Milestone Two Narrative

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

The artifact selected for the software design and engineering portion of the ePortfolio is the Inventory Manager application created as the final project of my mobile applications development course, originally created in 2022. The application offers an inventory management solution for individuals or organizations through the use of a simple mobile interface.

2. 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?

This artifact was selected as it displayed knowledge in multiple technologies while also providing opportunities to show growth in both software design and engineering and databases.

The primary improvements made to the Inventory Manager app are as follows: the app can now properly remember inventory information between sessions, the inventory can now hold more than four objects, objects can now be added to the inventory, all three inventory count buttons have been assigned proper functionality and all share a single callback function (greatly reducing

code redundancy), the XML files were modified to remove redundancy using styles, security was implemented, and thorough comments were added.

3. 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?

Throughout the program's modification, the two course objectives planned for this milestone were met. Through the elimination of program redundancy, and the creation of high quality and informative comments and formatting, strategies were employed for building collaborative environments that enable diverse audiences to support organizational decision making in the field of computer science. With the program modified in such a way as to have become much more readily modifiable, extensible, and understandable, working cooperatively with other software engineers or communicating program details with stakeholders is now much more intuitive.

Additionally, the implementation of the originally planned security features displayed 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. The program now successfully and safely requires the use of a user account in order to modify stock items and their respective amounts. Additionally, account creation, which originally remained unimplemented, was added in a way that

responded to invalid input appropriately. All security functions were protected from injection attacks though the parametrization of SQL queries, ensuring that no such attack vector can be used within the application.

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?

While trying to fix the issue of code redundancy, I learned how to better utilize XML and Java together to avoid repetition in callback functions or in the XML files themselves. Although I was unable to figure out how to fix this problem during the applications initial development, through the use of XML IDs and tags, this problem was able to be solved in an elegant and easy to understand fashion. With these newly found capabilities, it has become much easier and more intuitive for me to program using the MVC design pattern.

Another important lesson I learned during the program's modification was how to better reuse screens for the maximization of efficiency. During development, I was unsure how to display an indefinite amount of stock items. In an attempt to alleviate this issue, I chose to see if reusing the inventory screen would be an effective solution. By implementing additional navigation buttons, I was able to reuse the one screen in a way that made it seem like an indefinite amount of other screens, allowing the user to navigate around their entire

inventory. This was done by simply having the navigation buttons increment an index variable, and redrawing the screen using the inventory items selected using the index variable. In the end, I was able to turn a confusing problem into an elegant solution, resulting in an invaluable software engineering lesson learned.

The primary challenge that I faced with this milestone was its need for an early implementation of the SQL database. While I did attempt to create this milestone without the need for the premature addition of the database, the ultimate plans for the project were built upon the planned SQL implementation, and it eventually became apparent that I would save more time by doing most of the milestone four work early. As a result, the creation of this milestone necessitated doing much of the final milestone. As much of the program was to be based on the database, it was important to do this part early.