Monolithic architecture and microservices architecture are two distinct approaches to designing applications.

In monolithic architecture, the entire application, including all its features and functionalities, is built as a single codebase. This means that all the features are bundled together into one binary or image. While this approach can be straightforward to develop initially, it introduces challenges as the application grows. For example, if there is an issue with the application, developers and testers must analyze the entire codebase and all its components to identify and resolve the problem. This becomes more difficult as the application scales.

To address these challenges, experts introduced microservices architecture. In this approach, the application is divided into small, independent services. Each service is responsible for a specific feature or functionality and is built, deployed, and managed independently. Each service has its own binary or image and runs in separate pods (in containerized environments like Kubernetes).

This independence means that if one pod or service encounters an issue, you only need to investigate and fix that specific service. There’s no need to check the entire application, which significantly simplifies debugging and reduces downtime. Microservices also enhance scalability and flexibility, as individual services can be updated, scaled, or replaced without affecting the rest of the application.