Bryan Jimenez

Assignment 4 Design

Scan Bill/Check: In this use case, the camera is used to scan the restaurant receipt where the Google Android Vision API will aid in obtaining the recognized blocks of words and save them into a temporary arrayList that will be used to save the items before assigning them to anyone. The user will also have the opportunity to revise the items and prices obtained from the scan to ensure that all items were obtained successfully; if not they can retake the picture of the receipt. The important actors in this case are the user and the camera. For this application, the only external actor will be the user as the app only allows for interaction with one user at a time. This is the intended use of the application. If the OCR proves to be too complex of a challenge, the app may need to be downgraded in its capabilities (which will be shown in the UI mockups). Another option will be to have the user input the party size and names, and food items bought and their price individually on separate activity screens. The user would then be able to attach the item to the party member who bought that item, where the application would then divide all totals individually. This option would remove the camera function entirely from the functionality of this app.

Assign Person to Item: In this use case, the user will land on an activity screen that already contains all the list of items, along with their prices, where the user will then have to input the names of all party members that are chipping in for that item. If an item is split between two people, the user must input all names of those chipping in, which must be separated by commas so that the system will recognize the different party members. The important actors in this case will be the user who will input all necessary information and continue to the next stage part of the process.

Save Tab: In this use case, the user will be able to save the totals received after completing the process of splitting up the bill using a button from the “Totals” screen. This is used so that if someone is unable to pay at the moment, the user will be able to keep record for who owes what for that meal. If there exists a match in names in the current saved Tab, the system will update the total that belongs to that friend/ party member. The space available to the Tab will be dynamic, most likely a hashmap, that will allow for the system to have easy access to the saved items when the user needs to edit the saved Tabs. The important actors in this case will be the user who will determine when to save and not to save the calculated totals.

Edit Tab: In this use case, the user will be able to add new people to the Tab, associated with the amount they owe, and remove current people saved in the Tab. In this activity stage, there will be icons on the top and bottom of the screen that clearly state what is done by pressing each of the buttons. At the bottom of the screen the user will be able to cancel the edit, which will take them back to the original Tab screen, or save the updated totals. At the top of the screen icons that represent “Add” and “Delete” will be visible so that the user can perform the necessary functions. The important actors in this case will be the user who will add and delete items saved in the Tab as they feel necessary.