Most Meaningful Learning Objective
Leon Watkins
Georgia Southern University
WBIT 4030 Fall 2020

The purpose of my senior project was to learn how to develop a functioning IOS application using a MacBook with the XCode developer app and Swift programming language. My app called Padawan was intended to allow users interested in learning and users interested in teaching to connect with one another. My learning objectives for this project included:

- The ability to utilize XCode Integrated Development Environment (IDE) to develop a working application for IOS mobile devices.
- The ability to utilize the Swift programming language to develop a functioning IOS application.
- The ability to apply the knowledge learned from previous courses into my senior project.

As I had no previous experience creating an IOS application and therefore no experience utilizing the tools required I started out by following basic tutorials that demonstrated how to install and utilize XCode. I found documentation from Apple that walked me through how to download and install XCode, which is free to MacBook users. The Apple tutorial consisted of written instructions and several screenshots. The Apple tutorial did allow me to install XCode, but I couldn't follow that particular tutorial any further as it was for a previous version of XCode and many of the menu items had been moved or renamed making the tutorial inaccurate. I had some concerns at this point in time as I was worried if I could find tutorials that demonstrated more recent versions of XCode that would be easier to follow.

After some frustration with the outdated Apple tutorial I turned to YouTube and found a series of videos that provided an in depth explanation of XCode. This series covered everything from installation, which I had already completed, to frequently used menus and tools, as well as an explanation of all the different parts of window. This was particularly useful as the XCode user interface has several different columns that all can display multiple data sets and I was overwhelmed and concerned prior to this tutorial. After completing this series of tutorials, I was slightly more comfortable with the XCode user interface, but still hadn't practically applied the knowledge learned from the tutorial.

I then found myself at a crossroads, should I start to develop my application not really knowing where to begin, or follow additional tutorials? I located additional tutorials that walked me through how to build a simple application. Although this tutorial taught me how to develop a simple matching game, which was nothing like my application, it still proved to be extremely beneficial. This was because this series of tutorials allowed me to gain hands on experience with XCode. During these tutorials I learned how to place objects on view controllers, which are essentially the various screens that will be displayed on the iPhone. I learned how to place items on the view controllers, format them, and add constraints. These view controller tutorials provided me with experience that allowed me to design the user interface that Padawan users see in the app. The great thing about XCode is that once you have a basic design or scheme you can copy and rename that view controller leaving you to only change the applicable fields, so that items like buttons, page titles, or menus don't require custom formatting, although this can be achieved via the Swift programming language.

The next step in the tutorial after generating the user interfaces was to add functionality to the matching game app. This is where the Swift programming language is utilized as XCode runs on Swift. I followed the tutorials and discovered that XCode allows you to take an object created and displayed on the view controller and place that item into code. This is achieved easily as XCode allows multiple views and therefore I could have the main view controller view on the right side of the screen and the Swift view controller code on the left side of the screen. I then only had to drag the physical object from the view controller to the Swift code and the object was placed into the code and a reference created that connects the code to the object. Following the tutorial, I learned how objects could be placed into a class and that class given attributes similar how HTML can be formatted with CSS. I also learned how to add basic functions to the code and objects, which for the tutorial picked two random numbers, displayed the playing cards that corresponded to those numbers and declared a winner or draw between the player and CPU. As these tutorials were very detailed but segmented that helped me from becoming overwhelmed and proved to be valuable resources to reference if I needed a refresher while developing my app.

After designing a basic login and signup page I needed to understand how a user could sign up and login to the application. To accomplish this, I found a series of YouTube videos that discussed Firebase, which is a Google owned cloud based database that also handles user authentication. Luckily there was another tutorial series that walked me through the download and installation of Firebase and Cocoa Pods into my XCode project. Cocoa Pods is a framework provider that integrates into an XCode project that prevents app developers from having to recreate common features from scratch on each app. After following this series of tutorials, I had created an integrated database that allows users to create an account within the app. The user information such as email address which served as user id and user password was then stored in the user authentication table. Once a user had created an account up they then could login to the application with valid credentials and were granted access to the application after the app pushed the sign in information to the database and received a return notification. This was a big step and I was very proud of this accomplishment.

I then didn't have any tutorials that were specific to the needs of my application, so I had to lean upon the lessons learned from previous tutorials in order to add functionality to my application. This proved to be very difficult and time consuming and it was apparent that my goals for this project were too ambitious and I had to curb some of the features and functions that I desired for my app. While I was able to navigate through some of the obstacles from the lessons I had learned to this point the part of the project become very frustrating and it would have been nice to have a mentor to point me in the right direction.

Overall, I found this to be a valuable learning experience as I pushed myself to learn how to use several new tools such as XCode, Swift, Firebase, and Cocoa Pods. However, although I have developed some familiarity with these tools and now have a foundation that I can build upon I do believe that an internship may have provided me with more real world experience with IT use in the workplace. I am aware that an internship is only rewarding if the employee provides

resources to the intern and is willing to challenge them and also guide them, as my former employer had a steady stream of Industrial Engineer interns that quite often were left underutilized and assigned mundane tasks that were unrelated to their field. My suggestion for the senior project path is to provide a pool of mentors/ subject matter experts that could be utilized by students as it would be a great asset and really give complete the effectiveness of this learning experience.