Nicholas Rusinski

CS-470-15136

Mar-02-2025

CS 470 Final Reflection

Presentation YouTube link

This course has significantly contributed to my professional growth by equipping me with essential skills that are highly valued in the software development industry. I have gained practical experience in containerization, cloud computing, and full-stack development, all of which are critical for modern software engineering roles.

I have developed skills in containerization using Docker and Docker Compose. Learning this skill will help in the future when moving projects to a production environment. Another topic is cloud computing, specifically using AWS services architecture. This understanding has given me the skills needed to understand the benefits of cloud infrastructure and utilize them in my future decision-making as a software engineer.

As a software developer, my main strength is problem-solving. I have displayed this through creative solutions that have surprised my professors with new and innovative approaches to tasks. Dedication to completing difficult projects often involves sticking with problems to find better solutions and keeping a forward-thinking mindset when planning solutions.

I am prepared to take on roles such as:

- Backend developer

- Full-stack developer
- Embedded developer

Through this course, I have gained valuable knowledge about cloud services, and I am better prepared to design applications that are cost-effective, resilient, and scalable. Microservices like AWS handle a majority of the management required for hosting an application. Each service is separate, although they are used together, allowing for easy updates and management of each service separately. Serverless solutions completely eliminate the need for managing infrastructure. AWS offers tools like CloudWatch to closely monitor an application, making it easier to understand bottlenecks or shortcomings. Services can scale automatically when needed, making applications dynamic, which reduces server load and overhead.

In a previous course, I had to design a moisture tracking application. The application could take input from moisture sensors, send raw data to an API through AWS API Gateway or AWS IoT Core, which then triggers an AWS Lambda function. The Lambda function processes the readings in real time and stores the data in DynamoDB, updating plant health stats in the app. The serverless Lambda functions take a load off the CPU, allowing for more efficient response times.

The pros and cons I would consider when planning future cloud applications are:

Pros:

- Automatic Scaling
- Reduced Maintenance

- Faster Deployment
- Cost Scaling for low to mid-volume

Cons:

- Cost Scaling is bad for high volume
- Limited Execution Times

The pay model is the biggest consideration when deciding if serverless or cloud expansion is possible. Although the pay-for-use model is great for low-volume or short-term applications or microservices, it can quickly become more expensive than traditional server architecture for complete enterprise solutions. Elasticity and scaling ensure that the application runs smoothly at all times, which doesn’t have many drawbacks.