1 HOW DID WE LEARN NEW TECHNOLOGIES?

By far the leading factor of our team learning new technology was having each other as knowledge points. Some members of the team had previous knowledge working with a JavaScript framework, and others had experience working on a backend. One of the most beneficial qualities of our team was our constant communication. This made learning and asking questions very easy and fast, and aided in our own growth as developers.

Another major learning point was following documentation online. Unfortunately, the Aurelia framework isn't as polished as other JavaScript frameworks that have been around longer. This posed as a minor problem, but didn't stop us from accomplishing everything the team needed to.

2 WHAT WEBSITES WERE HELPFUL?

Aurelia Documentation - http://aurelia.io/hub.html

Aurelia Materialize - https://aurelia-ui-toolkits.github.io/demo-materialize/

Aurelia Chatroom - https://gitter.im/aurelia/Discuss

Aurelia Materialize Chatroom - https://gitter.im/aurelia-ui-toolkits/aurelia-materialize-bridge

3 WHAT REFERENCES BOOKS REALLY HELPED?

There were no resource books that were utilized during this project.

4 WERE THERE ANY PEOPLE ON CAMPUS THAT WERE REALLY HELPFUL?

Beyond the professors and those related to this course of this course, there were no really helpful people on campus.

5 WHAT I LEARNED: JOSH MATTESON

I believe the biggest thing I learned throughout the year is how crucial it is to have communication between everybody involved, whether that be other team members or our client. In the beginning of the year, there wasn't a whole of communication on anyones part. This caused for a lot of frustration, but thankfully it got better as we continued on.

5.1 What technical information did I learn?

While I already had a decent understanding of web applications are made, I feel that this skill was strongly reinforced by working on this project. JavaScript is one of the major languages that companies are looking for, and our entire application was built around JavaScript.

Another technical skill that I learned over the year is how to write papers in LaTeX. Our application is centered around books being built, and they are all stored in the backend as LaTeX. This skill proved greatly useful for other classes that required a paper with illustrations or math problems.

Many factors go into building a web application, and it's hard to nail down every I learned from it. Web development is one of the more popular desired fields of computer science, and having knowledge in this area greatly influences my chances of getting a job after college.

5.2 What non-technical information did I learn?

I would say the largest piece of non-technical information I've learned is how to work with a team and how to communicate properly. Organizing team meetings and following through with it is something I've learned as an important factor in the success of any group activity. Another important non-technical skill I've learned is learning to set deadlines and having other team members keep you accountable. Making sure that everyone is getting their work done on time is one of the biggest factors of success, as long as it doesn't come in the form of micromanaging.

5.3 What have I learned about project work?

This is a repeated theme throughout my writing, but communication is key. From the beginning, a member of the team set up a communication platform that we could all communicate over. I think this is largely why our team was able to do so well and actually have a working demo at expo. I've also learned that sometimes you just have to do more than you thought you would in order to succeed. This comes in the form of picking up the slack of other people, and sometimes other people pick up your slack.

5.4 What have I learned about project management?

Organizing everything beforehand and setting up deadlines is crucial to good project management. Otherwise, the project will get away from you and you'll be left wondering what happened.

5.5   What have I learned about working in teams?

I've learned that working in teams is a lot of give and take. Sometimes you have to rely on your teammates for certain task, and other times your teammates rely on you to get everything done. This is imparative to having a functioning team, because if you can't rely on your team to work, then there's no trust.

5.6   If you could do it all over, what would you do differently?

If I could redo the project differently, I would get a lead on the project as soon as possible. We started the biggest parts of the projects later than we should have, and this costed us when time was crunched for other classes and projects. I know that, for me personally, I could have been researching far more than I had early on. This could have saved us an incredible amount of time when we were stuck on various technical problems about the Aurelia framework.

6 WHAT I LEARNED: STEVEN POWERS

6.1 What technical information did I learn?

I learned a lot of technical information during the past year. As a result of the year of working on this project I have learned a lot more about newer technologies such as page routers. I also learned a lot about NodeJS, containers, and how JavaScript frameworks such as Aurelia are different than raw JavaScript that I am used to working with. Aurelia Materialize and Aurelia Single Page Applications are two examples of some of the front end technologies that I learned during development.

6.2 What non-technical information did I learn?

I learned a lot about the documentation portion of software development projects. The various document formats, specific IEEE standards specifications, etc. I also learned about how to write these documents and write about my project at a higher level. I also learned how to present myself and talk about projects I am working on with someone without a technical background.

6.3 What have I learned about project work?

I have learned a lot about how projects can become disorganized or put in disarray when some information is unknown. No matter how much planning is put into the various documents detailing your targeted goals and dates of delivery, this can cause your workflow to drastically change.

6.4 What have I learned about project management?

Managing a project is a tricky one. I was elected / volunteered to be team leader and as a result I had to stay extra vigilant on deadlines and make sure that things were getting finished on time.

6.5 What have I learned about working in teams?

Everyone has different strengths and weaknesses, and being understanding of the weaknesses and exploiting their strengths can still result in a successful project.

6.6 If you could do it all over, what would you do differently?

If I could start over and do it all over again, I probably wouldn't. There was a lot of time and effort put into this course and I wouldn't want to suffer with that right now. If I could change something that would put me into a better situation now, without effort, it would be to start working on the project earlier. We were put into a bad situation and had to wait before we could continue with development early on in Winter term. While other groups continued plugging away at their projects, we were trying to get clarification and guidance from our client. Working with improved client relations, such as how it was later in the term, and spring term. For the project itself, I would have tried using a different JavaScript framework, or see how much we could have got completed with just the addition of a router instead of a whole framework that we had to learn how to use.

7   WHAT I LEARNED: EVAN TSCHUY

At times fun, at times incredibly frustrating, the last year of capstone has been a great opportunity for me to get a look at the perils of project management, and dig into creating a technical product from nothing.

7.1   What technical information did I learn?

On a technical level, the project can be broken down into several sections:

- a NodeJS backend:

    - direct interactions with ElasticSearch
    - a custom-written library for interacting with `git` repositories
    - 3rd-party authentication integration with Google

- a frontend:

    - a single-page application using Aurelia
    - Materialize for user interface items

Every single one of these sections was its own learning experience for me. In the past, I have written web applications for work. These generally tended to be fairly self-contained, with simple user interfaces that would fit in well with the year 2009.

Instead, this project was hyper-modern. We purposefully chose a single-page application backed by a modern backend, a complicated datastore, and a powerful, real-world search engine. All of these things were new to me, and being exposed to them has prepared me in the event I decide to continue creating web applications.

7.2   What non-technical information did I learn?

Some people will need pushing to get things done. There's no way around that, and so it is simply necessary to know when to push and how. It is necessary all throughout the year to make tradeoffs in who does what, when, and why.

7.3   What have I learned about project work?

It is easier if you just accept that some people put in less effort. After weighing how much I wanted to finish the project and how much recognition I would get from my group, my peers, the school, and others, I simply went ahead and did parts that we had earlier divvied up otherwise. The other option would simply have been to let the project go uncompleted, which I was not alright with.

7.4   What have I learned about project management?

I think that what I learned about project management can be summed up in one small line:

Divide up work early, and check progress often. Readjust as needed.

7.5   What have I learned about working in teams?

Working in teams is a give and take situation. For parts of a project where you are the most technically familiar, it is often a good idea to take on that section while someone else works on things they are more familiar with. On the other hand, if you have a little more free time, it can be good to instead work together on both sections, so that you can learn what they know and they can learn what you know.

7.6   If you could do it all over, what would you do differently?

I have told every single one of my friends that are going into capstone next year the exact same thing: **make a team and contact a good project**, and **do it — don't just do the rankings on the website**. It is worth it to not get stuck with strangers on a project that you didn't choose, and you might get stuck with a project you didn't choose even if you submit your rankings. That has lead this past year to be much worse than it could've been, had I just had the energy to put in to corral some of my friends into doing a project we liked. It was an alright year but it could've been much better.