CC-499 Capstone Project

5-2 Milestone Five: Enhancement Three: Databases

Name: Tyler Strickland

Professor: Fitzroy Nembhard

1. **Briefly describe the artifact. When was it created?**

   The artifact is an Android studio application I made in winter of 2023 for my mobile development class.

   What is it?

   It's a weight tracking application that records your weight according to a selected date.

2. **Justify the inclusion of the artifact in your ePortfolio.**
   As a up and coming developer I believe this artifact helps show my knowledge of Android development which I believe is a great starting point for my career in software engineering.
   Why did you select this item?
   I selected this object because it fits well in showing off my capability of using my learned practices in coding to improve a mobile project. A majority of developers will show their abilities in more popular languages such as Python and Java, although I want to present my abilities in Kotlin + Java to stand out in the mobile development market.
   What specific components of the artifact showcase your skills and abilities in software development?
   My priorities with this artifact lie in it's ability to show off major programming skills that a majority of projects will use such as its use of databases, the flow and processing of data between multiple classes and functions, and it's modularity that allows parts of the code to be reused and repurposed for other projects.
   How was the artifact improved?
   The current database for this application is SQLite. I implemented a search that allows the system to search for the users specific table by their user ID rather than checking each hash table one by one. Since the user ID is incremental the system searches by filtering past a certain set of tables (ordered by their userID as they'd be created in the same order as user's creating their acconts) depending on their user ID. For example, if the user ID is 12 then it would filter past 12 tables (starting at 0) then check if the next one is 12. If it's 12, great it pulls it. If it's above 12 then it begins to work backwards (there should be no case in which this happens as there's not currently a way to delete an account, but for the sake of error handling it feels like a

safeguard), and if it's below 12 it continues with it's original search function of going one by one. This doesn't make a huge difference on a small scale but would cut down search times exponentially in a commercial large-scale use as thousands of users could be skipped at once.

3. **Did you meet the course outcomes you planned to meet with this enhancement in Module One?**
   I believe I didn't meet course outcome three but I did meet course outcome five, as the implementation of the userID does help prevent users from accessing anothers information by changing their username to someone's old username. However, I don't think this enhancement provides any trade-offs, aside arguably the processing time of assigning a userID that would be negligible and a one-time process upon creating the account.
   Do you have any updates to your outcome-coverage plans?
   I don't have any outcome-coverage plan updates.
4. **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?**

   The implementation of the search algorithm wasn't that difficult, overall this module was surprisingly the most straight forward despite personally being unsure of my handling of a database.