Jeremiah Kellam

CSD380 Module 6.1 Discussion

7/16/2024

Monolith vs. Microservices

Hello,

For the Module 6 discussion I will be covering the main topic of Monoliths vs. Microservices and when each of them might be used. Monoliths and Microservices are two different ways to approach software architecture. “A monolithic application is built as a single unified unit while a microservices architecture is a collection of smaller, independently deployable services” (Harris). Harris described how Netflix grew so big that its infrastructure was failing, so Netflix migrated to cloud storage and replaced its monolithic architecture that it built from the ground up to a microservices architecture that allowed diversifying what handled different aspects of the business.

Monolithic architecture is the traditional way that an application would be made that is a unified single structure handling all facets of the application. Making changes to the monolith involves dealing with every part of how the pieces of code fit together and how they interact. Their main use would be found for start-ups, like Netflix, that have low overhead and are easily deployed at the beginning. But much like Netflix, monolithic architecture is not easily scalable. It is not a flexible system and usually has a slower development speed due to its complexity (Harris).

Microservice architecture is a method that relies on independently deployable services. The use of Microservices decouple business into independent code bases that are domain specific and even though the complexity is still there, it is more visible to see the smaller processes to manage them and handle development or any issues that arise (Harris). Microservices are used by many large businesses today, like Atlassian where I found the article.

Advantages of Microservices include being agile to handle CI/CD initiatives, they’re flexible to changes as each service is independent, and they’re more reliable due to being separate and not having a systemic crash that Monolithic architecture is susceptible to. But their disadvantages may prevent their use due to the high overhead in not only development and use, but also maintenance and a lack of standardization that can blue lines of what handles which tasks. Both are applicable in different stages, and both have challenges to overcome. Hope you enjoyed.

Thank you for your time,

Jeremiah K.

Reference(s):

Harris, C. (n.d.-a). Microservices vs. monolithic architecture | Atlassian. <https://www.atlassian.com/microservices/microservices-architecture/microservices-vs-monolith>