

Aaron Shipley

CS 470: Final Reflection

Southern New Hampshire University

Link: <https://youtu.be/rsQKdHmI6aA>

Throughout this course, I have gained a better understanding of how the deployment of web applications can be taken from a local environment, containerized, and then deployed via cloud-based services like AWS services. Before this course, I had minimal knowledge that pertained to how an app was deployed and cloud services in general. Learning how to containerize a web application through docker Desktop has led to a better understanding of how applications can be deployed to multiple machines while keeping the proper versions and dependencies across all platforms. This was an area I was ignorant about before this and had shed light onto how this is done. Another skill learned was how to create buckets with S3 to store the data and files for other AWS services to access. Lambda functions are now an area I understand in their structure and how to properly utilize them as well for my cloud-based application. API Gateway has helped me in understanding APIs more in-depth as well.

Throughout my 3 years at SNHU, I have been able to identify strengths that I possess as a software engineer. Some of these strengths were developed in previous roles that were not related to software engineering but then applied to different aspects of the SDLC. With multiple years of customer service and employee training, I was able to incorporate these skills into projects. Systems analysis, UI/UX design and writing test cases were some fields that I was able to use these skills to convey my message to technical stakeholders and non-technical stakeholders alike. Along with these skills, I have identified niche areas of software engineering that I have excelled at and have gravitated toward. QA Testing, software development, Systems Analysis, and Database management are some roles that this degree pursuit has prepared me for. These areas are ones that I have investigated through job searches and have had the opportunity to interview for.

In terms of applying cloud services and the serverless portions of AWS can be applied to make an application more efficient, we can look at our application that we deployed. By using AWS services, I will be able to handle errors within my code that may arise when the application is updated or modified, AWS backups of the application can be utilized to deploy the application before the errors were present.

This will allow me to start back at the functioning version of the app and re-work my updates or modifications to avoid the same errors the next time. By using AWS Lambda, I will be able to write test cases for functions created to properly test them and examine the results from the functions to ensure they match the expected results. This is one way of ensuring the functions are working properly as well as working in an iterative manner to create and test code as it is developed.

By using AWS services that are serverless, scaling of the application will be done automatically. Along with scaling up and down for the application, AWS services will also maintain and manage the backend infrastructure of the application, letting me focus more on the frontend and the code itself. This is a convenience in efficiently deploying the application but must not be taken for granted. When it comes to running an application cost-effectively, it depends on the application itself. If the application were to be running constantly, for example a company's inventory system, the most cost-effective route would be to run it in containers locally. This could be done via local machines or through a VM on a cloud-service. By running the application non-stop, a local system would be the best option, keeping the application reliant on local machines to run it. If the application was a customer-facing web app, deployment through a serverless model, like AWS services would be the best option. The app would be charged based on its use. This pay for what is needed / used model is more cost-effective in this manner.

Some cons of running an application on local machines would range from machine failure, natural disaster, and maintenance costs. If the system is based off physical machine on location, a small failure of one of the machines could disrupt service and be quite costly. An instance of failure could be due to natural disaster. Also, having adequate staffing on hand to maintain the machines and system will be necessary to keep the application running properly as well. When using the serverless deployment, there will be some drawbacks as well. Services like S3 rely on having a stable and available internet connection. If an internet connection is not available, the user will not be able to properly access buckets and work with them as needed. By having an application deployed through the cloud, one is at the

mercy of the services offered as well. If there were to be an error in the services offered, one may not be able to use them correctly and may be subject to potential outside threats. This must be a constant concern for users and a proper mitigation must be in place if something like this were to happen.