4-2 Milestone Three: Enhancement Two: Algorithms and Data Structure

Vy Huynh

Southern New Hampshire University

CS499 Computer Science Capstone

Sherri Maciosek

November 22, 2024

The second artifact I have selected will be my Android Studio Project from CS 360. For this class, I built an Inventory Manager, this is an app that requires a user's username and password for login. After logging in, the user will be able to see their inventory list with all the items they have, as well as the total count of each item. Users can add, delete, quickly change the quantity of the item based on their choice. As well as overall delete their account entirely if they decided to. The goal of enhancement for this project is first checking if a username is a valid email. Then checking to make sure the password is a strong password that is at least 8 characters and also contains a capital letters and characters. Next is working on the interior of the application by adding a function to sort the table inside the application by name or quantity. This will make it easier for users to see which item is running low without scrolling or searching. Finally, I will add a search button that allows user to search for any item, this helps if there is a mass number of items and the user is looking for a specific item. I would say that I did meet the course outcome goal that I made during module one, my course outcome goal for this enhancement was "Design and evaluate computing solutions that solve a given problem using algorithmic principles and computer science practices and standards appropriate to its solution, while managing the trade-offs involved in design choices (data structures and algorithms)" and "Develop a security mindset that anticipates adversarial exploits in software architecture and designs to expose potential vulnerabilities, mitigate design flaws, and ensure privacy and enhanced security of data and resources." I would say I met both by first creating and sorting and searching algorithm, which was part of the course outcome. Furthermore, I also added a security feature which helps to migrate security flaws and make the overall applications more secure.

Reflecting on my overall experience of building and upgrading my inventory manager was a very fun yet challenging time. I had no idea where to start when I first decided to work on

this project. The last time I touched this code was probably last year, so I did a bit of a learning curve and a ramp up that I had to follow and learn. The first task that I tackled was figuring out how to add an email checker to my username and check for special characters in the password. Those two tasks were straightforward, as I found exact questions with the answer I was looking for on Stack Overflow. With that, I cleared the first two hurdles, which were the easiest hurdles I was going to clear in this enhancement. The next task I took on was making the search query for my inventory app. The first thing I did was add the search query to my XML file. There had to be a bit of tinkering because I had to make changes to make code that required me to rewrite the layout constraint, but that was a quick set of code to write and fix. The next task I took on was figuring out how to create the query function to search for the item in my database, I easily found the answer by referencing this page. With the knowledge of querying and my old SQLite function to reference to, I was able to easily write my query function. The challenges came when I realized that I needed to incorporate that with my search bar. I have never used SearchView before, so I had to do research on how to set it up and get it to connect to my application first. For this research, I referenced a page that I had all the basics of SearchView.

Breaking down what I have done so far for the search view, I add the search view to my XML file, I write the query for the SQL function, and I write the function to handle the search query. The one function I was missing was the function to actually filter the table. I didn't really have trouble with the function at first, I mainly copied and pasted the old code from forming the table originally and used "***cursor = inventoryDB.filterTable(username, sanitized Query);"*** instead of" ***cursor = {inventoryDB.getInventory(username)}*;**." Everything was technically working; however, I faced a couple of issues that I didn't know how to solve, the first issue that I faced was that the query duplicated once I searched for something and then exited the search

view, and the final issues that I faced was that my header would disappear once I delete the query and try to go back to the original table. At first, I tried to call“ ***setContentView(R.layout.activity_home_screen);*** ” before calling my function, but that broke my code, it caused my code to not be able to query more than once. I did a lot of research and googling, referenced a lot of different Stack Overflow pages (will be cited in the reference), and I found the solution to resolve my problem. The first issue about header missing, I just needed to add the header myself every time I move between different sections. The duplicate entry can be solved by clearing the whole table before pulling data and adding entries (link). Those two difficulties solved my entire issue with the search query, and I was done with this section, all I had to do was the sort query now.

For the sorting query, I didn't really have to do anything with the XML file, all I needed to do was give the "Item Name" and "Quantity" column an ID. Subsequently, I did a quick google where I stumbled on multiple articles and used the article to write my SQL function effortlessly. Thereafter, I created a text view for my item name and quantity with a flag called "sortOrder" this flag will be flipped every time the column is clicked. This click will sort the table in the opposite directions. It will then call the function "sortTable" taking the variable columnName (COLUMN_ITEM_NAME or COLUMN_QUANTITY) and (sortOrder). I didn't struggle to build the table or anything because the filterTable and the struggled I had when doing the SearchView did most of the heavy lifting for me. All I had to do was copy and paste and change a few things in my code. Overall, this artifact was a lot more challenging when compared to the first artifact that I worked on, this is because artifact one uses C++ and Python, two languages that I was very familiar with. Compared to this artifact, which uses Android Studio, I did not remember much about using Android Studio, so the ramp up was quite difficult. I had to

relearn how to code things like the XML files, and how to set up the button to get it working. Once I got a brief reminder and understanding how the files and code work, it was just a lot of research from there, just googling and trial and errors until the code worked as intended.

Reference

ADL. (2016, December 5). *Why is this activity missing its header?* Stack Overflow. https://stackoverflow.com/questions/40985395/why-is-this-activity-missing-its-header

andsec. (2017, October 24). *Android Room inserts duplicate entities*. Stack Overflow. https://stackoverflow.com/questions/46916388/android-room-inserts-duplicate-entities

Barger, J. (2009, November 9). *Android TableLayout Header row*. Stack Overflow. https://stackoverflow.com/questions/1698978/android-tablelayout-header-row

FarOoOosa. (2016, November 22). *Android listview header text disappear on refresh listview*. Stack Overflow. https://stackoverflow.com/questions/40743441/android-listview-header-text-disappear-on-refresh-listview

Fitzpatrick, F. (2011, September 4). *How to clear all rows from a TableLayout?* Stack Overflow. https://stackoverflow.com/questions/7302702/how-to-clear-all-rows-from-a-tablelayout

Forgia, P. (2019, January 17). *java - Check if a String contains a special character*. Stack Overflow. https://stackoverflow.com/questions/1795402/check-if-a-string-contains-a-special-character

Sachin Kainth. (2013, April 23). *Best Regular Expression for Email Validation in C#.* Stack Overflow. https://stackoverflow.com/questions/16167983/best-regular-expression-for-email-validation-in-c-sharp

SearchView Tutorial With Example In Android Studio | Abhi Android. (2019, April 8). Abhiandroid.com. https://abhiandroid.com/ui/searchview#gsc.tab=0

SQLite WHERE - Filter Rows in a Result Set. (n.d.). SQLite Tutorial. https://www.sqlitetutorial.net/sqlite-where/

ThatGuy343. (2015, January 21). *SQLite Select from where column contains string?* Stack Overflow. https://stackoverflow.com/questions/28059975/sqlite-select-from-where-column-contains-string

voila. (2011, May 5). *Need to show a SQLite query in descending order*. Stack Overflow. https://stackoverflow.com/questions/5900351/need-to-show-a-sqlite-query-in-descending-order