add permission for internet in the android manifest file. And also the AppController Java class is the first one to start after we open our application. So we have to add this class in the manifest file as well using name attribute for <application> tag. Here is a code snipped of the AppController singleton class.



**Figure 2- Request Queue Singleton.**

**Message API reference**

|  |  |  |  |
| --- | --- | --- | --- |
| **URL** | **Method** | **Parameters** | **Returns** |
| http://sfsuswe.com/413/get\_token | GET | Username, Password. | json- encoded object containing an access token labeled “token” |
| http://sfsuswe.com/413/get\_message | GET | Token= the token that you received upon valid login using get token. | Json- encoded object containing your “secret message” labeled “message”. |

**JSON Requests**

JSON requests can be of many types such as 1) Json Objects 2) Json Array 3) string request 4) image request.

Json objects can be two methods get and post methods. GET method is used to obtain JSON responses from the external API. And post method we can actually pass request parameters to the URL.

The project uses get request methods to accept data from the remote system, and respond to successful requests with JSON objects.

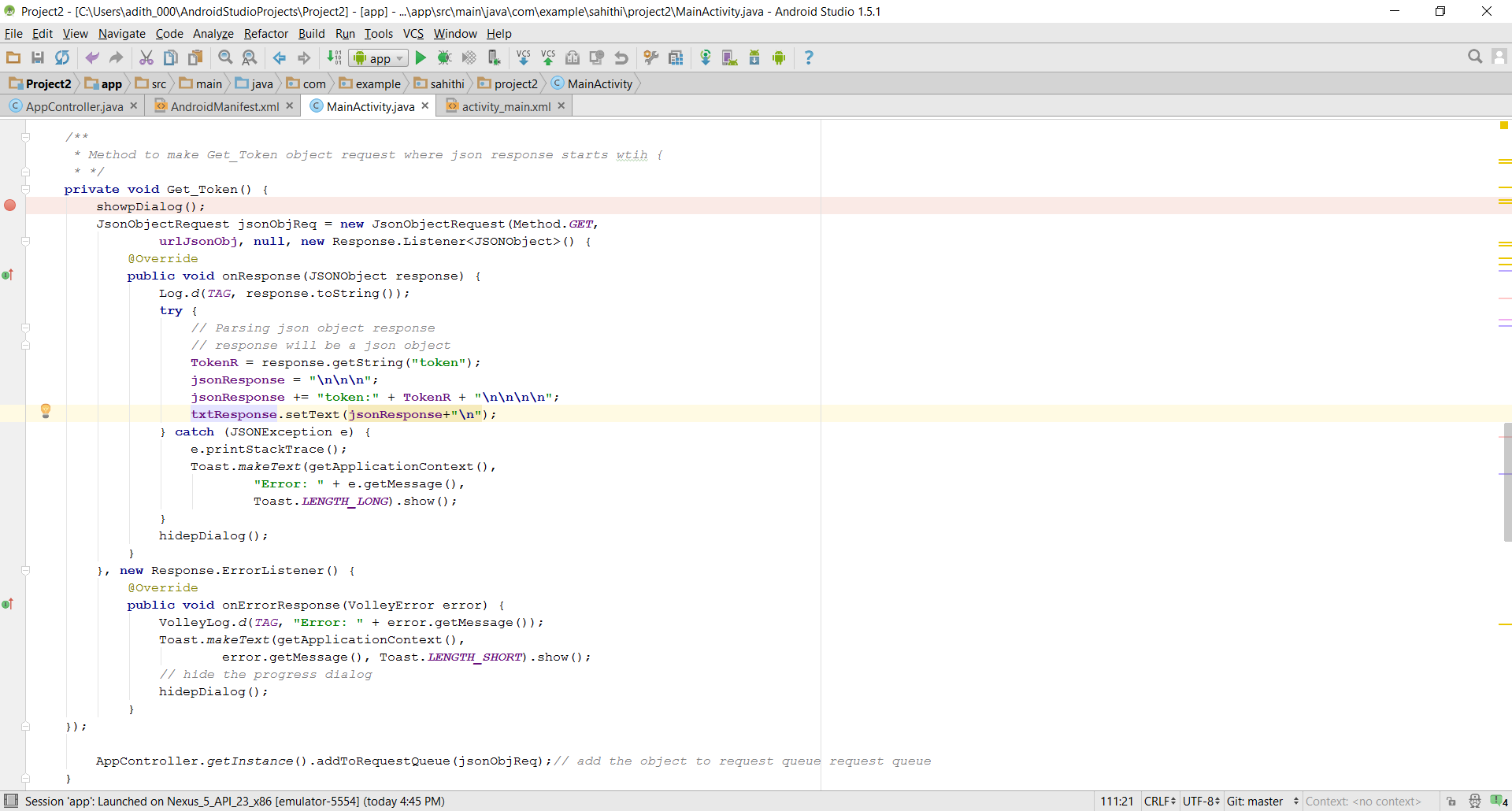
JsonObjectRequest jsonObjReq = **new** JsonObjectRequest(Method.***GET***,  
 **urlJsonObj**, **null**, **new** Response.Listener<JSONObject>(){

@Override  
**public void** onResponse(JSONObject response) {  
 Log.*d*(*TAG*, response.toString());

**A snippet of how to use JsonObject Get request method.**

**Get\_token**

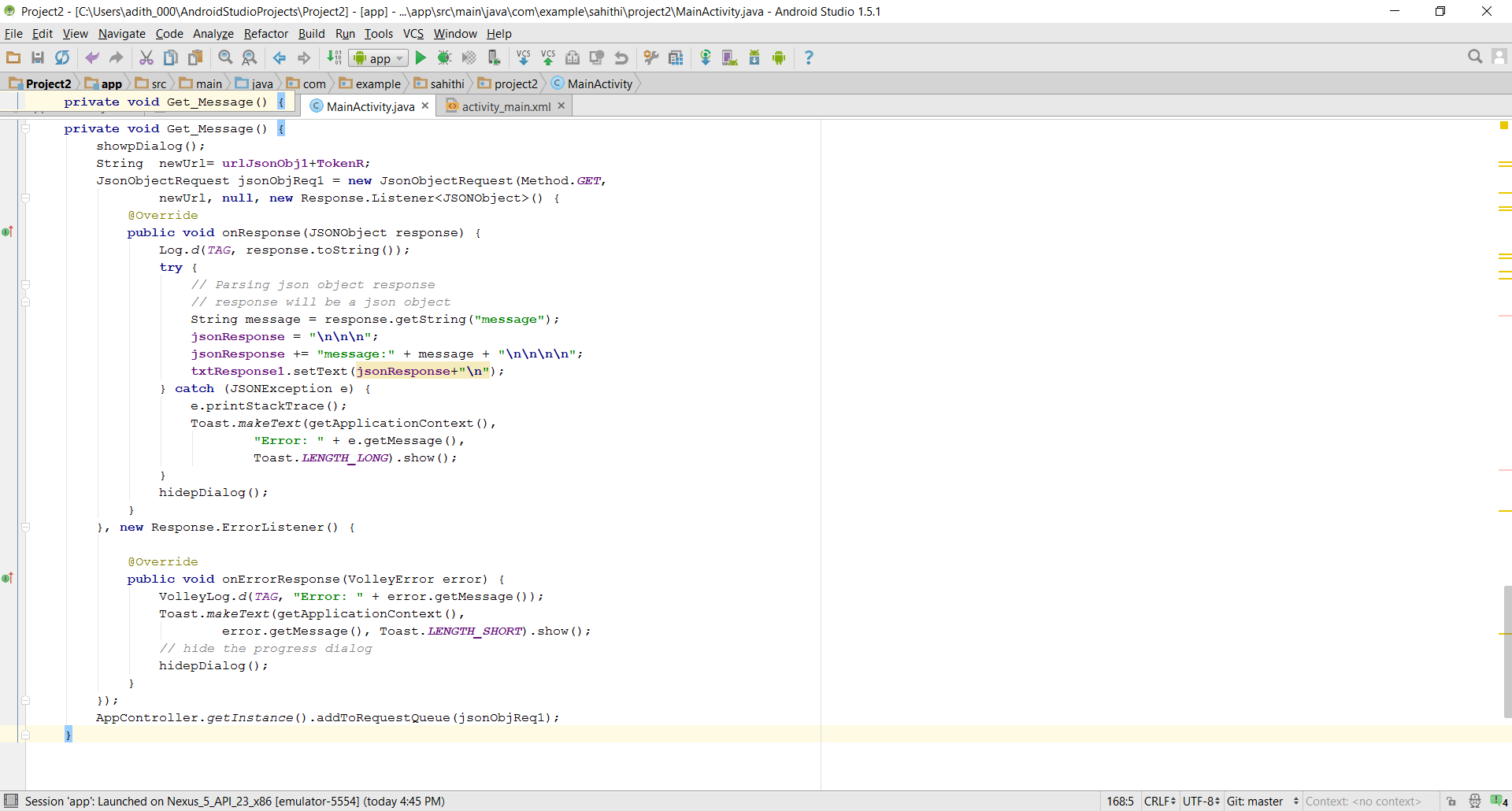
The purpose of the API was to obtain a token from the SFSU server via secure fashion. To obtain token we have to provide our username and password. If we send a request to the server it return a json response containing an access token labeled “token”. Volley provides JsonObject Request class to make json object request. I provided the user name and password in the URL itself for the Get\_token part. In the class we are fetching Json by making a call to http://sfsuswe.com/413/get\_token and parsing it. Finally the parsed token response is appended to a string and displayed on screen.



**Figure 4- A snippet of the get\_token class.**

**GET\_ Message**

To retrieve the message from the Json object we had to provide the json token response upon valid login. Getting the message was a tricky part. Instead of parsing the token in the url we have to concatenate the token received in the class. We have to do string concatenation. To do that we have to globalize the token and create a string with a new URL in the class and concatenate the globalized json response of the token. The message is same for all the randomized tokens generated.



**Figure 5- A code snippet of our GET\_Message class**

****

**Figure 6- Token and My “Secret Message”**

**Challenges and Hurdles Faced in the Project**

I was able to get the token part after understanding the concept of the JSON object and Request Queue Singleton. I spent most of the time on message part even though I kind of understood the concepts. Since it was confusing at first on how to parse the json response of token in the message class. But with the help of TA and useful online resources like stack overflow I was able to achieve it. I had few troubles with the text view scrolling at the beginning. But other than that I was able to get the token, and I can actually see the tokens changing with the click of the button. And I was able to get the correct secret message from the external API. Overall this project helped me to gain understanding and perspective of how to make external API calls using volley and JSON.

**Resources Used**

1. <http://www.tutorialspoint.com/android/android_textview_control.htm>
2. <http://developer.android.com/training/volley/requestqueue.html>
3. <http://www.sitepoint.com/volley-a-networking-library-for-android/>
4. <http://stackoverflow.com/questions/35678867/getting-json-response-with-volley-android-studi/35679150#35679150>.