Samuel Doan

State Farm RDC Intern (IT/Systems)

iOS Development Research for Application Availability Tracker App

***Swift vs Objective-C:***

Choosing the best programming language is among the first steps in beginning the iOS development process. The traditional language was Objective-C, which is a superset of the C programming language; keeping all of the syntax of C while adding Object-Oriented capabilities through structures for implementing classes and methods. The Objective-C language along with Cocoa and Cocoa touch (Objective-C framework for Apple OS development) have limited error-checking capabilities. Only exceptions for programming errors, such as out of bounds array access, will be detected. Runtime errors such as running out of diskspace is handled by NSError (an Objective-C class).

The programming language I have decided to use in developing the iOS version of the Application Availability tracker app is Swift, which is a fairly new language made specifically for iOS development and released in 2014. Swift is a language designed to be easy to pick up with its very simple syntax which makes code easy to read and debug; facilitating development. One of the interesting things about Swift is that it comes with an Apple tool called *Playgrounds* that provides an efficient way of learning the language using real-time feedback.

***Why Swift is better:***

There are a few main reasons why I chose Swift over Objective-C. The main reason is that Objective-C is a relatively old language (dating back to the mid-80s), and because of this, it is not going to be as easy to use or fitted to mobile app developing compared to Swift, which is designed specifically for that purpose. Furthermore, updates to the Objective-C libraries are slower and are reliant on C being updated as well. Swift, on the other hand, has experienced great support from Apple with frequent updates and in the few years that it has been released, Swift has grown tremendously in popularity. I would predict that Swift will continue to be the main programming language for iOS development in the coming years, which would ensure that this Tracker app will not have to be rebuilt anytime soon.

Another reason why Swift is the ideal language for this application is that it comes with efficient debugger tools. Swift would be able to detect the usage of NULL variables (where Objective C would not) and immediately prompt the user to fix it as to prevent unpredictable behavior later. Swift also handles memory management so that the developer does not have to. Traditionally, with C programming, the developer would have to keep mental note of all the memory that is being allocated and having to remember to de-allocate the memory when finished. With Swift, the developer does not have to worry about memory loss and can focus solely on the logic and implementation of the program.

Another notable feature about Swift is the runtime. Swift does not compromise between a compiler language (such as Objective C) and an interpreter language (such as Python). A compiler language involves time used to compile the program, but the executable created will run quickly. An interpreted language does not have compile time, but the execution of the program will run slower. Swift will not need to compile but it will still run quickly.

***iOS Limitations:***

The development of iOS applications requires the XCode IDE, which is only available for Mac OS. Because of this, iOS development requires an Apple computer. The other option is to install OS X and access it via a VM. HOWEVER, downloading Mac OS on non-Apple licensed devices is against Apple’s SLA (<http://www.apple.com/legal/sla/>). Also, uploading applications via iTunes may also require Mac OS.