ISTE456 – Week 3

Lab: Intents

Objectives: Create Intents and then use them to activate Android Activities.

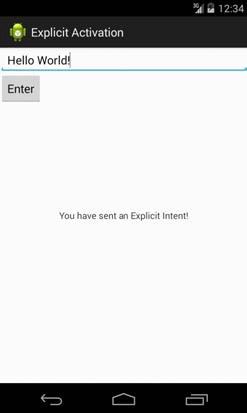
In this lab, you will use both explicit and implicit Intents to activate Activities. In one case, you will start an Activity using the startActivity() method. In another case, you will start an Activity by using startActivityForResult().

See http://developer.android.com/training/basics/intents/ for more information.

The main Activity for this application is called ActivityLoaderActivity. It should display two Buttons, one labeled “Explicit Activation” and one labeled “Implicit Activation.” It should also display one TextView, initially displaying the words, “No Text Entered”.

The application should behave as follows. When the user clicks the “Explicit Activation” Button, the ActivityLoaderActivity should launch a new Activity called ExplicitlyLoadedActivity. This activity should display a user interface containing an EditText and a Button. An EditText object allows a user to enter text.

ActivityLoaderActivity 🡪 ExplicitlyLoadedActivity 🡪 ActivityLoaderActivity



When the user clicks the “Enter” Button, this Activity should finish, returning the current contents of the EditText object (seen above as ‘Hello World!) back to the ActivityLoaderActivity. Once the ActivityLoaderActivity is again visible, any text that was returned from the ExplicitlyLoadedActivity should appear in the ActivityLoaderActivity’s TextView object.

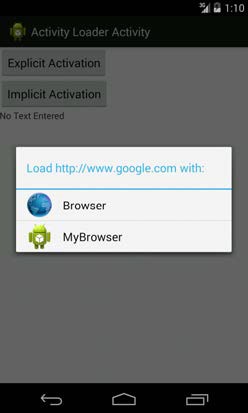
When the user clicks on the “Implicit Activation” Button, the ActivityLoaderActivity will create an implicit Intent, and then use it to implicitly activate a separate application to view the URL, “http://www.google.com”.

Because multiple applications may be able to handle this Intent, ActivityLoaderActivity will create and display an App Chooser (one example is shown below, but yours may have a slightly different layout), allowing the user to select the specific application to handle the Intent. For this lab, the Chooser should display two choices: Android’s built-in Browser and a separate application you’ve created called MyBrowser.

To create an App Chooser, start by creating an initial Intent to view a webpage, (as part of this process you'll need to use the Uri class’ parse() method). Then, create a second Intent, based on the first one, by calling the Intent class’ createChooser() method.

Finally, start a new Activity using this second Intent.

See http://developer.android.com/training/basics/intents/sending.html for more information about creating chooser Intents.



If the user selects the Android Browser from this chooser dialog, then that application will open and display the webpage at the given URL. If the user instead selects the MyBrowser application, then that application will open and simply display the text of the URL in a TextView.

In the Lab3a\_MyBrowser’s AndroidManifest.xml file, you will add the appropriate Intent Filter to MyBrowserActivity so that Android will know that this Activity can view web pages.

To indicate that the Activity can handle this type of Intent, you will need to add an <intent-filter> to the <activity> in Lab3a\_MyBrowser’s AndroidManifest.xml file.

Be sure to add the correct <action>, <category>, and <data> elements to an <intent-filter> element.

The skeleton code is located on MyCourses as 456IntentsLab. There are two projects inside this folder, Lab3a\_IntentsLab, and Lab3a\_MyBrowser

For Lab3a\_IntentsLab, the list of classes and methods you need to implement are explained below. You can find them in the code by looking for comments with the String TODO:

1. In ActivityLoaderActivity.java implement the following methods.

**private void** startExplicitActivation(): create an intent and start the ExplicitlyLoadedActivity**private void** startImplicitActivation(): create an Implicit Intent and use the Intent.createChooser() method to select an application for viewing a web page. What action string should you use? Activate the user-selected Activity using this Implicit Intent.

**protected void** onActivityResult(**int** requestCode, **int** resultCode, Intent data): get the data that is returned from the ExplicitlyLoadedActivity Activity and display it in the mUserTextView. How do you find and access that TextView?

2. In ExplicitlyLoadedActivity.java implement the following method.

**private void** enterClicked(): get user-entered text from the EditText, store it in an Intent, and return it to the ActivityLoaderActivity.

In Lab3a\_MyBrowser’s AndroidManifest.xml file, add the necessary intent filter tags. You'll need to define the right <action>, <category> and <data> elements. What values should be used for these? Remember that how you express certain values in Java can be different than how you express them in xml.

Show your working lab to the TA for sign off:

TA Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_