

Garrett Dunn

25th February 2023

CS 470

Southern New Hampshire University

CS 470 Final Reflection

- What skills have you learned, developed, or mastered in this course to help you become a more marketable candidate in your career field?

Before this class, I had no idea what Amazon Web Services was used for, nor could I conceptualize the power of the serverless cloud. It always seemed like some strange, hyper-technical black magic to me. Now, I can confidently say that I am prepared to deploy full stack web applications to the AWS platform using tools like S3, Lambda, API Gateway, and DynamoDB. There is clearly much more to learn about the AWS environment, and I look forward to looking into the many features that it provides.

- Describe your strengths as a software developer.

One strength that I have is that I can creatively solve problems to the best of my ability. As a hobby, I produce and mix music for myself and my friends, and I feel like having that background allows me to step outside of the box to break down problems in a more efficient way. Recently, I began the “Leetcode grind” to try and bolster my data structures and algorithms knowledge, and it has been really humbling. I try to solve a problem the best I can, and once the tests pass, I will optimize it from there.

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

I would like to try and go into full stack development, as the past two classes have shown me that it is the perfect mix of creativity and technicality. I am sure that as a fresh grad I wouldn't be thrown into anything too complicated or important. I am kind of anxious about the transition from school to the real world, as I have some idea of what to expect, but have no idea what to expect at the same time. I believe that SNHU has created a great foundation for me to stand on as I climb up the next rung of the ladder.

- How would you handle scale and error handling of a web application?

To handle scale and error handling of a web application in AWS, you can use Elastic Load Balancing to distribute traffic to multiple instances, Auto Scaling to adjust the number of instances based on traffic, and fault-tolerant architecture to ensure high availability. You can also use CloudWatch to monitor your application's performance and implement robust error handling and logging mechanisms within your application to capture errors and identify their root causes. Using AWS Lambda functions or Amazon SNS, you can send notifications when errors or other critical events occur within your application.

- How would you predict the cost of a web application in AWS?

To predict the cost of a web application in AWS, you need to identify the resources needed, estimate the usage of each resource, and use the AWS Simple Monthly Calculator to estimate the cost of each resource based on its usage and pricing. Additional costs such as data transfer, backups, and other services that may be needed should also be considered. Monitoring your application's usage and costs using AWS Cost Explorer or other tools is crucial to ensure that your actual costs are in line with your estimates.

However, it's important to note that costs can vary based on usage patterns, traffic fluctuations, and other factors. Thus, regular monitoring and optimization of your application's usage and costs is necessary to ensure that you're getting the best value for your investment in AWS.

- What is more cost predictable, containers or serverless?

In general, serverless computing is more cost predictable than container-based computing. This is because serverless computing is charged based on actual usage of resources, whereas containers require provisioning of a certain number of resources to run the application, potentially leading to unexpected costs if the resource requirements of the application are difficult to predict. Ultimately, the most cost-effective choice depends on the specific requirements of the application, with containers being more cost-effective for consistent workloads with predictable resource requirements, and serverless computing being more cost-effective for variable workloads or event-driven applications.

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

Expanding a web application can bring several pros and cons. On the pros side, expansion can increase the user base, improve functionality, provide scalability, and give a competitive advantage. On the cons side, there can be technical challenges, increased costs, user acceptance issues, and a risk of failure.

Technical challenges can include system stability and security, infrastructure upgrades, and backward compatibility. Increased costs can involve hiring development teams, upgrading infrastructure, and marketing and advertising expenses. User acceptance can be a challenge for new features and functionalities. The risk of failure can lead to significant financial losses and

damage to the company's reputation. Ultimately, the decision to expand a web application should weigh the potential benefits and risks, considering the company's resources, competitive landscape, and strategic goals.

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

Elasticity and pay-for-service play important roles in the planned future growth of a web application. Elasticity is essential for managing growth by ensuring that the infrastructure can handle the increasing demand for resources, without performance degradation. Pay-for-service allows companies to pay only for the services they use, rather than investing significant upfront costs in infrastructure. This model provides flexibility for scaling up or down as necessary and can help control costs. By considering these factors in the decision-making process, companies can create a scalable and cost-effective infrastructure that supports the growth of the web application while remaining financially viable.