Marcus Doucette

2025-09-17

CS 499

Milestone 2

1. Briefly describe the artifact. What is it? When was it created?
   1. This is a mobile app I created for the mobile architecture course in march, April, and may of 2025. It uses the android views framework to build a simple app with multiple types of views demonstrating some of the foundational requirements of mobile apps like an authentication page and a list view.
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?
   1. I think that mobile devices are a good example of my software engineering preferences for native programs or software that otherwise works close to hardware. I think compared to many of the other projects this one is also easier to show off. Like I said above it does demonstrate many different common use cases, and while it doesn’t demonstrate skills with the more common compose framework, in combination with some of my other projects that do it shows a good breadth of knowledge and ability. I think my improvements will make it much easier to present to others.
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?
   1. I believe that I did work towards the collaborative environments and improved communications with my usage of GitHub issues, the updates to the readme, and the visual overhaul of the app that makes it much easier to use and understand.
   2. I believe I contributed to both algorithmic computing solutions and well-founded computing practices with my search bar and new login features.
   3. My edits to the login page functionality best with the concept of security though I think this may be the weakest category for this project.
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?
   1. Adding the theming took a long time but was relatively easy all things considered. The biggest issue came when trying to center align the items on the grid because the docs were not agreeing so I had to figure out some work around. I feel like the app is much easier to work with now because there are not any conflicts where the text you need to read or edit is the same color as the background.
   2. Improving the Readme build instructions and screenshots was similarly enjoyable. While something like build instructions can feel like something important to do perfect right off of the bat, trusting in the iterative process of GitHub to allow you to make the project better one step at a time helps assuage a lot of that worry and just let you focus on finishing one improvement at a time. Beyond that, not only does android studio make it surprisingly easy to clone a git repo meaning my instructions didn’t have to be that long, but adding images especially after fixing the layout feels like a significant payoff not only for the little bit of work involved in taking the screenshots and putting them into the readme but also paying off the previous task of improving the theming by showing off the effect that work had front and center.
   3. I thought that the search bar would be one of the more difficult features to add but it seems like I had already put in a lot of the important features so I just needed to add a method to take some text and filter a list based on that search. Many of the java utility methods like filter or contains also helped make this super easy.
   4. The opposite was true for ethe page-rearranging/login. I decided to have the login page load the item page automatically if the user had logged in within the past hour. I thought it would be easy to just save a number somewhere quickly because it usually is but with android it seems to be a bit harder to get my app to interface with the disk. While I don’t think it’s optimal I added a new table to my database to store and fetch this data along with utility methods to help for accessing it which was more work than I was hoping to need. Once that was all done the actual comparison and redirection was really easy. This feature might work better if I also add something like a logout button; however I don’t particularly want to implement that and I don’t believe it is the kind of feature that will be missed if it is absent so I’ll just make an issue on the repo and worry about it if I ever come back to this project.

Improvements

* Theming
* Readme Build instructions & screenshots
* Search bar
* Page re-arranging