Jordan Burke
C196 Mobile Application
Student ID: 001179962

1. If I were to develop the application for a tablet, there would be many differences in layout, design, and functionality. With more space, there would be more opportunities to add more functional layouts within one screen, for example, a static side-panel navigation fragment. Rather than moving from one screen to the next, I would be able to add a vew of courses, assessments, and instructors all in one screen. This would allow the user to view all the courses, assessments, and instructors in any given term. I would use fragments to create uniformity by re-using the same fragment in different activities as well as using child fragments to make up split-view screens.

In addition, I would have tested the application in both portrait and landscape mode from the very beginning. I think it is very important to keep that in mind throughout the development process.

2. The minimum API is 21, while the target API is 31. Tested with Pixel_XL API 26. It is compatible with devices running Android 8.0 Google API

3. The main challenges that I faced while developing the mobile application were getting oriented to the Android Studio IDE, API, Libraries, and overall architecture. This IDE and its Libraries were almost completely new to me as well as the architectural components such as the Activities, Manifest, Resource Files, Design Palette, etc. It took time to learn how to configure and use the Room database, Recyclers, Live Data, Adapters, and other components that were necessary to develop this app to its specifications.

4. I overcame these challenges by consistently meeting with my course instructor and following along with the weekly live instructor sessions. I went through many online tutorials hosted by the Android Studio website, poured through official Android Studio documentation, Stack Overflow forums, and it was a lot of trial and error. I used the debug console as often as possible as well as filtered through Logcat logs. My experience developing apps with Java helped immensely, however, in order to make all the pieces fit together, it was necessary to read and ask questions.

5. If I did this project again, I would have spent more time developing the design and layout. Due to time constraints, I settled with a layout and design that was not what I had envisioned. I had to make a decision, spend more time perfecting my theme and layout, or make the functionality and logic work to fit the requirements of the project. I chose the latter because it was more important to me to finish on time and I would have the opportunity to make the app more aesthetically pleasing and user friendly later on.

6. Emulators are virtual machines that simulate an actual experience with a mobile device. Emulators help mobile developers develop and test mobile applications with a variety of

devices. They are free, easy to use, and completely portable. They are also dynamic in that you can provision an emulator that mimics any mobile device of your choosing, you can choose the make, model, version. The drawback of emulators is the lack of tangibility. You cannot feel and hold the device as if you were an actual user trying out a new application. This can be detrimental to development. For example, while a button placed at the top left of a device may seem fine, if a user cannot reach that button while they are holding the mobile device in their hand, that would be seen as a flaw in the design. With emulators, you cannot test for flaws of that sort. I believe developing with both emulators and a physical device would be the perfect solution.