Achieving High Availability - Fault Tolerance

In this lesson, we will learn about fault tolerance & designing a HA fault tolerant service.

There are several approaches to achieve HA. The most important of them is to make the system fault-tolerant.

What is Fault Tolerance?

> *Fault tolerance* is the ability of the system to stay up despite taking hits.

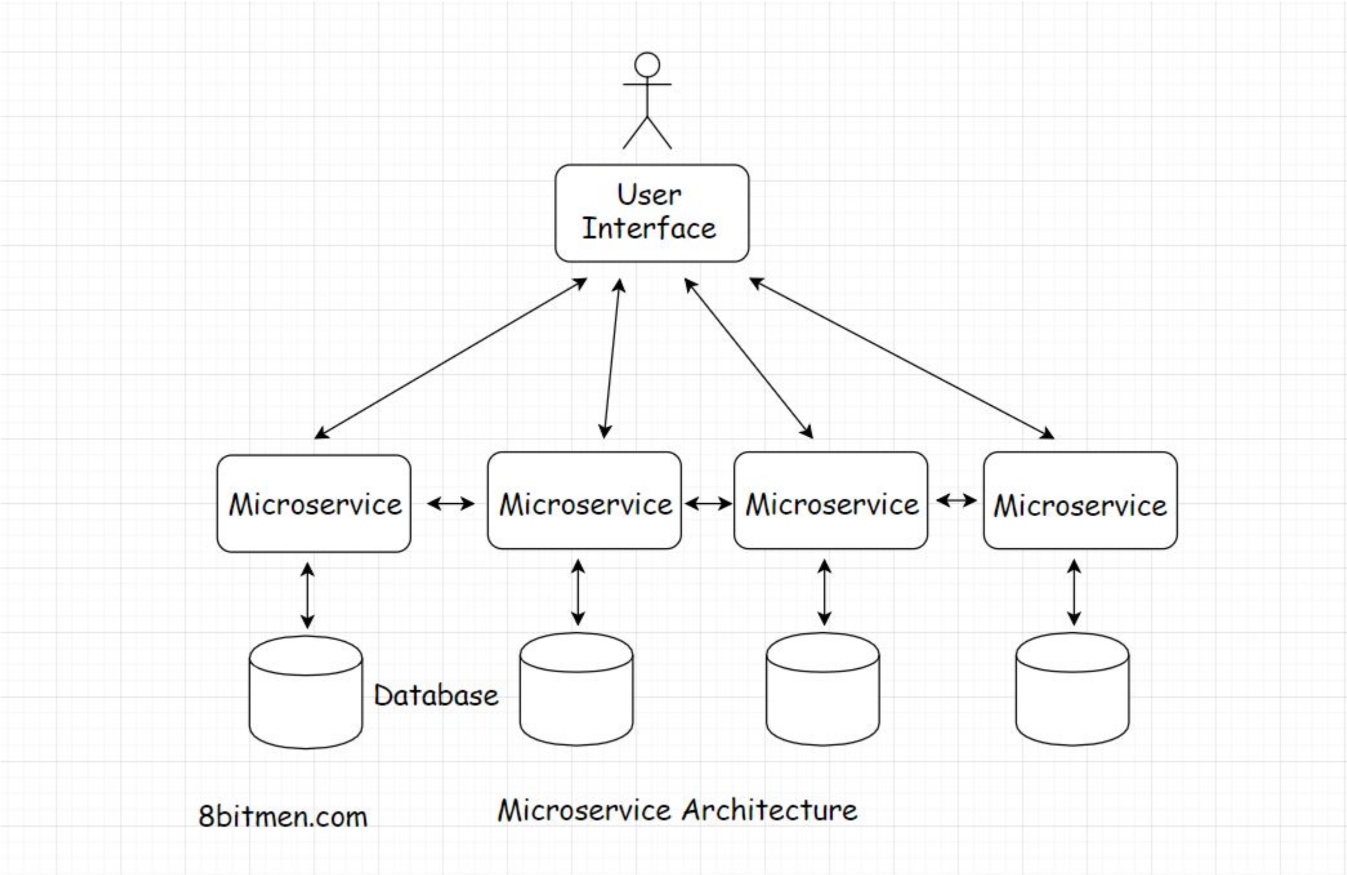
A fault-tolerant system is equipped to handle faults. Being fault-tolerant is an essential element in designing life-critical systems.

A few of the *instances/nodes*, out of several, running the service go offline & bounce back all the time. In case of these internal failures, the system could work at a reduced level but it will not go down entirely.

A very basic example of a system being fault-tolerant is a social networking application. In the case of backend node failures, a few services of the app such as image upload, post likes etc. may stop working. But the application as a whole will still be up. This approach is also technically known as *Fail Soft*.

Designing A Highly Available Fault-Tolerant Service – Architecture

To achieve high availability at the application level, the entire massive service is architecturally broken down into smaller loosely coupled services called the **micro-services**.



There are many upsides of splitting a big monolith into several microservices, as it provides:

- Easier management
- Easier development
- Ease of adding new features
- Ease of maintenance
- High availability

Every microservice takes the onus of running different features of an application such as image upload, comment, instant messaging etc.

So, even if a few services go down the application as a whole is still up.