Alex Alder

Module 8 Assignment

June 2025

I have learned a lot about how cloud infrastructure works and the various methods to achieve a result such as hosting a web application both front end and the API/database backends. Learning the differences between docker containers, and serverless infrastructure has been a great exercise. I understand that each has its own benefits and drawbacks. Having this knowledge will allow me to provide valuable input when considering a companies' cloud stack and which method might be best for the end result. As a software developer, my strengths are picking up on new technologies and helping to understand what methods might suit a certain application. There are often a number of options, so being able to decipher which method is best requires a good general set of knowledge.

With utilizing serverless it can help greatly with scaling, as a lot of it is already handled by AWS automatically scaling as more requests come in. If you are unsure of your traffic, it could be good to consider serverless. Containers may be less predictable to understand costs as it can be difficult to determine how many containers would need to be spun up to handle an increase in load. Whereas, serverless you have a clear understanding of how much each request will cost and the bandwidth required to handle these requests.

Pros for expansion would be more in the serverless area since AWS will automatically scale the application for you. However, this would also mean that you are locked in with AWS and should they decide to increaes costs or if for any reason AWS stops serving your data it would take a lot of time to restore service. you don't have much choice. Docker containers require a more complex method to scaling such as kubernetes to spin up and down containers as needed. This adds some complexity with the data layers, but you can take your containers anywhere with you quickly and easily. This would allow you to have deals with more than one provider or go bargain shopping for hosting.