# Homework 3 – Fragments and BroadcastReceiver

**Due Date**: June 7th, Friday, at 11:59pm

**Second Deadline**: June 9th, Sunday, at 11:59pm

**Submission**: [gt10e@my.fsu.edu](mailto:gt10e@my.fsu.edu),

**Note**: **I will give you a rubric which specifies the spec that works and not. It also needs to tell me what extra credits you implemented. Please submit the rubric.txt file along with your .zip file.**

**Title: Mobile Programming Homework 3**

**I will reply to your submission email with “Got it” when I see it**

Check back to this document later in case there are updates

For this homework you will build an application that catches SMS intents and searches them for several keywords.

**REQUIREMENTS**

* This Homework can only be done in Android 3.0 or higher, so you’ll have to do this on the emulator if you don’t have a supported device
* Make the orientation of your app Landscape Mode
* You will need
  + 1 Activity
    - **MainActivity**
      * this will control both (2) Fragments since Landscape mode is good to show two Fragments side-by-side
  + 2 Fragments
    - **UrlListFragment**
      * this will show a list of URLs
      * it should have at least 3 default entries in the list, e.g.
        + <http://mobile.cs.fsu.edu/android>
        + <http://www.google.com>
        + <http://my-favoriate-website.com>
      * clicking on an item in the List should open up the website in the **MyWebFragment**
    - **MyWebFragment**
      * this will load URLs that have been clicked in **UrlListFragment**
  + 1 BroadcastReceiver
    - **SmsReceiver**
    - use an intent filter to listen for **SMS\_RECEIVED**, register the receiver inside of the Android Manifest file, **not dynamically**
    - when an SMS is received, your code should get the message from the SMS
      * **We will assume that you will only receive messages containing URLs**
    - your receiver should then launch a new instance of your Activity, which will add the new URL to the default list AND open the website immediately in the WebView
    - all entries (including the new one) should still be clickable
    - Your app should support SMS messages with URLs in any valid format, i.e.
      * http://www.example.com (Don’t need to modify)
      * http://example.com (Don’t need to modify)
      * www.example.com (Need to prepend http://)
      * example.com (Need to prepend only http:// , NOT www)

**THINGS TO NOTE**

* I don’t want to take the joy of figuring things out away from you, but if you can’t figure out
  + Post on the QA board!
* If anything is unclear, please ask me. I’m very willing to help
* Successive SMS messages WILL NOT contain any URLs in the **UrlListFragment** except for the default entries and the new entry (So never more than 4 entries in this example). That is OK.
* For Example
  + if your default list contains
* www.google.com
* www.apple.com
* www.amazon.com
  + Receiving an SMS message with www.microsoft.com will cause your list to now be
* www.google.com
* www.apple.com
* www.amazon.com
* www.microsoft.com
  + Receiving any additional SMS messages such as www.samsung.com will cause your list to now be
* www.google.com
* www.apple.com
* www.amazon.com
* www.samsung.com
  + It is OK if your app works this way
* You can spoof SMS messages to the emulator.
  + In Eclipse, Click Window > Open Perspective > Other
  + Choose DDMS
  + Click Window > Show View > Devices
  + Click Window > Show View > Emulator Control
  + In the Devices View, select the Emulator
  + In the Emulator Control View, under Telephony Actions
    - enter an incoming number (doesn’t need to be valid)
    - Select SMS
    - Enter an SMS message (in this case, you’ll want to enter a URL for this homework)
    - Click Send
* When you want to “update” your listview, you will probably need to do it by replacing the current visible UrlListFragment with a new instance of it.
  + This means that when you create a new instance of UrlListFragment, you will need to pass the data to it at that point using setArguments
  + When UrlListFragment has been started (in its onActivityCreated method), you will need to get the new entry at that point, add it to the list, and then set the list adapter

**GRADING**

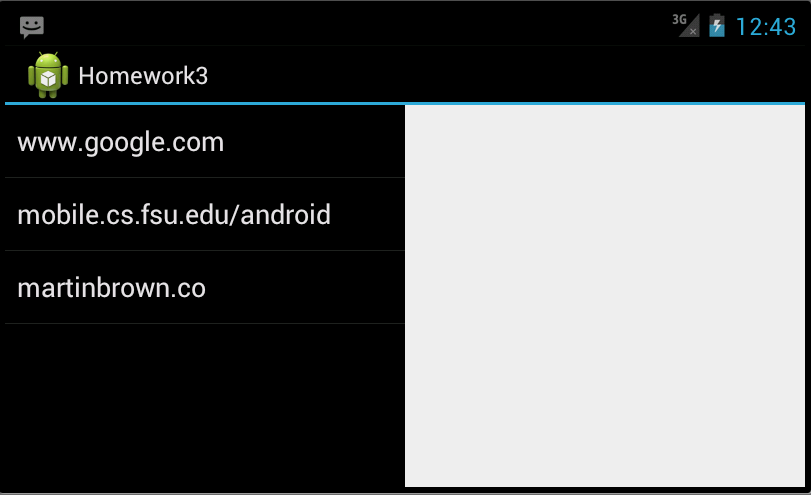
You cannot get a check-plus if your assignment is late

To get a check plus on this assignment

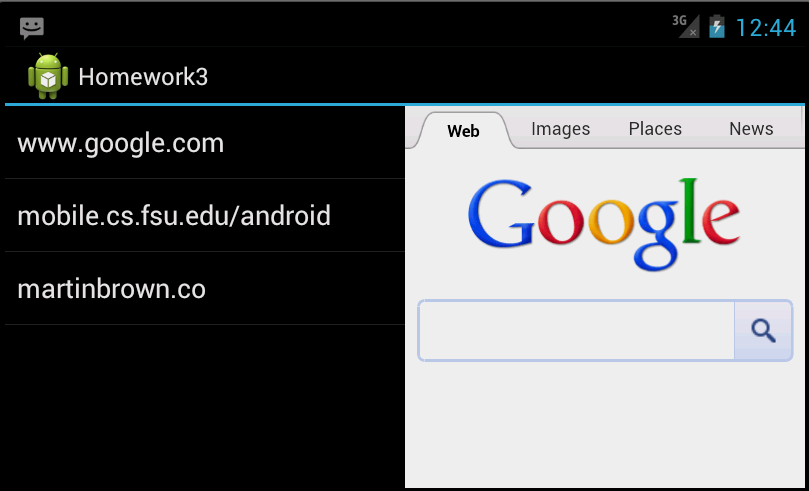
1. Have your app support Portrait Mode. In Portrait Mode, you can follow this design:
   * **MainActivity** will only support **UrlListFragment**
   * you should have a second Activity **WebActivity** to support the **MyWebFragment**
   * In Portrait mode, you should show the **UrlListFragment** alone at first
     + When an item is clicked, it should launch a new Intent to start **WebActivity**
     + **WebActivity** will in turn launch the **MyWebFragment**
       - either dynamically or by way of the fragment tag in XML
   * You will need 2 additional Layout XML files, 3 in total
     + one for supporting **MainActivity** and **UrlListFragment** in Portrait mode
     + one for supporting **WebActivity** and **MyWebFragment** in Portrait mode
     + one for supporting **MainActivity**, **UrlListFragment**, and **MyWebFragment** in **Lansdcape** mode
   * Switching to landscape mode should show both fragments, switching to portrait mode should show a single fragment
2. Use SharedPreferences to save the URLs that are in the list.
   * This means that even after you app is destroyed, even after the device is shut down and rebooted, that your list should contain the default URLs as well as the URLs received via SMS
3. If you have a different idea that requires extra work, email me about it and I will let you know if you can get a check-plus

**EXAMPLE**

**This is the app when it is first launched. The 3 entries in the list are the default entries.**



**This is the app after clicking on the first item in the list on the left**



**This is the app after receiving an SMS message where the contents of the message was “www.yahoo.com”**

