July Wellman

CS 470: Full Stack Development II

Southern New Hampshire University

October 19, 2025

Final Reflection

YouTube Link:

https://youtu.be/qmo78eCySek

Experiences and Strengths:

Skills that I acquired during this course include hands-on experience with full-stack web application development, containerization, and cloud deployment. This course provided valuable insight into how modern applications are designed, built, and maintained in real-world environments. I gained a stronger understanding of how front-end and back-end systems interact through RESTful API integration and how containerization with Docker streamlines deployment across multiple environments. I also developed proficiency in cloud-based services such as AWS, learning how tools like S3, Lambda, and API Gateway support scalable and secure applications. Through this experience, I improved my problem-solving and debugging skills, learned to manage complex application architectures, and strengthened my ability to collaborate effectively within a development team. Everything I learned in this course is directly relevant to a career in software development, particularly in environments that emphasize efficiency, scalability, and automation. Roles I could take on with this experience include Full-Stack Developer, Cloud Developer, or DevOps Engineer, where I can apply both my technical skills and my understanding of modern cloud-native workflows.

Planning for Growth:

In planning for future growth, cloud services provide the scalability and flexibility needed to efficiently manage expanding web applications. Implementing microservices or serverless architectures allows each service to scale independently, improving fault isolation and reducing downtime during updates (Amazon Web Services, 2023). To handle scale and error recovery, auto-scaling and monitoring tools can automatically adjust resources in response to traffic demands. Cost prediction depends on workload patterns, containers generally offer more predictable expenses for steady use, while serverless options are often more economical for variable workloads. Elasticity and pay-for-service models remain critical advantages of the cloud, enabling applications to dynamically adjust resources and maintain financial efficiency (Microsoft, 2023). These capabilities make cloud-native strategies essential for sustainable, long-term application growth.

References

Amazon Web Services. (2023). *AWS architecture best practices*.

https://aws.amazon.com/architecture

Microsoft. (2023). *Azure scalability and monitoring overview*.

https://learn.microsoft.com