Having had great success since 2013, App Academy definitely had a great curriculum cut out for us. On top of this, there is a constant desire to update areas where content may seem outdated or ill-fitted. Just based on my short time there, I noticed several occasions where the readings or projects were altered for the betterment of the program and learnings. As a student progresses through the course, he/she is asked for feedback many times and this feedback is definitely taken into heavy consideration as it has been emphasized to us many times that the student’s ability to learn will always be placed at the top of App Academy’s priority list. With that said, I will use this section to dive deeper into my day-to-day experiences during these several months.

A Typical Day:

Class starts promptly at 9am. There is roll call right on the dot and your absence leads to a strike (10 and you’re out). With the exception of assessment days, we usually begin the day with a lecture by the TAs based on the previous night’s readings and projects. There is plenty of opportunity to ask questions throughout this time so I made sure to ask questions that I had jotted down from the previous night.

After the lecture, we head to the computer workstations. We are assigned a partner to work on projects with for the rest of the day. The idea of pair-programming is to encourage not only collaborative work but also to strengthen our verbal skills when it comes to talking code. The whole class is divided into three smaller sub-groups and students are only paired with other students from the same sub-group. Anytime we were stuck or had questions about the projects, we could reach out to a TA for assistance by sending a request in an internal system (literally a simple button that puts our request in a queue). In regards to these daily projects, we are not encouraged to complete them but rather to take in as much knowledge and learn to the best of our ability without skipping over gray areas.

There will always be homework. This comes in the form of readings, videos, and projects. Get in the habit of using version control systems such as Git (get those green squares on Github!). You are also encouraged to review that day’s work in order to absorb as much information as you can out of the subject matter. With the completion of homework, it’s time to rest up and repeat this cycle the next day.

Structure:

My friend Raymond Zhang had previously warned me about the course by saying “you will always feel like you’re behind”. Even after day 1, this was absolutely the case for me. Though the curriculum jumps incredibly fast from subject to subject, I believe it was organized and taught incredibly efficiently. Prior to the start of the program, we are prepped on basic algorithms and simply understanding syntax, methods, and coding style of Ruby. This is a great transition into the first two or so weeks where we dive deeper into Ruby by exploring Object Oriented Programming at a much deeper level. After acquiring a solid foundation for Ruby, we dive into SQL and the topics of relational databases, MVC architecture, and HTTP/API Routing. This is fundamental in our understanding of backends. Pause.

We then jump into JavaScript. The beginning of this portion of the curriculum is very much like what we did in the Ruby portion but, we’re basically doing all those algorithm problems in JavaScript now. We also dive into more specific JS concepts such as callbacks and closures. These two ideas were incredibly frustrating at first (s/o to Jill De Los Angeles and our JS struggle day) but I’ve learned to appreciate how useful they can be, especially given that you’re able to do things in JS that you cannot do in Ruby. As we deepen our knowledge of JS, we jump into JavaScript libraries such as React and jQuery which allow us to really get started on frontend web development.

With all this under our belt by the end of week 7, it’s time to put everything we’ve learned together. We have all the necessary knowledge and so we now dedicate the next two weeks on our full stack projects.