

Tyten Perez

10/12/24

CS-470 Final Reflection

Presentation Video Link: <https://www.youtube.com/watch?v=vyNykgYThfk>

1. Experiences and Strengths: Explain how this course will help you in reaching your professional goals.

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

This course has helped me better understand cloud services and their underlying principles, like their dynamic scaling and approachable pay-as-you-go business model. The course tasked me with migrating and deploying a full-stack application to Amazon web services. Through this, I gained hands-on experience with AWS services like Lambda, API Gateway, S3 buckets, and DynamoDB.

b. Describe your strengths as a software developer.

The first CS-465 full-stack course helped me with developing a full-stack application locally. This course builds on this idea by migrating and deploying an application in a production environment like AWS. As a result, I have a strengthened understanding of both the front-end and the back end of a full stack application in a realistic cloud pipeline.

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

This course has made me more capable of assuming roles related to cloud software development and design. There are many careers involving cloud development. One example would be a Cloud Developer which designs and implements cloud applications using AWS services. Another example would be pursuing a DevOps Engineer role where I can test, manage, and deploy cloud application pipelines through a CI/CD framework. Other relevant opportunities might be a Cloud Solutions Architect for planning or Backend Server developer for API/database development.

2. Planning for Growth: Synthesize the knowledge you have gathered about cloud services.

a. Identify various ways that microservices or serverless may be used to produce efficiencies of management and scale in your web application in the future.

To handle scale, I would utilize serverless architectures like AWS Lambda, which automatically scales according to user demand, helping to ensure a consistent experience during peak times. Error handling might involve implementing robust monitoring and logging with services like AWS CloudWatch to monitor with real-time detection and can automatically resolve issues to maintain application reliability.

I could predict costs by analyzing application usage patterns and estimating resource consumption based on expected traffic. To assist in this process I could use AWS cost calculators and monitoring tools to provide detailed forecasts, allowing for budgeting and optimization of expenses according to actual usage.

Serverless architectures seem to be more predictable and manageable in terms of cost for variable workloads because you pay only for the compute time used. On the other hand, containers may incur costs even when idle, as they are likely designed to run continuously. This makes serverless a more economical choice for an application with dynamic demand.

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

Serverless architectures have the advantage of automatic scaling, which allows the system to adjust dynamically to user demand without manual intervention. Serverless also benefits from more predictable expenses with the pay-per-use model. Additionally, with the server infrastructure being handled by a different party, companies using the service can focus on development. A drawback of using a cloud service might be “vendor lock-in”, where a heavy reliance on a cloud provider's services can make future migrations difficult. Another drawback is how application availability and its security are at least partially dependent on the cloud service company since they host the content and services. It may also be more challenging to debug with multiple distributed micro-services, adding complexity.

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

Elasticity plays a part in decision making by adapting to moment-to-moment user demand. If future growth is expected, it can allocate more resources automatically or reduce resources if demand is low. The pay-for-service model provides a financially sustainable way to scale operations, as resource costs are more explicitly outlined in an easily understood manner. To help with budget planning, AWS offers cost-analysis tools as well. Together, these promote more strategic planning that balances performance needs with budget.