

ITIS/ITCS 4180/5180 Mobile Application Development
Homework 03

Basic Instructions:

1. In every file submitted you MUST place the following comments:
 - Assignment #.
 - File Name.
 - Full name
2. Each group is required to submit the assignment on Canvas.
3. Please download the support files provided with this assignment and use them when implementing your project.
4. Submission details:
5. The file name is very important and should follow the following format:
Group#_HW03.zip
6. You should submit the assignment through Canvas: Submit the zip file.
7. **Failure to follow the above instructions will result in point deductions.**

Homework 03 (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 3 activities, namely Main Activity, Apps Activity and Preview Activity.

Important App Requirements:

Points will be deducted if your application does not follow the below requirements:

1. The app should display correctly on Nexus 5. Your assignment will not be graded if it does not meet these requirements, and you will not be granted any points on your submission.
2. All API calls, XML parsing and image downloading should be performed using Threads (or AsyncTask) and your code should not block the main thread.
3. Your implementation should target the most efficient algorithms and data structures. You will be graded based on the efficiency of your implementation.
4. **For this assignment you will be provided with a project folder which includes the activity initial implementations and UI. Import the provided project and add your implementation.**

API Description

- The API used in this assignment is the iTunes RSS API. The api retrieves 100 top grossing applications provided by the iTunes store. Figure 1 shows the XML response and the required fields. To access the XML stream issue a GET request to the below url:
 - <https://itunes.apple.com/us/rss/topgrossingapplications/limit=100/xml>

```
<?xml version="1.0" encoding="UTF-8"?>
<feed xmlns:im="http://itunes.apple.com/rss/" xmlns="http://www.w3.org/2005/Atom" xml:lang="en">
  <id>
    https://itunes.apple.com/us/rss/topgrossingapplications/limit=100/xml
  </id>
  <title>iTunes Store: Top Grossing Apps</title>
  <updated>2015-02-24T15:57:42-07:00</updated>
  <link rel="alternate" type="text/html" href="https://itunes.apple.com/WebObjects/MStore.woa/wa/viewTopGrossing?uo=us&id=38&popId=38"/>
  <link rel="self" href="https://itunes.apple.com/us/rss/topgrossingapplications/limit=100/xml"/>
  <icon>https://itunes.apple.com/favicon.ico</icon>
  <author>...</author>
  <rights>Copyright 2008 Apple Inc.</rights>
  <entry>
    <updated>2015-02-24T15:57:42-07:00</updated>
    <id im:id="529479190" im:bundleid="com.supercell.magic">...</id>
    <title>Clash of Clans - Supercell</title>
    <summary>...</summary>
    <im:name>Clash of Clans</im:name>
    <link rel="alternate" type="text/html" href="https://itunes.apple.com/us/app/clash-of-clans/id529479190?mt=8&uo=2"/>
    <im:contenttype term="Application" label="Application"/>
    <category im:id="6014" term="Games" scheme="https://itunes.apple.com/us/genre/ios-games/id6014?uo=2" label="Games"/>
    <im:artist href="https://itunes.apple.com/us/artist/supercell/id488166216?mt=8&uo=2">Supercell</im:artist>
    <im:price amount="0.00000" currency="USD">Get</im:price>
    <im:image height="53">
      http://a3.mzstatic.com/us/r30/Purple1/v4/10/ad/12/10ad12f3-d9ad-47f0-8130-fb1af2e26206/mzl.edpjwkgp.53x53-50.png
    </im:image>
    <im:image height="75">...</im:image>
    <im:image height="100">
      http://a5.mzstatic.com/us/r30/Purple1/v4/10/ad/12/10ad12f3-d9ad-47f0-8130-fb1af2e26206/mzl.edpjwkgp.100x100-75.png
    </im:image>
    <rights>© 2012 Supercell</rights>
    <im:releaseDate label="August 2, 2012">2012-08-02T01:24:58-07:00</im:releaseDate>
    <content type="html">...</content>
  </entry>
  <entry>...</entry>
  <entry>...</entry>
  <entry>...</entry>
  <entry>...</entry>
  <entry>...</entry>
</feed>
```

Figure 1 shows the XML response and the required fields. Red boxes highlight specific fields, and red arrows point from labels to these fields:

- Developer Name**: Points to the `<im:artist>` field.
- App Price**: Points to the `<im:price>` field.
- App ID**: Points to the `<id im:id>` field.
- App Title**: Points to the `<title>` field.
- App URL**: Points to the `<link rel="alternate" type="text/html" href=...` field.
- App Small Photo**: Points to the `<im:image height="53">` field.
- App Large Photo**: Points to the `<im:image height="100">` field.

Figure 1, Required XML Components

- You are required to access the api and retrieve for each app the following fields: id, app title, developer name, url, small photo url, large photo url, and app price. The required fields are highlighted in Figure 1.

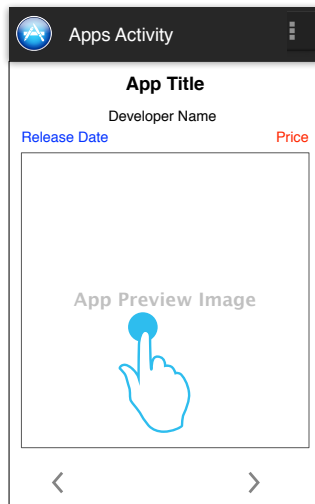


Figure 2: Apps Activity

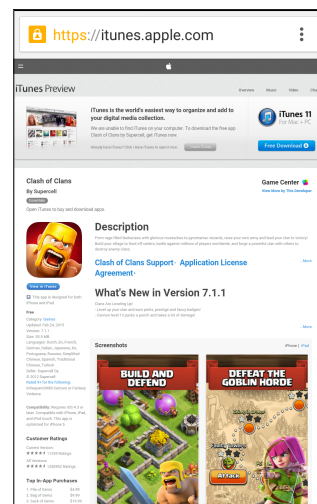
Part 1: Apps Activity

The interface should be created to match the user interface (UI) presented in Figure 2. The Apps activity is responsible for loading the list of apps retrieved from the api. The implementation requirements include:

- Use a thread pool (or AsyncTask) to communicate with the iTunes api and to parse the result. Do not use the main threads to download the api results. Care should be taken while parsing and should take into consideration the namespace.
- Display the details of the first app.
- For image loading/caching you must use the Picasso library. Use App large photo URL.
- For free apps, indicate the price as "0.0".
- Display Release Date
- Display Price
- Next and Previous navigates between apps details. Reaching the last app should navigate back to the first one. Navigating back to the first app navigates to the last app.
- Clicking on the "App Preview Image" ImageView should display the app url in the native browser, as shown in Figure 3(b).



(a) Apps Activity



(b) Native Web-Browser

Figure 3, Apps Activity