

In this article, author discussed about web application's architecture and he talked about two applications which are monolithic and microservices architectures. The monolithic application is built as a single and indivisible unit where developers can easily develop, deploy and testing of the application. There also have several weaknesses of monolithic application for example, difficult to understand when it scaled up, quite difficult to implement changes and the whole system is depended to each other. So that, it is quite impossible to apply new technology. On the other hands, microservices architecture breaks the application into a collection of smaller independent units which means the entire application split up into several independent components. The smaller components also easier to understand and it's also manageable. The benefit of the independent component that, it's given the better scalability, flexibility to choose new technology and higher level of agility. Using Application Programming Interfaces (API), whole application is communicated to each other. There also have some disadvantages of the microservices, which are testing complexity, multiple components and database connection and cross-cutting concern. But recently many giant tech companies are moved on microservices application system because of the scalability and independency of the application. Then author mentioned, when you have small team or your application are not big or you want to launch an application within a very short time, then you should go for the monolithic architecture. It also quite difficult to established a microservices when you don't have microservices expertise and you also need enough engineering skill to developed the microservices architecture.

In the conclusion, we can say that, monolithic application is less complex. If you have small plan or initial stage of plan then you go for the monolithic application. But if you want to create more complicated software where it's had thousands of transactions at a time, then you may go for the microservices architecture because it covered many user journeys and workflows.