8-1 Assignment: Final Reflection

CS – 470

Southern New Hampshire University

June 24, 2024

Dan Taylor

https://youtu.be/qvkVmmHuXEc

Completing CS 470 has equipped me with a diverse skill set crucial for achieving my professional goals as a software developer. The course's comprehensive focus on cloud development, containerization, and serverless computing has been instrumental in enhancing my technical acumen. By learning to migrate full-stack applications to the cloud using the Lift and Shift model, I have gained practical experience in ensuring efficient and minimal downtime transitions. This experience is vital for a career in software development, where seamless and efficient deployment processes are highly valued.

In this course, I mastered the use of Docker for containerization, ensuring consistency across development, testing, and production environments. Understanding how to use Kubernetes and Docker Compose for orchestration has also been a significant skill development. These tools are crucial for managing and scaling applications efficiently. Additionally, learning about serverless computing, particularly through AWS Lambda and S3 storage, has broadened my ability to design scalable and cost-effective solutions. The skills gained in handling serverless APIs, including automatic scaling and the pay-as-you-go model, make me a more marketable candidate in the software development field.

My strengths as a software developer lie in my ability to adapt to new technologies and apply them effectively to solve real-world problems. I am proficient in creating scalable, efficient, and secure applications using cloud-based principles. The ability to work with both document-oriented databases like MongoDB and key-value databases like DynamoDB highlights my versatility in handling different data models and requirements. Furthermore, my understanding of

cloud security, including the use of IAM roles, MFA, and encryption, ensures that I can develop applications that are both robust and secure.

I am prepared to assume roles such as Cloud Developer, DevOps Engineer, and Full Stack Developer in a new job. My experience with cloud-based development, containerization, and serverless architecture positions me well for these roles, where I can leverage my skills to design, deploy, and manage applications effectively.

Planning for Growth

Planning for the future growth of a web application involves synthesizing knowledge about cloud services to ensure scalability, efficiency, and cost-effectiveness. Using microservices and serverless computing can produce significant efficiencies in management and scale. Microservices allow for modular development, where each service can be developed, deployed, and scaled independently. This modularity enhances flexibility and resilience, as issues in one service do not necessarily impact others.

Serverless computing, such as using AWS Lambda, offers a model where the cloud provider manages the server infrastructure. This approach simplifies scaling and error handling since the infrastructure automatically adjusts to handle varying traffic levels. Additionally, serverless architectures reduce operational complexity and costs, as you only pay for the compute resources used. Predicting costs in a serverless environment involves understanding usage patterns and planning accordingly. While serverless computing offers cost predictability due to its pay-as-you-go model, container-based approaches using tools like Docker can sometimes be more cost-effective for predictable and sustained workloads.

When considering the pros and cons for expansion, serverless architectures provide automatic scaling, reduced management overhead, and cost efficiencies during low activity periods. However, they may incur higher costs during peak usage if not managed properly. Containers offer more control over the environment and can be more cost-effective for steady-state applications but require more management and orchestration effort.

Elasticity and pay-for-service models are critical in decision-making for planned future growth. Elasticity ensures that the application can handle varying loads efficiently, scaling resources up or down based on demand. The pay-for-service model allows for cost savings by paying only for the resources consumed. These principles ensure that the application remains both performant and cost-effective as it scales.

In conclusion, the skills and knowledge gained in CS 470 have not only enhanced my capabilities as a software developer but also prepared me to plan for and manage the future growth of web applications effectively. By leveraging cloud services, microservices, and serverless architectures, I am well-equipped to design scalable, efficient, and secure solutions that can adapt to changing demands and technologies.