

Devin Perry – 4/22/24

CS-470 Final Reflection

<https://youtu.be/fjUBQ6vM4x8>

Completing CS-470 has been instrumental in sharpening my skills and preparing me for my professional career in the tech industry. Throughout the course, I've acquired a vast understanding of the frameworks and tools essential for building comprehensive web applications. These practical skills, combined with theoretical knowledge, have significantly boosted my confidence in pursuing roles as a software developer. One of the key skills I've learned and mastered in this course is the ability to develop scalable and efficient web applications using AWS. From front-end technologies like HTML and Angular, to back-end frameworks like Node.js and Lambda, I've gained proficiency in building complicated and responsive user interfaces coupled with powerful server-side logic. Additionally, looking into database management systems like MongoDB and DynamoDB, equipped me with the skills to design and maintain efficient data storage solutions.

As a software developer, my strengths lie in my problem-solving abilities and attention to detail. Though I may not know how to read/write tons of coding languages, I thrive in dynamic environments where I can apply my technical expertise to tackle complex challenges. Moreover, my collaborative nature enables me to effectively communicate and collaborate with cross-functional teams to achieve project objectives. Upon completing this course, I feel nearly completely prepared to assume roles such as a Full Stack Developer, Software Engineer, or Web Developer in a new job. My diverse skill set and hands-on experience in developing end-to-end web applications make me a valuable asset to any tech team.

Taking all the knowledge I gained about cloud services, I now have the ability to provide insight into a few important points. I can envision leveraging various microservices or serverless architecture to facilitate future growth and scalability of my web application. By breaking down the application into smaller, decoupled services, I can achieve greater flexibility, agility, and ease of management. To handle scale and error handling, I would implement auto-scaling capabilities in my cloud infrastructure. This would allow resources to dynamically adjust based on demand, so there would be minimal need for manual efforts. Additionally, incorporating comprehensive monitoring and logging mechanisms would be important. Doing so would enable proactive identification and resolution of errors, ensuring uninterrupted service. In terms of cost prediction, while both containers and serverless offer scalability and cost efficiency, serverless architectures typically provide more predictable costs due to their pay-per-execution model. However, containers may offer more control and flexibility in resource allocation, but with potentially higher management overhead. When planning for expansion, several factors must be considered. Planning for things such as performance, reliability, security, and cost are important. While microservices offer agility and scalability, they may introduce complexity in management and communication between services. On the other hand, serverless architectures simplify infrastructure management but may create limitations on resource utilization. Elasticity and pay-for-service models play crucial roles in decision-making for planned future growth. Elasticity allows resources to scale up or down based on demand, ensuring optimal performance and cost efficiency. Pay-for-service models enable organizations to pay only for the resources consumed, aligning costs with usage, and minimizing waste. By considering all the above concepts and “this-or-that” questions, I can effectively choose when to use what based on structured knowledge.