8-1 Assignment: CS-470 Final Reflection

Michael Vail

Computer Science Department, Southern New Hampshire University

CS-470 Full Stack Development II

Professor Nizar Dajani

October 22nd, 2023

Video Presentation Link: https://youtu.be/W_1I5qO0kcs

Experiences and Strengths

Throughout this course, I have learned how to move a locally hosted application to a cloud-based environment. We first began by learning how to utilize Docker to create containers to hold portions of our application and the dependencies they require. We also learned how to use Docker Compose to enable multiple containers to be run with a single command. We also learned how to create S3 buckets and Lambda functions. Using API Gateway, we created APIs for our web application and learned how to connect them to our Lambda functions. Another skill that I learned was how to use DynamoDB for storing data. From there, we learned how to tie everything together to create a working cloud-based application. These skills will be necessary for my desired career as a software engineer. I think that what I learned in this class will set me up with the basic knowledge I will need to continue to build on my skills as a software engineer.

I think that overall, my strengths as a software developer are my ability to troubleshoot my code and find issues that are causing bugs/crashes. I also think that another strength of mine is the ability to quickly pick up on different coding languages. I feel that I have been able to quickly pick up on the different nuances between the coding languages. I would say that I am prepared to assume the role of a junior software engineer or software developer. I feel like I have a solid grasp of what goes into creating applications either locally or cloud-hosted. This is not to say that I wouldn’t need further training before I can confidently create applications. I do feel confident in the skills that this class and others have taught me.

Planning for Growth

Using a cloud service provider, such as Amazon Web Services, we can utilize the numerous services AWS offers to plan for expansion. One such service can help implement an automated scaling plan. Using the service AWS Auto Scaling we can build scaling plans for other AWS services that will automatically adjust to supply the user with the best possible performance. To handle errors, I would utilize AWS Step Functions. This will allow me to create error handling for my Lambda functions. To predict the cost of a cloud-hosted application, AWS offers a service called Cost Explorer. This service analyzes data of the application to create a predicted forecast of your expected costs over a set period. I would say that overall, serverless applications would be more cost-predictable than containers. Containers need to be constantly managed and scaled accordingly. If resources go unutilized, these costs can begin to add up. With a serverless architecture, we can utilize auto-scaling to increase resources when needed and help save costs when experiencing low traffic. This also helps reduce costs incurred from managing and maintaining the application.

There are several pros and cons when deciding where to host an application when thinking about expansion. I would say that there are several pros for hosting in the cloud when planning for expansion. The application can be set up for auto-scaling, the server can be managed by the cloud provider, and a lower cost of not having to house the server. Some cons can be seen when a server requires constant management due to the nature of the application. I would say that elasticity and pay-for-service play a large role in the decision-making process for planned future growth. The flexibility that a cloud service provider offers is a large benefit when thinking about future growth. Companies want their applications to always perform well, not just when they are experiencing low traffic. This helps keep user satisfaction high and can help keep

them using the application. I would say that pay-for-service also plays a large role when planning for future growth. From my experience, companies do not like wasted resources. This is where pay-for-service can be beneficial. Since you are only paying for what you use, there is little to no waste. Overall, all these points should be considered when determining the future growth of an application.