**Mobile Application**

**Reflection Document**

**Jarrod Crockett**

**04/10/2021**

I have just finished my Mobile Application project, and, although it wasn’t without its difficulties, I have learned lessons through experience that I can apply to future projects. This is my reflection on the challenges I faced, how I overcame them, and what I would change if I ever had the opportunity to do this kind of project again.

If I was designing this application for a tablet, I have changed the way I developed the app. I would also include more detail on each page because of the larger screen size. I would have designed the application to have a multi-paned layout. The detailed page would have been viewable with the selection of a specific course or assessment. Another import difference in tablet development verses mobile app development is with the use of fragments. When building an app for a mobile phone you use the fragment to pull up another activity that details the information from the fragment. Due to the larger screen size of a tablet a fragment can be viewed side by side with the detailed information.

The minimum operating system SDK is 26 and operating system is 30 is my target operating system.

During the development I found that a had several significant challenges. One of the major challenges that I faced was with my database. My goal was to make a button that auto-populates my tables into my database using sample data. When I created the population class, I made a mistake in my code. Instead of adding assessments to the corresponding course, I was accessing an ArrayList outside the indexes available. This caused a run time error that I could not find until viewing the tables of the database though a third-party application.

In addition to making this mistake, I also failed to correct it promptly. I moved on to other parts of the code before sufficiently testing my code, which caused a day of wasted time hunting for my error. After correcting this error, the application moved much faster.

The way in which I corrected my index out of bounds error was downloading a third-party application for SQLite databases. I was then able to see the assessments table was never populated and was inaccessible to access because it did not exist. I then changed the pre-populated data to make it add correctly.

Although I could not remedy the wasted time, I did learn a valuable lesson about frequently testing my code before moving to the next step. I will remember this for future projects and will not repeat this mistake.

If I had this project to do again, I would do some things differently. The index out of bounds error caused me to become overwhelmed early in the project, so I tried to move on to other things. This led to many aspects of the project being half-finished. It made the project last longer and require more debugging.

In the future, on any project like this, I will try to slow down and define more specifically how I want the app to work before moving on. I will correct problems in my code, or if I cannot, I will take a few hours (or even a day or two) off. This will save in development time which will help the project to move more quickly and smoothly. Another aspect I will use to mitigate problems like this in the future is to use integration testing. “Integration test verifies that multiple app components interact correctly.” (McCown, 2018, p.11.1) Mobile Application Development.

Emulators are used to allow a programmer to run the program on a computer when they don’t real device in hand. One pro for emulator use is that a developer can create software for a device they do not actually have. As useful as emulators can be, though, they still are not exactly like a mobile phone. When you can hold the actual device in your hand, you have a much clearer picture of what works and doesn’t work, and you can see exactly what your customers will see.