Secally Barbosa

March 2nd, 2025

CS 470 Final Reflection

YouTube Presentation Link: https://youtu.be/g19TEp1Ibzk

Experiences and Strengths

I have acquired useful technical and problem-solving abilities throughout CS 470 that will help me achieve my software development career objectives. My capacity to develop and implement full-stack applications in a cloud setting—a crucial ability in contemporary software engineering—has improved as a result of this training. I've become proficient in serverless architectures, containerization, API development, and cloud computing services—skills that are highly valued in the tech sector.

I have improved my capacity to create scalable and maintainable apps by working on the last version of my cloud-based web application. I gained practical expertise with troubleshooting and performance optimization by testing the entire system and debugging API integrations. Making a presentation to document the project also improved my technical communication abilities, which are essential in work environments.

My strengths as a software developer are my flexibility, problem-solving abilities, and rapid acquisition of new technologies. I feel comfortable taking on positions where my understanding of cloud computing, automation, and software architecture will be useful, such as cloud developer, full-stack software engineer, or DevOps engineer.

Planning for Growth

I will use serverless computing and microservices to improve cost, performance, and management in order to guarantee the future scalability and effectiveness of my online application.

Handling Scale and Error Management

Strategic planning is necessary for cloud scalability. In order to keep the application responsive while keeping expenses under control, I would put auto-scaling policies into place to dynamically modify resources based on demand. A load balancer would also efficiently distribute traffic and avoid outages. I would use distributed logging and monitoring (like AWS CloudWatch and Datadog) for error handling in order to identify and fix issues instantly.

Predicting Cost and Choosing Between Containers vs. Serverless

Predicting costs is essential for cloud-based applications. Due to its pay-per-execution basis, serverless computing (such as AWS Lambda and Google Cloud Functions) may be more affordable for applications with varying workloads. However, because resources are allotted ahead of time, containerized solutions (like Docker with Kubernetes or AWS Fargate) could offer superior cost predictability for applications with consistently high demand.

Pros and Cons of Microservices vs. Serverless

When considering scalability, serverless computing and microservices (containers) both offer benefits and drawbacks. Microservices are perfect for applications with consistently high demand because they offer more control over infrastructure and more predictable performance. But compared to serverless systems, they need more setup and upkeep. However, serverless

computing removes the requirement for server maintenance and provides cost-effectiveness for applications with fluctuating workloads. On the other hand, it offers less control over the execution environment and may cause latency because of cold starts. The needs of the application will determine which of these approaches is best; microservices are more appropriate for sophisticated, long-running programs that need greater stability and customization, while serverless is best for event-driven jobs.

Elasticity and Pay-for-Service Considerations

When making plans for future expansion, elasticity is essential. I can make sure that the application adapts dynamically to traffic spikes by utilizing auto-scaling technologies, which will save over-provisioning and needless expenses. Instead of maintaining pricey on-premise equipment, I just pay for the resources I use thanks to cloud computing's pay-for-service approach, which offers flexible cost management.

As the application grows, I can balance cost and performance by using containerized microservices for core business logic and serverless computing for event-driven tasks.

Conclusion

I'm ready for a job in cloud-based software development thanks to the abilities I acquired in CS 470. I am capable of creating effective and flexible online apps because of my knowledge of cloud services, API development, and scalable architectures. I can make sure that my applications continue to be efficient and economical as they expand by carefully considering scalability, error management, and cost optimization.