|  |
| --- |
| PROG 2200 |
| Learning Narrative |
| Gap Analysis |

|  |
| --- |
| Omar Colón Silva  4-10-2017 |

Contents

[Introduction 2](#_Toc479712207)

[On choosing AngularJS 2](#_Toc479712208)

[The Learning Process 2](#_Toc479712209)

[Screenshots 3](#_Toc479712210)

[Quizzes 3](#_Toc479712211)

[Projects 5](#_Toc479712212)

[Bolt Network 5](#_Toc479712213)

[Pizza Planet 6](#_Toc479712214)

[Move Log 6](#_Toc479712215)

[Gameboard 7](#_Toc479712216)

[Feedster 7](#_Toc479712217)

[Outbox 7](#_Toc479712218)

[Calendar 8](#_Toc479712219)

[Reader 8](#_Toc479712220)

[NearMe 9](#_Toc479712221)

[All Done! 9](#_Toc479712222)

[Additional Resources 10](#_Toc479712223)

# Introduction

In this document, I will go over some of the details about why I became interested in learning AngularJS as part of my gap analysis project, as well as reflect on the process of learning more about this popular JavaScript framework.

# On choosing AngularJS

I was excited to learn more about JavaScript in our Rich Internet Applications course. It had been more than a decade since I wrote anything in JavaScript, and I wanted to get back into it. I was surprised about how much the language had grown since the old Internet Explorer days, especially with the advent of jQuery. Suddenly, things I remember being extremely tedious to code were finally made much easier.

I learned a lot through the RIA course. Until this last semester, I wasn’t that interested in front-end website development, but the things I learned opened my eyes to the possibilities of the modern web platform. I’ve heard about JavaScript frameworks like AngularJS and React, but I didn’t quite understand what made them so popular, or what their purpose was in the front-end ecosystem. I was hoping we could touch on those frameworks in our RIA class, but unfortunately we were unable to due to time constraints.

After doing some research, it became apparent that frameworks like AngularJS were becoming dominant technologies in web development. I feel that as a professional developer, it would be prudent to keep up with the latest technologies, as it would reflect favorably in my résumé. Due to this, I decided to learn AngularJS, particularly because it is the most widely adopted JavaScript MVVM framework to date.

# The Learning Process

I wanted the process of learning AngularJS to be structured. I learn better by following along and working on code rather than just listening to theory and then applying what I had learned. While trying to decide what resources to use for learning AngularJS, I weighted various options that ranged from tutorial videos on YouTube, to various tutorial sites like w3schools.com. I finally found out that Code Academy had a whole AngularJS tutorial series, however, most of the lessons were locked to non-premium members. I decided to take the plunge and pay for a membership so I could have access to all the lessons. There are also other subjects available on the site that I’m interested in learning, so I just thought of it as an investment.

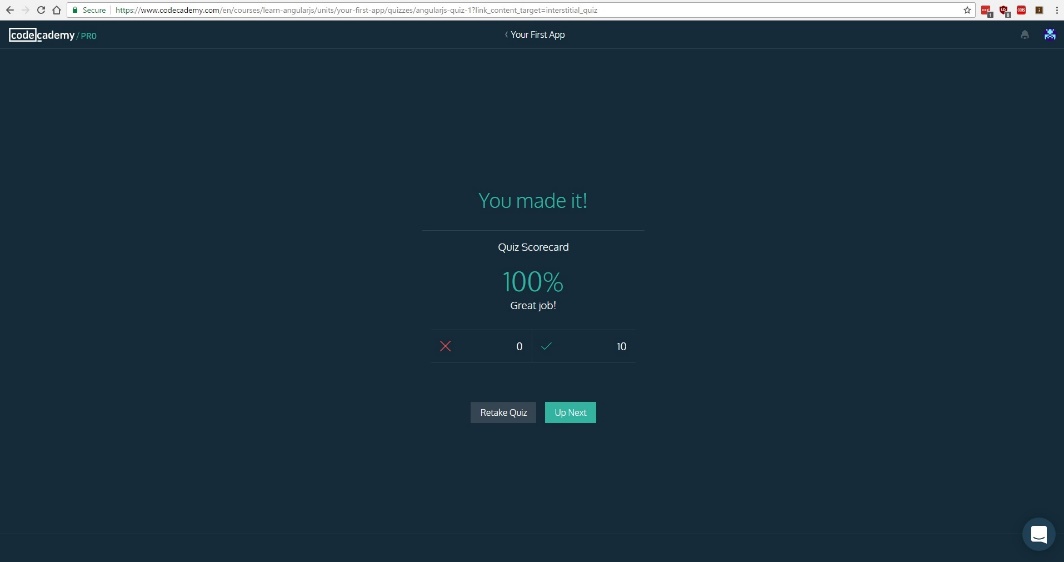
I liked how Code Academy delivers their lessons: Each section focuses on a concept, starting with an interactive lesson meant to teach the basics of that concept. The lesson is then followed by a quiz. After that, there are several free-from projects that allows the student to apply what they had just learned. Each section introduces another subject, but it also requires the student to apply what was learned from the previous one. That made the process of learning more engaging and enjoyable.

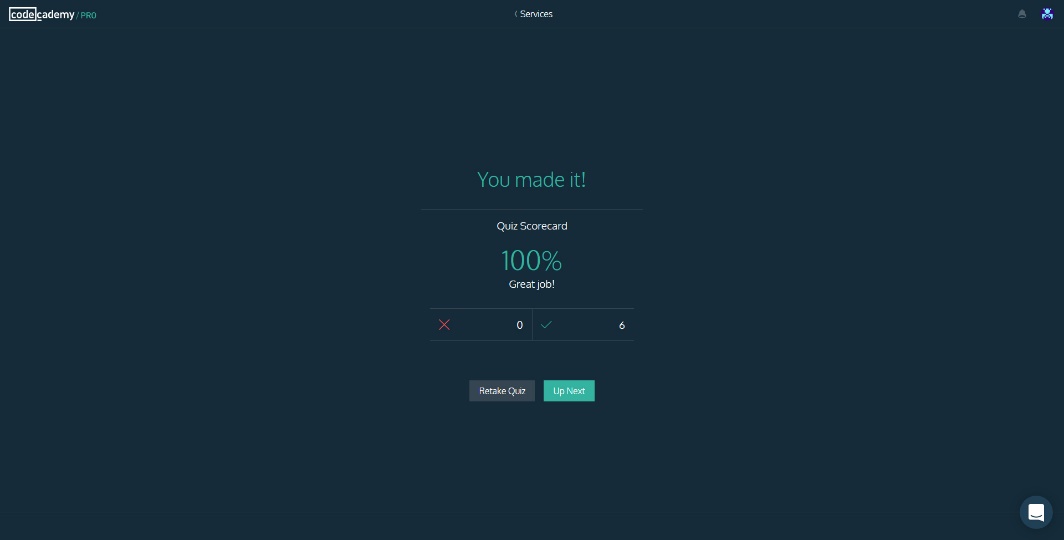
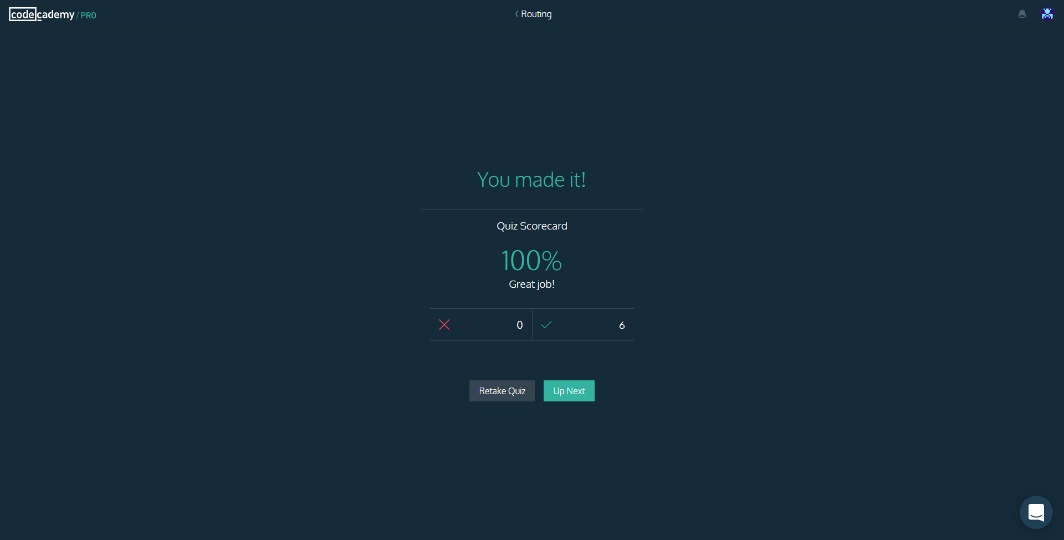
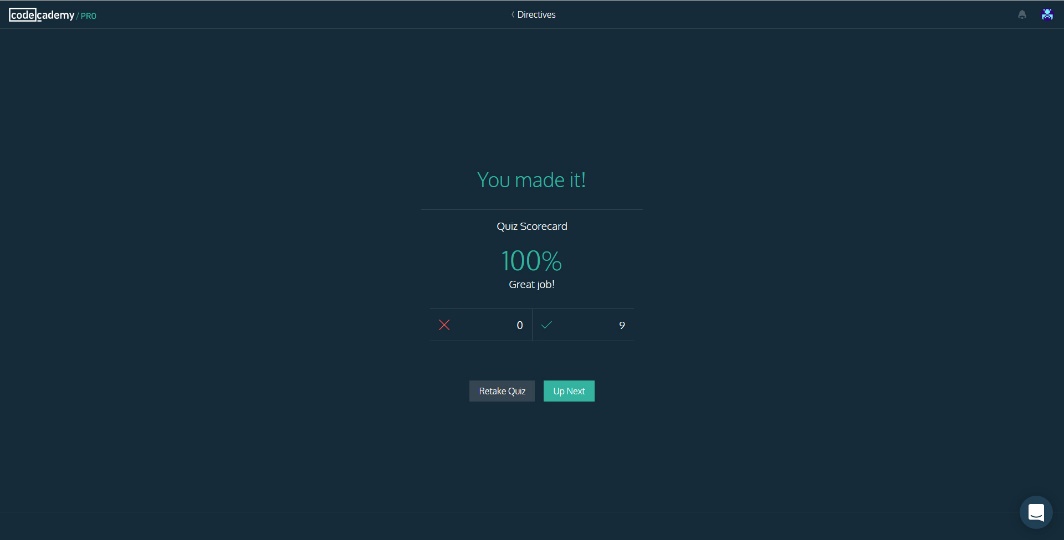
# Screenshots

Here I will talk about some of the screenshots I gathered while learning AngularJS. All of it is from Code Academy since that was my main source of learning. Due to space limitations, all screenshots had to be shrunk to fit in this page, but full size versions are available at <https://drive.google.com/drive/folders/0B0mdfFYGuIMVb1FVOGJoeXZZa28?usp=sharing>

## Quizzes

There was a total of four quizzes, one for each section in the series and each with unlimited amounts of re-tries. I had to repeat some of them several times, but I feel that allowed me to cement what I had just learned from the introductory lesson.

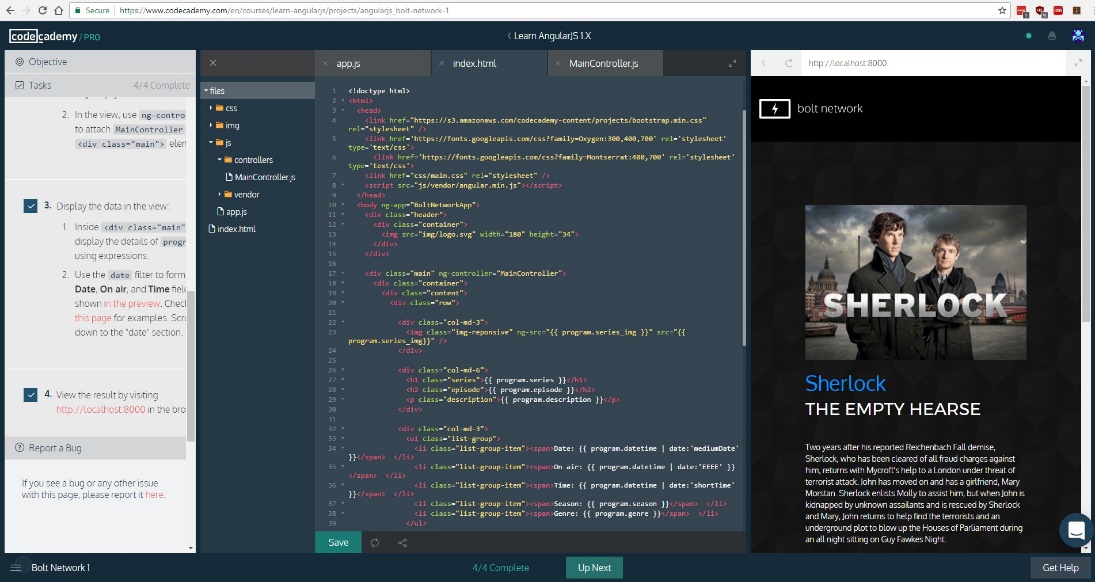


## Projects

Each section in the series contained about three free-form projects. Each project focuses on that section’s concept, and each one has different scenarios that demonstrate how the concept can be applied. Some of the projects are revisited in later sections so that the student can apply newly learned concepts.

### Bolt Network

This project simulates a website for a TV network, using AngularJS to display TV programming information. The first part of this project uses data from a controller to display information directly into the main page using expressions. The second part uses a custom directive to display information from a template.

### Pizza Planet

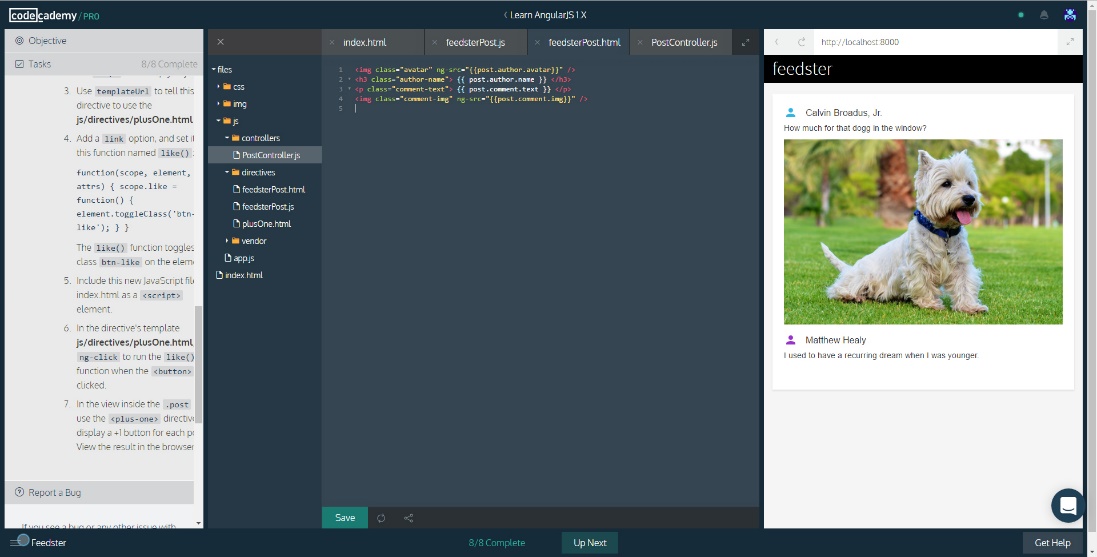
This project uses data from a controller and the ng-repeat directive to display restaurant menu items. It also introduces filtering by using the *currency* filter to display decimal data as currency.

### Move Log

This project simulates an exercise app that allows the user to increment and decrement repetitions by clicking on the plus or minus buttons. All functionality is stored inside the main controller, which has methods that can be called by using the *ng-click* directive on each button. It also uses *ng-repeat* to display each exercise.

### Gameboard

Gameboard uses a custom directive and a template to display a scoreboard. All the information is static and is saved inside the main controller.

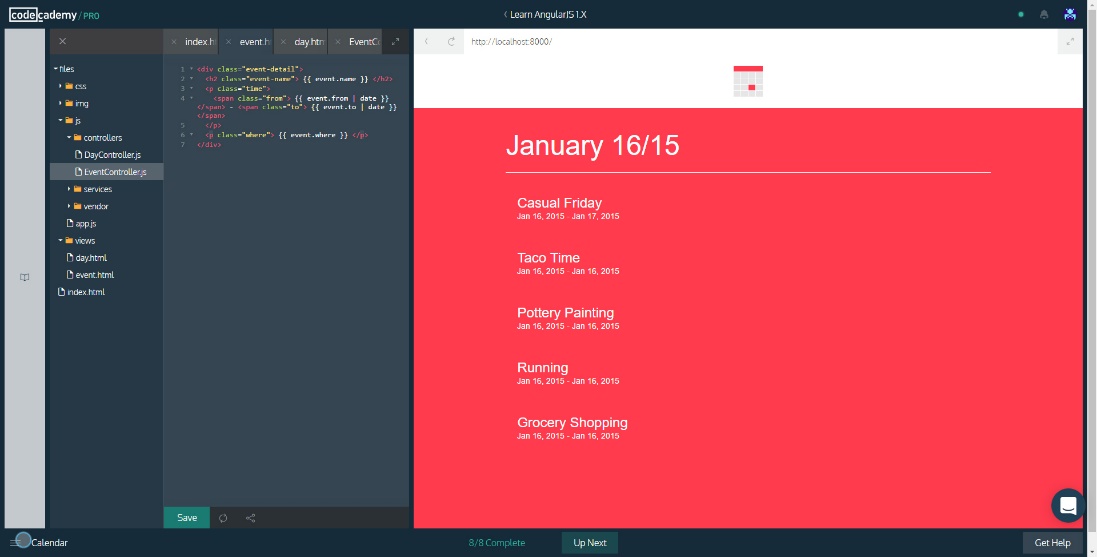


### Feedster

This project also uses custom directives to simulate a blog-like application.

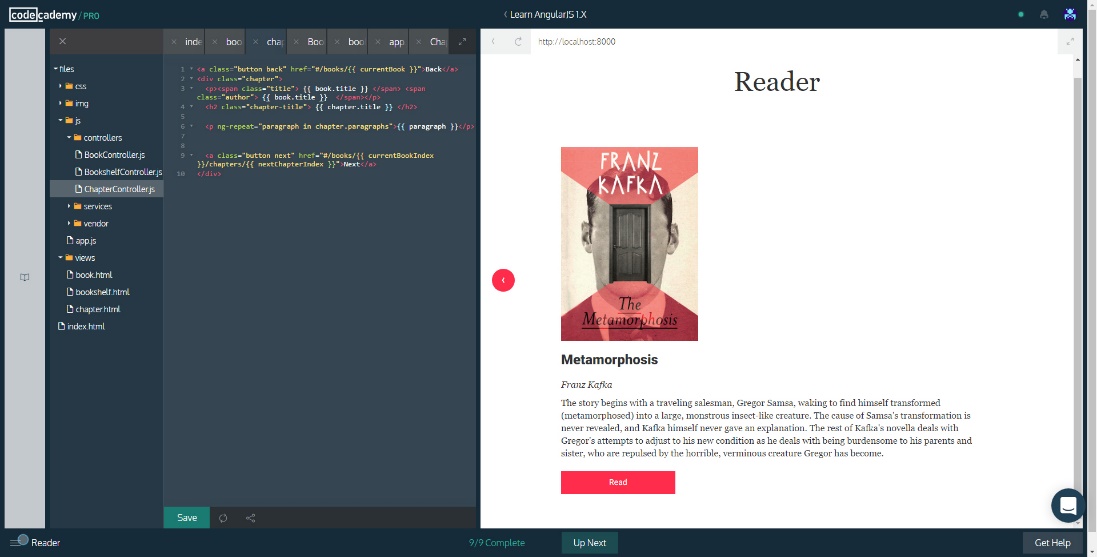
### Outbox

Outbox uses a custom-made service to fetch JSON data from a remote resource using AngularJS’ $http service. The second part of the app utilizes routing to fetch and display a full email when an item on the list is clicked.



### Calendar

Calendar uses a custom service to fetch JSON data and routing to display even details when an event item is clicked on the calendar list.



### Reader

Another app that uses routing and services. This one uses sub routes to fetch and displays chapters from each book.

### NearMe

This is the final project of the series; it uses every concept learned, including directives, services, and routing. It uses an external directive called leaflet, that can be found at: <https://github.com/tombatossals/angular-leaflet-directive>. Leaflet displays a map which can be interacted with. The service fetches Geolocation data from a Wikipedia API, and displays it as map markers. Routing is used to display an About page.

# All Done!



# Additional Resources

While re-building my portfolio in AngularJS, I also used the following resources:

* <https://material.angularjs.org/latest/> - A CSS framework counterpart to AngularJS. Based on Google’s material design specification.
* <https://www.w3schools.com/angular/default.asp> - Good AngularJS tutorials by w3schools. Good for quick reference.
* <http://stackoverflow.com/> - Where every programmer spends 90% of their time.