**Milestone 3**

**Briefly describe the artifact. What is it? When was it created?**

This project was the first actual Android App I wrote for the final project for CS-360. It was designed to be an Inventory Management System, aptly named “Inventorio.”

I wrote it in February(?) of this year, so I was still somewhat familiar with the codebase.

The app takes a user log in, then allows the user to view all products within the local DB, which could be expanded to implement a web service call to an external DB since the table is dynamic.

This table will generate based on what products it pulls back and provides a way to Create, Read, Update, and Delete products, utilizing the necessary features of a Database. The app also provides a way for the user to enable “Low-Stock” alerts that will send an SMS message when a product falls below the threshold specified by the user.

**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?**

The app demonstrates my ability to work within the constraints of Android. This involves writing code that must request permissions to work within the Android Sandbox, and adhere to the security of the OS.

The app also demonstrates my ability to work with a database, by relying on a custom DB Helper Class that manages communication between the app itself and a self-contained MySQL database that is contained on the Device. This allowed me to design a modular product system that can ingest data from any source, as long as an interface exists that converts the data points to a Product Object. With this functionality, I can also update and delete data from the database (either internally or externally) thanks to the Primary Key being used within the product object as a key identifier (ProductID). This allows for the main functionality of the app to be independent of the data source, whether it is a Database or a Web Service Call.

Improvements include:

* More Metadata on Products
* Delta Updates on Update Calls for the Product
* Data Truncation to allow for better readability in Portrait
* Dark Mode fix so that Odd-row table entries can be read on Physical Devices

**Did you meet the course objectives you planned to meet with this enhancement in Module One? Do you have any updates to your outcome-coverage plans?**

Yes, I met the course objectives for this. There were a couple alterations I made to my initial plans to fit time constraints. I was able to add metadata to the products to expand their usability. I updated the table output to be more readable in both orientations. I also implemented delta updates to products, that only write datapoints if they differ from what is already stored. This will allow for less resources being used during a Write cycle.

**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?**

After spending the last 4 or so weeks writing in C++, it took me some time to switch gears back to Java. Having a way to compare against Null again, without the scare that the app will implode on itself was nice for detecting when products exist or not, but man… the boilerplate code is real…

I write SQL code daily for work, so that part came naturally. I did have to relearn the intricacies of modals and UI/UX development on Android, but other than that, there weren’t any challenges worth mentioning. Most issues I ran into was just mis-referencing or typos or realizing that rendering on a physical device rather than the emulator was a very different beast and that most of my design with illegible on Dark mode, so I had to spend quite a bit of time re-evaluating the theme.

**References**

Google. “Material Design.” Material Design, 2023, m3.material.io