TEAM ID: PNT2022TMID07083

9.1 Performance Metrices:

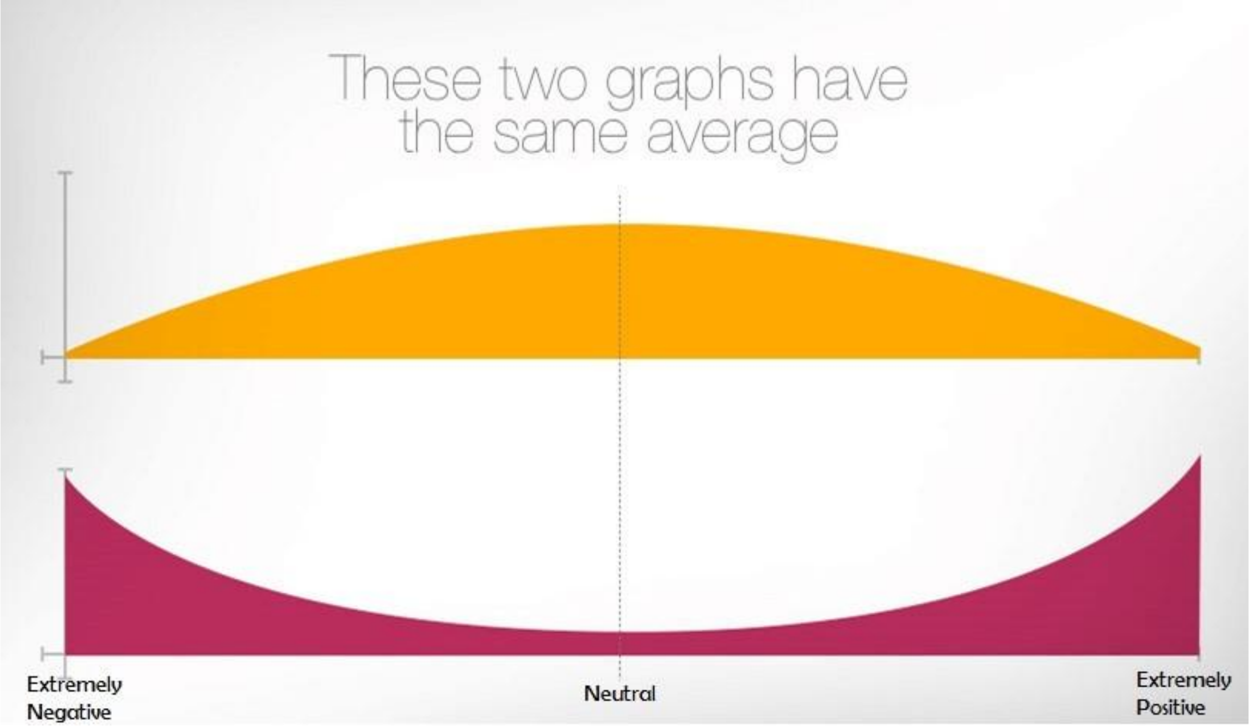
1. User Satisfaction Scores

The application performance index, score, has become an industry standard for tracking the relative performance of an application.

It works by specifying a goal for how long a specific web request or transaction should take.

2. Average Response Time

Let me start by saying that **averages suck**. I highly recommend using the aforementioned user satisfaction scores as a preferred way to track overall performance. That said, averages are still a useful application performance metric.



3. Error Rates

The last thing you want your users to see are errors. Monitoring error rates is a critical application performance metric.

There are potentially 3 different ways to track application errors:

1. HTTP Error % – Number of web requests that ended in an error
2. Logged Exceptions – Number of unhandled and logged errors from your application

4. Count of Application Instances

If your application scales up and down in the cloud, it is important to know how many server/application instances you have running. Auto-scaling can help ensure your application scales to meet demand and saves you money during off-peak times. This also creates some unique monitoring challenges.

5. Request Rate

Understanding how much traffic your application receives will impact the success of your application. Potentially all other application performance metrics are affected by increases or decreases in traffic.

6. Application & Server CPU

If the CPU usage on your server is extremely high, you can guarantee you will have application performance problems. Monitoring the CPU usage of your server and applications is a basic and critical metric.