

Joseph Valle

CS 470

Final Reflection

3/2/24

Taking this course has helped me in many ways. From learning the functions that are needed to make an application live to learning the functionality of AWS cloud services. A huge part of this course was the introduction of AWS cloud services opportunities. Come into this course I thought professionally I would be taken the route a full stack developer. With the knowledge of HTML, CSS, JavaScript and node.js, I have also gained the interest of what AWS clouds services professional roles I could follow. We first began by creating a full stack application. I have gained expertise in writing and developing code. I have also begun to learn the importance of testing and using tools such as POSTMAN. In this course I have also learned the ability to develop and test APIs. Some of my biggest strengths include the ability to adapt and learn new processes. AWS services were a completely new tool to me that I ended up enjoying. Learning something new and consistently growing with the industry is something I see as a strength. Technology is always evolving. A huge piece of my strengths also includes that ability to work in a team and take on leadership roles. Certain roles I'm currently looking for that I am prepared to take on are Cloud architect roles. I plan on taking on and learning all AWS certifications to make sure I'm prepared, and I make myself more marketable beyond a developer. My goal right now is becoming an IOS engineer as I have worked on writing code app as I continue to learn the development process of applications.

To handle scaling, I would design an application that's able to scale horizontally using serverless functions. In the event of error handling my main attack of defense would be to introduce logging mechanism to record and identify errors. This would help fix the problem in the moment rather than letting it lag on. Logging allows me to circle back if issues like this arise again and have a starting point to fix it. Predicting the cost, I would see how often certain requests are made, how much storage I need to begin with and use tools such as AWS Cost Explorer. With everything we've learned using a serverless server would be the direction I would take. It offers models such as pay-as-you-go. This model allows you to pay for the services you use. You may need certain services but if they are not in use they are not charged. The scalability is also quite endless and no reliant on any physical storage data centers that can be quite costly. All these are huge benefits of serverless that usually outweigh any cons people may see. Elasticity also plays a huge role as it can scale resources as the demand is raised while also only paying for them as they're needed. Here as developer, I can safely say I am optimizing my performance of the application built while also keeping costs to what is needed.