**Microservices Assignment**

**1) What is microservices?**

Microservices - also known as the microservice architecture - is an architectural style that structures an application as a collection of services that are

* Highly maintainable and testable
* Loosely coupled
* Independently deployable
* Organized around business capabilities
* Owned by a small team

The microservice architecture enables the rapid, frequent and reliable delivery of large, complex applications. It also enables an organization to evolve its technology stack.

**2) Challenges with monolithic oriented architecture**

Following are the drawbacks of monolithic oriented architecture:

* This simple approach has a limitation in size and complexity.
* Application is too large and complex to fully understand and made changes fast and correctly.
* The size of the application can slow down the start-up time.
* You must redeploy the entire application on each update.
* Impact of a change is usually not very well understood which leads to do extensive manual testing.
* Continuous deployment is difficult.
* Monolithic applications can also be difficult to scale when different modules have conflicting resource requirements.
* Another problem with monolithic applications is reliability. Bug in any module (e.g. memory leak) can potentially bring down the entire process. Moreover, since all instances of the application are identical, that bug will impact the availability of the entire application.
* Monolithic applications have a barrier to adopting new technologies. Since changes in frameworks or languages will affect an entire application it is extremely expensive in both time and cost.

**3) Any three advantage and disadvantage of microservices**

Advantages of microservices:

* It tackles the problem of complexity by decomposing application into a set of manageable services which are much faster to develop, and much easier to understand and maintain.
* It tackles the problem of complexity by decomposing application into a set of manageable services which are much faster to develop, and much easier to understand and maintain.
* It tackles the problem of complexity by decomposing application into a set of manageable services which are much faster to develop, and much easier to understand and maintain.

Disadvantages of microservices:

* Microservices architecture adding a complexity to the project just by the fact that a microservices application is a distributed system. You need to choose and implement an inter-process communication mechanism based on either messaging or RPC and write code to handle partial failure and take into account other fallacies of distributed computing.
* Microservices has the partitioned database architecture. Business transactions that update multiple business entities in a microservices-based application need to update multiple databases owned by different services. Using distributed transactions is usually not an option and you end up having to use an eventual consistency based approach, which is more challenging for developers.
* Testing a microservices application is also much more complex than in case of monolithic web application. For a similar test for a service you would need to launch that service and any services that it depends upon (or at least configure stubs for those services).