

ITIS/ITCS 4180/5180 Mobile Application Development  
In Class Assignment 7

**Basic Instructions:**

---

1. In every file submitted you **MUST** place the following comments:
  - a. Assignment #.
  - b. File Name.
  - c. Full name of all students in your group.
2. Each group should submit only one assignment. Only the group leader is supposed to submit the assignment on behalf of all the other group members.
3. Please download the support files provided with this assignment and use them when implementing your project.
4. Export your Android project as follows:
  - a. From eclipse, choose "*Export...*" from the File menu.
  - b. From the Export window, choose *General* then *File System*. Click *Next*.
  - c. Make sure that your Android project for this assignment is selected. Make sure that all of its subfolders are also selected.
  - d. Choose the location you want to save the exported project directory to. For example, your *Desktop* or *Documents* folder.
  - e. When exporting make sure you select *Create directory structure for files*.
  - f. Click Finish, and then go to the directory you exported the project to. Make sure the exported directory contains all necessary files, such as the .java and resource files.
5. Submission details:
  - a. When you submit the assignment, compress your exported Android project into a single zip file. The format of compressed file name is InClassAssignment#.zip
  - b. You should submit the assignment through Moodle: Submit the zip file.
- 6. Failure to follow the above instructions will result in point deductions.**

## In Class Assignment 7 (100 Points)

In this assignment you will make simple HTTP requests and will parse the iTunes RSS feed data. The Apple iTunes store provides a RSS feed for its different provided media types. In this assignment you will focus the iOS Apps RSS feed. The app is composed of 2 activities, namely **MainActivity**, and **PreviewActivity**.

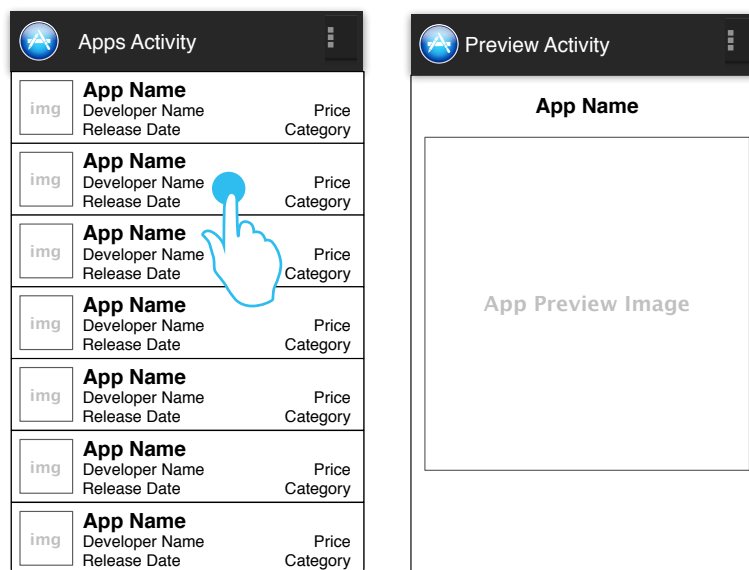
### Notes:

1. The recommended Android Virtual Device (AVD) should have minimum SDK version set to 11 and target SDK at least 17. The app should display correctly on 3.2" QVGA (ADP2) (320x480: mdpi).

### API Description

- The API used in this assignment is the iTunes RSS API. You should create a developer account and acquire an API key to be used for this application.
- The api used for this assignment should retrieve 300 top grossing applications provided by the iTunes store. Use the following api:

<http://itunes.apple.com/us/rss/topgrossingapplications/limit=100/xml>



(a) ListView MainActivity

(b) Preview Activity

**Figure 1, Application User Interface**

### **Part 1 (80 Points): MainActivity**

The interface should be created to match the user interface (UI) presented in Figure 1. You will be using layout files, and strings.xml to create the user interface. The **MainActivity** is responsible for the loading the top grossing apps provided by the iTunes store using the api. The implementation requirements include:

1. Use a thread pool (or AsyncTask) to communicate with the iTunes api and to parse the generate result. Do not use the main threads to download the api results. The parsing should be done careful and should take into consideration the namespace.

2. The retrieved apps' information should be displayed as list item in a ListView, see Figure1(a). You should setup the ListView and the required adapter.
3. Note that, you will need to create a custom layout and adapter in order to display the required list view items as shown in Figure 1(a). The application thumbnails should be retrieved by the getView method in your adapter by using an AsyncTask.
4. Tapping on a list item for a specific app item should setup the required intent to start the **PreviewActivity** and to pass it the app object.
5. Tapping on the device's back button should finish this activity.

### **Part 2 (20 Points): PreviewActivity**

This activity should display the application title, and the application preview image. The PreviewActivity should receive the app information from the MainActivity. Figure 1(b) shows the PreviewActivity.