



# Achieving High Availability - Fault Tolerance

In this lesson, you will learn about fault tolerance and designing a HA fault tolerant service.

## We'll cover the following



- What is fault tolerance?
- Designing a highly available fault-tolerant service – Architecture

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 a system's ability 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.

Out of several *instances/node*, running the service, a few go offline and bounce back all the time. In case of these internal failures, the system can work at a reduced level without, going down entirely.

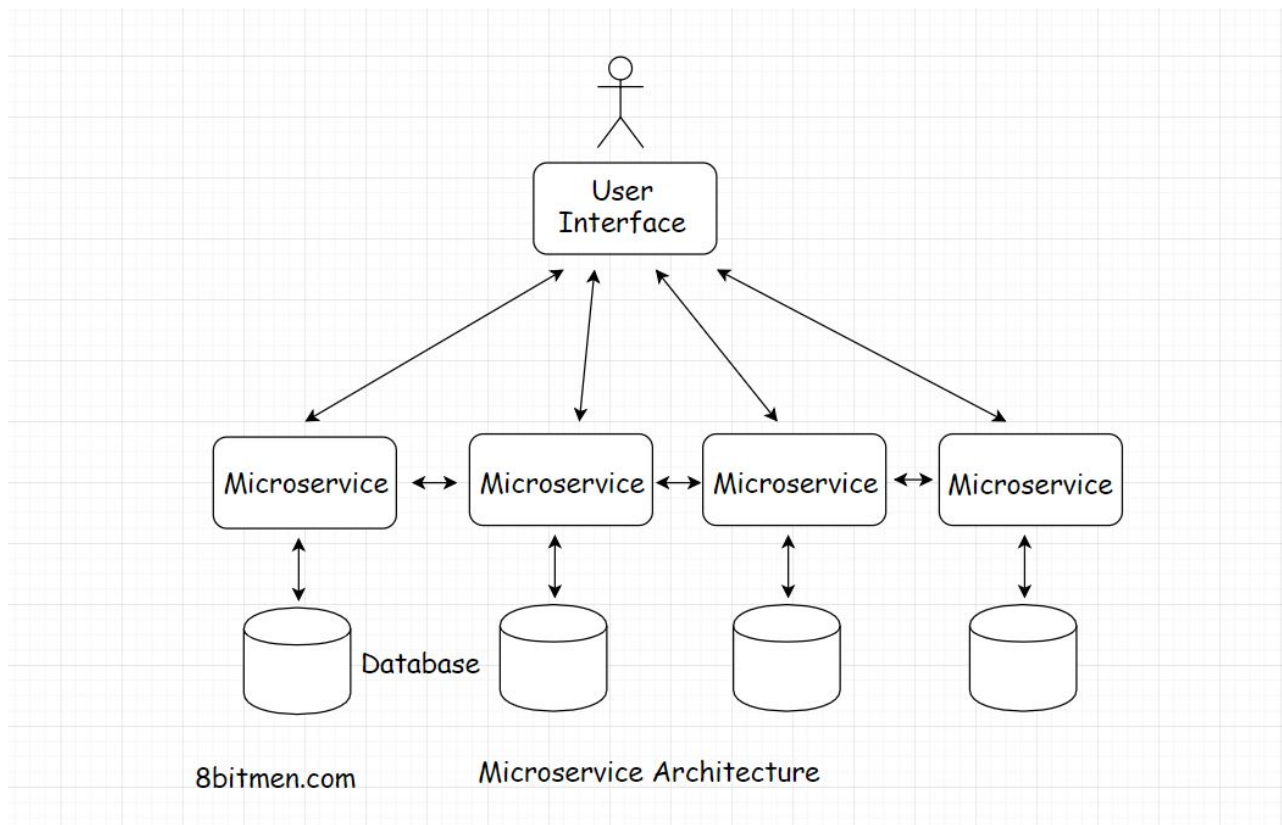
A very basic example of a system that is fault-tolerant is a social networking application. In the case of backend node failures, a few services of the app such as image upload, post likes etc. may stop working.

However, 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 **micro services**.



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

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

- High availability



Every micro service 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.

← Back

Reasons For System Failures

Next →

Redundancy

☒ Mark as Completed



Report an Issue