**Current Situation**

Microservices is still a new concept, although some companies have already migrated to the Microservices architecture. There are many companies facing many issues that could be solved by this new architecture but still are hesitant in moving forward and migrating their application to the Microservices architecture.

This hesitation doesn’t come from one or two reasons but the idea of refactoring an existing application into a new one built using another architecture is not easy and brings with it many challenges that are still waiting to be addressed. The process of migration itself is still not clear.

How can an application be divided into smaller parts? What norms and standards should be followed when making a decision about such division? How big can be any microservice? How can Microservices interact with each other and exchange data? How can those microservices be merged together? What if the Frontend is becoming also complex and needs to be divided into Micro Frontends? How can Micro Frontends and microservices work together? How can microfrontends work together and send and receive data? How should a microfrontends interact with a microservice? How can microservices trust each other? What if the developers had to use ready-made microservices solutions? How can each microservice trust the behavior of a ready-made microservice? How can sensitive data such as credentials exchanged securely among those small parts of the application? And many more.

Questions are many and numerous when it comes to migrating an existing application into a Microservices architecture, so many questions are waiting to be answered. The architecture itself is new and a great deal of research is needed to help those who want to move their application into a Microservice architecture or even if developers want to build their applications from scratch based on the Microservices architecture.

There are still no standard definition of Microservice architecture and there’s no clear guideline of how an application based on Microservices should be built. Over the last few years some characteristics for a microservices-based application have been developed and some general basic outlines are now commonly used.

* A Microservices-based application should consist of more than one component: Unlike monolithic-based applications, a system built using Microservices architecture should be composed of multiple components, each component is self-contained. This way the application can be changed, updated and modified whenever is needed since each change will be applied to only the concerned component itself and not the entire application.
* Simple Routing: Components in a Microservice-based application will have a simple workflow, they will take an input process it and then forwards the result. But one should not forget that there are complications and challenges regarding interaction and securities between microservices.
* Decentralization: An application based on Microservice architecture is built out of many different components and each has its role, but the application in total is not a one unit and there’s no on big unit moderating the communications between the microservices.
* Different technology stack: The development cycle of a Microservices-based application involves having different teams working on different microservices. Since each microservice is a self-contained component, each team can then choose the most suitable development technologies and tools that are most suitable for their own microservice.