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

   This artifact comes from the course CS360: Mobile Architecture and Programming at Southern New Hampshire University. The initial purpose of the application was to provide the user with access to an inventory with functionality to view, add, edit, and delete items. The application stores users and items in SQLite databases, and it can prompt the user via SMS when item quantities reach a certain threshold. The project began in August of 2025 and was completed for submission in September of 2025. I took this course the term prior to this Capstone course.

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?

   I have chosen to include this artifact in my ePortfolio because of its capabilities of displaying a complex software that I coded entirely from scratch. While it is an extensive application to be coded for one course from scratch, this also means that it contains lots of vulnerabilities and issues. I figured it would be capable of showcasing my skills through development of code applicable to both the front end and back end, and I equally decided that it had glaring areas of improvement. I saw that there were many structural issues that needed to be addressed upon the artifact's point of expansion, and it will continue to grow as I have other substantial changes to implement with the other categories (which is why there still may be noticeable areas of improvement in other structural parts of the project).

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

   So far, I have met the course outcome I associated with Software Design and Engineering. Initially, the outcome I chose was:

   Design, develop, and deliver professional-quality oral, written, and visual communications that are coherent, technically sound, and appropriately adapted to specific audiences and contexts.

   My initial intention with associating this outcome with this category was that I would be able to display changes visible to both technical and nontechnical users through changes that affect the back-end functionality as well as the user interface of the project. These changes to the user interface indicate tailored software depending on the user, and this is extended to the overarching view of the project through my significant enhancements of comments and documentation as related to the software. Before this milestone submission, my project was lacking substantially in

more generalized comments related to the functionality of the software and were more directed to its previous baseline iteration. I implemented substantial change to the structure of the code, and in doing so I also added to what is being communicated to the audience in terms of explaining each component of the software in a way that different audiences can interpret from which is supplemented by my reflection and narratives for the ePortfolio.

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?

   Throughout enhancing and modifying the artifact, I quickly noticed I was asking myself more questions than I was giving myself answers in terms of the strategy of development, but I expected this since I knew I was going to have to make some significant changes to not only implement this category's enhancement but also to create the groundwork for a more expansive and structurally sound project. I learned even more extensively about how to interact with Java in the context of Android application development; during CS360, I learned quite a bit about Java when coding Android applications, but diving into and improving this project resulted in a depth that required a far greater comprehension of how to use Java in this context. I have learned how to further implement SQLite into the application and I also have more clarity on how to connect many complex components together to create a cohesive system.

   The implementation of this complexity also resulted in breaking earlier functionality on numerous occasions. For example, passing along user preferences through multiple activities that were not previously established upon login so they were not a thought before. I was also having trouble when hardcoding certain values for testing. Since we are implementing Firebase in the future, I didn't deem it logical to introduce role reassignment into this structure, so when testing user roles, some of the implementation of that for rough testing is what opened up a lot of the issues of ensuring that all of the gaps were filled with refactoring the methods associated with the databases, implementing user permissions, and those permission changes' effects on the user interface.