**5-2 Milestone Three: Enhancement Three: Databases**

Joshua Brown

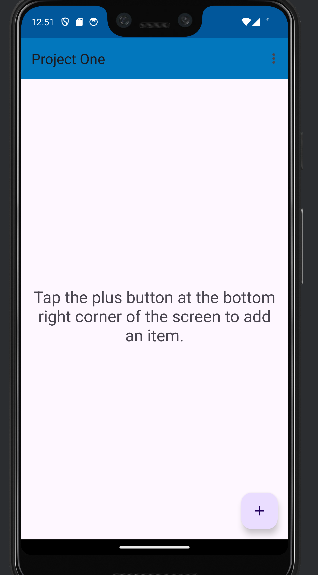
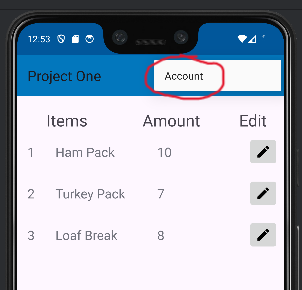
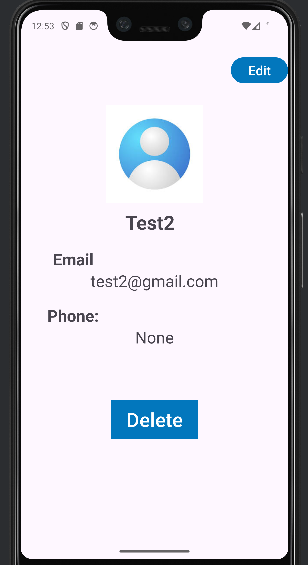
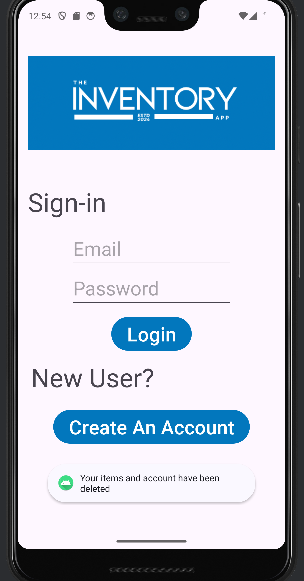
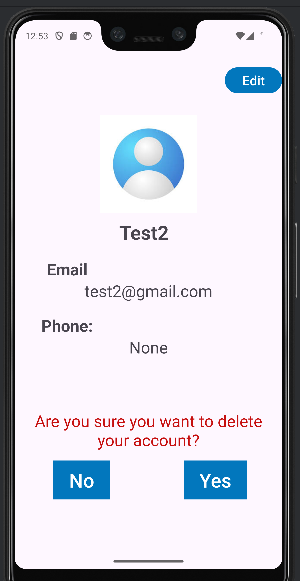
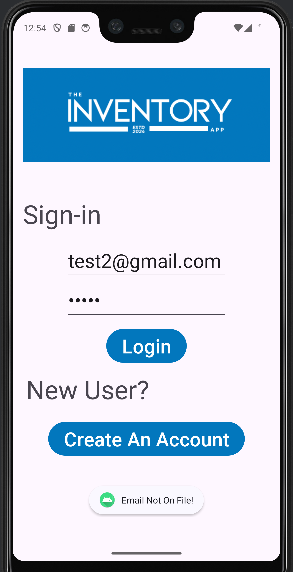
Department of Computer Science, Southern New Hampshire University

CS – 499: Computer Science Capstone

Mike Alesso, M.Sc.

December 7, 2024

**5-2 Milestone Three: Enhancement Three: Databases**

* **Briefly describe the artifact. What is it? When was it created?**
  + The artifact for this milestone is from the final project of Mobile Architecture & Programming Class. In this project, I chose to create an inventory app that allowed a user to create and track their inventory, as well as be notified when an item is out-of-stock. The original project was created on June 30, 2024.
* **Justify the inclusion of the artifact in your ePortfolio. Why did you select this item? What specific components of the artifact showcase your skills and abilities in software development? How was the artifact improved?**
  + I chose to include this artifact, because the proper implementation of the flawed designs would not only showcase my work as a developer, but also show my dedication to a secure and functional application. For this particular Milestone, I created a user account interface page that can be accessed from the menu on the Inventory screen. The user account displays the user’s name, email address, and phone number, along with the ability to edit and delete their account. If a user hits the edit account button, they are able to change their name, email, phone, and password associated with the account. In the user repository I created several update functions to handle the different cases of updates that would be required and tied those functions to the interface along with an update all items function from the inventory repository. This allows both the user’s items and their account to be updated accordingly when changes are made to the account. I also tied the delete all items and delete user account functions to the delete user button interface, so that all items would be properly deleted from the database along with the user’s account upon deletion.
* **Did you meet the course outcomes you planned to meet with this enhancement in Module One? Do you have any updates to your outcome-coverage plans?**
  + Everything went as planned for this enhancement. For the last Milestone, I started working on some of the user repository functions ahead of time that I thought I might need for this week. These functions would be needed to update and delete a user account and update the items; however, I ended up having to change quite a few of the functions along with adding some new functions in order to incorporate all of the different cases a user would need to update their account. I anticipated having a hard time with changing the user’s email and password in particular, but in all actuality, I only really had issues with changing the user’s password. When a user changes their password, I had it where the user’s items are decrypted from the database using the old password and then re-encrypted using the newly generated password and salt combination and had it updated in the database all in one integrated step. Afterwards the user’s password is then updated, having the user sign back in again; however, I kept having an issue where the encryption got mixed up, and old encryption remained along with the new one, making the text impossible to decrypt. I eventually fixed this issue by making the function decrypt all of the user’s items and put them in a list, and from there encrypting the items with the new password as they were updated in the database once again
  + I also added some cosmetic fixes to the inventory activity screen where it prompts the users to add an item if the list is empty. Finally, I made the SMS feature optional depending on whether a user has a phone number saved on file or not, to prevent the system from crashing from trying to send a text to a phone number that didn’t exist.
  + **Screenshot prompting user to add an item:**
  + ****
  + **Screenshot showing access to user account from main menu:**
  + ****
  + **Screenshot showing user account screen:**
  + ****
  + **Screenshot showing deleting a user account:**
  + ****
  + **Screenshots showing trying to access an account after deletion:**
  + ****
* **Reflect on the process of enhancing and modifying the artifact. What did you learn as you were creating it and improving it? What challenges did you face?**
  + **Item Class & Table**
    - Done in previous milestone: update item’s class and table to include creator and item ID. These changes make the changes in the software and algorithms milestones possible.
  + **User Class & Table**
    - Done in previous milestone: updated user class and table to include phone number and salt. These changes make the changes in the software and algorithms milestone possible.
  + **User Repository Update**
    - Created 3 new different update methods for User accounts:
      1. Update Correction Email: to update a user’s email account (along with name and phone number), but password and salt remains the same.
      2. Update Correction Password: to update a user’s password (along with name and phone number), but email remains the same.
      3. Update Correction All: to update all of a user’s information (email, password, and salt).
    - Thoroughly commented code
  + **Inventory Repository Update**
    - Refactored Update Correction Password, to take a list as an argument due to issues I was having with encryption when using a new password to encrypt an item with update all.
    - In the previous milestone, I developed the other update functions that I would need for this milestone.
    - Commented code thoroughly.
  + **Inventory Activity Update**
    - Included a text view that would prompt a user to click the plus button to add an item, if there are no items in the user’s list
    - Removed hardcoded phone number from SMS feature and had it use the current logged-in user. If the user’s phone number field is empty, then it will forgo utilizing the SMS feature, until the user updates their phone number field on the user account screen.
    - Added a taskbar on the layout screen, that included a main menu option to take the user to their account.
  + **User Account Activity Creation**
    - Created the visual interface for a user account, that includes the user’s name, email address, and phone number; along with an edit button to edit the user’s account and a delete button to delete the user’s account.
    - Created a user dialog fragment to handle user input and output the necessary strings to change a user’s account.
    - Tied user and item repository functions to update changes in the user and item database based on what fields the user updated in the edit screen.
    - Tied the inventory repository delete all function and the user repository delete user function to the delete button, so that a user can properly delete all items created by the user along with their account.
    - Thoroughly commented code.