Kyle Gaskill
06/25/2022
CS 470 Final Reflection
https://youtu.be/fX6im0Bg_Kw

Taking Full-Stack II I got hands on experiences with AWS with creating API's with Lambda functions and connecting a frontend and backend with AWS services. It was a great experience learning the steps in creating a front end hosted on AWS with a database, but I will need to continue to practice developing my skills where everything becomes second nature. Like all things, practice will improve my flow process of setting up the various lambda functions. Converting an application to a serverless architecture was a skill I feel would benefit me as working with how serverless architectures work could help broaden my career paths. I found the overall experience rather enjoyable and never gave much thought about being a cloud engineer or a full stack developer until this course. My strengths as a developer are by ability to debug and critical thinking to solve complex problems. I feel these are good strengths to have as a developer since solving complex problems and finding solutions to errors is a trait strong software developers possess. I am ready to assume roles currently working in network engineers and hope to gain more hands-on experience to become a security analyst.

AWS offers a lot of features to handle a web service or application with on demand scaling. AWS offers Auto Scaling automatically and AWS's step functions with lambdas can be designed around workflows that are fault-tolerant with built-in try/catch functions. This will be beneficial for handling errors. Amazon's web services have multiple features that make designing and implementing your web site or application seamless. AWS offers a Pricing Calculator that can create estimations of costs per month around the webs services you choose to configure inside AWS portal. This will give you a good estimate upfront but with auto scaling, costs can fluctuate based on peak demand of your website. Luckily, these costs will fluctuate back down during off peak hours. Serverless is more cost efficient over containers, due to paying for your usage and code not having to constantly run on a serverless architecture.

When planning for an expansion of your web application, research should be done to understand more on the market you are launching your product into. Businesses that can expect a lot of traffic, like social media hosting applications, business websites for online orders, etc. should expect a decent number of services running upfront. This would require a decent number of trained staff onboard to handle the application. Cons of expanding too big too early would result in increase operation costs that could take a hit of your net profits. Having too much demand with little workforce would also result in rollout delays due to not enough manpower programming, updating, and handling logistics of the web application. Elasticity plays an important part in serverless cloud systems by utilizing the necessary resources to operate to efficiently run the web application and not overusing resources when traffic is lower. The pay-for-service plan plays an important role in the future growth plan with elasticity as it will only charge you the resources you utilize. When the application scales up to handle more traffic, you will serverless architecture will expand and only be charged those resources at time of roll out. This will help operating costs for your growth plan by saving money due to the efficient design.