1. Mobile applications are developed on desktop computers but built to operate on mobile OS platforms. I developed my application on two Windows 10 computers but was designed to run on any Android device running the Android OS version 16 or higher although I have targeted devices operating on at least version 24. Similar to mobile hardware, it is usually important to build applications to target the greatest number of possible users. When developing an application, it is important to not exclude older OS devices unless certain features require packages that are not supported on them. Most devices running a compatible version of Android will not have OS related issues running applications designed for that version but could have hardware limitations.

Hardware limitation issues would be more common to discover when developing a mobile application. Mobile devices with an older OS quite often have older and limited hardware resources which, can also change the choice of the target Android version. Mobile devices have many architectural differences between device models. On mobile devices, available RAM can be very limited on some devices while others may have more than needed to run your application. Mobile devices have varying screen resolutions, varying camera availabilities and qualities, and varying built in sensors. While developing mobile applications, it is important to consider the hardware limitations of the greatest number of devices available unless the application is being designed for a specific device model. Many consumers will have the newest hardware but many more consumers will be running older devices that cannot handle the same requirements. Each mobile application must be designed to run multiple screen resolutions with the ability to determine the best resolution available for each device. Adding features that take advantage of internal sensors without considering devices that may not have them can alienate large groups of device owners. These features may add a great benefit to

2. Version control and managing my development environment were my greatest problems I faced while developing this application. I used two devices in different locations to develop this application and I discovered small differences in my environment were causing compiler errors. On one computer my Gradle version was different than the other and my package dependencies were also different versions. Every time I switched machines, I had to perform project cleans and Gradle syncs as well as settings being reverted. While developing this application, I found that I had implemented methods that worked fine when I wrote the code on one computer but at some point I had upgraded software versions or dependencies on the other computer and a method was now deprecated. Using a cloud service to have the ability to access the project from any device seemed like a great idea until I found files were not being updated properly and causing compiler errors as well as taking an extremely long time to sync on each machine properly. This was a time waster I had to remove from my development time.

3. After discovering some of my issues created by using two development devices, I was forced to uninstall Android Studio and Java from each device and ensure I would re-install the same versions of the Java and Android Studio to create mirror development environments. Having a clean environment removed many issues I had been suffering from. I also stopped utilizing the cloud storage service I was using to access the project because it was improperly syncing data and wasting time. I began copying my project to a new directory from a thumb drive before working on my project then copying the newly edited project to the thumb drive to allow me to work at home. I always made copies at this point to allow me to revert to the previous state in case of a major problem.

4. I developed my project using two computers because my free time was divided between work and home. I developed this application using a Dropbox location to store my workspace so I could sync the project between the two devices. Unfortunately, I quickly discovered the two environments were not identical and the syncing process was time consuming. I had also reverted the project back to a previous state accidentally at one point, losing days of work because of this method. If I were to start this project again, I would start by ensuring each computer I would be using had the identical setup. Then I would save each day’s work onto a location accessible to each computer and start each session by making a working copy of the project. I would do all my work in that copy until I was done then copy it again to the same accessible location so I could work on the project from my other computer. Version control was my biggest failure during this project.

5. An emulator simulates the hardware and software of a given device model to allow a developer to test their application’s performance on any device and discover potential unexpected behavior before release. The emulator allows a developer to quickly test their applications on many different popular devices without actually having access to one. This allows for an inexpensive method of testing the greatest number of devices. Emulators allow the application to run unmodified but can be slow and do not factor in any battery drain or heating issues that may occur on a real device. Conflicts with other applications will also be ignored on an emulator. I was only able to use an emulator to develop my application but there are many pros and cons to testing with actual devices as well. Real devices allow developers to test their applications in real world scenarios. Performance on a real device is usually faster than an emulator and lighting levels and screen resolutions can be better tested. Built in sensors in a real device help when testing many features as well as testing on an actual manufacturer modified OS rather than a vanilla Android version. Bugs are easier to replicates but there are thousands of devices making creating a test pool expensive. Maintaining a test pool of devices can also be tedious and time consuming.