1. What is meant by the phrase Monoliths vs. Microservices? When might each be used?

Monolithic applications are built as a single unified unit. It is the traditional model that of most software programs. It is independent of other applications and has one codebase that puts together all of its services. When updated, the developers will revise all stacks and would deploy an updated version. It allows early projects ease of code management, low cognitive overhead, and easier deployment.

Microservices is an architectural method that is made up of a series of independently deployable services. They are like mini monolithic applications that have their own business logic and database with specific goals. The microservices can update, test, deploy, scale within each iteration. A good note about microservices is that they don’t reduce complexity but make that complexity visible by separating the tasks into smaller processes. It is huge in continuous delivery practices and allows teams to adapt quickly to user requirements. It allows software updates to be pushed frequently which can improve reliability, uptime, and performance. Microservices allow agility, flexible scaling, continuous deployment, highly maintainable and testable, independently deployable, technology, flexibility, has high reliability, and happier teams.

https://www.atlassian.com/microservices/microservices-architecture/microservices-vs-monolith#:~:text=A%20monolithic%20application%20is%20built,one%20is%20right%20for%20you%3F