

Experiences and Strengths: Explain how this course will help you in reaching your professional goals.

- What skills have you learned, developed, or mastered in this course to help you become a more marketable candidate in your career field?

This course has helped me to become a more marketable candidate in my career field by teaching me skills related to full-stack web development, containerization and introducing me to resources like Docker and Amazon Web Services (AWS). These skills without a doubt make me more appealing to hiring managers, as in the last 3 weeks alone I have seen multiple job postings asking for experience with containerization and products like AWS.

- Describe your strengths as a software developer.

My strengths as a software developer are that I can learn very quickly, I am very good at researching, troubleshooting and problem solving and I am very thorough with a high attention to detail and ability to properly comment out code. I can work good in a solo environment as well as a team environment and I hold myself accountable for my actions and work.

- Identify the types of roles you are prepared to assume in a new job.

The types of roles I am prepared to assume a new job in would be a full stack developer, front-end developer, back-end developer, cloud architect or a software engineer.

Planning for Growth: Synthesize the knowledge you have gathered about cloud services.

- Identify various ways that microservices or serverless may be used to produce efficiencies of management and scale in your web application in the future. Consider the following:

Microservices can be utilized to make large applications easier to manage and scale as they break down the application into smaller independent services. They can be used to make changes to individual services offered by the application without influencing other aspects of the application. The serverless architecture means that I or the team I am working with will not have to deal with managing the servers which will reduce vulnerabilities and save time and money that can be utilized elsewhere. When it comes to handling scaling and errors a monitoring system can be utilized for identifying errors and performance-based issues. For predicting pricing there are cloud pricing calculators that can be used to get an estimate on the cost of running an application. Containers will be more cost predictable than serverless due to their cost being based on how much resources they use, while serverless is based on the number of requests received. This does mean that when your application is seeing less traffic serverless will help you save money since you are not paying for what you are not using, but it can also mean you sometimes will have a larger bill then you were expecting.

- Explain several pros and cons that would be deciding factors in plans for expansion.

Some of the pros of serverless when planning for expansion are that you will not be charged more unless you begin to have more traffic, you do not have to worry about updating and maintaining the database and if something goes wrong with the expansion launch you can go back to an older version of your application while you make corrections to the expansion.

- What roles do elasticity and pay-for-service play in decision making for planned future growth?

Elasticity and pay-for-service help to keep costs down when making plans for future growth, as you only need to pay as the growth starts to happen. Prior to serverless if you were planning for future growth there was a lot of upfront cost as you needed your application to be able to handle all the extra traffic and therefore had to pay for it up front. You can very easily scale up or down the resources you need based on your average usage and if you do receive random spikes in usage, you can feel safe knowing that your application can automatically scale to handle those extra requests and then scale back down to your baseline and you just need to pay for the scaling up while you were using it.