Marcus Bullock

08/22/2021

CS 470 Final Reflection

<https://www.youtube.com/watch?v=8j0vxbf4oqY>

I would say being able to navigate AWS efficiently was a new skill that would be beneficial to my career and future path. Knowing how to deploy a full stack application and migrate one to the cloud is absolutely needed anymore as everyone is starting to slowly move to a serverless model. As for my strengths as a software developer, my biggest one is having to adapt. While I’ve only been a developer for a couple of years, I have had to wear many hats and figure things out on a rather small team. While this would be completely different at a bigger firm, this has given me the opportunity to learn about the full development lifecycle. I have recently submitted my two weeks and am taking a new job here soon. Things I’m preparing myself for is a bit less stress and having a bigger team to call on for issues I may not know how to solve. Another big thing I’m preparing my self for is simply not knowing the schema and having to put in some extra work the next couple of months to ramp my self up for the team.

The way I would handle scale is observe how much traffic is used, since you can scale on the fly with serverless it doesn’t hurt to start with lower resources and ramp yourself up. You can also scale down during the night if needed as well to save some money. For error handling, I would probably implement a few more errors responses for questions when an answer is given that shouldn’t belong in the field. I would predict my cost off how many users are active on the given application. I would say that serverless would also be more predictable and easier to manage. After watching Silicon Valley, and currently running into issues at my company with growing pains. I know all too well the many pros and cons of expansion. One of the biggest pro’s to expanding is you get more money. However, the biggest con is the money needed to expand. You need large amounts of money to even bring on bigger companies or to even think about expanding. You’ll run into issues where it’s not always as simple as throwing more money into resources. It could simply be that a lot of the application itself isn’t optimized to handle such a large customer. You’ll have to spend quite a bit of time optimizing several queries to handle such large datasets. You’ll also have to work continuously and slowly bring this customer on if they have several stores or users. Pay-for-service has definitely been the biggest money saver at out company, as you can just scale when needed and set up different instances fairly quick.