CS 470 8-1 Assignment: Final Reflection

Mathew Dawson

December 17, 2022

Southern New Hampshire University

CS 470 Project Two Presentation

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?

Throughout this course, there are a couple of valuable things that I have learned which assisted me to appreciate software development and beyond. During this course, we learned about static webpages and how to migrate them to the cloud, in this case to Amazon Web Services (AWS). Specifically, for me, this was somewhat new and exciting, though I have had experience with Amazon Web Services in the past as I am a Certified Cloud Practitioner through AWS. I still learned a lot about how there are multiple ways that a database server, website server, or even APIs can have access beyond the AWS world. When using AWS, you can setup access through APIs to a database by utilizing HTTP requests and responses and those are used in conjunction with roles, policies, and permissions. These roles, policies, and permissions are setup by the root account user and the policies and permissions define what access is granted to the roles. By going further in depth with AWS and learning all the intricacies of the platform, this has made me a more marketable candidate and gives me even more skills for a better career choice and a better tomorrow.

Describe your strengths as a software developer.

One of the great strengths of mine as a software developer is my attention to detail. When I work on a program, I read every line of code that I have written and make sure there are no errors in what I have typed and included everything that is needed for the program to work correctly. In addition to this, I go above and beyond what is needed and try to include extra parts of code to help with the efficiency of the program.

Another great strength of mine is my organization skills. When working with a particular program, I make sure that all the code that is grouped together is shown and setup in a logical and efficient way. Furthermore, I make sure that the code is readable and that another user can fully understand what is going on in the code.

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

The types of roles that I would like to become are either a software engineer or a software developer as I love to work with the user and what they will interact with regarding the application.

Planning for Growth

How would you handle scale and error handling?

For correct scaling in a serverless environment such as AWS, you need to look at the requirements that your application needs to function correctly. After assessing the requirements such as the size of the hard drive space needed and other resources, then you need to assess the cloud-based environment that would support the application's needs. After the application is migrated to the cloud, the scaling of the application becomes much easier, especially with AWS. The resources can be setup in a way using various systems in the cloud. In addition, migrating to another location is effortless. Using a serverless environment, the migration of resources is used to scale.

For error handling, Amazon Web Services has something called AWS Step Functions that can be utilized with AWS Lambda. These Step Functions can generate a workflow that will manage errors. This can be done utilizing conditional logic to find exactly where the error is coming from. This lets you make an error handler API that is utilized as a link with a role and

permissions directed to this handler. This not only lets you catch any coded errors, but also catch anything out of the ordinary as well.

How would you predict the cost?

As said before, to predict the cost, you would have to look at the current resources that are utilized for the application. After this, any dependencies that are needed that could utilize resources would need to be looked at as well. Once the total amount of resources utilized is found, the predicted cost would be based upon the current available choices out there and their services. This is a great method to provide a rough prediction as to what requirements are needed.

What is more cost predictable, containers or serverless?

When comparing containers and serverless and which one is more cost predictable, this really depends on the application. Serverless can be more cost predictive over containers, but this really depends on the application. With a container, with each version of the application, comes with packaging it for distribution. With having an unpredictable size for future versions comes with a greater cost. In addition, a container must be managed, which will add to its cost. On the other hand, with a serverless environment, you only pay for what is used, which means that if the application is running but there is no traffic happening, then there are no resources being utilized. This same concept for serverless also goes for scaling, meaning that the more resources that are utilized, the more the cost goes up.

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

A pro that would be a deciding factor in the plans for expansion would be that if the application was a static type of website, this would mean that scalability is needed. So then, serverless would be the best choice. On the other hand, a con of this is the latency for a website of

this type. Due to the website being at a different location while being online, the time to get to the website would ultimately shift and customers and users would see a lag in response time.

Another pro that would be a deciding factor would be the capability to have your application up and running faster with a cloud-based service such as AWS than you would with a container. Using a container would require the building or having a space made, then installing all the needed software before the application can be used. A pro for serverless is the cost that would be needed to build and maintain the application and would cost a lot less than having a container.

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

Elasticity, regarding a serverless environment, means that it can adjust to the workload and its changes by modifying resources as needed. In addition to this, pay-for-service means you only pay for the resources used. Once combining both for planned growth, this decision is made based upon how quick the growth will be. There are a couple of questions to be considered when thinking about this like, how much growth? Do we move to a separate network to create space for the traffic, or is the current bandwidth location enough? With a serverless environment, the scalability factor is quicker because the network provider has better elasticity.