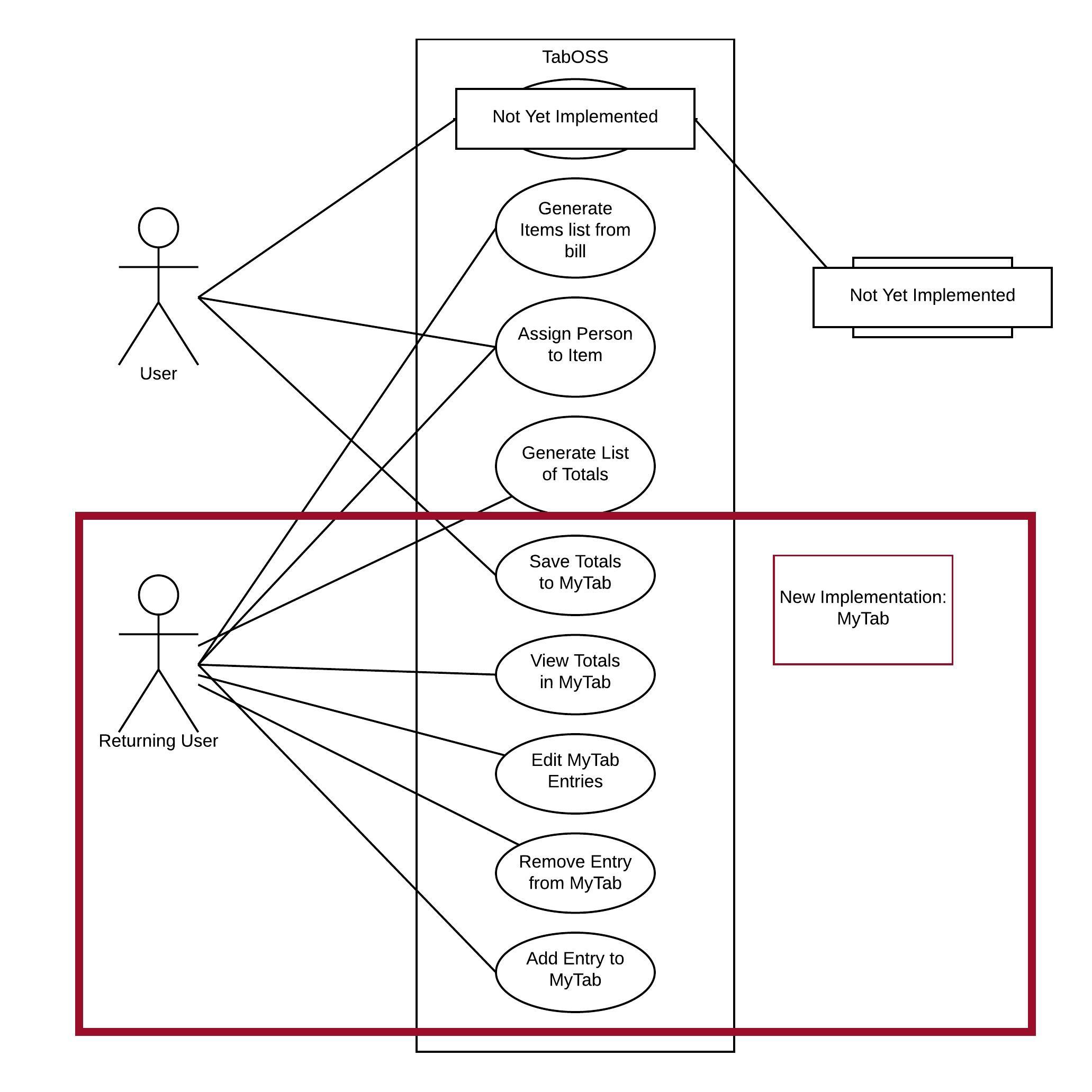
Bryan Jimenez

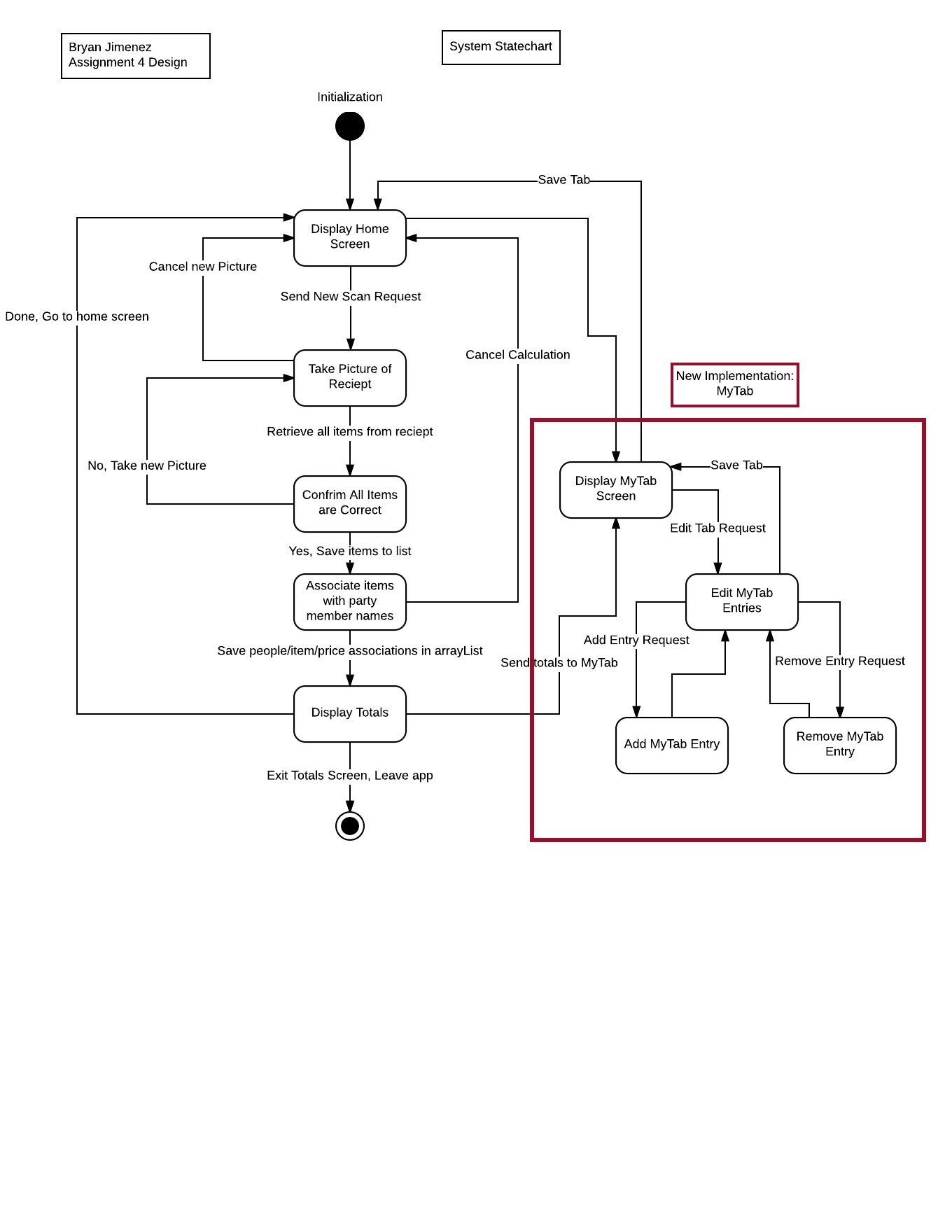
Final Evaluation Take Home

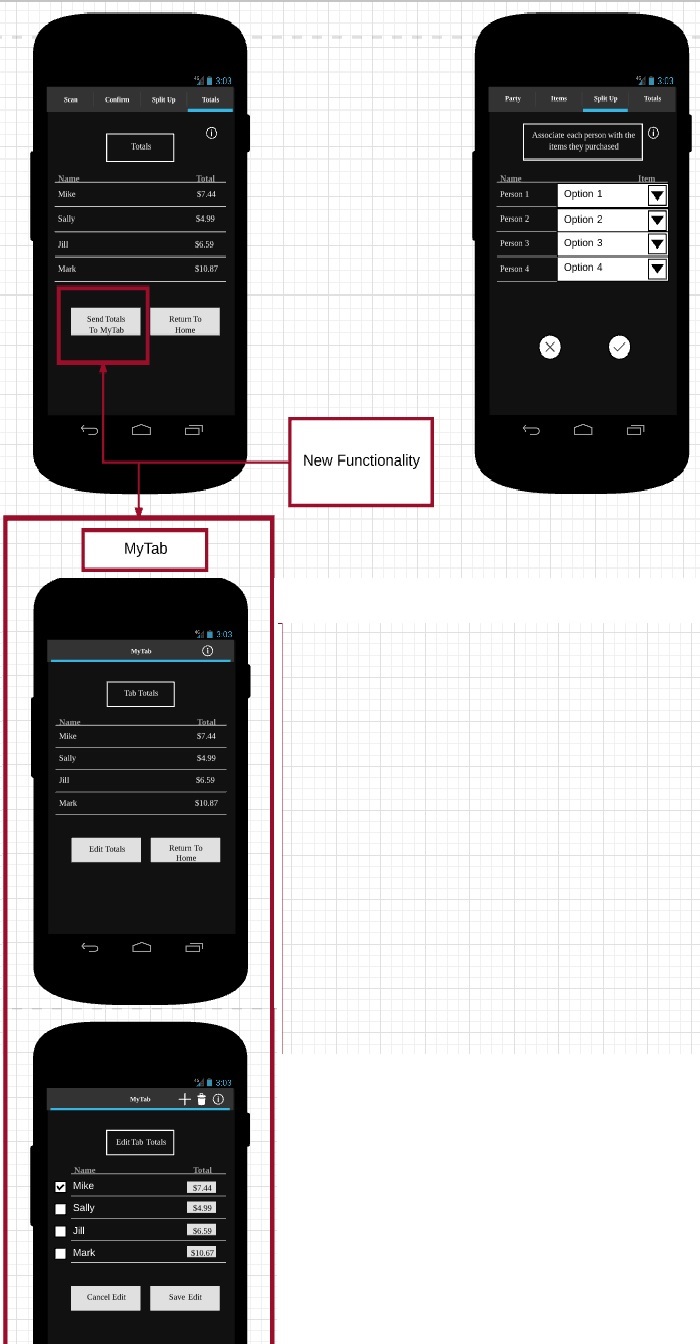
Project: TabOSS

Development Platform: Android Studio

Submitted: 12/16/16

1. Given the “moral imperatives” outlined in the ACM Code of Ethics, the intent of the two outlined points aims to encourage developers, and people of similar profession, to maintain the integrity of their work at every step of the development process. The “professional imperatives” also aim to outline more professional responsibilities when it comes to computational development. Computing professionals must strive for excellence in their work.
   1. **1.1 Contribute to society and human well-being.**
      1. This imperative identifies the need for computing professionals to affirm a safe, yet quality environment in the implementation and design of their work. Human wellbeing is a main concern as some products of computational efforts may at times be neglectful of the span of their impact in social environments. In addressing this imperative, I can confirm that the development and integrity of the TabOSS conforms to the ideas presented. By implementing a practical user interface, the user is ensured ease of use of the system as it outlines the intent of the application’s functionality to guide the user through a calculation of a party’s totals.
   2. **1.6 Give proper credit for intellectual property.**
      1. This imperative outlines the importance of maintaining integrity when using ideas gained from the work of others. It also defines how developers reserve the rights to their intellectual property. This imperative has been addressed in the development of the TabOSS as the License I am using for development is an Apache 2.0 License which preserve copyright and license notices. In addition, all code has been developed by my hand.
   3. **2.1 Strive to achieve the highest quality, effectiveness and dignity in both the process and products of professional work.**
      1. This imperative defines the need for creating high quality work when developing computational systems. Developers must recognize the problems that may arise with the creation of poor quality systems. This imperative has been addressed in the development of the TabOSS as the system works efficiently with the user as the main focus. In order to provide excellent quality, the user interface offers guided interactions so that the user knows exactly how to use the application to its fullest extent, whether they are a first-time user or a returning user. Clean, uncluttered UI demonstrates quality development practices.
   4. **2.2 Acquire and maintain professional competence.**
      1. This imperative outlines the need for computing professionals to take responsibility for maintaining professional competence. This means that professionals must stay up to date on new technologies and continue to expand their knowledge and skill base. By attending professional organizational meetings, one can develop their knowledge through collaboration of ideas. This imperative has been addressed in the development of the TabOSS as I had taken the time to learn a whole new programming language along with many new techniques for manipulating data between activities in a manner that depicts quality software, such as handling only specific data types for input fields to avoid system malfunctions.
      2. Project: TabOSS
      3. Development Platform: Android Studio
      4. Stakeholders of Original System: As Open Source Software, major stakeholders include myself, developers interested in the idea, Dr. Pulimood, the public, and possibly Venmo once the full intended system has been developed.
      5. Original Users & Functionality: The intent of the original system aims to serve socially active individuals to calculate individual totals for a restaurant bill/check.
      6. New Users & Functionality: The MyTab feature will add more functionality to the system allowing for users to reuse the system, instead of simply processing a single calculation and exiting the application until the next time they eat out, which may be frequently.
      7. GitHub Repo: <https://github.com/jimeneb1/TabOSS>
   5. New Implementation: MyTab
      1. The "MyTab" feature allows users that have previously used the application to save their calculated totals from the "Calculation" feature in order to view them at a later time. Returning users can access the MyTab feature from the application home screen by pressing the "Go To MyTab" button. In the MyTab screen, users at first will only view the current saved totals. By pressing an edit button on the bottom of the screen, the user can now change the total for an entry or multiple entries, remove an entry, and add a new entry to the running tab. At the end of the edit session, the user can save changes made to the saved tab. The information is then sent into a file that is saved locally on the device.
   6. From “TabOSS State Charts.pdf” on GitHub Repository:

****

* 1. ****From “TabOSS UI Mockup.pdf” on GitHub Repository:

|  |  |  |  |
| --- | --- | --- | --- |
| Functionality Tested | Inputs | Expected Outputs | Actual Outputs |
| Edit Entry Total in MyTab | Double values, strings, integers, and special characters will be inputted into the “Total” section of a MyTab entry to ensure that only double values can be stored while handling all errors effectively. Also test to ensure that user can exit edit menu without actually editing any entries. | Only double values and integers can be saved in the “Total” input field of a MyTab entry when saving changes made to MyTab list. User is notified for invalid entry and given suggestions for how to correct error. | Only double values are saved to update totals. User is notified for invalid entry and given suggestions for how to correct the error. |
| Add Entry in MyTab | Multiple entries for people from the initial use. Entries that are added after a calculation has been completed. Entries with the same name will be tested to ensure that the system will have a way to handle different people with the same name. | Saves all entries during an individual total calculation as well as maintaining running totals to names listed in MyTab. People with same names get stored as different entries. | Entries can be added through the add entry option on the “Edit Totals” screen and the entry list is updated according to input. |
| Delete Entry in MyTab | Select names to remove from MyTab. | Data updates successfully, Entry removed, MyTab updated. | Data updates successfully, Entries removed successfully, MyTab updated correctly. |

* 1. From “TabOSS Test Case Design.docx” on GitHub Repository: