John Doe

Spring Semester 2019

Learning Experience Final

For our senior design project, my team was placed to develop a tool for LA County Animal Care and Control. They asked us to develop something that could help deliver information to their clients and reduce the time spent waiting to be helped. To help solve this issue, our team decided to develop a mobile application that would contain simple information clients needed and also deliver a ticketing system to help reduce queues at facilities. We needed to develop something that would be easy to use and understandable for a large age range. In order to achieve wide success with our project, we quickly realized we would need a cross-platform application. But, we did not have the experience or time to code two separate applications for both iOS and Android. We needed a solution that allowed us to work on both simultaneously and effectively. This meant we would have to turn towards finding something new to work with that could meet our development needs.

In order to tackle this problem, we dived into a week’s worth of research to find a language or framework that could satisfy our requirements. Eventually, we found and decided upon a new mobile framework called Nativescript. Nativescript enables programmers to write a single set of code and compile that code into an iOS and Android version. The development process is similar to that of a website. Nativescript allows for the code to be written in either Angular, Javascript, or Vue.js. Alongside some specific tagging that Nativescript calls for, it compiles the code and replaces the necessary tagged areas with either iOS or Android code. The application for each device then looks as if it was natively made for that platform. Nativescript also applies any styling that is done, meaning you get well-designed application interfaces. Nativescript also has plugins available for use granting control over some device functions, such as location services, microphone access, and camera access. This means we would be able to implement some features that we weren’t sure we would have the time to develop. Nativescript was essentially a dream come true. It would be giving us access to the features we needed, and allowing us to do cross-platform development all in a timely manner.

When deciding upon which of the three languages we would write our code in, our team decided to use Vue.js due to the experience one of our team members, Franky, had with it. In order to bring us up to par, Franky walked us through several projects he worked on that used Vue.js. He explained the overall capabilities of it, how to work with it, and why it was a good fit. Vue.js provides strong control over design and looks, which would help us create an easy to navigate UI. Through the use of some its routing features we would also be able to create services that give access to our backend. We would also be able to maintain a modular approach to the project, allowing for easier expansion in the future. But first, we still needed to learn how to use Vue.js within Nativescript. Together as a team we sat down and read through the documentation bundled with Nativescript, taking note along the way of specific things we would be using. While it was mostly straightforward, there were still some things we would have to do differently than if we were just using purely Vue.js. After noting some of these differences, we each spent some time making simple hello world applications to make sure we grasped the basics. From there, we developed some more robust test applications by following tutorials online, in order to truly learn some of the capabilities of Nativescript. At this point, we then began work on our actual project with enough confidence in the use of these technologies.

By the end of this, we had spent several weeks doing research and just learning the framework we decided upon. In comparison to the structured learning environment of a classroom setting, this was a much more hands-on experience. It was much faster paced and a more interactive overall. As a team, we realized we had deadlines to meet and we had to learn this framework as quickly as possible. This opened a greater line of communication between the five of us. I had to be comfortable approaching my teammates for help and clarification where I needed it. Since this framework is the core part of our project, I couldn’t afford falling behind or not understanding something. With this, I realized that I needed to stay on top of my end of things, as this is what a real work environment would feel like. If I were to fall behind, it would only hold my teammates back. I plan on keeping this same type of mentality when I have to learn something new in the future, be it for personal use, or for a workplace.

Walking away from senior design, I’ve truly learned a better approach to learning. This project provided me with an opportunity to learn and experience something new. Not only on my own but also alongside a team. Our major is based entirely around working with new cutting edge technology. This means you have to have the right state of mind and the ability to learn and adapt at any moment in time. Falling behind means holding others back, which is simply unacceptable in a working environment.