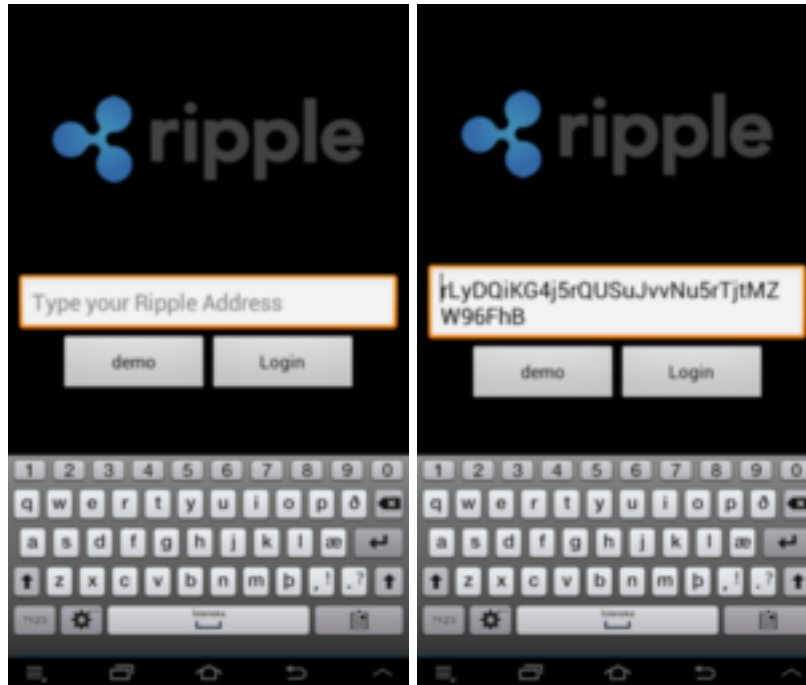# RippleWallet.java


```java
package org.icehat.ripplewallet;

import java.net.URI;
import java.net.URISyntaxException;
import java.io.IOException;
import java.util.List;

import android.app.Activity;
import android.os.Bundle;
import android.util.Log;
import android.view.View;
import android.widget.EditText;
import android.widget.TextView;
import android.content.Intent;

import com.codebutler.android_websockets.WebSocketClient;

import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;
import org.apache.http.message.BasicNameValuePair;
```

```java
/** Starting activity with a log in screen.\ It contains a Websocket
 *  client that detects incoming JSON messages from the server and decides
 *  what to do with them.\ It also contains methods to extract relevant
 *  information from them and pass it on into other activities.
 *
 *  @author Matthías Ragnarsson
 *  @author Pétur Karl Ingólfsson
 *  @author Daniel Eduardo Pinedo Quintero
 *  @version Thu Oct 17 16:33:37 GMT 2013
 */
public class RippleWallet extends Activity
{
    public static final String TAG = "RippleWallet";
    private static final String rippleServerURI = "wss://s1.ripple.com";
    private WebSocketClient client;
    List<BasicNameValuePair> extraHeaders;
    public EditText address;
    private static final int ID_ACCOUNT_INFO = 100;

    /** Shows the log in screen, initializes the websocket client and
     *  establishes a websocket connection with the Ripple server.
     */
    @Override
    public void onCreate(Bundle savedInstanceState)
    {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);

        client = new WebSocketClient(URI.create(rippleServerURI),
                                     new WebSocketClient.Listener() {

            @Override
            public void onConnect() {
                Log.d(TAG, "Connected!");
            }

            @Override
            public void onMessage(String message) {
                Log.d(TAG, String.format("JSON received... %s", message));
                TextView login_msg = (TextView)findViewById(R.id.login_msg);
                try {
                    String balance = parseBalance(message);
                    Log.d(TAG, String.format("Balance is: %s", balance));
                    toBalance(balance);
                } catch (JSONException e) {
```

```java
                Log.e(TAG, "Parsing error.");
                login_msg.setText("Address not found!");
            }
        }

        @Override
        public void onMessage(byte[] data) {
            Log.d(TAG, "Got binary message!");
        }

        @Override
        public void onDisconnect(int code, String reason) {
            Log.d(TAG, String.format("Disconnected! Code: %d Reason: %s",
                                        code, reason));
        }

        @Override
        public void onError(Exception error) {
            Log.e(TAG, "Error!", error);
        }
    }, extraHeaders);
    client.connect();
}


/** Extracts and returns the string inside a JSON response to account_info call.
 *
 * @param message JSON response to account_info.
 * @return The amount of XRPs specified in message.
 */
public String parseBalance(String message) throws JSONException{
    JSONObject json = new JSONObject(message);
    return json.getJSONObject("result")
                .getJSONObject("account_data")
                .get("Balance").toString();
}

/** Shifts to Balance Activity with balance in Intent.
 *
 * @param balance Balance of the account.
 */
public void toBalance(String balance) {
    Intent intent = new Intent(this, Balance.class);
    intent.putExtra("address", address.getText().toString());
    intent.putExtra(TAG, balance);
    startActivity(intent);
```

```java
    }

    /** Sends a request for account information to the server.
     *
     * @param address Address of account.
     */
    public void getAccountInfo(String address) throws JSONException {
        JSONObject json = new JSONObject();
        json.put("command", "account_info");
        json.put("account", address);
        client.send(json.toString());
    }

    /** Populates address field with dummy address for easier demoing
     */
    public void populateAddress(View view){
        address = (EditText) findViewById(R.id.address);
        String dummy_address = "rLyDQiKG4j5rQUSuJvvNu5rTjtMZW96FhB";
        address.setText(dummy_address);
        Log.d(TAG, "PopAddress: "+ address.getText().toString());
    }
    /** Logs a user into a wallet with an address and password
     *  specified in the activity's address and password fields.\ When
     *  that occurs, appropriate information is stored and the activity
     *  switches to wallet's balance activity.\ If the address or password
     *  are invalid, an error message is displayed on screen.\
     *  </p>
     *  Note: Method in the making, now used only to show account balance.
     *
     *  @throws JSONException
     */
    public void logIn(View view) {

        address = (EditText)findViewById(R.id.address);
        //EditText password = (EditText)findViewById(R.id.password);
        TextView login_msg = (TextView)findViewById(R.id.login_msg);

        try {
            getAccountInfo(address.getText().toString());
            Log.d(TAG, "Attempting to log in to address: " + address.toString());
        } catch(JSONException e) {
            Log.e(TAG, e.toString());
            login_msg.setText("Error: " + e.toString());
        }
    }
}
```

main.xml

```xml
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="match_parent"
    android:gravity="center"
    android:orientation="vertical" >

    <ImageView
        android:id="@+id/imageView1"
        android:layout_width="249dp"
        android:layout_height="0dp"
        android:layout_weight="1.04"
        android:contentDescription="@string/login"
        android:src="@drawable/logo" />

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="vertical" >

        <EditText
            android:id="@+id/address"
            android:layout_width="302dp"
            android:layout_height="wrap_content"
            android:layout_gravity="center_horizontal"
            android:ems="10"
            android:hint="@string/address_hint" >

            <requestFocus />
        </EditText>
    </LinearLayout>

    <LinearLayout
        android:layout_width="238dp"
        android:layout_height="wrap_content" >

        <Button
            android:layout_width="119dp"
            android:layout_height="wrap_content"
            android:onClick="populateAddress"
            android:text="@string/demo" />

        <Button
```

```xml
            android:layout_width="119dp"
            android:layout_height="wrap_content"
            android:onClick="logIn"
            android:text="@string/login" />

    </LinearLayout>

    <TextView
        android:id="@+id/login_msg"
        android:layout_width="wrap_content"
        android:layout_height="0dp"
        android:layout_weight="1"
        android:text="" />

</LinearLayout>
```
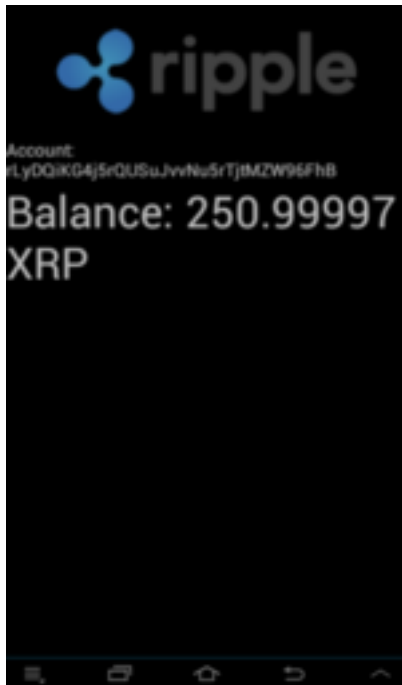
Balance.java


```java
package org.icehat.ripplewallet;
```

```java
import android.app.Activity;
import android.os.Bundle;
import android.widget.TextView;

/** Shows the account and the XRP balance RippleWallet passed it in
 *  Intent.
 *
 *  @author Matthías Ragnarsson
 *  @author Pétur Karl Ingólfsson
 *  @author Daniel Eduardo Pinedo Quintero
 *  @version Thu Oct 17 16:54:38 GMT 2013
 */
public class Balance extends Activity
{

    /** Fills the relevant fields with information from RippleWallet
     */
    @Override
    public void onCreate(Bundle savedInstanceState)
    {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.balance);
        String balance = getIntent().getStringExtra(RippleWallet.TAG);
        String address = getIntent().getStringExtra("address");
        TextView balanceView = (TextView) findViewById(R.id.balance_view);
        TextView accountView = (TextView) findViewById(R.id.account);

        double b = Double.parseDouble(balance)/1000000;

        balanceView.setText("Balance: " + b + " XRP");
        accountView.setText("Account: " + address);
    }
}
```

balance.xml

```xml
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="fill_parent"
    android:gravity="center|top"
    android:orientation="vertical" >
```

```
<ImageView
    android:id="@+id/imageView1"
    android:layout_width="240dp"
    android:layout_height="106dp"
    android:src="@drawable/logo" />

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical" >

    <TextView
        android:id="@+id/account"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Account: " />

    <TextView
        android:id="@+id/balance_view"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_weight="0.17"
        android:text="Balance: XRP"
        android:textSize="15pt" />
</LinearLayout>

</LinearLayout>
```

strings.xml

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <string name="app_name">Ripple Wallet</string>
    <string name="address_hint">Type your Ripple Address</string>
    <string name="password_hint">Password</string>
    <string name="login">Login</string>
    <string name="demo">demo</string>
</resources>
```