Cameron Carey
10/27/24
CS 470 Final Reflection
https://youtu.be/lv3k0DI4EGk

**Experiences and Strengths**

This course will help me reach my professional goals by giving me the skills to be successful in a career in computer sciences. The skills that I have learned and developed in this course include creating full-stack applications using MEAN stack methodology, use command line prompts to build, deploy, and containerize the application, and utilize AWS S3, Lambda, API Gateway, and DynamoDB services to deploy the web application via the cloud.

In addition to the aforementioned skills, my strengths as a software developer will help set me apart from other job candidates. These strengths include the ability to collaborate effectively with other people, unshakable perseverance, and a positive attitude. I plan to utilize these skills and strengths in a new job where I would see myself starting in an entry level position where I would continue to learn and develop my skills in order to progress my career to higher positions.

**Planning for Growth**

There are numerous ways that microservices and serverless features can produce efficiencies of management and scale within the development and deployment of the application. In particular, scaling the application is done automatically via AWS services such as S3, Lambda, and API Gateway. The error handling will be done in the code itself, which is for me, the developer, to implement. Estimating the cost of deploying the application will be difficult as AWS employs the pay-per-use pricing model. In other words, the application is billed per visitor and how much processing power they use, which means I won't be able to plan a budget without an idea of how much traffic my application will see. For this reason, predicting the cost of containerization is easier than the cost of serverless, even though this may allow serverless to cost less. Scaling and

error handling may be less efficient using containers, but the costs are more upfront and easier to predict.

There are a number of pros and cons to expanding an application via the cloud. One of the pros to using serverless services is automatic scaling, meaning that AWS will expand the storage capacity to accommodate more data and higher usage. Another pro is the pay-per-use pricing as mentioned above, which means that I don't have to worry about paying too much for coverage I don't need. A con to expansion would be a need for increased security. A larger application and increased traffic means there's more data to protect and more potential attackers to protect it from. Another con is that I will need to reassess the roles and permissions for the newly expanded application, not only for security purposes, but also to ensure that each role is able to perform only the actions and access only the resources needed to function properly.

To help in deciding for future growth, considering elasticity and pay-for-service will play a role in that decision. Serverless resources that offer elasticity as a feature help accommodate for both high-traffic and low-traffic periods of use. These services reduce wasted computing power by increasing capacity at moments of high traffic but also dropping the capacity in moments of low traffic. In tandem with elastic service, pay-for-service pricing will bill the application based on the number of users and the amount of time they spend utilizing the application's services, which will reduce the cost for expansion. When the traffic to my application drops, elastic services will reduce the capacity for low traffic, which reduces the services I will have to pay for.