GroceryListManager:
An App for Managing Grocery Lists
(Deliverable 1–Preliminary Work)

Background

Brad and Janet are tired of doing grocery lists with pencil and paper and are unhappy with each and every existing app for managing grocery lists. Because they are rich and bored, they decide to hire a team of programmers to develop *GroceryListManager*, an Android app for managing grocery lists that works exactly the way they want it. Your team has been contacted by Brad and Janet, who provided the set of requirements listed below and want to see a possible design for the system before moving forward and formally hiring your team. Because Brad and Janet studied computer science in college and are familiar with UML, they want the design to be represented using a UML class diagram. Luckily, the members of your team already have experience designing systems of this kind, so the team should be able to hit the ground running and produce a good design quickly.

Requirements

See the requirements in Assignment 5.

Instructions

1. Create a directory called "`GroupProject`" in the **team repo** we assigned to you. Hereafter, we will refer to this directory as `<dir>`.
2. Create two directories, `Design-Individual` and `Design-Team`, under `<dir>`.
3. Each team member must copy both the UML design and the design information document he or she created for Assignment 5 in a directory called `<student's Github username>` under `<dir>/Design-Individual`.
4. Discuss and critique the different designs within the team.
   Important: Avoid being confrontational or defensive; keep in mind that the goal is not to judge each other's work, but rather to come up with a good design for the system.
5. Based on the results of the discussion, create a design on which the whole team agrees. The team design can be one of the team members' designs, possibly updated, a combination of two or more of the team members' designs, or a completely new design based on what the team members learned while doing Assignment 5 and during the design discussion.
6. Save the team design in directory `<dir>/Design-Team` as a PDF file named `design-team.pdf`.

7. Create, also in directory `<dir>/Design-Team`, a document in MD format called `design-discussion.md` with the following content:
    a. One section for each of the individual designs, called "*Design 1*", "*Design 2*", and so on, that shows that design (as an embedded figure) and discusses its main pros and cons as they emerged during the team discussion.
    b. One section for the team design, called "*Team Design*", that shows the team design (as an embedded figure), discusses the main commonalities and differences between this design and the individual ones, and concisely justifies the main design decisions.
    c. One final section, called "*Summary*", that concisely summarizes the lessons learnt in the process of discussing the designs, in terms of design, team work, and any other aspect that the team members consider relevant.
8. Commit and push `<dir>` to your remote repository and submit the commit ID on Blackboard. **Only the current project manager should submit the commit ID. The other team members should submit the full name of the project manager on Blackboard, to avoid confusion and possible inconsistencies.**

Important:

- Team members who have not submitted their individual design yet (i.e., late submitters) should not be involved in the discussion until they are done with their individual submissions.
- We will **not** use this deliverable to grade Assignment 5; in fact, we will not even look at it before we return your Assignment 5 grades. In other words, please be candid in your discussion about the individual designs and in reporting such discussion.
- Although you may want to get feedback on your individual designs before producing the team design, this would completely defeat the purpose of this deliverable, as our comments would clearly and heavily influence the discussion within the team.
- The fact that the system will be implemented on the Android platform should not affect your design, which should not contain Android specific elements (e.g., activities).