Andrew Riley

February 25, 2024

CS-470 Final Reflection

https://www.youtube.com/watch?v=LNlT23huoI0

Experiences and Strengths

- What skills have you learned, developed, or mastered in this course to help you become a more marketable candidate in your career field?
    - This course has given me many new skills that I will be able to carry over into the Software Engineering field. More and more applications are becoming serverless, cloud-based applications, so it is important to have familiarity with developing, deploying, and maintaining a cloud-based environment. Everything learned in this course, from developing an S3 bucket to deploying APIs will prove to be valuable as I pursue a career as a Software Engineer.
- Describe your strengths as a software developer.
    - As a new software developer, my biggest strengths are my critical thinking and problem-solving abilities. Throughout my educational career, many new programming languages, concepts, and principles were thrown at me, and at times it could be overwhelming. However, each time I was presented with a new challenge, I was able to learn quickly and overcome the challenge. While I still have a lot to learn as a software engineer, this has shown me that I not only possess the desire to be a software engineer, but that I can learn quickly in this ever-changing profession and continue acquire new skills that will allow me to remain successful.

- Identify the types of roles you are prepared to assume in a new job.
    - As I break into a new profession, I am prepared to assume the role of a software engineer. However, I am uncertain as to exactly what type of engineer I would like to be and am open to all possibilities. I feel that as a new software engineer, it is important to gain experience in many different roles to find the one that you not only enjoy but excel at. Therefore, I would be open to any role, whether it be as a frontend engineer, a backend engineer, mobile app developer, full stack engineer, etc.

Planning and Growth

- Identify various ways that microservices or serverless may be used to produce efficiencies of management and scale in your web application in the future. Consider the following:
    - How would you handle scale and error handling?
        - With the use of microservices, scaling can be done automatically with services such as CloudWatch through AWS. This would allow for the dynamic scaling of resources based on traffic. Error handling can also be automated using AWS Step Functions.
    - How would you predict the cost?
        - The first step would be to utilize a cloud price calculator, such as AWS Price Calculator. This would allow me to estimate the cost based on the

services I would be using. From there, I could compare various plans and providers and determine the most cost-effective approach.

    - What is more cost predictable, containers or serverless?
        - While costs for both containers and serverless can be predictable, containers are more predictable. This is because containers typically allocated a fixed amount of resources, therefore the cost would remain fixed and if additional resources need provisioned the cost would be known. Conversely, serverless allows for dynamic scaling, which causes the cost to fluctuate depending on traffic and is therefore more difficult to predict.

- Explain several pros and cons that would be deciding factors in plans for expansion.
    - When planning for expansion, some deciding factors would be scalability, cost, performance, and security.
        - In terms of scalability, a pro of cloud development is that scalability is easy and often done automatically. However, a con of scalability is that scaling can vary between cloud providers, sometimes leading to configuration issues.
        - When considering cost as a factor, cloud developments pay-per-use model enables cost efficiency as you only pay for resources being used. However, this can lead to excess expense if resources aren't optimized or if there are inconsistent traffic rates.
        - Performance is another deciding factor when planning for expansion. Cloud platforms offer numerous optimization features which can improve

performance. However, when utilizing the cloud, performance can be hindered by variables such as network connection, latency, or improper configuration.

        - Security is another thing to consider when deciding whether to expand. Cloud providers make security a priority, offering a wide range of security features. However, improper configurations could lead to gaps in the security, making the system vulnerable. Also, since the cloud is accessible virtually anywhere, it allows potential threats to try and access the system without needing to be physically connected to it.

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

    - Elasticity and pay-for-service play an important role when deciding for future growth. Elasticity is the system's ability to dynamically and automatically scale resources based on traffic. A system that optimizes elasticity would benefit the most from a pay-for-service model. A pay-for-service model means that the developer is only paying for the service, or resources, they are using. This means that if the system's elasticity is optimized, they won't be paying for unused resources during times of low traffic. This could cut costs immensely, meaning more funds would be available for expansion.