Douglas Buckley
Professor Sandifer
CS-470-R3374
February 21, 2024

CS470 Final Reflection:

https://youtu.be/SJ-VwWzIp4s

- **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?

Throughout this course, I have learned valuable skills such as containerizing an application with Docker, orchestrating the containers with the help of tools such as Docker Compose, and deploying the containers from a local host environment to a legitimate online website with the tools provided by AWS. Being able to develop and deploy a serverless application or website will make me a more marketable candidate in obtaining a tech job, especially if it's a job in full-stack or any end of web development.

- Describe your strengths as a software developer.

As a software developer, I would say that I possess strengths in general understanding of systems and application function. Throughout my degree path, I have acquired multiple skills consisting of UML design making, reverse engineering, object-oriented programming, and debugging/testing programs that I'm developing. I also have basic understandings of security implementation both through code and policies created with AWS services such as IAM.

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

I believe that I have a few roles that I could assume in tech jobs. For instance, I have quality assurance knowledge and Scrum-Agile methodologies which make me an essential member of a team since I understand the importance of Scrum meetings and documentation. Having knowledge in cyber-security and JUnit/Google testing makes me suitable for a quality assurance role as well. Having knowledge in front-end and back-end web development can allow me to fill either of the roles, or even become a full-stack developer. I believe that I could also qualify for a cyber-security job since I do have knowledge in it. One concern that I have with my degree,

however, is since I'm just a "software engineer" is that I'm seen as a "jack-of-all trades, master of none" which could result in me not being picked over candidates with concentrations in fields such as cyber-security, quality assurance, or web development.

- **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 were to host my own web applications, I would more than likely start on a small scale and utilize local machines to host the application. I would image that only friends and family would be the first customers to interact with my system, so starting out small I don't scalability being an issue. However, if my application were to have a growing customer base, or even if it fluctuates between high demand and low demand, I would resort to utilizing services provided from AWS as I found it really interesting to accomplish everything for my website assignments for a literal dollar. Error handling would involve extensive testing and debugging, almost to an exhaustive level, however that's easier to say than perform.

- How would you predict the cost?

Predicting cost would simply entail doing extensive research on cost for up-time of a website, as I would assume that the rates are different depending on the service I'm attempting to use. For instance, I would to have consider prices for services used, such as if I intend to use MongoDB to host my data or another type of NoSQL database.

- What is more cost predictable, containers or serverless?

I would want to believe that containers are better to predict costs since they are constantly running, so it would have a constant rate attached to it. Since serverless development follow a pay-for-use model, I would need to have a correct estimate on the volume of customers that will

interact with the service as well as how long each customer interacts with the servers. This could be dictated through average count of users as average uptime per user, which can have varying results since this is an average estimate.

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

As far as pros go, having a successful application or service can result in a profit for myself or having more funds to improve the functionality or features included in my application. With an increasing audience, however, demands increased attention and resource requirements for the application, as customer will surely want more features included. If utilizing local servers for hosting the application, this would mean more research is needed to predict how much resources I would need to ensure that every customer is capable of interacting with the service. A serverless approach could help solve this, however with the increase of interactivity would mean that I would need the finances to pay for the hosting of the service.

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

Pay-for-service and elasticity play a huge role in the decision making of planned future growth. This is because I can ensure that I would only be paying for the ‘uptime’ of the website and not having to pay for ‘idle-time’, which depending on the usage of the application could actually never happen. Nevertheless, only paying for the users that actually are interacting with the service rather than a predetermined rate is a more cost-effective approach in the long run. This can also help with the scalability issue so that there aren’t issues of maxed-out servers which would result in customers being turned away from the application due to server overflow issues.