

# CSE494/598 Mobile Health and Social Networking (Sp2009)

## Solution to Homework 1

There are two ways to define the View objects in Android: through programming using java, and through specification using the XML file "main.xml." This solution use the second option. The main.xml file's content is

```
<?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"
    android:id="@+id/backgroundView">
<Button android:id="@+id/actionButton"
    android:text="Action"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />
<TextView android:id="@+id/textDisplay"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:text="" />
</LinearLayout>
```

Basically, in the file above, there are three entities: (i) a layout, which basically defines the way the views are arranged on the screen (this is similar to the layout manager objects in original Java), (ii) a button view object, named actionButton, and (iii) a text view object, called textDisplay. The order they are defined more-or-less defines the order they appear.

The idea behind this configuration is to create an Activity program which changes the textDisplay text whenever the actionButton is pressed. (Alternatively, other programmers could pre-create two or more text view objects and hide/reveal them dynamically, or dynamically create text view objects, all based on button presses)

(please turn over)

For this purpose, we compose a program that alters the text whenever the actionButton is pressed. The ChangeText.java file looks like this:

```
package com.asucse494.ChangeText;

import android.app.Activity;
import android.os.Bundle;
import android.view.View;
import android.view.View.OnClickListener;
import android.widget.Button;
import android.widget.TextView;

public class HelloWorld extends Activity {
    /** Called when the activity is first created. */
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);
        final Button bb = (Button) findViewById(R.id.actionButton);
        bb.setOnClickListener(new OnClickListener () {
            public void onClick(View v) {
                TextView tv = (TextView) findViewById(R.id.textDisplay);
                if (bb.getText() == "Action") {
                    bb.setText("Reset");
                    tv.setText("Thank you! Now push the button to reset!");
                }
                else {
                    tv.setText("");
                    bb.setText("Action");
                }
            }
        });
    }
}
```

The logic in this program is to create a listener on actionButton and then change the text when the listener is activated by an onClick event.