Bryan Jimenez

Assignment 3 – OSS Proposal & Specifications

10/18/16

**Project Title:** Tab or MyTab (Work in Progress)

**GitHub Repo:** https://github.com/jimeneb1/TabOSS

**Option:** Option 2

**Platform:** Android OS Mobile Application

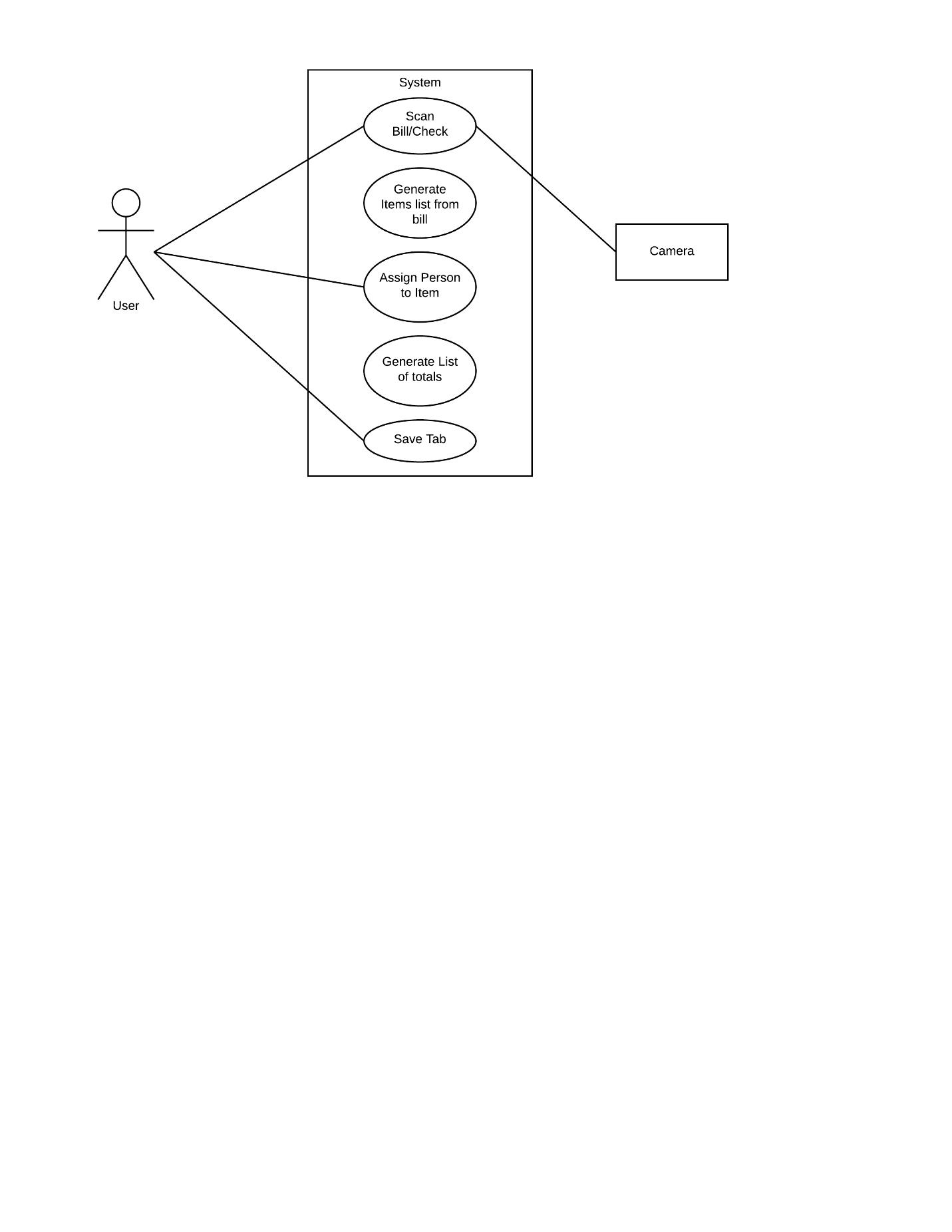
**Project Idea:** This application will use the device’s camera to obtain an image of a restaurant bill/check which is then split up between all parties involved.

**Rationale for Idea:** This idea is innovative because it eliminates the lengthy and confusing process of splitting up a bill total when the restaurant claims that they cannot provide individual checks for when you are eating out at a restaurant with friends or colleagues. When going out to eat, it is possible that one person may not have cash or a credit card on them at the moment, which is usually solved by one person paying for everyone’s meal, and later having the rest of the group pay that person back with how much they spent on food for themselves. Normally when this happens, you and your colleagues break out the pens and calculators to split up the bill, including tax and the tip amount, to see who is paying what. This application aims to eliminate that irritating process by simplifying the process which will be completed at the touch of a button. Added functionalities may include being able to keep track of who owes you or someone else, as well as how much, even after leaving the restaurant.

**Algorithm:** The algorithm I plan to implement involves using the Android Vision API for text recognition to obtain the items and their values from the bill/receipt. The reason I plan on implementing an algorithm for this function is because not all bill/receipt layouts are the same; they vary from restaurant to restaurant, so the camera must know what exactly to look for. The layout for most receipts, however, often contain the body/middle part of the receipt. With this knowledge I can narrow down the requirements for which pieces of text are items that were purchased. The algorithm aims to explicitly obtain only the items that were purchased along with their value, while disregarding all other information. This is necessary because there must be a set of rules for what text the camera identifies as items purchased, to avoid errors in functionality.

**Data Structure:** The data structure I plan on implementing for this project is a ArrayList. The reason I am choosing to use a ArrayList is because I will simply be storing the received data from the bill/check inside of a list which will allow easy access when it needs to be accessed again. I chose to use an ArrayList rather than a LinkedList as it will allow for more efficient use of the get() and set() methods, which work faster in ArrayLists than LinkedLists. Although LinkedList allows for faster execution of add() and remove() methods, the implementation in this case does not require an emphasis on these methods as the information will only be added in the stage of reading the bill, in which the data can be stored in a stack or list structure.

**New Concepts:** In creating this application, I will be working with Google’s Android Vision API’s to gain access to the functionality of the phone camera for optical character recognition (OCR). Using this API, I expect to learn new concepts around how to use computer vision to execute image to text conversions, which I expect to be very useful for future projects. I also expect to learn how to use a multitude of features in Android Studio, as well as reinforce my understanding of object-oriented programming.

**Use Case Diagram:**

**Plan of Action:** As I have had previous experience with object-oriented programming in C++ and Python, I plan to first acquaint myself with the syntax of the Java language. By exploring the nuances of the Java language I can reduce the amount of errors in syntax when developing. Once I have been acquainted with the language, I plan to explore the Android Studio IDE to familiarize myself with the interface and different functionalities. To do this, I plan on finding tutorials online that will provide me with more knowledge on how to navigate through different screens of an application, as well as tutorials that work with the flow of data between screens. After obtaining a good understanding of the IDE I will be working in, I will explore Google’s Android Vision API to learn how to use the camera’s text recognition functionality to develop the methods necessary to complete the given tasks. To do this, I have found the API on GitHub that includes samples for implementing the text recognition capabilities to add new functionality to the camera which I plan to explore.

**Open Source License:** When creating an open source software, it must be given a license that establishes the permissions allowed to other developers. Some popular licenses include GNU General Public License, Apache License, and MIT License. The GNU license applies to projects that anticipate heavy levels of contribution from outside developers as it gives them the freedom to make improvements and distribute the software under the same conditions of the original license. The MIT license is a simple, permissive license that gives outside developers the ability to use the code in any way they want, so long as they provide attribution back to you and don’t hold the original developer liable. Much like the MIT license, the Apache license is permissive but also provides an express grant of patent rights from contributors to users. After exploring the different licenses, I chose to implement an Apache license for my project as it will allow me to have the rights to the software reserved. I am not too concerned with having other developers create new contributions on the software, as it aims to have a straightforward functionality, therefore, the GNU and MIT licenses do not provide the best fit for my concerns for this project.