ITIS/ITCS 4180/5180 Mobile Application Development
Homework 5
Date Posted: 10/24/2012 at 01:30
Due Date: 11/06/2012 at 23:55

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. Your assignment will be graded for functional requirements and efficiency of your submitted solution. You will loose points if your code is not efficient, does unnecessary processing or blocks the UI thread.
4. Please download the support files provided with this assignment and use them when implementing your project.
5. 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.
6. 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 HW#.zip
   b. You should submit the assignment through Moodle: Submit the zip file.
7. **Failure to follow the above instructions will result in point deductions.**

Homework 5 (200 Points)

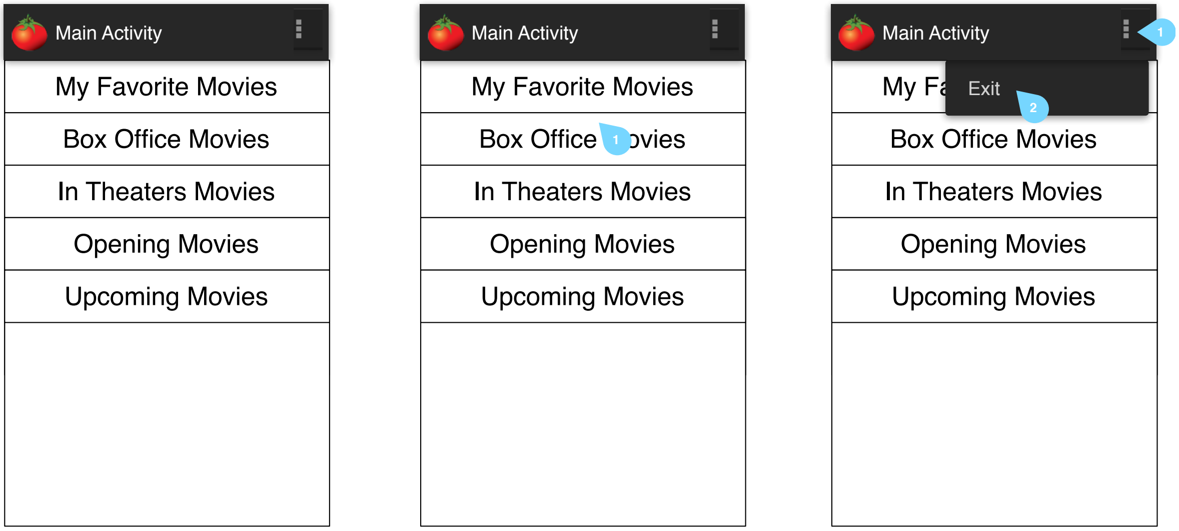
In this assignment you will develop a simple Rotten Tomatoes client application (see http://www.rottentomatoes.com/). The application should be able to load information about box office, in theater, opening, and upcoming movies. Users of the application will be able to add/remove movies to their local favorites. The favorites should be stored into a SQLite database. For this homework you will be using a JSON parser as data loaded from the rotten tomatoes API is in the JSON format.

Notes:

1. The recommended Android Virtual Device (AVD) should have a Target = Android 4.1, and a Skin = HVGA. All string values should be used via strings.xml.
2. All API calls, JSON parsing and image downloading should be performed using Threads (or AsyncTask) and your code should not block the main thread.
3. For further information about the rotten tomatoes API check the documentation available at http://developer.rottentomatoes.com.

Part A: Main Activity (20 Points)

The main activity displays a ListView with 5 options to the user, which include “My Favorite Movies”, “Box Office Movies”, “In Theaters Movies”, “Opening Movies”, and “Upcoming Movies”. Figures 1(a) and 1(b) show the Main Activity and the ListView. Clicking on any of these options should start the Movies Activity. Also note that clicking the options menu should show the “Exit” option when clicked the application should exit.



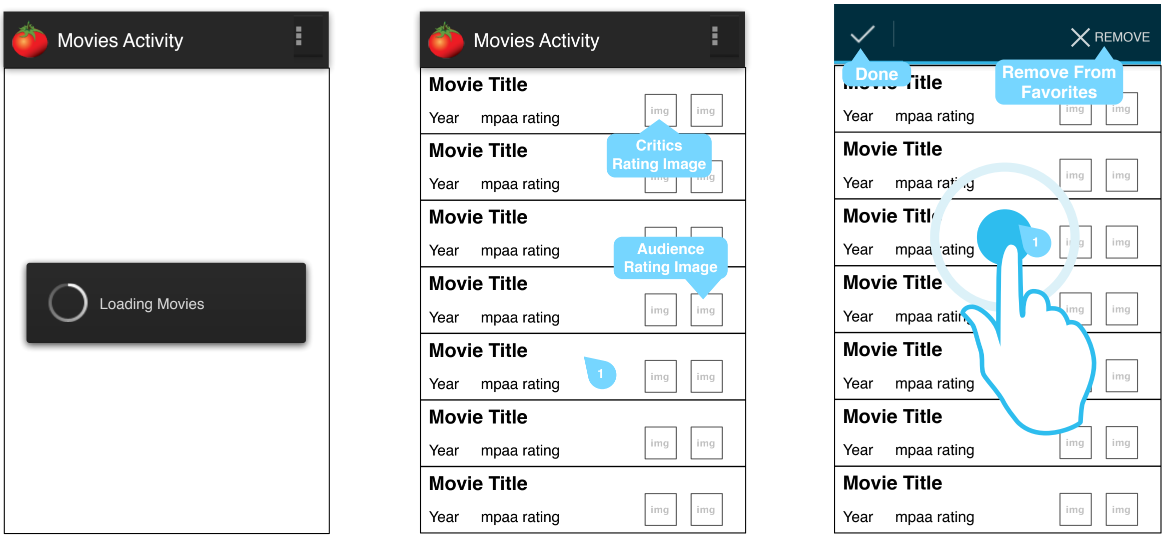
(a) Main Activity Starting (b) ListView can click options (c) The menu and exit

Figure 1, The Main Activity Wireframe

Part B: Movies Activity (120 Points)

The Movies activity is responsible for the loading the list of movies based on the selected option in the Main activity.

1) If “My Favorite Movies” was selected then the movies list should be loaded for the device local SQLite database. If the “Box Office Movies”, “In Theaters Movies”, “Opening Movies”, or “Upcoming Movies” options are selected then the corresponding JSON rottentomatoes APIs should be used.
    a) Check the API documentation (http://developer.rottentomatoes.com/docs).
2) The content of the JSON API should be retrieved by establishing a HTTP connection to the service, you should request at least **50 movies**. All the parsing and HTTP connections should be performed by a worker thread or an AsyncTask and should not be performed by the main thread. While, the movies are being loaded you should indicate that using the progress alert as indicated in Figure 2(a).
3) The loaded movies should be displayed in a ListView. The ListView items should be displayed to include the movie title, year, MPAA rating, critics rating image and the audience rating image, as shown in Figure 2(b). The audience and critic rating images are provided in the support files.
4) Clicking a movie item should display the movie details by starting the Movie activity.
5) When the Movies activity is displaying the Favorite Movies, then upon long pressing any of the favorite movies the activity should display an ActionBar as shown in Figure 2(c). The ActionBar should include a “Remove” menu item, which if clicked should remove the selected movie from the Favorite Movies. Note, removing a movie from the Favorite Movies should remove the movie recored from the SQLite database and should refresh the ListView to indicate this change.
    a) ActionBar info http://developer.android.com/guide/topics/ui/menus.html#CAB
    b) ActionBar icons http://developer.android.com/design/downloads/index.html



(a) Movies Activity Loading Movies

(b) ListView can on the different movies

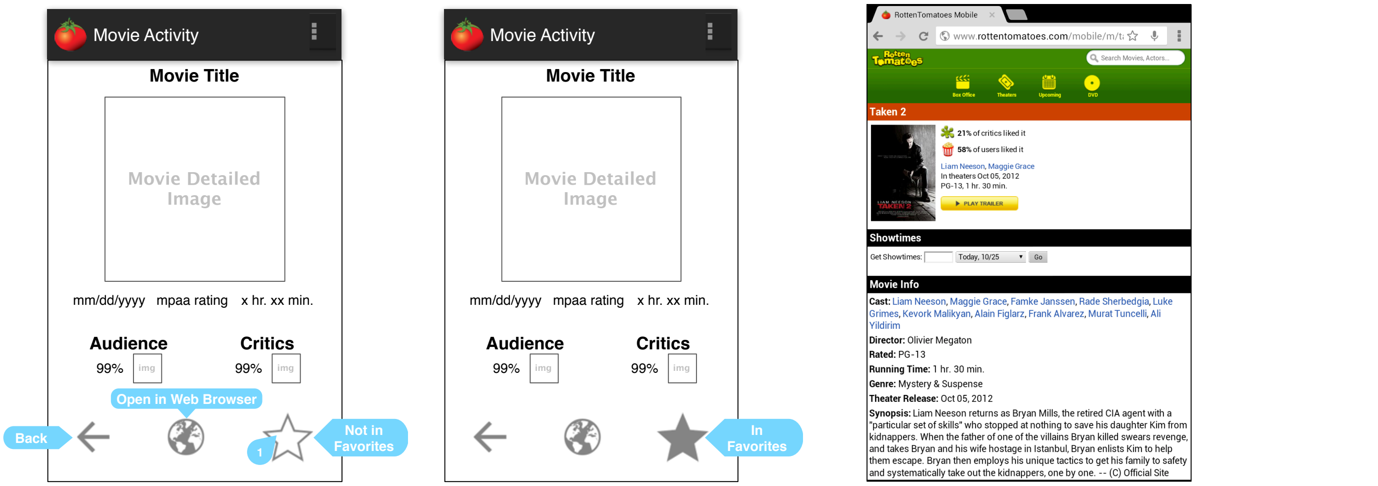
(c) Long Press displays the ActionBar

Figure 2, The Movies Activity Wireframe

Part C: Movie Activity (60 Points)

The Movie activity will display more details related to the movie selected in the Movies activity. The Movie activity wireframe is shown in Figure 3.

1) Loading the movie detailed image should be performed by a worker thread and not by the the main thread. If no poster is found use the “poster_not_found.png” image.
2) The audience and critics images are similar to ones used in the Movies activity.
3) The loaded movie is either in the favorite list or not. If the movie is not in the favorites this should be indicated by using the icon in Figure 3(a), and if it was in the favorites it should be indicated using the icon in Figure 3(b). Clicking not in favorites icon should add the current movie to the favorites and should change the icon to the in favorites icon. Similarly clicking the in favorites icon should remove the current movie from the favorites and should change the icon to the not in favorites icon.
4) Upon clicking the back icon (left arrow icon), the Movie activity should exit.
5) Upon clicking on the globe icon, the movie website should be loaded in the web browser, as indicated in Figure 3(c).



Taken 2

21% of critics liked it

58% of users liked it

Liam Neeson, Maggie Grace
In theaters Oct 05, 2012
PG-13, 1 hr. 30 min.

Showtimes

Get Showtimes: Today, 10/25

Movie Info

Cast: Liam Neeson, Maggie Grace, Famke Janssen, Rade Sherbedgia, Luke Grimes, Kevork Malikyan, Alain Figlarz, Frank Alvarez, Murat Tuncelli, Ali Yildirim

Director: Olivier Megaton

Rated: PG-13

Running Time: 1 hr. 30 min.

Genre: Mystery & Suspense

Theater Release: Oct 05, 2012

Synopsis: Liam Neeson returns as Bryan Mills, the retired CIA agent with a "particular set of skills" who stopped at nothing to save his daughter Kim from kidnappers. When the father of one of the villains Bryan killed swears revenge, and takes Bryan and his wife hostage in Istanbul, Bryan enlists Kim to help them escape. Bryan then employs his unique tactics to get his family to safety and systematically take out the kidnappers, one by one. -- (C) Official Site

(a) Movies Activity Loaded for movie not in favorites

(b) Movies Activity Loaded for movie in favorites

(c) Movie website loaded upon clicking globe icon

Figure 3, The Movie Activity Wireframe

Bonus:

1) (15 Points) Display the movie thumbnail when displaying the movie items in the Movies activity. To get credit for this bonus, you must load the thumbnails using threads and should not block the main thread. You should investigate reusing the views provided by the ListView adapter from the (scarp) recycle pool. In addition, the thumbnail images should be loaded beside their corresponding movie items.
2) (35 Points) In the Movies activity, for any selection made in the Main activity, upon long pressing any ListView item in the Movies activity should activate an ActionBar, which has menu items that enable the user to filter the movies displayed by their MPAA ratings. For example, selecting a menu item “PG-13”, would only include the movies that are “PG-13” in the ListView. The menu items should be dynamically generated when the Movies activity is first loaded by finding the list of unique MPAA ratings of the loaded movies. Also include a menu item “ALL” that will list all the initial retrieved movies.