Trent Hesler

CS470

10/27/24

8-1 Assignment: Final Reflection

In this class, I learned many skills that I can absolutely see myself putting to use in my future career. For instance, before this course, I had never done any work with containers. I was aware that containers existed, but I really didn’t have an in-depth understanding of how they worked and how they could benefit my life and workflow.

We also learned about the AWS ecosystem, which I was hitherto unfamiliar with. I know that many companies use AWS for their backend and it was always something I had wanted to learn, so this class was the learning opportunity I had been waiting for! I feel confident creating and manipulating software in AWS and will carry this skill set with me as I grow in my software development career.

One of my biggest strengths as a developer is that I am able to learn new things—be they programming languages, concepts, or workflows—very quickly. There have been occasions where I needed a piece of open source software, written in a language I was unfamiliar with, to do something that it was not designed to do. In these instances, I just forked the project and learned the programming language as I went along adding functionality to my fork.

Serverless technology is extremely popular in the current software development zeitgeist. Knowing how to leverage this technology will be critical to breaking into and staying relevant in my career field, so I am very grateful to have learned about it. It is a technology that allows software projects to be spun-up quickly and scale effortlessly, which can be very useful in today’s increasingly cloud-based world.

Completing this class, I feel confident in my ability to be a full-stack developer, or work on projects hosted within the AWS cloud. I have a solid understanding of how frontend and backend applications communicate with one another to form a complete workflow. I also understand how to transition a piece of software from a self-hosted environment to the cloud. These skills will be invaluable to me as I move towards being a software developer.

In a serverless environment, nearly all concerns regarding scaling and load balancing are handled by the cloud provider. This would free the development team up to focus strictly on development. It also might ease or completely offload some of the growing pains as a company expands.

One the other hand, if one was self-hosting, one would have to closely monitor resource usage to maintain stability as demand increases. If demand increases beyond a certain level, new hardware would have to be acquired to accommodate it.

In my estimation, serverless is more predictable, in terms of both resource management and costs. The cloud provider handles the majority of the concerns regarding infrastructure and scaling, as well as many concerns around security.

For example, if a piece of server hardware breaks down when one is self-hosting, it could cause a major interruption in service, as well as potentially a major expense. In a serverless setting, the customer is unlikely to even know when a server has gone down, as there are redundant systems in place to compensate. Furthermore, the serverless customer would not be expected to shoulder the expense of replacing any hardware, meaning no unexpected repair bill.

One factor that would make someone against moving a project into the cloud could be concerns surrounding data privacy and security. If an organization was handling sensitive or highly secret proprietary information, it might be safer to self-host. One of the biggest downsides to using a serverless service is that you have less direct control over how things operate under the hood. If an organization has to meet strict security standards or maybe is just wary of hosting their data on the public internet, self-hosting may be the better option.

Link to my final presentation: https://youtu.be/OBIoaLM54kI