Jon Way

4/21/24

CS470 Project Reflection

https://youtu.be/1aqZP8xhdok

The skills I've developed during this course include developing an application for the cloud. I've learned how to migrate a program to an AWS server by uploading files, creating Lambda functions, setting up API gateways, and establishing a database through AWS services. I think as a software developer one of my strengths is my attention to detail. I try to be very consistent when I write my code so that it is easy to follow and read if you are looking at my program with no prior knowledge. I'm also very creative and can find alternative solutions to problems. As a new developer in the field, I will be able to work in all aspects of application development as I have worked on full stack projects both on a computer and on a cloud-based server.

Handling scale and errors in serverless systems requires some knowledge of how a serverless system works. As with any software development, I would need to work on debugging an issue, tracing it through the program's run-time to figure out where the issue is being caused and what is causing the issue to resolve it. With development, predicting the cost of the application would involve getting pricing from the serverless system and identifying and estimating how much usage your application is going to use. I think it can

be tough to predict the cost for either the usage of containers or the usage of serverless systems, but I think containers might be more predictable if they have flat fees. You would at least be aware of the exact amount that you would be paying when you do your upgrades compared to serverless systems that charge you different amounts based on usage and run-time as well, which can fluctuate.

I think elasticity and pay-for-service are important to consider when planning for growth because they can really change how your application progresses. If you predict that your application will grow quickly, or you don't have the manpower to expand your servers in a timely manner yourself, then the elasticity of a serverless system would be better for your application. The pay-for-service is helpful to keep the cost efficient as well. Depending on your application, if you make more money from the people using your application, then you can simply increase the payments made to the serverless system as it grows instead of needing to shell out a large sum for a server capacity increase.