Ser423/Cse494	**Mobile Computing**

Lab 7 Android Movie Collection Database

Due by midnight Tuesday March 29, 2016. No late submissions will be accepted.

Changes since distribution:

- No changes.

Description

Recast your Android Movie Library app to work with a **SQLite** database. Your app should include a minimal database in the app bundle, which is used to initialize the app's database should it not already exist in internal storage. Your app should provide the ability to add new Movie Descriptions, and to remove Movie Descriptions. You should also provide the ability to search Open Movie Database API. Use the **Json** representation returned by a successful search, providing the user the ability to add new descriptions into the apps database. For example with the search http://www.omdbapi.com/?t=Frozen&y=&plot=short&r=json returns the Json of an entry, whose details should be displayed by your app. Include at least the following fields: **Title, Year, Rated, Released, Runtime, Genre, Actors, Poster,** and **Plot**. You may limit the Genre you save to the first in the list returned by the **API**.

What To Hand-In

The same packaging, documentation and solution formatting constraints as with the prior assignments apply to this assignment as well.

1. Place all of your application source code into the package named: **edu.asu.bsse.asuriteid.appname** where **bsse** designates your academic program: **bscse**, **msse**, **bsse**, **bscs**, which stand for **BS Computer Systems Engineering**, **MS Software Engineering**, **BS Software Engineering**, or **BS Computer Science** respectively. **asuriteid** is your asurite id.
2. As part of all class header comments, that you create and turn-in this semester, include a copyright notice, such as: **Copyright 2016 Your Name**.
3. As part of the class header comments, include a **right to use** statement. The examples presented in class use the Apache License Version 2, but you should put whatever rights you prefer. At the very least, you must provide the instuctor and the University with the right to build and evaluate the software package for the purpose of determining your grade and program assessment.
4. As part of the class header comments, include a reference to the software's author, such as: **@author FirstName LastName   mailto:FirstName.LastName@asu.edu**.
5. As part of the class header comments, include an indication of the software version, such as: **@version March 2, 2016**

These comments are required on all code that you generate this semester. If you hand in an assigment that does not include them, it will not be graded.

Structure your project as a sub-directory of the folder named **Assign7MyASURITEID**. Use the same folder structure as used in the database example studentCourseDatabase.jar. The sub-directories should separately contain each of the following:

- **The Android app,** and
- **The Database Initialization files**.

You will submit this project, by first cleaning its Android app to remove all generated files. Then create a **jar** or **zip** archive of the project (Assign7*) directory. You can create a **Java Archive (jar)** by executing the following command from a terminal in the directory which is parent to the project directory:

jar -cvf Assign7MyASURITEID.jar Assign7MyASURITEID/

That archive will then be submitted via **Blackboard**. See the Content section.

Grading Criteria

- **5 points**. Your database definition and initializtion files.
- **5 points**. Your **Android** app should define and properly use network requests off of the UI/Main Thread to search the open movie database by using either AsyncTask or a separate thread with a Handler created on the Main Thread.
- **5 points**. Your Android app solution includes comments described above as headers for each class used in the apps.
- **5 points**. Your app provides the ability to list all movie titles, display the details of a movie description, search by movie title and display successful results, to add and remove movie descriptions.

Email: Tim.Lindquist@asu.edu | Ser423 Home