

# CS478: Software Development for Mobile Platforms

## Project #2

Due time: 7:00 pm on 2/24/2017

Submit using Blackboard web site

Total points: 100

Instructor: Ugo Buy

TAs: Venkat Sathyanarayanan and Jennifer Jesuraj

For this project you will design and code a new Android app that allows an interactive user to create a play list interactively and play songs from the play list. The main activity in the app displays a list of songs from various artists and/or bands. These are the songs from which a user will select the play list; the minimum number of songs is six, but feel free to have more. The activity allows a user to select one or more songs. The user can then choose to create a play list consisting of the selected songs from the original list. When this happens, the app displays a new activity containing a grid view with thumbnails of the (covers of the) songs selected by the user in the main activity. However, if no songs are selected, the main activity just displays a toast message alerting the user that she must select at least one song in order to create a play list.

Each item in the grid view can be clicked by a user. When this happens, a new activity containing web view is opened that will play a video segment playing the song. (You can choose from any video segment publicly available on youtube.) The device user can now watch the video and listen to the song. In addition, each item in the grid view supports long click functionality. When a user long clicks on a grid item, a context menu appears, which gives the user three options: (1) Play the video clip (similar to a simple click); (2) View a browser's page with the song's wikipedia page; and (3) View a browser's page with the artist (or band) wikipedia page. A user would return to the parent activity by pressing the device's "back" soft key when the video clip is complete or when done reading the wikipedia page.

The design of your project is subject to the following main constraints.

1. The list of songs in the main activity uses a checkbox for each list item. The user checks one or more items before creating the play list.
2. The main activity supports an options menu with the following functionality: (1) Create a play list out of the items selected by the user; (2) Clear all selections; (3) Invert selections; and (4) Check all items in the list. Option 1 should *always* be displayed as a shortcut in the app bar. Options 2 through 4 should be displayed in the app bar only if room is available. Otherwise, Options 2 through 4 are displayed in the overflow area in the right side of the app bar.
3. Feel free to add a button to the main activity to create and display the grid view with songs selected. (This functionality can also be invoked using the options menu.)
4. When launching the browser activity, your app should use the OS to find an appropriate browser, rather than choosing a specific browser hard-wired in your app.
5. Use a publicly available web site to play the song. Here is an example of a song by a band: (1) Comfortably numb, (2) Pink Floyd, (3) <https://www.youtube.com/watch?v=gBUTbHHH3I8>. Here is an artist's song: (1) Misirlou, (2) Dick Dale, (3) [https://www.youtube.com/watch?v=IRH\\_70\\_Foow](https://www.youtube.com/watch?v=IRH_70_Foow).

6. You should open the wikipedia pages in a way similar to the video clip, that is, using any available browser on your device, rather than a specific browser.
7. The list view in the main activity should contain no less than six items.

**Hints.** You may use the existing class *ArrayAdapter* to define the list view's adapter in the main activity. Use the list view's method *getChildAt()* in order to retrieve a check box (e.g., to find out if the check box is checked or not when building the play list). However, you will have to define your own adapter for the grid view as we saw in class for *ImageAdapter*.

**Implementation notes.** For this project use a Nexus 5 device running the usual Android platform (API 23—Marshmallow). Design your list view in such a way that it will display optimally in portrait mode, and reasonably well also in landscape mode. You are not required to provide backward compatibility with any previous Android versions.

*You must work alone on this project.* Submit the entire Studio project as a zip archive using the submission link in the assignment's page on Blackboard. No late submissions will be accepted.