

## **Final Reflection: Cloud Development**

### **Assignment 8-1**

Connor Wallace

Computer Science, Southern New Hampshire University

CS-470-X6157 Full Stack Development II

Dr. James O. Webb, Jr.

August 16, 2023

Video Link: [https://youtu.be/acvu\\_POkgjA](https://youtu.be/acvu_POkgjA)

**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?

Skills		
Learned	Developed	Mastered
Creating Docker containers	Developing APIs to allow front end and back end to communicate	Researching documentation and related media to accomplish work goals
Orchestrating multiple containers with Docker-compose	Constructing NoSQL queries	Communicating with technical and nontechnical audiences
Using Angular to build an application for deployment	Using microservice architecture	
Creating, deploying, and configuring an application on an AWS S3 bucket	Refactoring code in a sizable codebase	
Deploying code on AWS Lambda		
Implementing a REST API on AWS Gateway API		
Creating and integrating a NoSQL database on AWS DynamoDB		
Using roles and policies for cloud security		

This class built upon skills we had developed in previous classes by having us move a project similar to one we had created to an AWS serverless cloud. This class was focused on architecture, hosting, security, and connecting services. Many of these concepts were new to me, and have continued to round out my computer science education.

**Describe your strengths as a software developer.**

As a developer I am able to quickly come up with a solution to many types of problems. I also am skilled at exploring existing solutions for improvements, either by building upon or replacing what exists. My previous work experience has taught me how to communicate effectively, research to accomplish work goals, and most importantly when and how to ask for assistance.

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

While I am most comfortable designing solutions for logical problems, my CS degree has prepared me for any position in a tech stack. I also have the mathematical experience to fulfil entry level data engineer and AI/ML positions.

**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?**

Serverless is designed to scale based on demand. During periods of high traffic, the backend database can often become the bottleneck. I would consider the following solutions to scale:

- Refactoring the DynamoDB to ensure the single table is well implemented and operating efficiently.
- Hosting the data in a self-managed database to better store our data and serve our customers.
- Moving to an asynchronous model utilizing Amazon Kinesis to throttle and batch high amounts of requests to be better served by our system (Mao, 2020).

For error handling AWS documentation recommends utilizing a state machine to handle and address errors (*Handle Errors in Serverless Applications*, 2023).

**How would you predict the cost?**

Many cloud services use a pay-per-use billing system. This can depend on several factors, including the memory used by the application, the duration of the code execution, and the number of incoming requests. If a projection of these numbers is available, predicting the cost should be simple based on pricing from the provider.

**What is more cost predictable, containers or serverless?**

Serverless applications tend to be cheaper to run than container-based projects. This is due to the pay-per-use system mentioned previously. Containers generate cost throughout their lifecycle while serverless applications are only billed while they are in use.

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

Pros	Cons
Lower cost	Cost of migration
Lower maintenance	Loss of control/Lower security potential
Simple scalability	

While there are many advantages to serverless hosting, there are some situations where it may not be the best fit. Some projects may benefit from a specific structure that can not be implemented or maintained on the cloud. Some projects may contain sensitive data that can not or should not be hosted out of house. Other projects may not gain enough benefit from a migration to justify its cost. However, many applications would benefit from making the change to serverless hosting.

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

The elastic nature of serverless applications allows for companies to save cost during low traffic while being prepared for high request volume. The pay-per-service plan is what makes this

elastic nature beneficial for the application owners. The agility of serverless application costs allows application owners to pursue growth with much lower concerns about handling traffic or downtime.

### References

*Handle errors in serverless applications.* (2023, February 1). Amazon Web Services, Inc.

<https://aws.amazon.com/tutorials/handle-serverless-application-errors-step-functions-lambda/>

Mao, G. (2020, April 16). *How to design your serverless apps for massive scale* / Amazon Web

*Services.* Amazon Web Services. Retrieved August 15, 2023, from

<https://aws.amazon.com/blogs/architecture/how-to-design-your-serverless-apps-for-massive-scale/>