

CS 499 Milestone Four Narrative

This enhanced artifact is an inventory management application for android mobile devices running API 31+. I originally developed this application as part of my work in the CS 360 Mobile Architecture and Programming course. The application was made using Android Studio. It allows users to register for and sign into accounts and keep track of various inventory entries. The original program solved this problem in a simple sense, but there were some issues, namely when a user registered and logged into their account, they would have access to inventory items added and edited by other accounts. Additionally, the inventory management screen did not have any of the basic search and sorting functions that consumers have come to expect from tools that allow them to keep track of and analyze lists of items. I enhanced this artifact by first tying each user-created entry in the inventory database to a user ID unique to that user. I also rebuilt the queries used to populate the inventory screen with item entries so that only those items with a matching user ID to that of the currently logged-in user could be displayed and edited. I updated the inventory screen UI to support easy-to-use and straightforward search and sort functionality, and I wrote new queries to display the appropriate entries in the appropriate order based on the user's search and selections. While I originally also planned to add filtering, I quickly realized it didn't make sense to include filtering in this case and that search and sort functions would be more than adequate.

I chose to include this artifact in my ePortfolio under the Databases category because it highlights my proficiency in designing database solutions and managing and improving upon an existing DBMS. I also felt that the enhancements would demonstrate a security mindset in my work and specifically in properly securing the data entered and edited by individual users. These enhancements not only improved the usability of the inventory management application but also

ensured that user entries would be better protected from tampering by other accounts. From the moment a user logs in, their unique ID is tracked and used across the program to ensure that they only have access to their information. There are some minor tweaks I still want to make in polishing, including one issue I discovered late in testing where setting a sorting option and then using the search function can cause the items to revert to listing in ascending order by their item ID numbers, and this breaks the selected sort option until another sort option is selected. Overall though I believe I was able to clearly demonstrate my proficiency in database management, Android development, and UI design.

I was not initially looking forward to this project because I remembered it being a pain to figure out. However, after doing some initial research and refamiliarizing myself with Android Studio and this project specifically, I was able to divide my planned enhancements up into chunks and tackle everything I set out to do. Throughout the process, I also learned a lot more about Android Studio, and I am now far more comfortable with the IDE as a result of this project. The biggest challenge I ran into was during my first testing phase. My application continued to crash again and again on the registration/login screen when I attempted to register a new user. I ultimately ended up throwing a try-catch into the registration process and after finding the exception entry in Logcat, I realized I had not updated the version counter on my initialization of Room Database. Any time you make changes to the schema, which I had done, you are required to increment the version counter, and I had forgotten to do this, which resulted in data corruption issues when attempting to register. Updating the counter fixed the issue. This project provided me with some much-needed practice in bug fixing, troubleshooting, and secure database concepts.