Farzad Hayatbakhsh

# Code Smells in SharingApp

## 1. Excessive Use of Comments

**Location:** EditContactAcitivity; SaveContact().
**Code Smell:** Excessive Use of Comments. The comments in the method are explaining functionality of the code that should be self-explanatory by just looking at the contents of the code itself.
**Reasoning:** This code smell is a problem because it is a sign that the code was not written clearly and needed to be supplemented by long comments. These comments can create unnecessary problems if a developer were to make changes to the method since the comments would also need to be updated to reflect the changes and a comment may be overlooked.
**Potential Solution:** Remove the excessive comments, and instead make the code easier to understand by using appropriate variable and method names that describe what is happening at each step.

## 2. Duplicated Code

**Location:** ItemList and ContactList; loadItems() and loadContacts(), saveItems() and saveContacts().
**Code Smell:** Duplicated Code. The contents of the loadItems(0 and SaveItems() methods in ItemList are nearly identical to loadContacts() and saveContacts() in ContactList aside from the difference in stored file type.
**Reasoning:** This will become a problem for anyone who is maintaining the code or making changes to the storage system of the project. Any changes that affect the loading and saving of files will need to be reflected across both sets of functions. This is both a time consuming process and leaves room for mistakes by potentially forgetting to update a piece of duplicated code.
**Potential Solution:** We can utilize separation of concerns by extracting any of the data reading/writing functionality into its own class. This class would handle the save and load methods which would accept any object type and filename. We can then use this class by calling it from inside the ItemList and ContactList classes.