Project #5 (120 points)

Due Date

Monday, April 25, by 11:59pm.

Submission

Submit **a single zipped file** to Canvas. To ensure you have all the files needed for the grader to grade your Project 5. You can **zip the project folder**, which should include the AndroidManifest.xml, *.java, the gradle files, and all the resource files *.xml. When you submit the zip file, write a comment on Canvas to identify the API level of the app and the AVD you use to run your app.

- The zipped file MUST include the following grading items.
  (1) All files needed to run your project [115 points]
    - Java files (Activity files and Fragment files if you have any)
    - Layout files, i.e., *.xml files for the View Model.
    - Other resource files under the **res** folder, e.g., drawable/image files, values/string.xml, values/color.xml, etc.
    - AndroidManifest.xml and Gradle files.
  (2) Javadoc folder. [5 points]
- The submission button on Canvas will disappear after **April 25, 11:59pm**. It is your responsibility to ensure your Internet connection is good for the submission. **You get 0 points** if you do not have a submission on Canvas. DO NOT send the project to me or the graders through the emails.

Project Description

Develop an Android app based on the requirements of Project 4. You must meet all the functional requirements stated in Project 4 description, excluding the **Export** of store orders.

Project Requirement

1. You MUST follow the Software Projects Coding Standard and Ground Rules posted on Canvas under "Modules" "Week #1". You will lose points if you are not following the rules.
2. Each Java class must go in a separate file. **-2 points** if you put more than one Java class into a file. Inner classes defined for event handlers are fine.
3. You can use any Views to design your GUIs. However, you MUST use the following Views or Java classes, or **-5 points** for each violation.
    a) Toast
    b) AlertDialog
    c) ImageView
    d) Spinner
    e) ListView
    f) RecyclerView for ordering donuts
        - You must provide a list of at least 12 donut flavors, with all donut types combined
        - **-5 points** for not using the RecyclerView
        - **-5 points** for not having a list of 12 different donuts flavors in the RecyclerView
4. At least 2 Android Activities (*Activity.java) and their associated layout files (*.xml), or you will **lose 10 points**.
    a) If you need to share data between the Android activities, you use **public static** to define the variables.
    b) If you are using Fragment, then you can have a single Activity with multiple Fragments. Fragment is optional for this project.

5. You must remove ALL “hardcoded text” warnings. In other words, you must define all the texts to be displayed on the GUIs in the **string.xml** resource file. **-1 point** for each violation, with a **maximum of 5 points off**. You will not lose points on other warnings, however, try your best to fix all the warnings.
6. You MUST define the launcher icon for your app, or **-5 points**.
7. You are required to **generate the Javadoc** after you properly comment your code. Your Javadoc must include the comments for the constructors, private and public methods. You must include class comments with **@author** tags for all .java files. Generate the Javadoc in a single folder and include it in your project folder to be submitted to Canvas. You will **lose 5 points** for not including the Javadoc, OR, the grader cannot navigate your Javadoc through the “index.html”.

System Testing

1. Test design document is not required for this project. However, you are responsible to thoroughly test your software and ensure your software is meeting the requirements listed under **Project Requirement** and in **Project 4 project description.** You will **lose 2 points** for each incorrect implementation, incorrect amount, incorrect information shown on the GUIs, or each buggy behavior of your app.
2. Your software must always run in a sane state and **should not crash in any circumstances**. You must catch all Java Exceptions. Your program shall continue to run until the user stops the program execution or closes the window. **You will lose 2 points** for each exception not caught.