

A Breakdown of the Apprenticeship

By: Joe Sak

May 28, 2013

Well, we've got quite a program formulated from our first run here at Neoteric. Not only are we vastly proud of Matthew for all of the work he's accomplished here, but we're excited about what has only just begun! The following is a breakdown of the beta program, and some thoughts to go along with.

[Read all the blog posts from the Neoteric Design Beta Apprenticeship Program \(http://www.neotericdesign.com/blog/tagged/apprenticeship\).](http://www.neotericdesign.com/blog/tagged/apprenticeship)

[Full list of resources \(books, screencasts\) used \(http://www.neotericdesign.com/blog/2013/4/apprenticeship-resources\)](http://www.neotericdesign.com/blog/2013/4/apprenticeship-resources)

For the first two weeks, we introduced Ruby, Rails, and some programming/workflow basics with hourly lessons every day of the program. This included working with the command line, Git, Rspec, Cucumber, Bundler, and RVM. These topics were introduced, their documentation was provided, and then practice was incorporated more-or-less daily. Teaching Rails takes some more zooming out. Because there is so much "magic" happening due to its structure and inheritance, it's best to teach it from the outside-in. Matthew gained a better understanding when I used the client-server example to explain what part of Rails was being activated at each step and how it was passing instructions onto the next bit. We ended with a two-

week lunch review to make sure everyone wanted to continue with the program.

The third week was when we started the "client" project: an internal application to help track production schedules. The daily lessons were substituted with pairing and coding by example. Railscasts were used during this week to help explain Rails conventions about associations and nested models, and Peepcodes were used to introduce the command line.

Week four was about front-end integration. HTML, Sass, and Coffeescript were all introduced. The apprentice spent most of the time, obviously, working on the web interface. Prototyping and designing in the browser are important skills to learn when you need to iterate fast against user feedback.

The fifth and sixth weeks were complete independent study. The apprentice worked with more client feedback and fixed bugs and code design problems, and a code review was given by the mentor. The issues to look for included checking for code format against the Github style guide, looking for refactoring that can be done, ensuring clarity in variable and method names, and checking that code is logically separated into the areas where it belongs (or seems to belong at the time).

The project went live in the seventh week. Heroku and its documentation were introduced, along with some of the standard add-ons used in Neoteric projects. More client feedback was given as real content loading began in production. The office happened to have an internal day during this week, and the apprentice took it upon himself to brush up on topics that he was personally curious about. The mentor also curated a package of screencasts from destroyallsoftware.com (<http://destroyallsoftware.com>).

The final week was devoted to writing the tests. Of course any good

software craftsman will write his or her tests first, but in the interest of allowing the apprentice to learn the conventions of Ruby and a Rails app quickly, we saved it for the end. This was useful because learning to test is almost like learning to program again. I think it helps to see the production code and then to be shown "okay, how do I test this code?" Of course, major bugs and code design issues were found out through these tests; ones that may have been prevented in a test-first environment. This served to teach the importance of TDD. Moving forward, the apprentice must now test-first.

And that's how it all went down! Matthew's time has been extended not only to continue improving the UI of his application, but also to add features to our time-tracking app! So it just goes to show how valuable his work has been and continues to be. They say the best way to learn a topic is to teach it, and I've certainly learned a lot in the process, myself. You really find out how much you know, how much you didn't know you knew, and how much you still don't know. Matthew has been a champion through it all, though, and I couldn't have asked for someone who is more diligent, quick to learn, independent, and able to demonstrate new skills so readily.

Congratulations, Matthew, and thank you for an excellent two months!



Joe Sak

Joe was Neoteric's lead developer. When not working, he's playing the drums, spending quality time with friends and family, and riding his bike.

« **OLDER**

Creating Character, a tool for using web fonts in mockups (</articles/2013/5/creating-character-a-tool-for-using-web-fonts-in-mockups>)

NEWER »

How to start an art loan program at work (</articles/2013/7/how-to-start-an-art-loan-program-at-work>)

29 East Madison Street Suite 450

Chicago, IL 60602

(312) 245-9795

[Careers \(/contact/careers\)](#)