Joseph Les

October 21, 2024

CS 470 Final Reflection

https://www.youtube.com/watch?v=zKiWgWpmu5Y&t=1s

In this course, I had the opportunity to take a pre-existing codebase, edit it to deploy on my local machine, then build a cloud environment, move the application to the cloud, and make it live. I learned how to read through an existing project, understand what it is doing, then revise it to my needs. This is directly related to my current career as a site reliability engineer, where I am continually tasked with improving existing codebases or deployment methods. Therefore, this course made me more marketable by expanding my experience of using AWS, since I currently work for Oracle Cloud Infrastructure.

My strengths as a software developer are being able to jump into ongoing projects, get caught up to speed, and quickly start adding value to the project. With that said, I am motivated to start working on projects from scratch. My strengths here are understanding the task, reading documentation, and achieving the goals I am working towards. My short and long-term goal is to become a freelancer, creating websites and personalized tools, and taking advantage of the cloud to host websites and SaaS services.

I personally enjoyed slowly building the cloud services out over the past several weeks. I took advantage of the time to read and better understand the services and why we chose that approach to host our website. Therefore, when I get ready to deploy my own websites, I would take a similar approach. I would use a serverless approach with S3, DynamoDB, and Lambda functions to handle scaling up and down. For error handling, AWS Lambda automatically

monitors and logs entries to Amazon CloudWatch. Therefore, it would be good practice to review logs and metrics on a regular schedule. For issues that require a quicker response, I will rely on alarms to push notifications. Using Cost Explorer with forecasting, I can help predict the cost of usage for various situations, such as less or more traffic demand. If I were to go the container route, though, that is easier to predict since it offers a fixed amount. Serverless costs can be unknown due to unpredictable traffic spikes that can happen.

When planning for expansion, several pros and cons need to be considered to make informed decisions. One of the main advantages of using a serverless strategy is its ability to automatically scale resources to meet demand. This elasticity ensures that the application can handle varying levels of traffic without manual intervention, which is especially beneficial for unpredictable workloads. However, there are cons to consider as well. One significant drawback is the unpredictability of costs due to variable usage, particularly during unexpected traffic spikes. This can make budgeting challenging. Moreover, using serverless services often means less control over the underlying infrastructure, which may not be suitable for applications requiring specific configurations or compliance standards.

Elasticity and pay-for-service models play crucial roles in decision-making for planned future growth. Elasticity allows an application to scale resources up or down automatically based on demand, ensuring optimal performance without manual scaling efforts. This is particularly important for applications with variable or unpredictable traffic patterns, as it ensures users have a consistent experience regardless of load. When planning for future growth, considering these factors helps balance performance needs with cost efficiency. By leveraging elasticity, businesses can ensure their applications remain responsive and reliable as they scale.