Branden Langhals

CS470 Final Reflection

October 8th, 2023

YouTube Video: https://youtu.be/Zw57cG-MZx0?si=lNVW-6pY9cu4PuH

Experiences and Strengths:

It has been a long journey for me. 4 years ago, I spent over $12,000 becoming certified in full stack software development at Case Western Reserve University only to find out that I cannot use any of it towards real college credit. Fast forward to the last class to earn my BS of Computer Science at SNHU and this course is the final checkbox of a massive achievement for me. As the first in my family to earn a degree, I can now use this time and experience and apply it towards shopping myself as someone with a bachelor's degree in the field I already work in. This knowledge directly applies to my current field working on a team of AGILE developers and QA Analysts. I learned a lot about using AWS services for hosting apps, and now feel confident using docker and containerization in my work, or in a testing environment.

Describe your strengths as a software developer.

My strengths as a developer are positive discontent in a constantly changing field paired with the willingness to learn. As new technology arrives and replaces the old ways, I will always look at the new plan with a fresh and supportive approach. Too many students will list specific languages or frameworks to answer this question, but I believe that a computer scientist should be aware of the entire picture. I have experience with a litany of technologies, and a general understanding of how to learn and apply new technology to creating software.

Identify the types of roles you are prepared to assume in a new job.

I will continue to work in my current role as a QA Engineer and bring new ideas to my team with greater confidence. I am also prepared to work as a developer and will be considering that move as well. Now that I have real experience with migrating an app to AWS, I do feel that I am more marketable than I was prior to this course.

Planning for Growth

- How would you handle scale and error handling?

A: Utilizing AWS Auto Scaling, microservices like AWS Lambda, for serverless scaling. And Amazon CloudWatch would provide sufficient monitoring for scaling based on CPU utilization. As for error handling, AWS Step functions can be used to build error-handling logic within serverless workflows, and CloudWatch logs can help capture data and assess my applications performance within the system.

- How would you predict the cost?

A: AWS Cost explorer can help guide me based on utilization. Ultimately my team would have to gauge how much data we will be passing back and forth and how much that would slurp up our usage.

- What is more cost predictable, containers or serverless?

A: Containers are more cost predictable as they are usually billed with a fixed amount of resources. Meaning you will pay for these resources regardless of if they are used or not. Serverless platforms charge based on calls and resources every time data is used. Potentially this could lead to increased costs if a workflow is executed more than anticipated. This is why many companies restrict the use of their API to a fixed number of calls, or don't allow public use at all.

- Explain several pros and cons that would be deciding factors in plans for expansion.

A: The first thing to consider is how many API calls we are going to be making with this app. If the application has hundreds of users making thousands of calls per day, we need to evaluate costs accurately so that we can make the right decisions. Another factor is the cost of microservices. While many will help with scaling, adding many microservices can add up quickly.

- What roles do elasticity and pay-for-service play in decision making for planned future growth?

A: Elasticity is crucial for working with fluctuating workloads and helps ensure that the app can handle both high and low traffic periods.

Pay-For-Service is great for cost transparency and control that helps us allocate resources where we need them and when we need them.