

Application Monitoring tips for IT pros

Doctors are losing 1 hour per day with EHR information overload, and it got me thinking. How is IT implementing these EHR applications and how are they discussing requirements with their users?

Regardless of your industry or vertical, your applications should ensure end-user satisfaction and boost productivity for employees and partners. Therefore, IT pros implementing or monitoring applications should take the time to understand how end-users interact with their application, share the proper amount of information with the right stakeholders, implement the right workflows and ensure they are performing top-notch. Here are some quick tips to help you get started.

Do #1: Prioritize which applications should be monitored first. With an increasing number of employees bypassing IT and going rogue to the cloud, it is the wild, wild, west out there. Plus counting legacy applications, Citrix and Terminal server hosted apps, CRM, EHR, custom-built applications, accounting, invoicing, HR, email and collaboration tools, the list of applications your employees, partners or customers rely on is long! Your applications fuel your business, so they must consistently perform well, and ultra-fast. Since you have to start somewhere, identify those critical applications that must perform well in order to run your business (e.g. applications migrated to the cloud, CRM, ERP, EHR systems), and monitor them first. You know better than anybody else what is critical to your business and users.

Do #2: Identify critical transactions to monitor. Put on your “think from an end-user perspective hat” and map out common functions used by your power users (e.g. those using your applications the most, those driving the most revenue, key function, etc.). Or better yet, schedule a meeting with your business counterparts and stakeholders to identify critical functionality from their perspective. If you recently went through the process of implementing a new app, you should have your workflows already mapped, right?

For example, with Epic EMR, both patients & doctors access MyChart records, so you should monitor response time from both patient and doctor perspectives, as shown below.

- **Doctor**– MyChart log-in → Access a patient record → Add visit log → Save → Logout
- **Patient**– MyChart log-in → Access my record → Search for recent test results or next appointment date → Logout

Similarly, thinking about a CRM system and how your users interact with it

- **Sales Rep**- CRM log-in → Open a customer record → Add a sales log → Log-out
- **Sales Manager**- CRM log-in → Access sales pipeline → Generate a report → Log-out

As you document critical transaction paths or workflows for your application users, this is a great time to fine tune your processes and minimize the number of steps needed for common functions.

Do #3: Proactively monitor your applications from an end-user perspective. End-users are more impatient than ever before. Therefore, you should continuously monitor each one of these critical transactions from a user perspective, taking response time measurements for each step to ensure user SLAs are met. It is unacceptable that in 35% of cases IT learns that there is an issue when a

user opens a helpdesk ticket or calls (Source: Forrester Research). Change the game and get ahead; find and resolve bottlenecks, errors and constraints, problems *before* your users are impacted.

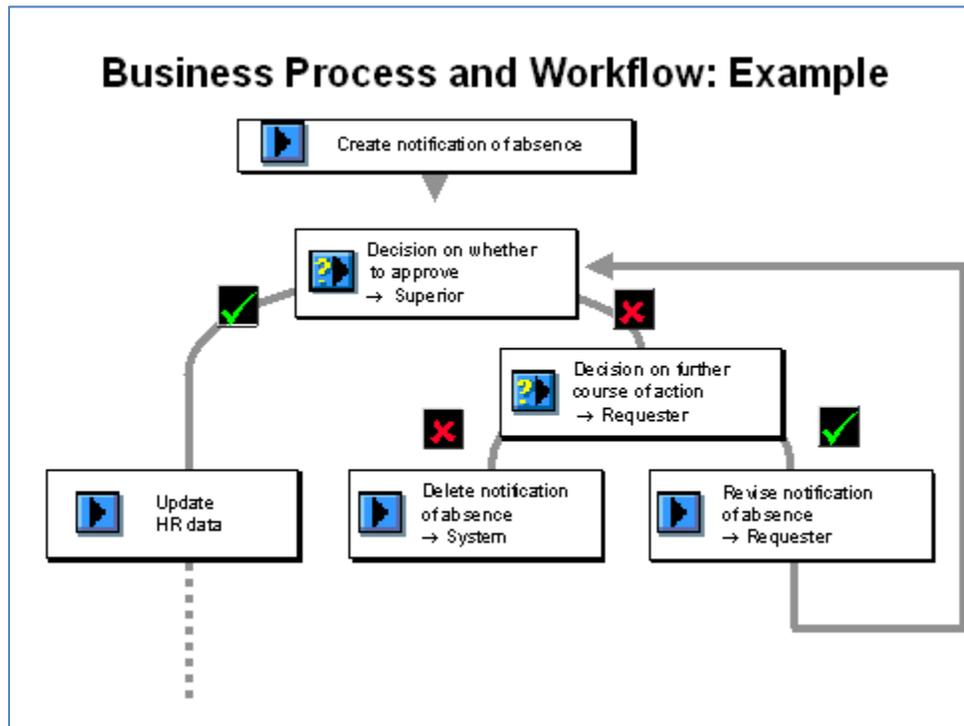


Figure 1: Map common workflows for your users, in this case HR personnel. These transactions should be proactively monitored 24x7x365 to find problems before your end-user do

Do #4: Decide polling frequencies and alerting policies. A good rule of thumb is to monitor key transactions more frequently (e.g. being able to send a sales proposal is more critical than reporting on sales pipeline, or being able to sell online is more important than reading a product review) to identify performance degradation signs earlier. Take the time to define who should be alerted in the event of specific threshold violations, and configure the number of response time violations that will trigger an alert to eliminate false positives and alert storms. Don't forget to look for key monitoring functionality like scheduling monitoring tests or disable alerting on scheduled maintenance periods or when you are on vacation. You should be in control of your monitoring.

Do #5: Identify geographical response time discrepancies early on. Employees at remote offices could experience slower response times than those accessing your applications from headquarters; legacy application could underperform for some offices or branches. Get ahead of user complaints. The faster way to find and resolve problems like this is to monitor and compare availability and response time of your applications across multiple monitoring locations (Headquarters, Boston, NYC, remote office locations, etc.).

Do #6. Define your custom reports. Since different metrics are important for different stakeholders, take the time to map out role-based reports with custom information for each team (per

application, per transaction, per functionality, etc.), and automatically distribute reports on an on-going basis (daily, weekly or monthly basis) to keep everybody informed and aligned.

Do #7: Centralize IT response procedures and workflow. From legacy applications, to client server applications, to web applications, to home-grown custom applications, cloud-based or green screen apps, most large enterprises have a complex portfolio with 250-500 applications to support. The cost of purchasing, configuring and maintaining several monitoring products to support individual applications is too high. Plus lack of integration across monitoring consoles results in islands of uncorrelated information which leads to wrong conclusions, hinders troubleshooting and increases Mean-Time-To-Resolution (MTTR). Instead, look for one solution that lets you test and monitor all applications, so you can quickly identify problem root cause.

Do #8: Keep everybody in the loop. In a new era where customer satisfaction rules, you need to continuously validate and demonstrate your SLAs, so go ahead and periodically share your SLA reports with your users and stakeholders. Provide a quick summary dashboard with a drill-in so that they don't have to peruse voluminous reports. Plus since user satisfaction is the ultimate measurement of IT success (your success), this is the best metric to promote the value that IT provides to your organization.

Do #9: Review results on an on-going basis. Do you need to fine-tune? Do you need to optimize application performance? With a metric-driven strategy in place you can keep all stakeholders in the know, and take informed business decisions that directly impact your bottom line (e.g. quickly ascertain if you need to focus on performance optimization or not, change cloud providers, etc.).

Do #10: Ensure quality. Build a culture where application quality is not an afterthought. You should include testing (functional testing, regression testing, performance testing, load testing) in all application development/application implementation cycles right from the beginning to ensure quality. Being able to reuse your testing scripts for production monitoring will also help streamline processes.

In summary, your end-users have the ultimate say on whether they are satisfied with the speed, availability and performance of your applications, so implement and monitor your applications from your end-users' perspective.

<http://tevron.com/application-performance-monitoring-citratetest-apm.aspx>