

# 3 CRITICAL END USER EXPERIENCE METRICS FOR APPLICATION PERFORMANCE



The popularity of web-based applications increases every year. And the percentage of companies that run most of their processes through software as a service (SaaS) products [is predicted to increase](#) from 38% in 2016 to 78% in 2022. Web-based applications are therefore fueling the growth in cloud services, with [SaaS responsible for](#) almost 70% of overall public cloud market share.

This trend shows an unstoppable shift in how organizations and individuals use applications every day. From consumer apps on smart devices to enterprise SaaS that powers a global workforce, web-based applications are quickly becoming the default way we work, communicate, and interact.

Employees and consumers want to make the most of these applications. Correct development and implementation of SaaS applications will maximize productivity, minimize operational costs, create a competitive advantage, and make it easier to scale.

Measurement, analysis, and improvement is critical to how well web-based applications function, and has a direct impact on user satisfaction, interaction, and output. Users want SaaS products that are reliable, fast, and easy-to-use. Translate each of these into a metric you can measure, track, and improve and you have three critical end user experience metrics: **availability**, **responsiveness**, and **usability**.

# 1. Availability—App Is Available When Users Need It

The most basic requirement of all is the app is available and working when employees or consumers need to use it. The availability of an app relies on several different areas working together correctly to provide a reliable end product.

- **Map the IT Elements That Deliver a Working App**

Map and audit all of the various areas that an app relies on. This includes local and cloud-based infrastructure, backend, middleware, and frontend.

- **Understand Interdependencies Between IT Elements**

An app relies on many interdependent systems. Understand the links between these systems and how failure in one area can create availability issues elsewhere.

- **Use Availability Incidents to Resolve Underlying Problems and Risks**

Identify and resolve system failures, then use a robust problem management approach. This will identify the root cause and resolve underlying issues to prevent incidents from recurring.

- **Create Robust Resilience, Recovery, and Failover Options**

Create fast, comprehensive plans for restoring availability quickly if you have an outage. For especially critical systems, implement complete failover systems for continual availability.

# 2. Responsiveness—App Is Fast and Reactive Across All Devices

Next to availability, the speed of your app is just as critical. Apps that are slow to react will impact productivity and create end user frustration.

- **Measure App Responsiveness Across Multiple Use Cases and Devices**

Various devices, use cases, and scenarios can impact on response times. Test your app across all devices and configurations to ensure a fast and satisfying end user experience.

- **Align Response Times with Demand and Capacity Management**

Demand and [capacity management](#) are two crucial ITIL disciplines for maintaining good response times in apps. Get an excellent understanding of likely future demands on web-based apps and have proactive capacity management in place.

- **Monitor Response Times Throughout the IT Ecosystem**

Monitor speed and responsiveness across all systems and interdependencies, similar to how you measure availability. Expand responsiveness monitoring across your wider IT networks, as events throughout your IT ecosystem can impact your app's responsiveness.

# 3. Usability—App Is Simple and Easy to Use

Good apps are simply a pleasure to use. The design, interface, navigation, and features of an app must come together to provide an outstanding user experience.

- **Learn How Users Navigate and Interact with Your App**

Set up monitoring throughout your app. Explore how long users spend in certain parts of the product before they move on. Analyze the entry and exit points in the app and get insight into how users engage with the product.

- **Ask Users What They Think of the App**

Use technical insight into how users interact with the app to guide questions and feedback. Set up focus groups with different types of users. Ask these users what they do and don't like

about the app, and the improvements they would like to see.

- **Baseline, Improve, and Re-measure**

Establish a strong baseline for how users interact with the product. Make changes to enhance the app based on metrics and user feedback, then re-measure to see if engagement improves. Carry out split-testing in the app to learn how effectively the product meets user needs.