**WHAT IS MICROSERVICES AND WHY IT IS IMPORTANT IN TODAY’S IT WORLD?**

In today’s technological world, IT has become a crucial part for any business to rapidly grow and flourish. The more you are connected via networks, the more is the tendency of a business to grow. This is because for any business, if it has the larger audience to attract and most importantly the target audience, it could reach to the list of the top notch companies.

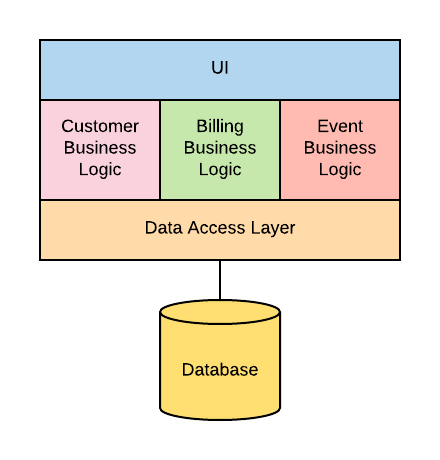
Companies that run businesses either via selling products or providing services or the ones that render services all they need is to develop their own computerized application systems where they can set up their databases, communicate with their departments via networks and provide platforms for the customers to do businesses in order to generate revenues.

**Here the importance of Microservices arises!** For any computerized application or software system, it is necessary to understand what kind of architecture it must encompass in order to fulfill its requirements. Here the architecture refers to the fundamental structure or block of a software system, how it manages its sub-systems? And how its sub-systems relate each other. In a nutshell, architecture acts a blueprint for the whole software system just like in the case of construction of a building where you first need the designs for laying out the necessary tasks.

But before going into the details of Microservices, we must first go through monolithic architectures which were the first developed in the early software development.

**Monolithic Architecture- A Predecessor to Microservices**

A monolithic architecture is a single application where all the functions are put together in a single unified package (tightly coupled) which serves as a one place to store everything. Let’s have a look at the following business model to understand this architecture:



As shown in the above image, all the components of the system resides in one single place including the User Interface layer, the business logic layer and the data access layer. Building applications in a monolith is an easy and natural process, and most projects start this way. But adding functionality to the codebase causes an increase in both the size and complexity of the monolith, and allowing a monolith to grow large comes with disadvantages over time.