CSE 660 Lab 7

Submitted by: Arumugam Thendramil Pavai

1)**Simple Remote Calculator**

Server is created using ServerSocket class of java

Server.java

```
import java.io.IOException;
import java.net.ServerSocket;
import java.net.Socket;
import java.util.Scanner;
import java.lang.*;
import java.io.*;

public class Server {

    public static void main(String[] args) {
        try{
            ServerSocket serverSocket = new ServerSocket(4444);
            System.out.println("Server Started...");
            while(true){
                new Thread(new
ClientConnectionThread(serverSocket.accept())).start();
            }
        }catch(IOException e){e.printStackTrace();}
    }

}
class ClientConnectionThread implements Runnable{
    private Socket socket;
    public ClientConnectionThread(Socket socket){
        this.socket = socket;
    }
    @Override
    public void run(){
        try{
            DataInputStream dIn = new
DataInputStream(socket.getInputStream());
            DataOutputStream dOut = new
DataOutputStream(socket.getOutputStream());
            String message = dIn.readUTF();

            System.out.println("Client Request : " + message);
            String[] input = message.split(" ");
            String result = input[0] + " " + input[2] + " " + input[1]
+ " = " + calculate(Integer.parseInt(input[0]),
Integer.parseInt(input[1]), input[2]);
      System.out.println("Server Response : " + result);

      dOut.writeUTF(result);
      dOut.flush();
      dOut.close();
      socket.close();
    }catch(IOException e){e.printStackTrace();}
  }

  public static String calculate(int num1, int num2, String
operator) {
    Integer result = 0;
    switch (operator.charAt(0)) {
      case '+':
        result = num1 + num2;
        break;
      case '-':
        result = num1 - num2;
        break;
      case '*':
        result = num1 * num2;
        break;
      case '/':
        result = num1 / num2;
        break;
      default:
        break;
    }
    return Integer.toString(result);

  }
}
```

Android Client App :

Changes were made to the AndroidManifest.xml file

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
  package="example.calculator"
  >
  <uses-permission android:name="android.permission.INTERNET" />
  <uses-permission
android:name="android.permission.ACCESS_NETWORK_STATE" />
  <uses-permission
android:name="android.permission.ACCESS_WIFI_STATE" />
  <uses-permission
android:name="android.permission.CHANGE_WIFI_STATE" />

  <application
    android:allowBackup="true"
```

```
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">
        <activity android:name=".MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category
android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>

</manifest>
```

MainActivity.java

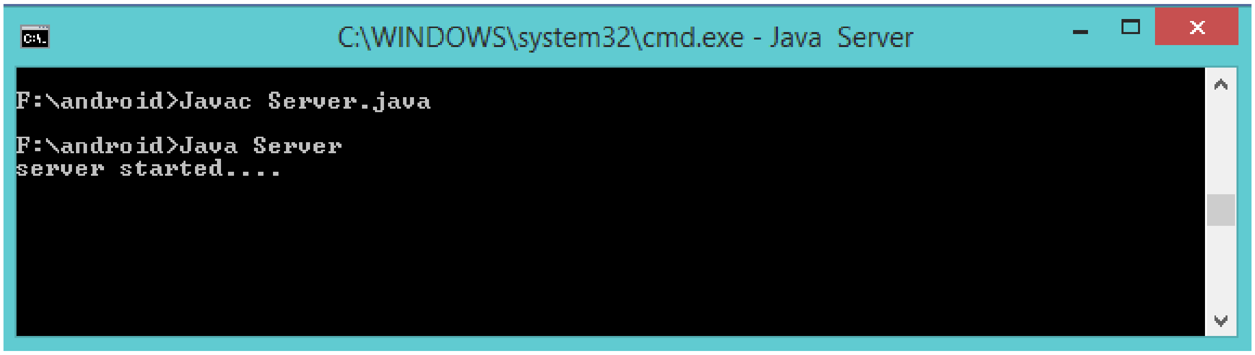
```
package example.calculator;

import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.text.TextUtils;
import android.view.View;
import android.widget.EditText;
import android.widget.ImageView;
import android.widget.TextView;
import java.io.IOException;
import java.net.Socket;
import java.io.*;
import java.net.UnknownHostException;
import java.lang.*;


public class MainActivity extends AppCompatActivity implements
View.OnClickListener{
        EditText t1;
        EditText t2;
        MainActivity activity;
        ImageView plus;
        ImageView minus;
        ImageView multiply;
        ImageView divide;

        TextView displayResult;


        String oper = "";
        Socket socket;
        String response = "";
        /** Called when the activity is first created. */
        @Override
        public void onCreate(Bundle savedInstanceState) {
            super.onCreate(savedInstanceState);
            setContentView(R.layout.activity_main);

            // find the EditText elements (defined in
res/layout/activity_main.xml
            t1 = (EditText) findViewById(R.id.t1);
            t2 = (EditText) findViewById(R.id.t2);

            plus = (ImageView) findViewById(R.id.plus);
            minus = (ImageView) findViewById(R.id.minus);
            multiply = (ImageView) findViewById(R.id.multiply);
            divide = (ImageView) findViewById(R.id.divide);

            displayResult = (TextView)
findViewById(R.id.displayResult);

            // set  listeners
            plus.setOnClickListener( this );
            minus.setOnClickListener( this);
            multiply.setOnClickListener( this);
            divide.setOnClickListener( this);

        }

        // @Override
        public void onClick( View view ) {

            // check if the fields are empty
            if (TextUtils.isEmpty(t1.getText().toString())
                    || TextUtils.isEmpty(t2.getText().toString())) {
                return;
            }
            // perform operations
            // save operator in oper for later use
            switch ( view.getId() ) {
                case R.id.plus:
                    oper = "+";
                    break;
                case R.id.minus:
                    oper = "-";

                    break;
                case R.id.multiply:
                    oper = "*";
                    break;
                case R.id.divide:
                    oper = "/";
                    break;
                default:
                    break;
            }
            new Thread(new Runnable() {
                @Override
                public void run() {
                    try {
                        response = "";
                        socket = new Socket("10.0.2.2", 4444);
                        DataOutputStream dOut = new
DataOutputStream(socket.getOutputStream());
                        DataInputStream dIn = new
DataInputStream(socket.getInputStream());
                        dOut.writeUTF(t1.getText() + " " +
t2.getText() + " " + oper);
                        dOut.flush();
                        response = dIn.readUTF();

                        runOnUiThread(new Runnable() {
                            @Override
                            public void run() {
                                displayResult.setText(response);
                            }
                        });

                        dIn.close();
                        dOut.close();
                        socket.close();
                    }
                    catch (UnknownHostException e) {
                        e.printStackTrace();
                        displayResult.setText("UnknownHostException: "
+ e.toString());
                    } catch (IOException e) {
                        e.printStackTrace();
                        displayResult.setText("IOException: " +
e.toString());
                    }

                }
            }).start();
        }
}
```

UI - activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent">
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:id="@+id/linearLayout1"
        android:layout_marginLeft="12pt"
        android:layout_marginRight="12pt"
        android:layout_marginTop="4pt">
        <EditText
            android:layout_weight="1"
            android:layout_height="wrap_content"
            android:layout_marginRight="6pt"
            android:id="@+id/t1"
            android:layout_width="match_parent"
            android:inputType="number">
        </EditText>
        <EditText
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:layout_marginLeft="6pt"
            android:id="@+id/t2"
            android:layout_width="match_parent"
            android:inputType="number">
        </EditText>
    </LinearLayout>
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:id="@+id/linearLayout2"
        android:layout_marginTop="4pt"
        android:layout_marginLeft="6pt"
        android:layout_marginRight="6pt">
        <ImageView
            android:layout_height="wrap_content"
            android:layout_width="match_parent"
            android:layout_weight="1"
            android:src="@drawable/add"
            android:id="@+id/plus">
        </ImageView>
        <ImageView
            android:layout_height="wrap_content"
            android:layout_width="match_parent"
            android:layout_weight="1"
            android:src="@drawable/minus"
            android:id="@+id/minus">
        </ImageView>
    </LinearLayout>

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:id="@+id/linearLayout3"
        android:layout_marginTop="4pt"
        android:layout_marginLeft="6pt"
        android:layout_marginRight="6pt">
        <ImageView
            android:layout_height="wrap_content"
            android:layout_width="match_parent"
            android:layout_weight="1"
            android:src="@drawable/multiply"
            android:id="@+id/multiply">
        </ImageView>
        <ImageView
            android:layout_height="wrap_content"
            android:layout_width="match_parent"
            android:layout_weight="1"
            android:src="@drawable/divide"
            android:id="@+id/divide">
        </ImageView>
    </LinearLayout>
    <TextView
        android:layout_height="wrap_content"
        android:layout_width="match_parent"
        android:layout_marginLeft="6pt"
        android:layout_marginRight="6pt"
        android:textSize="12pt"
        android:layout_marginTop="4pt"
        android:id="@+id/displayResult"
        android:gravity="center_horizontal">
    </TextView>
</LinearLayout>
```

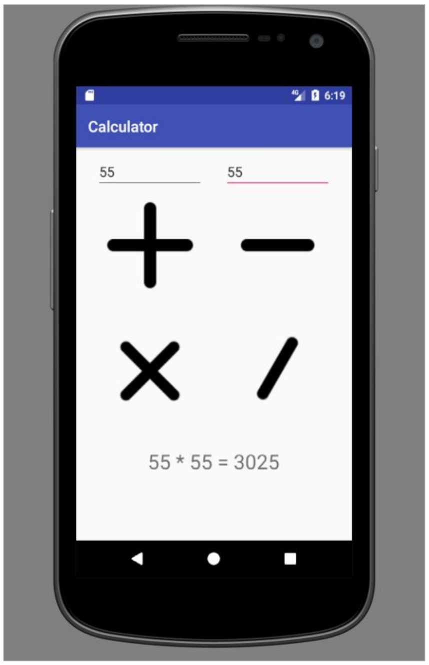
Output:

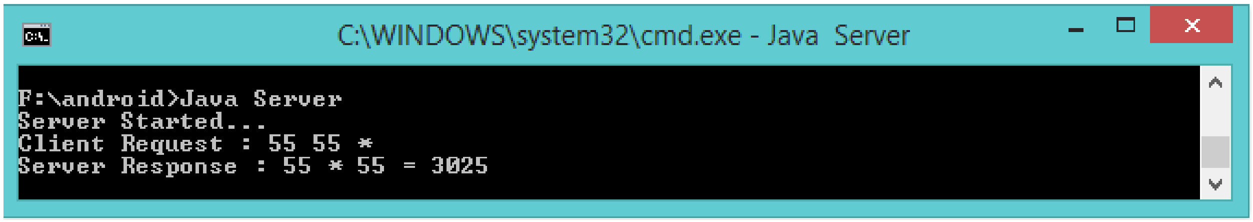
Server execution



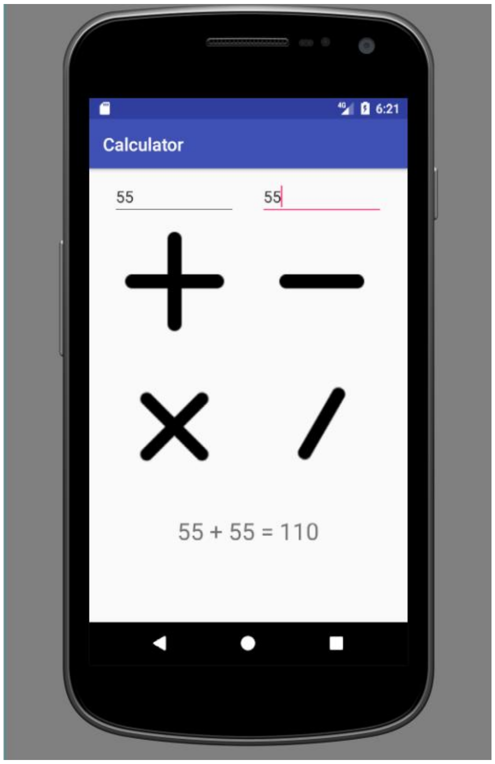
```
F:\android>Javac Server.java

F:\android>Java Server
server started....
```

Client :



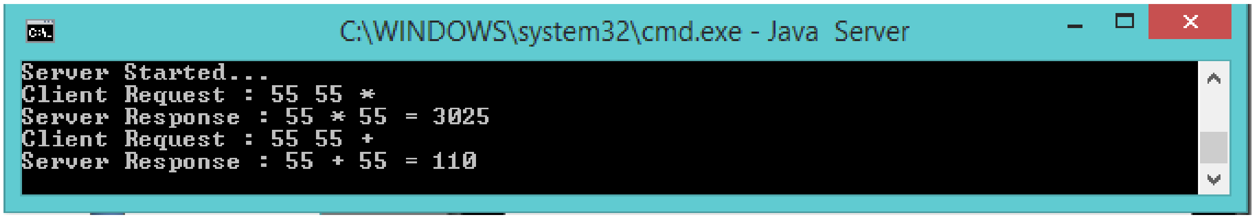

C:\WINDOWS\system32\cmd.exe - Java Server

```
F:\android>Java Server
Server Started...
Client Request : 55 55 *
Server Response : 55 * 55 = 3025
```



55 55

55 + 55 = 110



```
Server Started...
Client Request : 55 55 *
Server Response : 55 * 55 = 3025
Client Request : 55 55 +
Server Response : 55 + 55 = 110
```

Learning and Observation:

Learned about Socket Connections.

Learned about android communication via TCP.

2) **<u>Remote Random Number Generator</u>**

<u>RandomNumberServer.java</u>

```
import java.io.IOException;
import java.net.ServerSocket;
import java.net.Socket;
import java.util.Scanner;
import java.lang.*;
import java.io.*;
import java.util.Random;
import java.util.Arrays;

public class RandomNumberServer {

    public static void main(String[] args) {
        try{
            ServerSocket serverSocket = new ServerSocket(4444);
            System.out.println("Server Started...");
            while(true){
                new Thread(new
ClientConnection(serverSocket.accept())).start();
            }
        }catch(IOException e){e.printStackTrace();}
    }

}
class ClientConnection implements Runnable{
    private Socket socket;
    public ClientConnection(Socket socket){
        this.socket = socket;
    }
    @Override
    public void run(){
        try{
            DataInputStream dIn = new
DataInputStream(socket.getInputStream());
            DataOutputStream dOut = new
DataOutputStream(socket.getOutputStream());
            String message = dIn.readUTF();

            System.out.println("Client Request : " + message);
            String[] input = message.split(" ");
            String result = generateRandom(Integer.parseInt(input[0]),
Integer.parseInt(input[1]), Integer.parseInt(input[2]));
            System.out.println("Server Response : " + result);
            dOut.writeUTF(result);
            dOut.flush();
            dOut.close();
            socket.close();
        }catch(IOException e){e.printStackTrace();}
    }

    public static String generateRandom(int num, int min, int max ) {
        int range = (max - min) + 1;
        String response ="";
        for(int i=0;i<num;i++) {
            response += Integer.toString((int)(Math.random() * range)
+ min) + " ";
        }
        return response;
    }
}
```

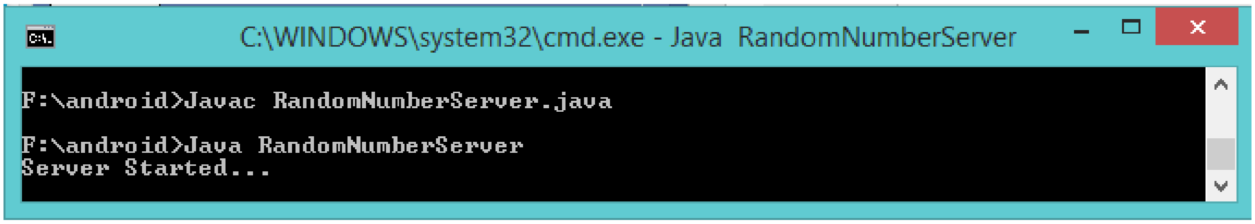
Android Client

UI - activity_main.xml

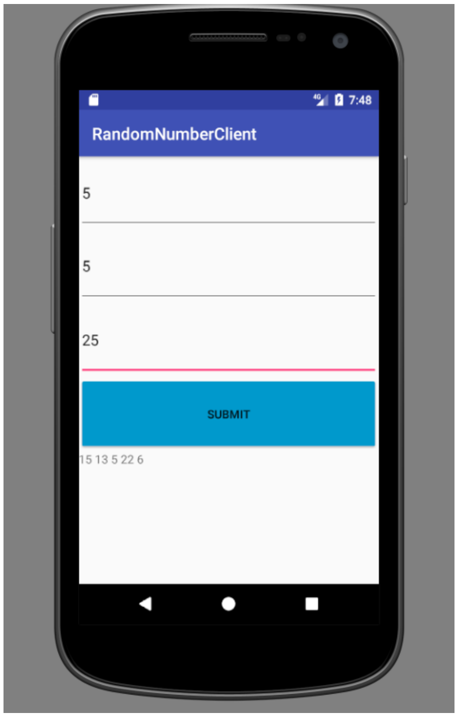
```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent">

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

        <EditText
            android:id="@+id/editText"

            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_weight="0.99"
            android:ems="10"
            android:inputType="textPersonName"
            android:text="Enter Number" />

        <EditText
            android:id="@+id/editText2"

            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:ems="10"
            android:inputType="textPersonName"
            android:text="LowerBound" />

        <EditText
            android:id="@+id/editText3"

            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:ems="10"
            android:inputType="textPersonName"
            android:text="UpperBound" />

        <Button
            android:id="@+id/button"

            android:layout_width="match_parent"
            android:layout_height="46dp"
            android:layout_weight="1"
            android:backgroundTint="@android:color/holo_blue_dark"
            android:text="Submit" />

        <TextView
            android:id="@+id/displayResult"

            android:layout_width="match_parent"
            android:layout_height="116dp"
            android:layout_weight="1"
            android:text="TextView" />

    </LinearLayout>
</LinearLayout>
```

MainActivity.java

```
package example.randomnumberclient;

import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.text.TextUtils;
import android.view.View;
import android.widget.EditText;
import android.widget.Button;
import android.widget.TextView;
import java.io.IOException;
import java.net.Socket;
import java.io.*;
import java.net.UnknownHostException;
import java.lang.*;


public class MainActivity extends AppCompatActivity implements
View.OnClickListener{
    EditText n;
    EditText max;
    EditText min;
    TextView displayResult;
    Button submit;
    MainActivity activity;
    Socket socket;
    String response = "";
    /** Called when the activity is first created. */
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        // find the EditText elements (defined in
res/layout/activity_main.xml
        n = (EditText) findViewById(R.id.editText);
        min = (EditText) findViewById(R.id.editText2);
        max = (EditText) findViewById(R.id.editText3);
        submit = (Button) findViewById(R.id.button);


        displayResult = (TextView) findViewById(R.id.displayResult);

        // set  listeners
        submit.setOnClickListener(this);


    }

    // @Override
    public void onClick( View view ) {

        // check if the fields are empty
        if (TextUtils.isEmpty(n.getText().toString())
                || TextUtils.isEmpty(min.getText().toString()) ||
TextUtils.isEmpty(max.getText().toString())) {
            return;
        }


        new Thread(new Runnable() {
            @Override
            public void run() {
                try {
                    response = "";
                    socket = new Socket("10.0.2.2", 4455);
                    DataOutputStream dOut = new
DataOutputStream(socket.getOutputStream());
                    DataInputStream dIn = new
DataInputStream(socket.getInputStream());
                    dOut.writeUTF(n.getText() + " " + min.getText() +
" " + max.getText());
                    dOut.flush();
                    response = dIn.readUTF();

                    runOnUiThread(new Runnable() {
                        @Override
                        public void run() {
                            displayResult.setText(response);
                        }
                    });

                    dIn.close();
                    dOut.close();
                    socket.close();
                }
                catch (UnknownHostException e) {
                    e.printStackTrace();
                    displayResult.setText("UnknownHostException: " +
e.toString());
                } catch (IOException e) {
                    e.printStackTrace();
                    displayResult.setText("IOException: " +
e.toString());
                }

            }
        }).start();
    }
}
```



```
F:\android>Javac RandomNumberServer.java

F:\android>Java RandomNumberServer
Server Started...
```



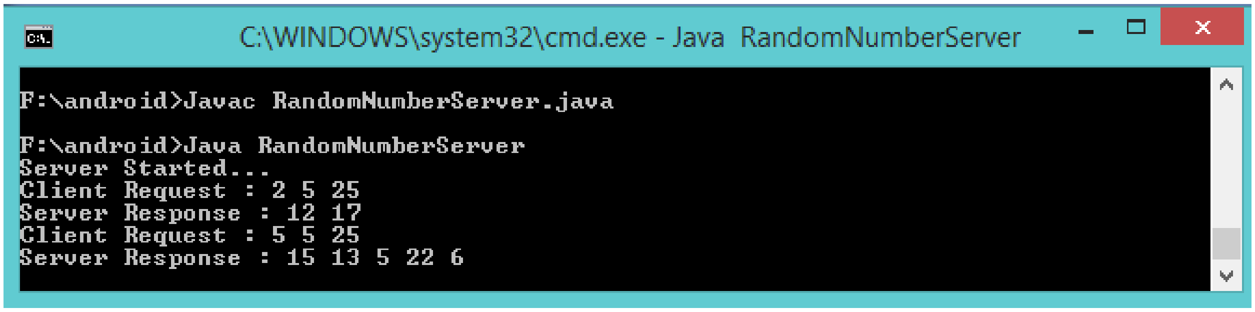
RandomNumberClient

5

5

25

15 13 5 22 6


C:\WINDOWS\system32\cmd.exe - Java RandomNumberServer

```
F:\android>Javac RandomNumberServer.java

F:\android>Java RandomNumberServer
Server Started...
Client Request : 2 5 25
Server Response : 12 17
Client Request : 5 5 25
Server Response : 15 13 5 22 6
```

I have successfully completed all parts of this assignment.