Part F – Reflection

1. Tablet Development Difference  
   If I developed this app for a tablet instead of a phone, I would design it so that I can make the most use of the larger screen size. For tablets, using fragments would be key. I could use Fragments to break the UI down into reusable pieces that can be arranged in different ways. This would make it easier to support various screen sizes and orientations. For layouts on a tablet, I could use a master detail layout which could show a list of terms/courses on the left side of the tablet and then on the right side I could have the selected course’s/term’s details.
2. Minimum/Target Operating System  
   The Term Tracker application was developed with a minimum operating system requirement of Android 8.0 (Oreo) and targets Android 14. The minimum SDK version for the application is 26, and the target SDK version is 34.
3. Project Challenges  
   One challenge I faced during the development of this application was creating the SQLite database within the application as well as writing data to it. It was difficult for me to understand how it was being built without being able to visually see the tables, as I was able to in the Java 2 course.   
     
   Another challenge I faced was implementing the notification alerts for course start and end dates. Making sure that the alerts would trigger, even when the app was not running, required an understanding of Android's AlarmManager and BroadcastReceiver.
4. Overcoming Challenges  
   I overcame the database challenge by watching a WGU webinar that specifically dealt with creating and manipulating data within SQLite tables. The instructor provided a clear explanation of how to set up and access the database within Android Studio, which helped me understand the process and implement it into my project.  
     
   I overcame the notification alerts challenge by researching Android's alarm and notification systems via the course material as well as on developer.android.com. I followed documentation and tutorials to implement the AlarmManager and create a BroadcastReceiver that would handle the alerts. Testing the functionality was important as there would be no other way the I would know if it was working or not. I tested it by setting the trigger times to be 30/60 seconds from the time of creating the alert. I had to debug several issues related to permission requirements and the correct handling of pending intents.
5. What I would do different  
   If I did the project again, I would change how I started the project. I didn’t create a wireframe or mockup which increased the amount of time it took to create this application because I frequently had to make changes throughout the project. A wireframe would have provided a clear guide in creating the layouts and functionality within the application. This would have led to a reduced number of revisions and decreased overall time.
6. Emulators vs. Physical Devices  
   Emulators are an amazing tool for development because they provide a quick way to test the application across Android versions and device sizes without needing physical devices. I enjoy how easy Android Studio makes it to run the app on the virtual device each time a change is made so I can visually see the change on the device. However, emulators can sometimes be slower than a physical device, and they may not mimic the behavior of real hardware at times. Emulators are somewhat limited in what they can do because they are not real devices with a phone service attached to them. Using a physical device allows for more accurate testing on an actual device that it would be used on. The biggest drawback of using physical devices for development is the need for multiple devices to cover all target devices and versions. This can be very costly.