CS 478 **Project #3** **Fall 2019**

Lists, Adapters, Menus

Due Date: Sunday, November 10th, 2019 @11:59pm

Project Details:

This project consists of three communicating apps, A1, A2 and A3, described below. The apps are supposed to work together in order to allow a user to shop for smart phones.

1. Application A1 contains two activities and a broadcast receiver. The main activity contains a welcome message and a button. When the button is pressed, the activity checks whether app A1 has obtained dangerous permission edu.uic.cs478.s19.kaboom. (This permission is defined in app A3 below.) If A1 does not have the permission, it requests it of the user. If the user grants the permission, processing continues by creating and registering its broadcast receiver programmatically. Subsequently, A1 launches the main activity in app A2 and enters the stopped state.
The broadcast receiver catches broadcast intents sent by A3 and launches the second activity in order to display the image of the smart phone.
The second activity is intended to display the web page of smart phones as requested by A3. The broadcast intent will contain an extra specifying the smart phone whose page must be displayed. The activity should be destroyed when the user starts a new activity on top of it. (It is automatically destroyed, when the user presses the “back” button.)

2. Application A2 consists of a single activity and a broadcast receiver. The activity is started by A1; however, this activity requires that the A1 activity have permission edu.uic.cs478.s19.kaboom. If A1 does not have the permission, A2’s main activity is not started. Otherwise, the main activity of A2 displays a welcome message and a button. When the button is pressed, the activity checks whether A2 was granted

permission edu.uic.cs478.s19.kaboom. If A2 does not have the permission, it requests it in a way similar to A1. If the permission is granted, A2 registers its receiver and then starts the main activity in application A3. If the permission is denied, A2 displays a toast message and terminates itself.

3. Application A3 contains a single activity that consists of two fragments. In addition A3 defines permission edu.uic.cs478.s19.kaboom. This application starts when its main activity receives the intent sent by A2, if A2 has permission edu.uic.cs478.s19.kaboom. In this case, A3's main activity is displayed with its first fragment.

App A3 maintains an action bar. The action bar shows the name of the application and an icon associated with the application (your choice). The action bar has an options menu that displays just two menu options: (1) launch applications A1 and A2 and (2) exit A3. When the first item is selected, A3 broadcast a single, ordered intent to start A1 and A2. Assuming that A1 and A2 have the permission, A2 will first display its toast message, then A1 will display the web page of the currently selected smart phone. If no smart phone is selected yet, no intent is sent and a toast message is displayed instead.

The main activity in A3 contains two fragments. The first fragment displays a list of smart phone names. (Each list item consists of a single string.) The device user may select any point from the list; when this happens, the selected item is highlighted. The second fragment shows an image of the selected item.

When the device is in portrait mode, the two fragments are displayed on different screens. First, the device will show only the first fragment. When the user selects an item, the the first fragment disappears and the second fragment is shown. Pressing the "back" soft button on the device, will return the device to the original configuration (first fragment only), thereby allowing the user to select a different smart phone. When the device is in landscape mode, application A3 initially shows only the first fragment across the entire width of the screen. As soon as a user selects an item, the first fragment is "shrunk" to about 1/3 of the screen's width. This fragment will appear in the left-hand side of the screen, with the second fragment taking up the remaining 2/3 of the display on the right. Again, pressing the “back” button will return the application to its initial configuration. The action bar should be displayed at all times regardless of whether the device is in portrait or landscape mode.

Finally, the state of application A3 should be retained across device rotations, e.g., when the device is switched from landscape to portrait configuration and vice versa. This means that the selected list item (in the first fragment) and the page displayed in the second fragment will be kept during configuration changes.

As for the order of execution of A1 and A2’s receivers, you should configure these apps in such a way that a receiver in A2 is always executed before the receiver in A1, after A3 sends a broadcast.

Implementation Details:

For this project use the usual Pixel 2 device running Android 9 (API 28—Pie). You are not required to provide backward compatibility with previous Android versions. Use of deprecated fragment classes is acceptable without penalty. Use method setRetainInstance() to prevent fragments from getting deleted when a configuration change occurs, resulting in the destruction of the containing activity. Check out the app Fragments Static Config Layout from Adam Porter’s Coursera course to see how to work fragment retention in A3 and app Fragments Dynamic Layout to see how to configure fragments dynamically. Don’t forget to add fragment transactions to the backstack.

Submission Details:

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

Academic Integrity:

Unless stated otherwise, all work submitted for grading *must* be done individually. While we encourage you to talk to your peers and learn from them, this interaction must be superficial with regards to all work submitted for grading. This means you *cannot* work in teams, you cannot work side-by-side, you cannot submit someone else's work (partial or complete) as your own. The University's policy is available here:

https://dos.uic.edu/conductforstudents.shtml.

In particular, note that you are guilty of academic dishonesty if you extend or receive any kind of unauthorized assistance. Absolutely no transfer of program code between students is permitted (paper or electronic), and you may not solicit code from family, friends, or online forums. Other examples of academic dishonesty include emailing your program to another student, copying-pasting code from the internet, working in a group on a homework assignment, and allowing a tutor, TA, or another individual to write an answer for you. It is also considered academic dishonesty if you click someone else's iClicker with the intent of answering for that student, whether for a quiz, exam, or class participation. Academic dishonesty is unacceptable, and penalties range from a letter grade drop to expulsion from the university; cases are handled via the official student conduct process described at https://dos.uic.edu/conductforstudents.shtml.