



# helpful human

CSS 497: Undergraduate Capstone — Report #4

Last Updated: November 22, 2014

Weekly updates here:

<http://students.washington.edu/rennurb/>

**Student:**

Josh Brunner

[Josh@HelpfulHuman.com](mailto:Josh@HelpfulHuman.com)

**Faculty Advisor:**

David Socha

[DSocha@UWB.edu](mailto:DSocha@UWB.edu)

**Company Sponsor:**

Mark Sandeno

[Mark@HelpfulHuman.com](mailto:Mark@HelpfulHuman.com)

## Introduction

As a student in University of Washington Bothell's (UWB) Computer Science and Software Engineering (CSSE) Department, the last requirement I have to fulfill in order to receive my degree is to complete one of several Capstone requirements. The requirement that I chose is to complete a 400 hour internship at a company of my choice. I have chosen to split up the 400 hours over the course of two quarters at UWB. However, due to a recent scheduling conflict where the presentation date change to a later date, I will no longer be able to present in December. While my internship will, in fact, be "complete," I will present this coming March.

The faculty advisor for my internship is Professor David Socha. He has been my professor for two quarters so far at UWB. Throughout his courses, I have enjoyed working with him on his research projects as well as other various class-related projects. The company that is sponsoring my internship is called [Helpful Human](#). Helpful Human is a small (6 person) startup company located in Seattle. We are capable of producing full front-end software solutions for clients' needs ranging from simple websites to complex [full-stack](#) web applications.

The following report will provide a brief overview of what it has been like at Helpful Human since [my last report update](#). I will discuss what kind of things I've been working on as well as what kinds of new things have happened. It's all very exciting and I think you'll enjoy it!

## Report #3 Recap

In my last report, I was really getting to the point where the work I was doing was mine. What I mean is, the things, processes, clients, styles, etc. were my creation and from my doing. I was working on projects that I had started with clients. In my last report, I was excited to be learning how all of these parts come together to make a business.

Another thing that was going on was the redefinition of what our internal processes looked and acted like. This dealt with us moving towards 2 week sprints, defining programming methodologies in Extreme Programming, and more.

In addition to the items mentioned above, I began working remotely because I am back in school and am required to be on campus 5 days a week. This has proven to be both difficult and beneficial to my learning. I will discuss more about this in this report.

Lastly, I highlighted things I was proud of like learning iOS app development in my spare time as well as presenting the work that I had done to one of the clients - in person. I also discussed things that were challenging to me like fully understanding all of the complexities that come together to form a complete web application. In addition, I talked more about a continuing

struggle that I face in understanding Git and the different way of thinking that goes along with it. All in all, since my last report, I have grown tremendously and I think that the following report will prove that.

## **Overview**

In this section, I aim to cover the major things that happened during this last month. Among these things, we moved to a new office with more space because we are growing so fast. We also redefined what/how we go about communicating with clients and getting client projects moving forward.

## **Full Page Designs**

First off, I wanted to talk about the full page designs that I have been working on and the modularity and extensibility that goes along with it. In particular, I wanted to discuss a major difference in how I was programming in my last report compared to how I am programming now. In my last report, I had been working alongside different developers as they created pages and I added different modules. In contrast, during this last time period, I have been working completely on my own page designs. This means that I would receive a design specification from our client's designer and I would have the full say in how that page was implemented as well as what technologies would be used to accomplish specific functionality. I've been responsible for designing the logic in JavaScript as to how content was displayed as well as designing the components to allow for interaction between the web application and user.

## **Specific Modules**

As usual, I am not allowed to say who exactly my client is as well as post and share screenshots of our client's work. However, I will talk about three of the specific modules that I took great pride in creating. The first module was a drop-down menu that allows the user to filter results of a search query. I called this a "Filter Dropdown." First off, designing this component required me to really hone in on my understanding of positioning an HTML element either relative or absolute. If you're interested, you can learn more about what the different positioning types are in HTML and CSS [here](#). Anyhow, it then required me to write up an Angular directive to manage a controller as well as various scope variables. Don't worry about what those mean, just know that it is complex and requires quite a bit of logic and reasoning in order to make sense of it.

Think about when you click on a button which then has a drop-down menu on any of the websites you visit. Consider the expected functionality. For example, when you click outside of the box, you expect it to close. If you click on a close button within the box, you're expected to close. However, when you click on a checkbox or another item in that drop-down, you would, naturally, not be expected to close. These are the states and interactions that I had to account for.

They do not just come out of the box — they have to be designed and implemented. I spent a total of 14 to 15 hours on this one module. However, once it was done, I felt very accomplished because this was one of the most complex things I had ever designed in the web space.

The other two modules were and include from the Angular team but needed some major re-working in order to fit the new designs from the client. These modules were a time picker and a date picker. The time picker allowed a user to click on an icon which then had a drop-down allowing the user to select a specific time for a meeting. This one was not too complex but it requires me to work more on the JavaScript side because some of the logic had to be re-thought to allow users to do various things in selecting a specific time. The next module was a date picker. This one was extremely complex and required 18 hours of effort to complete. It was complex because the entire date picker module (initially designed by the Angular team) was not, by any means, a complete product and most certainly required extra effort on the implementation side. Especially when I went to design the styling format NCSS, the true complexity was shown— Requiring me to rework how a user interacted with it as well how information was saved to the database.

### Screen Sizes

On top of all this work that I had to put in towards specific modules as well as many other components, another thing we have to be conscious of is how the web application and the user interact when the user is viewing the application on a mobile device, tablet, or desktop. Since the screen size and resolution varies greatly between desktop all the way down to mobile, I, as the developer, need to do extra work in the form of media queries to change how content is displayed depending on the screen size the app is loaded on. In the past few weeks, since my last report, I have had to do quite a bit of this specific work. In the past, since I had worked on somebody else's work, I would not have to worry so much about how content was implemented because the logic behind how it played with different screen sizes was already determined. However, this time I was able to get hands on experience with what to do and what not to do. There are different things you have to consider when implementing for mobile and desktop. One of the major things that took me a while to comprehend was the usefulness of tables in an HTML webpage structure. The reason you would use tables is because tables have a “magical” ability allow the developer to implicitly create the rules so that content displays a certain way depending on how they make it.

Let's take one of the components that I had worked on which was a “utility bar” containing multiple buttons, a search bar, and a drop-down all on one bar. While this may look to the average user like it is a simple thing, it really is not. As it turns out, it is actually very complex and requires some heavy logic when designing how the HTML and CSS played together. The problem that I was running into was that when I designed the “utility bar” to look right on

Chrome it didn't look right on Firefox and Internet explorer. In fact, it varied on all three browsers. The reason for this is because the browsers run on different "engines" which load and display content in various ways. And because of the complexity in the "utility bar," the best solution turned out to put all of the modules into different cells within the table. Now that this is how the finalized implementation is, the various components comprising the "utility bar" just work. They resize and scale correctly, they no longer jump around depending on what browser is loading the content and it easily goes away and when the user browse is the app on a mobile device. While this learning lesson most certainly took time, the end result was me furthering my understanding of how different technologies come together to provide inexperience for an end user.

## **Methodologies**

As I've mentioned in my previews weekly blog posts and reports, the methodologies that we use in practice are invaluable when it comes to establishing consistency both internally and externally. Consistency is important because it allows for people to be able to rely on your results after completing a task. The following section is dedicated to highlighting a few of the methodologies that we've recently implemented at Helpful Human.

### **Continuous Integration**

Due to a previous situation with one of our clients, we had to do some serious rethinking to the importance of quality control when software is released. Basically what happened was some software got released when it hadn't gone through quality assurance testing. This resulted in our code completely crashing when it would brought up on specific browsers and operating systems. The new Continuous Integration (CI) testing that we've been working to include in our practice is a software module called [Magnum CI](#) and it isn't quite yet finished but we are getting close. On top of Magnum CI testing, we've been allocating 8 hours of each sprint to QA-ing code and making sure that what we're releasing is up to standard and in a working state.

### **Browser Testing**

On top of CI testing, it's also important to make sure that you (as a developer) are aware of what kinds of browsers and operating systems your users plan on using. Since the last report, we had neglected to do our research and ask the client what types of devices needed to be supported. As a client, you are hiring us (the developer) to think these kinds of problems through. That said, we had gone 2-3 months in developing a web application for only the bleeding edge of device versions. The lesson learned was that this type of thing need to be done during the inception stage. Fortunately, the repercussions of this mistake were not too huge. One of the other developers I work with has decent experience in browser compatibility coding so we were able to fix a lot of the broken components.

## Device Testing

To the same tune as the difficulties in browser testing above, it is vital to understand and embrace the types of devices users of your web application will have. In the typical web application space, a user will often begin working on your web application on, say, their laptop and then I have to leave. At this point, the web app should automatically save their location and what they were working on to be picked up later. Then, as time goes on, the user will go back to the web application on their mobile device or even a tablet and continue working. In the above scenario, my mention three categories of devices. Each of these devices has hundreds of variable types of devices. You have iPhones, android phones, and even Windows phones. And that is not to mention the vast amount of desktop computers and tablets that people have access to these days. Each of these devices and categories of devices uses a different web browser and different interpreter engines to render the web application. It is up to the developer (me) to account for each of these devices and write code that will work across the board.

This is another thing that I have been working hard on since the last report. For example, when I was developing the web application for mobile devices, I was using Google Chrome's emulation tool for an iPhone 5. everything seemed to be working just fine up until we actually ported the web application onto an iPhone 5. The difference was drastic, things would zoom in because font size was too small and some things were not “touchable.” What I learned from this experience was that you cannot be sure that emulation tools are doing the best job. The way to be sure that things will work is to actually load the application onto the physical device and do your testing there.

## Highlights

### Understanding Git

Since the beginning of my internship, I I've always struggled with understanding the Git repository version control methods that are used so heavily in the software development realm. In the past, I have always mentioned that this was a struggling area for me because of the complexity in knowledge that it requires as well as the lack of being able to formulate a valid question when things go wrong. Fortunately, in this report, I am proud to say that I feel that I have a firm grasp on Git repository version control. In the past 4 to 5 weeks, I have found myself learning about branches, resets, forks, and much more in the realm of version control. I have found myself having to revert to a previous state in my code as well as fast forward to modularized code. Just this last week, I even pulled down code from one of my teammates Remote repositories and merged his code with mine so that I could get his functionality that he had been working on and implemented in my work I did all of this without having to be guided because I knew what to look up online and because I have done bits and pieces in the past. As I

mentioned above, I'm very proud that I am beginning to feel confident in understanding version control in software development. I can only hope that this kind of topic will be taught in future computer science classes at the University of Washington because I find it extremely valuable.

### **Being Proud of My Work**

Over the course of my internship, I have grown to learn quite a lot about web development and how useful the understanding of it is. Looking back, I remember the first couple of weeks in my internship where I was struggling to wrap my head around the different languages and how they all work together. I remember building a WYSIWYG editor from scratch and experiencing the trial and error process that came along with it. Since these days, I have grown to be working on full-page designs as well as communicating directly with clients. I am personally very proud of where I am and where I am headed with Helpful Human. The opportunities that seem to be coming to us are great and I expect them to push me to become a better developer in general. As I explore different languages and syntax-styles in the future, I feel like I will become more and more valuable to the job market. I am very happy with where I am and am excited to see the company that I was hired on growing so fast.

### **Challenges**

#### **Testing Everything**

As I mentioned in the above paragraphs, one of the things that I know that I struggle with is knowing what exactly to test and knowing what works and what doesn't in terms of delinquents that I am working then. Since web development is so vast and requires specific instructions for each device and browser, there seems to be so much that I need to keep in my head as I am writing code. I am sure that as time goes on, I will get better and better at it just like my teammate, Ryan. I work closely alongside Ryan and he is teaching me very many skills and tricks as I go. I would imagine that as I have time, I will be investing my personal free time into working with web applications on my own so that when I go to work, I have more and more experience. They say that it takes 10,000 hours to become an expert in something. I know that at this point, I am only a few hundred hours into web development - but that 10,000 hour mark is only a few years away.

#### **Keeping Healthy**

Another one of the things that I tend to battle with is knowing when I need to take a break from work and walk around to get myself moving. As a developer, it is very easy to get so involved with the specific task at hand that you are unable to keep track of the time that you spend doing so. The problems that you faced at work are so complex and require so much effort to wrap your mind around that sometimes you just don't want to do that again. Nevertheless, this is something that I am working very hard to create into a habit. Recently, I have been trying to take a break from the computer every 30 to 40 minutes. Whether this be simply standing up and looking out the window or actually going and walking to get a drink, as long as I am

stepping away from the computer, I am happy. This also ties into eating and drinking healthy. It is so easy to grab another cup of coffee from the office instead of making tea or drinking plain water. However, there comes a point where too much coffee is bad because your body is loaded with caffeine. So, I have been making it habit so that for every glass of coffee, I have one or two glasses of water. So far, this has helped me to be more involved and feeling healthier when I go home so I don't think it will be a problem to continue doing so.

## **Conclusion**

All in all, this month has been great for me as a developing software engineer. I have seen the company grow and opportunities present themselves. I have learned even more than I expected in the realm of software development for web applications. This is will be my last report as an intern at Helpful Human. There is one more, final, report that I will write which will summarize the entire internship into digestible chunks so that you (the reader) and I (the writer) I'm able to easily reflect upon what it was like to become an intern in a space that I had previously considered in a serious sense. I feel like I have learned an incredible amount of information and I have been given the opportunity to apply what I've learned in college, In my computer science classes, to a real world aspect. This gives greater value to my degree and I am proud to have worked hard to accomplish it. You can expect my final report by the end of 2014. I will be doing my internship presentation at the end of winter quarter, in March 2015.