

Mobile Application Development

Higher Diploma in Science in Computer Science

Produced
by

Eamonn de Leastar (edeleastar@wit.ie)

Department of Computing, Maths & Physics
Waterford Institute of Technology

<http://www.wit.ie>

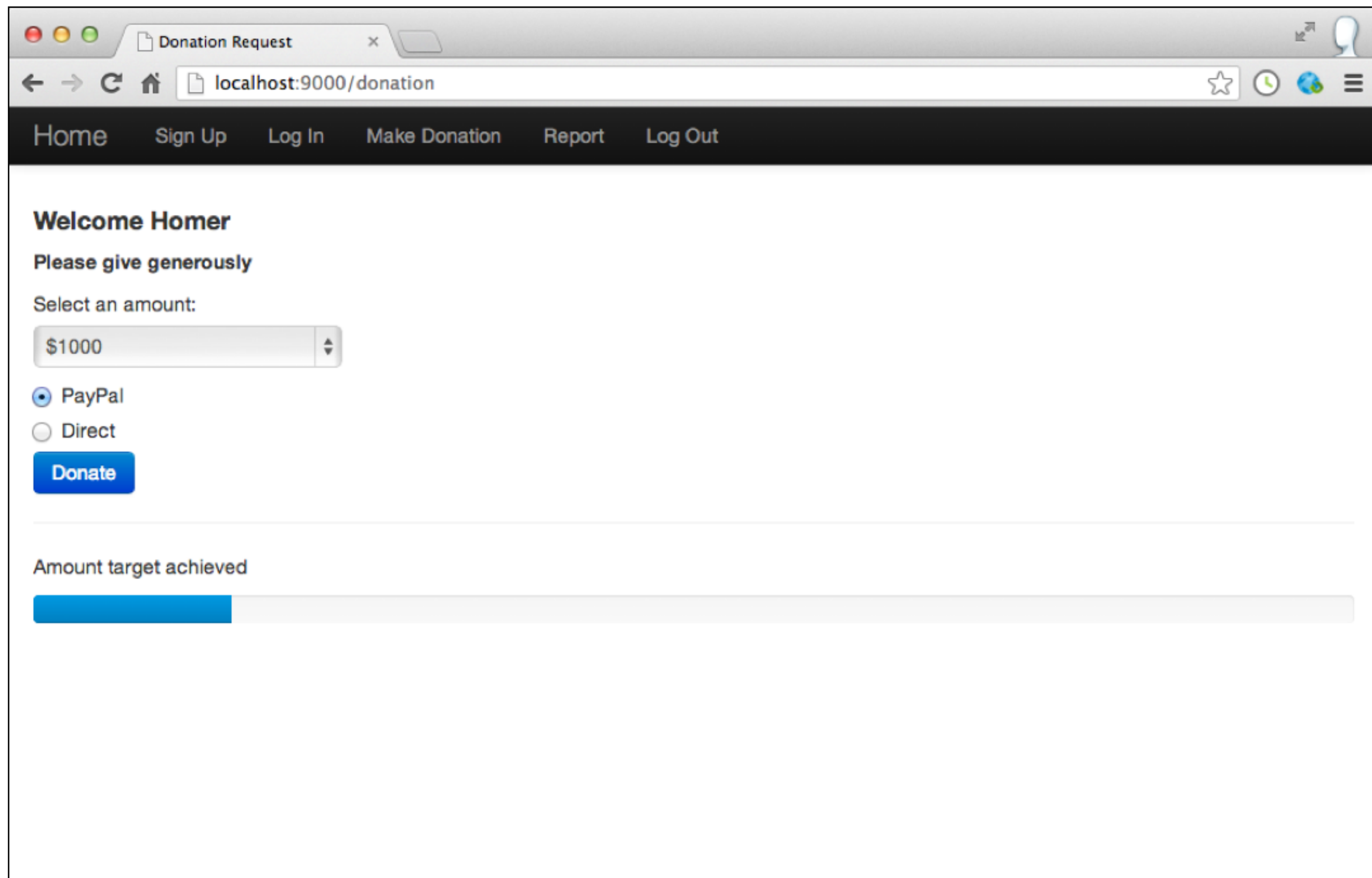
<http://elearning.wit.ie>

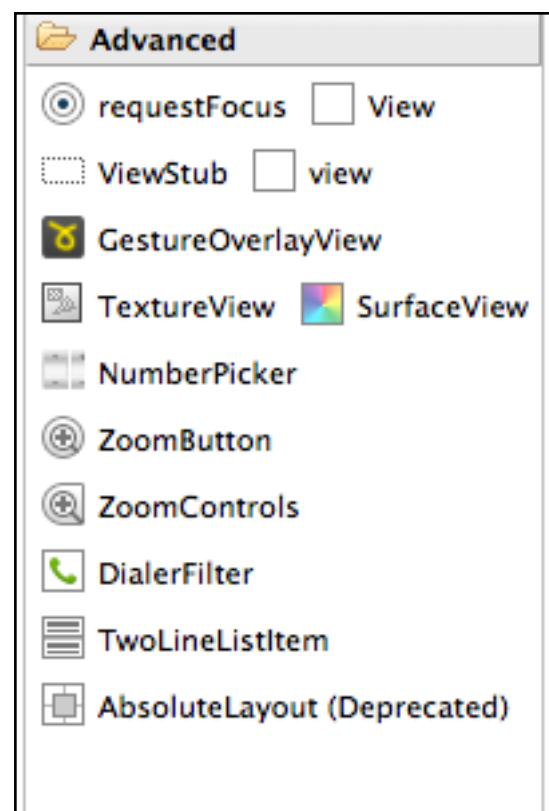
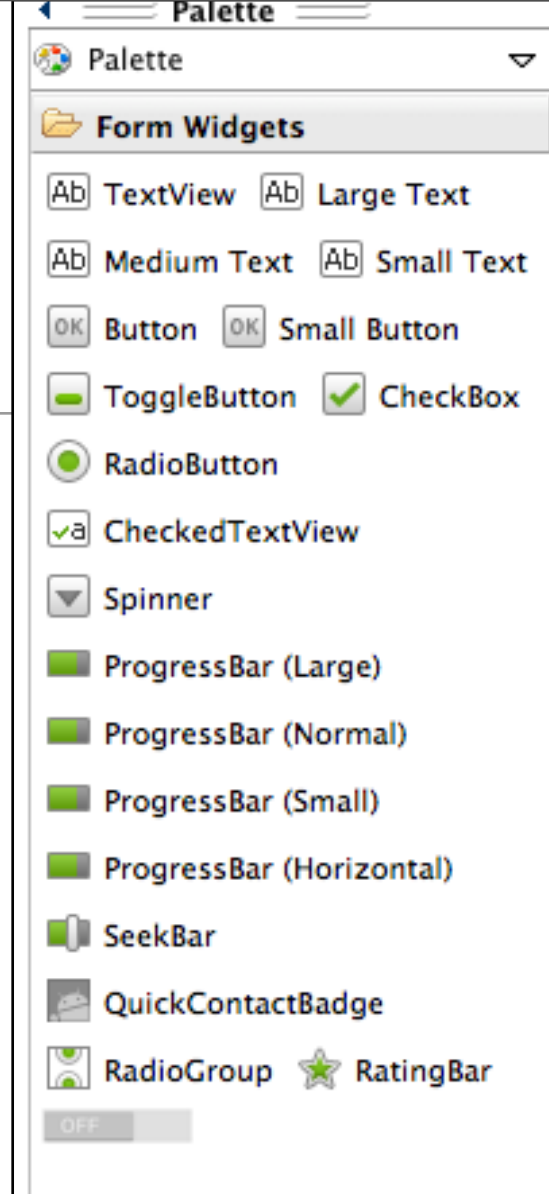


Waterford Institute of Technology
INSTITIÚID TEICNEOLAÍOCHTA PHORT LÁIRGE



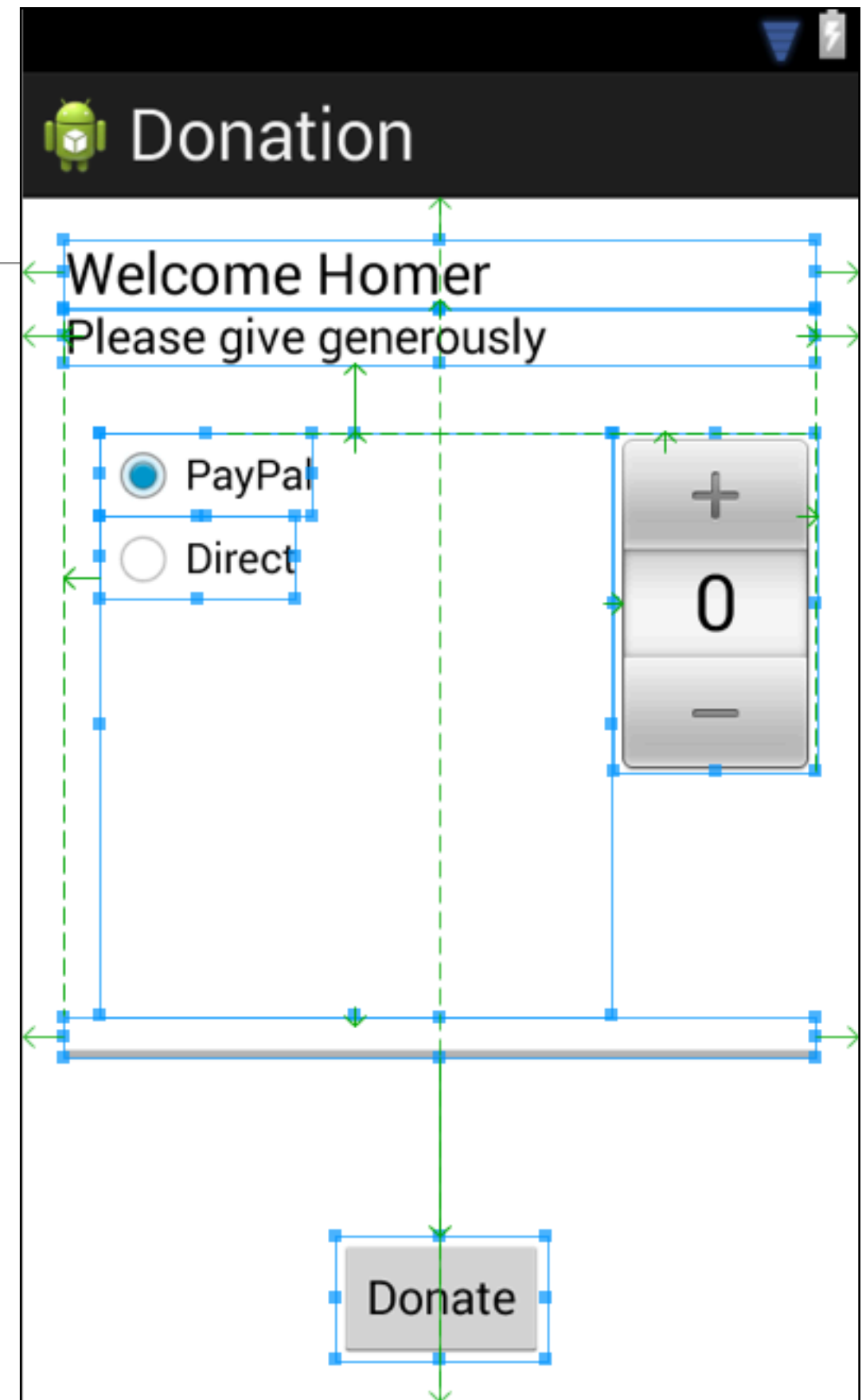
donation-android case study





Layout

- TextView
- Button
- RadioGroup
- ProgressBar
- NumberPicker



```

<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    tools:context=".Donate" >

```

```

<TextView

```

```

    android:id="@+id/donateTitle"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentLeft="true"
    android:layout_alignParentRight="true"
    android:layout_alignParentTop="true"
    android:text="@string/donateTitle"
    android:textAppearance="?android:attr/textAppearanceMedium" />

```

```

<TextView

```

```

    android:id="@+id/donateSubtitle"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentLeft="true"
    android:layout_alignParentRight="true"
    android:layout_below="@+id/donateTitle"
    android:text="@string/donateSubtitle"
    android:textAppearance="?android:attr/textAppearanceMedium" />

```

```

<Button

```

```

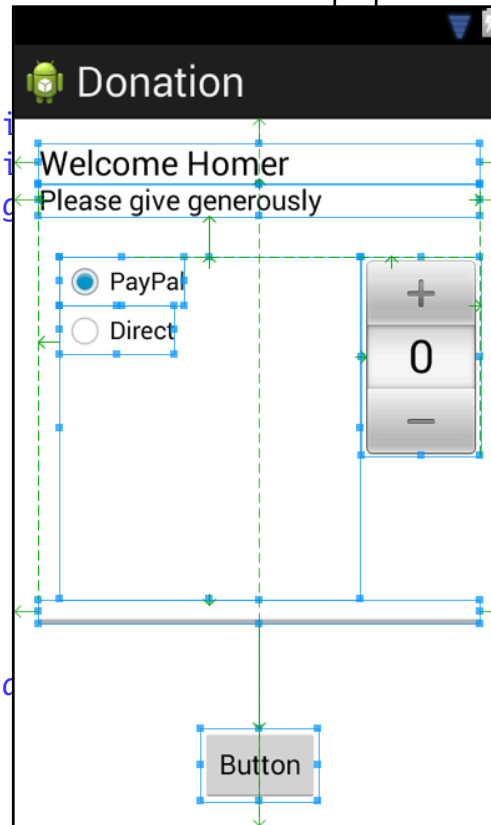
    android:id="@+id/donateButton"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentBottom="true"
    android:layout_centerHorizontal="true"
    android:onClick="donateButtonPressed"
    android:text="@string/donateButton" />

```

```

<?xml version="1.0" encoding="utf-8"?>
<resources>
    <string name="app_name">Donation</string>
    <string name="action_settings">Settings</string>
    <string name="donateTitle">Welcome Homer</string>
    <string name="donateSubtitle">Please give generously</string>
    <string name="donateButton">Donate</string>
    <string name="PayPal">PayPal</string>
    <string name="Direct">Direct</string>
</resources>

```



```

<RadioGroup

```

```

    android:id="@+id/paymentMethod"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_above="@+id/progressBar"
    android:layout_alignLeft="@+id/donateSubtitle"
    android:layout_below="@+id/donateSubtitle"
    android:layout_marginLeft="14dp"
    android:layout_marginTop="26dp"
    android:layout_toLeftOf="@+id/amountPicker" >

```

```

<RadioButton

```

```

    android:id="@+id/PayPal"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:checked="true"
    android:text="@string/PayPal" />

```

```

<RadioButton

```

```

    android:id="@+id/Direct"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="@string/Direct" />

```

```

</RadioGroup>

```

```

<ProgressBar

```

```

    android:id="@+id/progressBar"
    style="?android:attr/progressBarStyleHorizontal"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_above="@+id/donateButton"
    android:layout_alignParentLeft="true"
    android:layout_alignParentRight="true"
    android:layout_marginBottom="67dp" />

```

```

<NumberPicker

```

```

    android:id="@+id/amountPicker"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignRight="@+id/donateSubtitle"
    android:layout_alignTop="@+id/paymentMethod" />

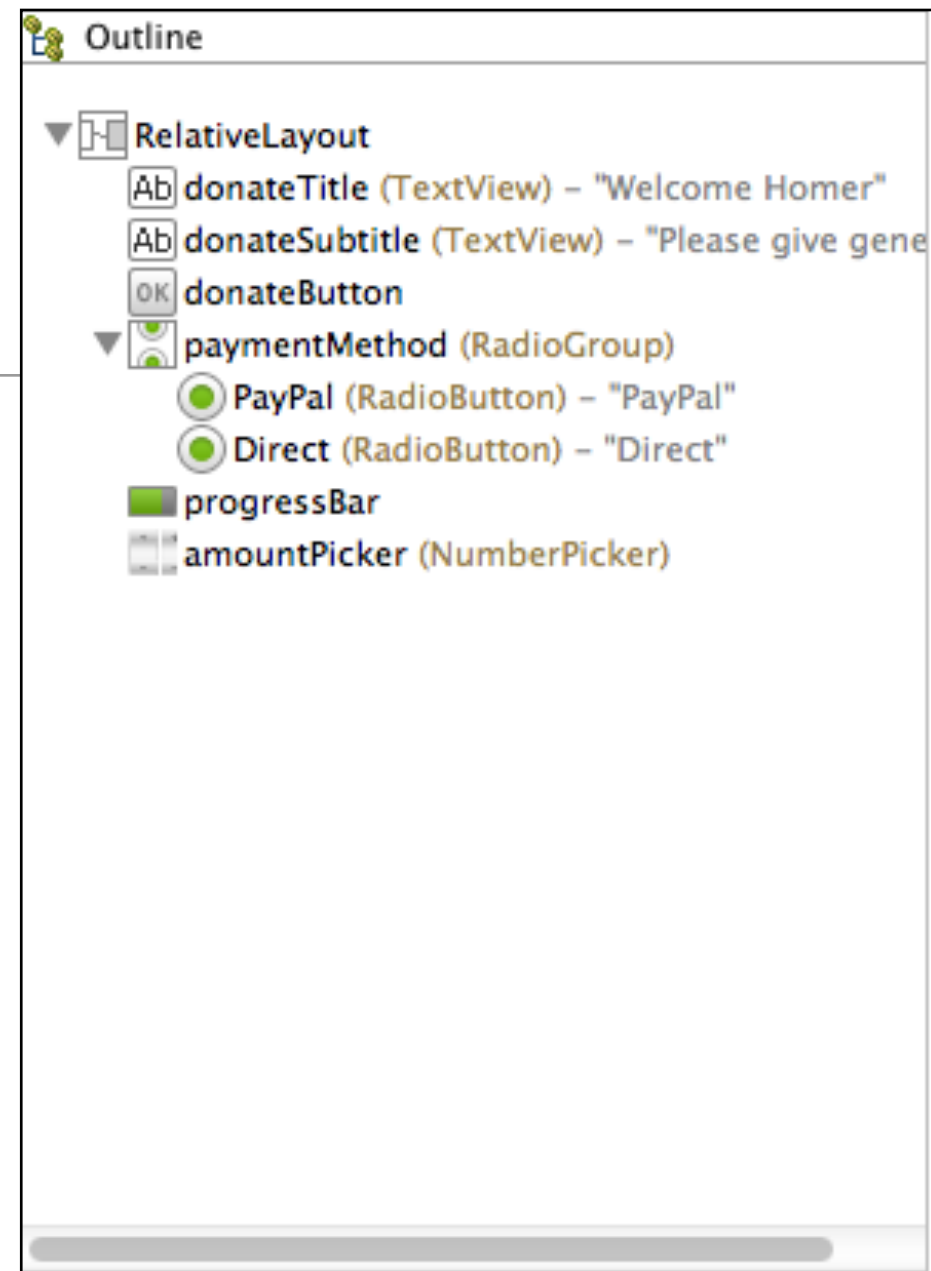
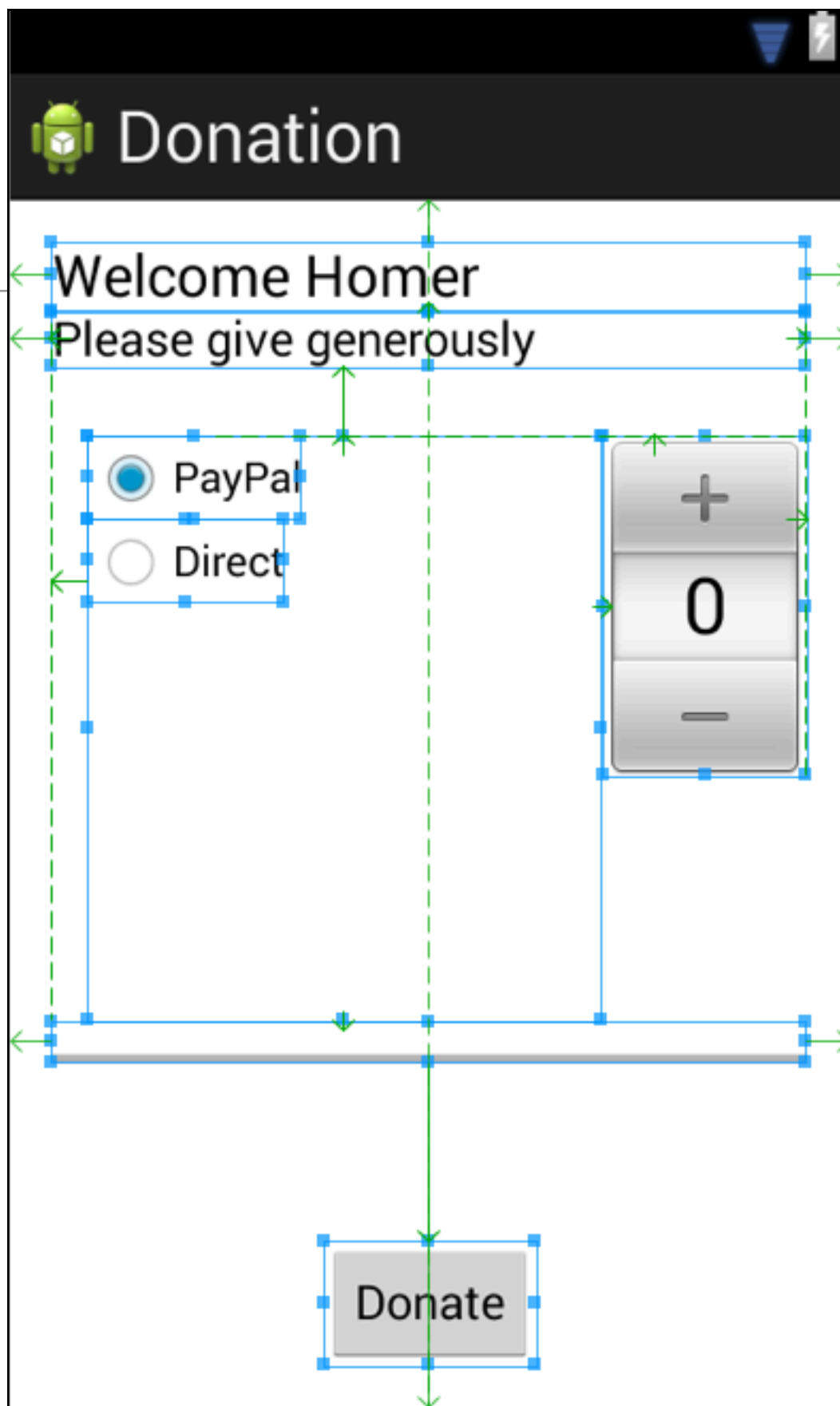
```

```

</RelativeLayout>

```

xml files



- Keep track of the outline view
- Give appropriate names to each control

```

public class Donate extends Activity
{
    private int            totalDonated = 0;

    private RadioGroup    paymentMethod;
    private ProgressBar    progressBar;
    private NumberPicker    amountPicker;

    @Override
    protected void onCreate(Bundle savedInstanceState)
    {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_donate);

        paymentMethod = (RadioGroup) findViewById(R.id.paymentMethod);
        progressBar = (ProgressBar) findViewById(R.id.progressBar);
        amountPicker = (NumberPicker) findViewById(R.id.amountPicker);

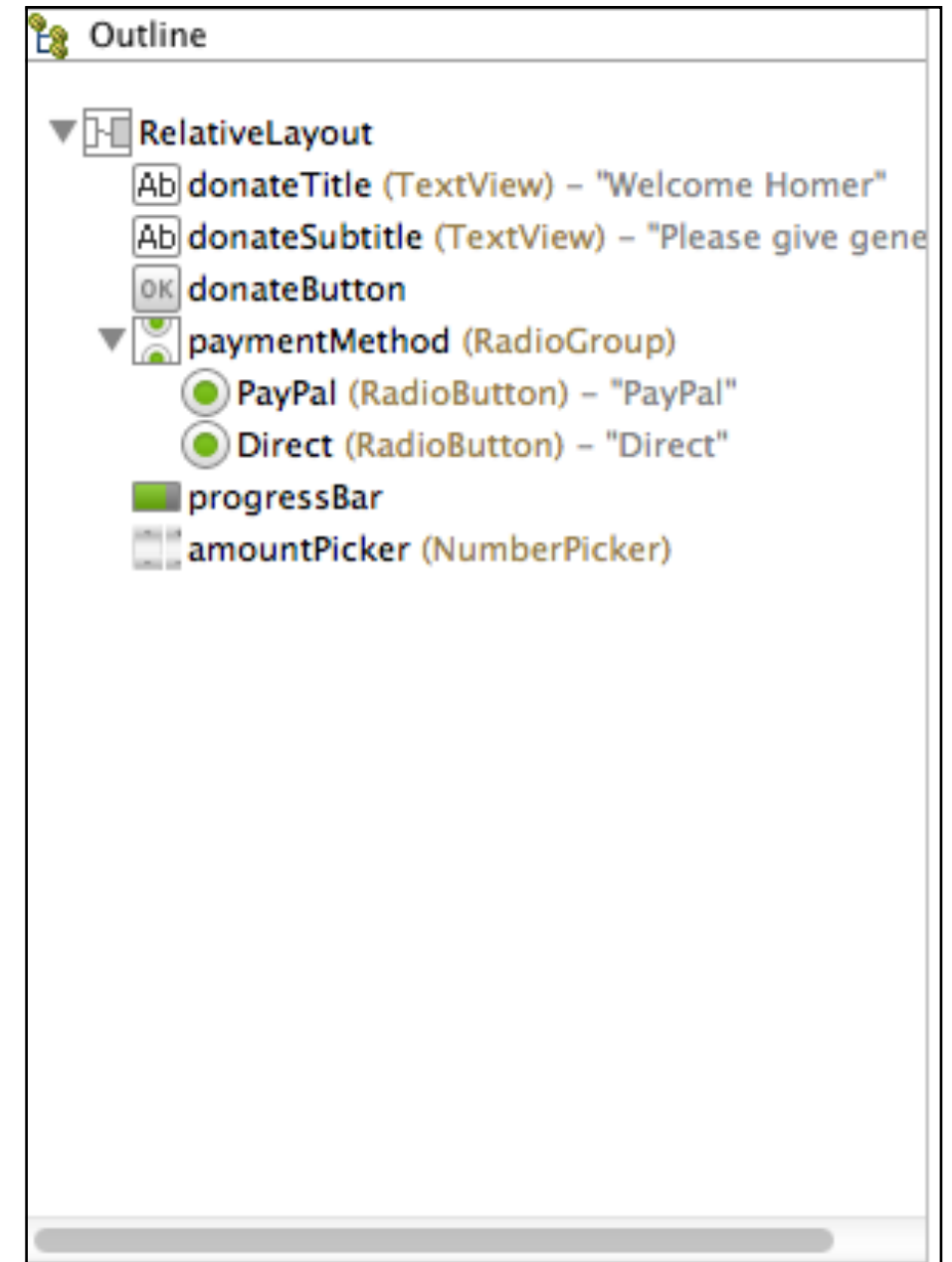
        amountPicker.setMinValue(0);
        amountPicker.setMaxValue(1000);
        progressBar.setMax(10000);
    }

    @Override
    public boolean onCreateOptionsMenu(Menu menu)
    {
        getMenuInflater().inflate(R.menu.donate, menu);
        return true;
    }

    public void donateButtonPressed (View view)
    {
        totalDonated = totalDonated + amountPicker.getValue();
        String method = paymentMethod.getCheckedRadioButtonId() == R.id.PayPal ? "PayPal" : "Direct";
        progressBar.setProgress(totalDonated);

        Log.v("Donate", amountPicker.getValue() + " donated by " + method + "\nCurrent total " + totalDonated);
    }
}

```




```

public class Donate extends Activity
{
    private int            totalDonated = 0;

    private RadioGroup    paymentMethod;
    private ProgressBar    progressBar;
    private NumberPicker    amountPicker;

    @Override
    protected void onCreate(Bundle savedInstanceState)
    {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_donate);

        paymentMethod = (RadioGroup) findViewById(R.id.paymentMethod);
        progressBar = (ProgressBar) findViewById(R.id.progressBar);
        amountPicker = (NumberPicker) findViewById(R.id.amountPicker);

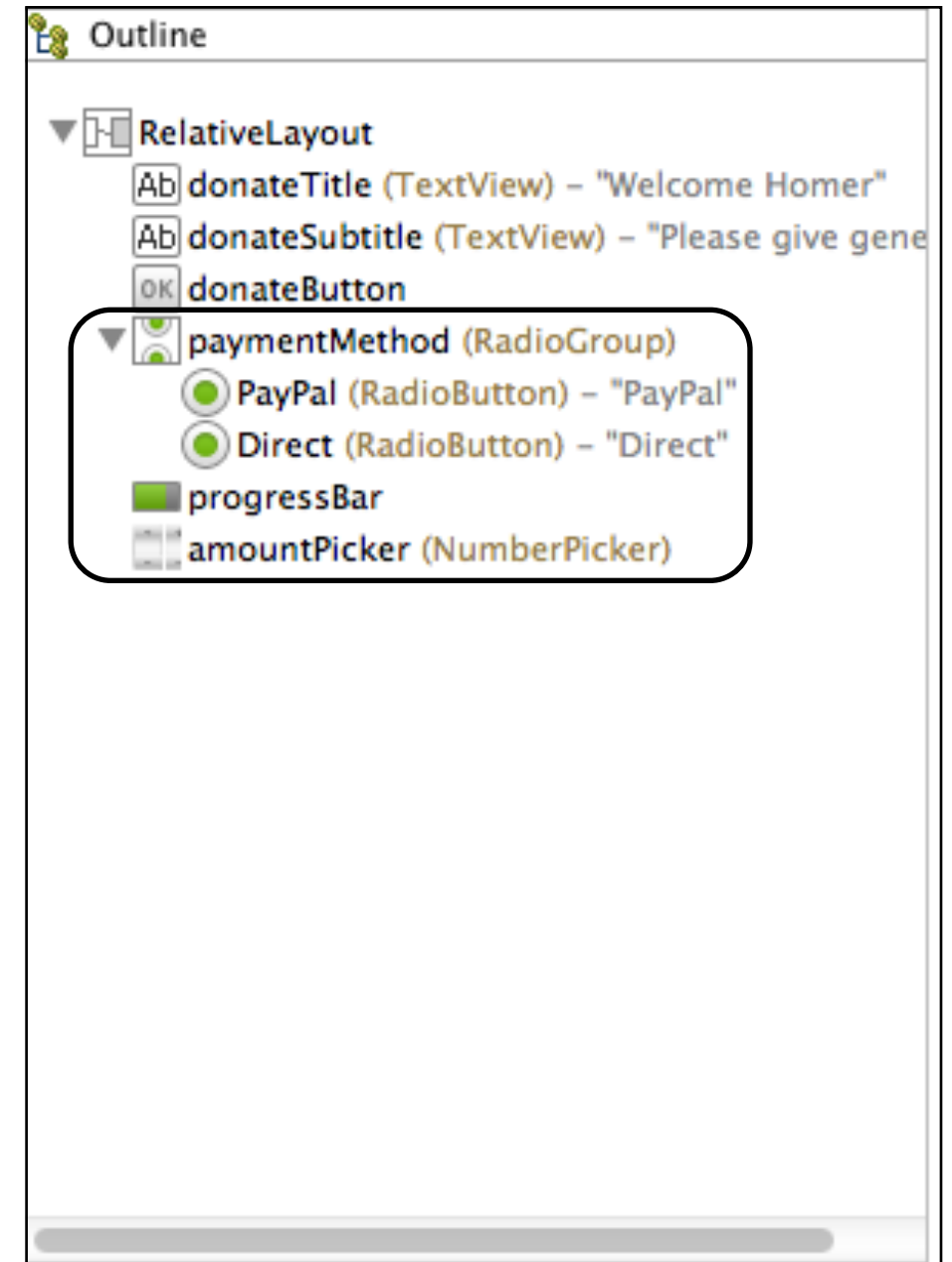
        amountPicker.setMinValue(0);
        amountPicker.setMaxValue(1000);
        progressBar.setMax(10000);
    }

    @Override
    public boolean onCreateOptionsMenu(Menu menu)
    {
        getMenuInflater().inflate(R.menu.donate, menu);
        return true;
    }

    public void donateButtonPressed (View view)
    {
        totalDonated = totalDonated + amountPicker.getValue();
        String method = paymentMethod.getCheckedRadioButtonId() == R.id.PayPal ? "PayPal" : "Direct";
        progressBar.setProgress(totalDonated);

        Log.v("Donate", amountPicker.getValue() + " donated by " + method + "\nCurrent total " + totalDonated);
    }
}

```




```

public class Donate extends Activity
{
    private int            totalDonated = 0;

    private RadioGroup    paymentMethod;
    private ProgressBar    progressBar;
    private NumberPicker    amountPicker;

    @Override
    protected void onCreate(Bundle savedInstanceState)
    {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_donate);

        paymentMethod = (RadioGroup) findViewById(R.id.paymentMethod);
        progressBar = (ProgressBar) findViewById(R.id.progressBar);
        amountPicker = (NumberPicker) findViewById(R.id.amountPicker);

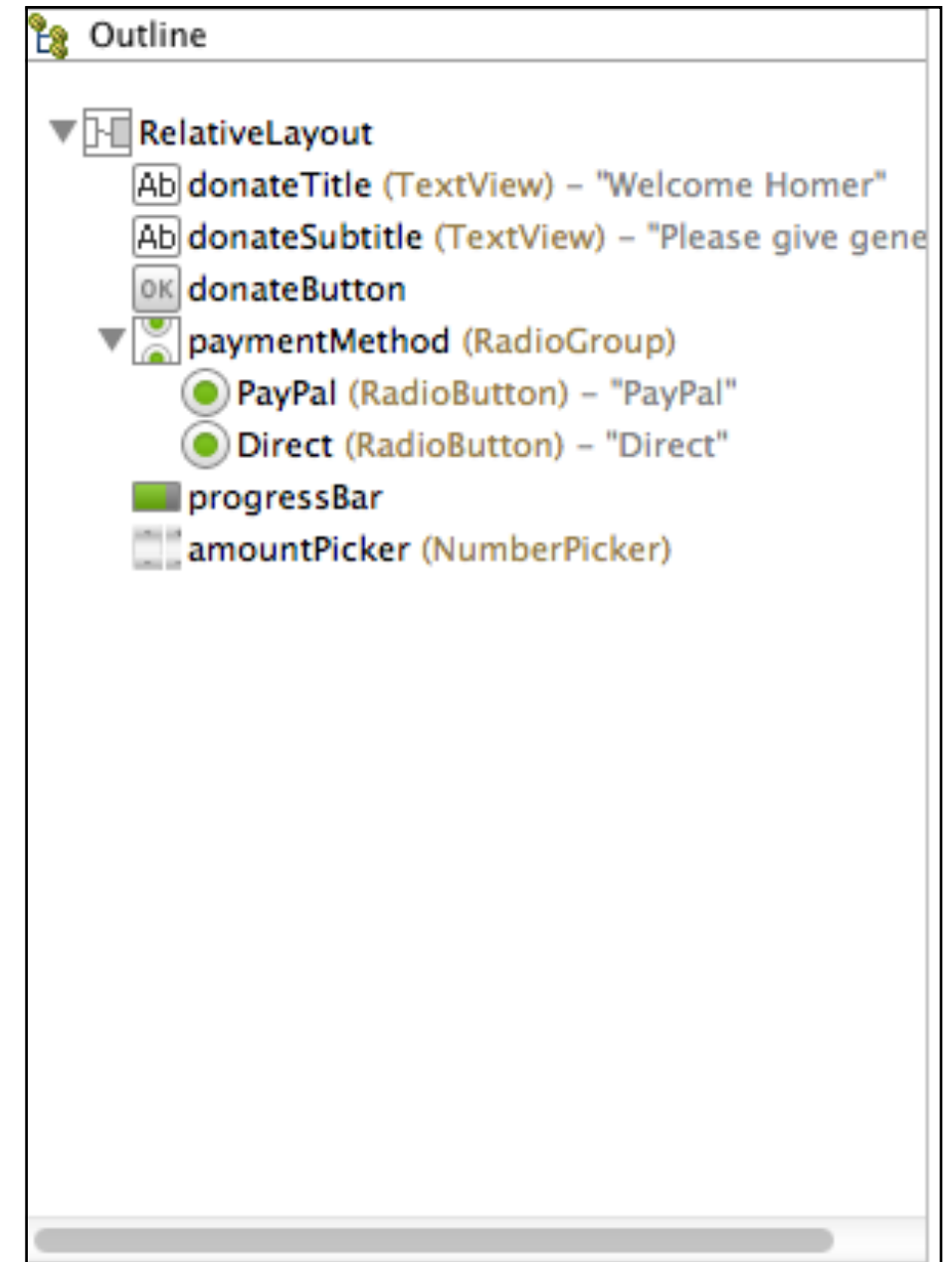
        amountPicker.setMinValue(0);
        amountPicker.setMaxValue(1000);
        progressBar.setMax(10000);
    }

    @Override
    public boolean onCreateOptionsMenu(Menu menu)
    {
        getMenuInflater().inflate(R.menu.donate, menu);
        return true;
    }

    public void donateButtonPressed (View view)
    {
        totalDonated = totalDonated + amountPicker.getValue();
        String method = paymentMethod.getCheckedRadioButtonId() == R.id.PayPal ? "PayPal" : "Direct";
        progressBar.setProgress(totalDonated);

        Log.v("Donate", amountPicker.getValue() + " donated by " + method + "\nCurrent total " + totalDonated);
    }
}

```



Donate Button Event Handler

```
<Button
    android:id="@+id/donateButton"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentBottom="true"
    android:layout_centerHorizontal="true"
    android:onClick="donateButtonPressed"
    android:text="@string/donateButton" />
```

```
public class Donate extends Activity
{
    //...

    public void donateButtonPressed (View view)
    {
        totalDonated = totalDonated + amountPicker.getValue();
        String method = paymentMethod.getCheckedRadioButtonId() == R.id.PayPal ? "PayPal" : "Direct";
        progressBar.setProgress(totalDonated);

        Log.v("Donate", amountPicker.getValue() + " donated by " + method + "\nCurrent total " + totalDonated);
    }
}
```



Getting Started ^

Building Your First App ▾

Adding the Action Bar ▾

Supporting Different Devices ▾

Managing the Activity Lifecycle ▾

Building a Dynamic UI with Fragments ▾

Saving Data ▾

Interacting with Other Apps ▾

Sharing Content ▾

Building Apps with Multimedia ▾

Building Apps with Graphics & Animation ▾

Building Apps with Connectivity & the Cloud ▾

Building Apps with User Info & Location ▾

Getting Started

Welcome to Training for Android developers. Here you'll find sets of lessons within classes that describe how to accomplish a specific task with code samples you can re-use in your app. Classes are organized into several groups you can see at the top-level of the left navigation.

This first group, *Getting Started*, teaches you the bare essentials for Android app development. If you're a new Android app developer, you should complete each of these classes in order:

Building Your First App



After you've installed the Android SDK, start with this class to learn the basics about Android app development.

[Creating an Android Project](#)[Running Your Application](#)[Building a Simple User Interface](#)[Starting Another Activity](#)

Adding the Action Bar



The action bar is one of the most important design elements you can implement for your app's activities. Although first introduced with API level 11, you can use the Support Library to include the action bar on devices running Android 2.1 or higher.

[Setting Up the Action Bar](#)[Adding Action Buttons](#)[Styling the Action Bar](#)[Overlaying the Action Bar](#)

App Components ▴

App Fundamentals

Activities ▾

Services ▾

Content Providers ▾

Intents and Intent Filters

Processes and Threads

Permissions

App Widgets ▾

Android Manifest ▾

User Interface ▾

App Resources ▾

Animation and Graphics ▾

Computation ▾

Media and Camera ▾



App Components

Android's application framework lets you create extremely rich and innovative apps using a set of reusable components. This section explains how Android apps work and how you use components to build them.

[APP FUNDAMENTALS](#) >

BLOG ARTICLES

Using DialogFragments

In this post, I'll show how to use DialogFragments with the v4 support library (for backward compatibility on pre-Honeycomb devices) to show a simple edit dialog and return a result to the calling Activity using an interface.

TRAINING

Managing the Activity Lifecycle

This class explains important lifecycle callback methods that each Activity instance receives and how you can use them so your activity does what the user expects and does not consume system resources when your activity doesn't need them.



App Components ▾

User Interface ^

Overview

Layouts ▾

Input Controls ^

Buttons

Text Fields

Checkboxes

Radio Buttons

Toggle Buttons

Spinners

Pickers

Input Events

Menus

Action Bar

Settings

Dialogs

Notifications

Toasts

Search ▾

Drag and Drop

Accessibility ▾

Styles and Themes

Input Controls

Input controls are the interactive components in your app's user interface. Android provides a wide variety of controls you can use in your UI, such as buttons, text fields, seek bars, checkboxes, zoom buttons, toggle buttons, and many more.

Adding an input control to your UI is as simple as adding an XML element to your [XML layout](#). For example, here's a layout with a text field and button:

```
<?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="fill_parent"
    android:orientation="horizontal">
    <EditText android:id="@+id/edit_message"
        android:layout_weight="1"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:hint="@string/edit_message" />
    <Button android:id="@+id/button_send"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="@string/button_send"
        android:onClick="sendMessage" />
</LinearLayout>
```

Each input control supports a specific set of input events so you can handle events such as when the user enters text or touches a button.



Android APIs API level: 18 ↕

[android.text](#)
[android.text.format](#)
[android.text.method](#)
[android.text.style](#)
[android.text.util](#)
[android.util](#)
[android.view](#)
[android.view.accessibility](#)
[android.view.animation](#)
[android.view.inputmethod](#)
[android.view.textservice](#)
[android.webkit](#)
[android.widget](#)
[dalvik.bytecode](#)
[dalvik.system](#)
[TabHost.OnTabChangeListener](#)
[TabHost.TabContentFactory](#)
[TextView.OnEditorActionListen](#)
[TimePicker.OnTimeChangedLis](#)
[ViewSwitcher.ViewFactory](#)
[WrapperListAdapter](#)
[ZoomButtonsController.OnZoo](#)

Classes

[AbsListView](#)
[AbsListView.LayoutParams](#)
[AbsoluteLayout](#)
[AbsoluteLayout.LayoutParams](#)
[AbsSeekBar](#)
[AbsSpinner](#)
[AdapterView](#)
[AdapterView.AdapterContextM](#)
[AdapterViewAnimator](#)
[AdapterViewFlipper](#)
[AlphabetIndexer](#)

public class

Button

extends [TextView](#)

[java.lang.Object](#)
 ↳ [android.view.View](#)
 ↳ [android.widget.TextView](#)
 ↳ [android.widget.Button](#)

- ▶ Known Direct Subclasses
[CompoundButton](#)
- ▶ Known Indirect Subclasses
[CheckBox](#), [RadioButton](#), [Switch](#), [ToggleButton](#)

Class Overview

Represents a push-button widget. Push-buttons can be pressed, or clicked, by the user to perform an action.

A typical use of a push-button in an activity would be the following:

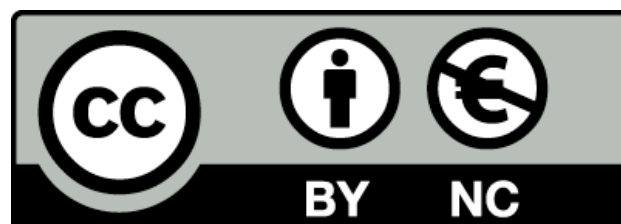
```

public class MyActivity extends Activity {
    protected void onCreate(Bundle icle) {
        super.onCreate(icle);

        setContentView(R.layout.content_layout_id);

        final Button button = (Button) findViewById(R.id.button_id);
        button.setOnClickListener(new View.OnClickListener() {
            public void onClick(View v) {
                // Perform action on click
            }
        });
    }
}
    
```

[Summary](#) |
 [Inherited XML Attrs](#) |
 [Inherited Constants](#) |
 [Inherited Fields](#) |
 [Ctors](#) |
 [Methods](#) |
 [Inherited Methods](#) |
 [\[Expand All\]](#)
Added in API level 1



Except where otherwise noted, this content is licensed under a Creative Commons Attribution-NonCommercial 3.0 License.

For more information, please see <http://creativecommons.org/licenses/by-nc/3.0/>



Waterford Institute of Technology
INSTITIÚID TEICNEOLAÍOCHTA PHORT LÁIRGE

