Thomas Cogley
18 December 2022
CS 470 Final Reflection
https://youtu.be/cGyzdAyl9yY

Cloud-Full-Stack Development

- **Experiences and Strengths: Explain how this course will help you in reaching your professional goals.**
  - **What skills have you learned, developed, or mastered in this course to help you become a more marketable candidate in your career field?**

I have learned how to take an application made with containers utilizing Docker and uploading it to a cloud service such as AWS. I also learned how to use different services on AWS. I learned how to use Amazon's Simple Storage Services (S3), Lambda API Logic, Amazon's DynamoDB. I also learned how to perform many different Queries on the Databases within the serverless cloud. Finally, I learned how to secure my cloud application by setting roles and permissions.

  - **Describe your strengths as a software developer.**

The biggest strength I have as a software developer is finding a solution. I am particularly good at using my resources to look for different solutions to a problem that may be giving me issues. Another strength I have is understanding how different programs connect with each other. How the front-end speaks to the back end and pulls information from the database to display to the front-end.

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

In a new job I am prepared to get a job as a software developer or a cloud developer. If a job requires me to use AWS cloud services or Docker containerization, I am prepared for that.

- **Planning for Growth: Synthesize the knowledge you have gathered about cloud services.**
  - **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?**

If I am using AWS, I would use Amazon Elastic Compute Cloud (Amazon EC2). Amazon EC2 Auto Scaling helps you make sure that you have the right number of EC2 instances available to manage the load for your application. As for error handling, I would use AWS Step Functions and AWS Lambda. You can use the step functions to create a serverless workflow that supports function error handling. The Lambda Functions can be created to mock an API.

- **How would you predict the cost?**

I would predict the cost by monitoring what I am using my application for. I would monitor what the user is selecting and how often. As I scale my application, I would continue to monitor everything and see what resources in my application are costing the most. For scaling I would use a pay-as-you-go model that AWS has. This will ensure that I only pay for what is used.

- **What is more cost predictable, containers or serverless?**

Containers are more cost predictable. This is because you pay for the entire service whether it is used or not. However, serverless applications can be more cost efficient. This is because you only pay for the time your application is executing rather than the entire time your servers are running. Also, serverless applications automatically scale to meet demand.

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

When deciding to expand it is important to weigh the benefits and downsides of expansion. Expanding will require hiring more employees, training new employees, and paying more for resources. As for pros, more revenue can be earned. If the product is already creating a profit, expanding the application can result in more profit.

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

Elasticity allows you to acquire resources as you need them and release resources when you no longer need them. Therefore, you only pay for what you use. This is beneficial when expanding. This will help you or the company save money by only growing the resources that need growth.