Final Reflection

Jonathon Gaspers

https://www.youtube.com/watch?v=Ec3O1TTXy4E

February 25, 2022

Final Reflection

Experiences and Strengths

The skills that I have learned, developed, and mastered within this course that will help me become a more marketable candidate in the software development industry are docker and docker-compose, serverless cloud environment development, and various services within the Amazon Web Services platform. I learned how to develop a full stack application, how to "lift and shift" the application into a docker container to be hosted in a cloud environment. I also learned how to move an application from a local computer to be hosted on Amazon Web Services using the S3, DynamoDB, Lambda, and IAM services on the platform.

My biggest strength as a software developer is my ability to quickly learn new technologies and adapt to new code bases. Being adaptable to new positions and being able to quickly get up to speed with the new role is something I consider one of my biggest strengths as a software developer. Other strengths involve having worked with many different technologies, for example, Android development, OpenGL programming, and even a little bit of TensorFlow for machine learning in Python.

Currently, the type of role that I feel most prepared for in a new job would be a entry to mid-level backend engineer working with either Springboot, NodeJS, or AWS Lambda/API Gateway. I really enjoy creating backend restful APIs, I have even started creating a backend for a service that works in conjunction with a mod for a game called Minecraft. I find it fun to improve the efficiency of the backend's we created in this course and in CS465, even to the point where I have changed the Single-Page-Application that we created in CS465 into an interactive database that keeps track of the books and shows I'm currently reading/watching. I would say that I feel comfortable working with backend API development.

Planning for Growth

Serverless computing is a great option for web applications that have an unknown amount of potential traffic because the service can allocate or deallocate resources based on the application's resource needs. Being able to allocate and deallocate resources to an application allows it to scale with the needs of the application rather than the developers needing to request a specific number of resources that the application may only use half of for each pay cycle. If we are using a serverless platform like Amazon Web Services, then we can set up lambda functions to handle the potential errors.

There are a few pros and cons that would be deciding factors when planning to expand the application, a major con could end up being costs, while a major pro would be possible increase profits. When expanding, it is important to look at the possible costs of the expansion, will the application need significantly more resources than it currently uses, or will we be able to continue with the current pay-for-use model when looking at hosting. Expansion would mean possible new users and traffic, which could lead to an increase in profit for the business, this is one major benefit to expanding.

Elasticity is the ability for a host to allocate and deallocate resources to a hosted application based on the application's needs. If an application is low-traffic and ends up having extremely high traffic for a few days, that wouldn't be a problem when hosting on a serverless platform that supports elasticity. Pay-for-service models are great as well for applications that have an easily predictable resource needs for an application. If the application will only ever need a certain number of resources, then a pay-for-service hosting service might be the best option. It is important to weigh both options when planning future growth as they each have their use-cases.