

In this project, I have learnt many of the different aspects of web development, and how they all work together to create a responsive web application.

Some Difficulties I Faced

For me as a beginner, the massive amount of libraries was puzzling and confusing. I was constantly bombarded by all these terminologies with zero idea of their purposes. What does jQuery do that JavaScript cannot? What is CoffeeScript and why is RoR using it? What is Yarn and what is Rake? These are few of the many questions I had, and I cannot say that I have figured them all out. The amount of assumed knowledge when people talk about technologies was often overwhelming when dependencies are built atop other dependencies.

Being used to small projects with full understanding of the entire structure, I was daunted by the fact that I simply had to accept a black-box where it does what I expect it to. Sometimes it feels like I am doing nothing productive, simply gluing different libraries together with no control over the outcome. Often I would be guilty of copying and pasting code without actually understanding the problem.

Popular libraries were developed to tackle problems that many others had faced. Without having experienced such problems, I would not understand the purpose of such libraries. Without using JavaScript to navigate the DOM, I would not understand the struggles of pinpointing the right elements to attach events to, and how jQuery solves many of these problems. Without understanding what problem the library solves, I would not be able to utilize the library properly.

Sometimes it can be hard to decide on whether to include certain libraries. For instance, I had this conflict when deciding whether to include jQuery into the project or not. On one hand, using jQuery on top of JavaScript could help improve the readability of my code and is easy to use. However, the drawback is the additional dependency in the project, which would be another burden in maintaining the compatibility.

One persisting difficulty I had throughout the project was the inaccuracy of online information. Web technology is consistently improving, and what seemed like best practice previously is now viewed as bad design. It is often difficult to discern if a particular piece of advice is still viable, especially if it is rather dated.

Even beginning with a basic understanding of HTML, CSS and JavaScript, I was still woefully unprepared for the real-world web development process. Ruby has an unconventional syntax that I was not familiar with. SQL had to be learnt from scratch. RoR's "Convention over configuration" is tough when you don't even know the convention.

Some Feeling About the Project

I certainly learnt much, including gluing libraries together, and how to google for answers efficiently. Besides that, I also grew familiar with the model view controller framework and learnt about the workings of the request response cycle.

Web development is definitely more difficulty than it seems. Simple modifications to the code may lead to unforeseen results. Testing would have to be done regularly to accurately pinpoint which modifications resulted in a bug. There are various gotchas that are unanticipated and very frustrating as these problems are rather unobvious, which leads to long hours of debugging for a very minor issue.

How I Designed the Web Application

I disliked the endless redirecting and rendering for every single request that the user makes in my initial design, which was unlike my usual web browsing experience. Therefore, I designed it to mimic my observation of what most other sites are doing, which led to me deciding to create a single page application (SPA), which in my opinion creates a better user experience.

Creating a SPA turned out to be much more difficult than I anticipated. I had to convert most requests from the HTTP requests to AJAX calls and use JavaScript to attach various event to modify the HTML and CSS. Many problems were cropping up from the change in design and had to be fixed. A lot of work is spent figuring out how to do things the RoR's way, how to fit into RoR's asset pipeline.

All in all, I managed to learn much about web development and appreciate the difficulty in creating a good website. Having to be careful about every single nitty gritty detail to make the website look nice. Web development is a tough job.