

End-to-End Testing und Monitoring mit Sakuli

Automatisierte Applikationstests durch die Anwenderbrille



Das Produkt im Überblick

- UI-Tests für Web-, native und betriebs-systemspezifische Benutzeroberflächen
- End-to-End Monitoring der Anwendungsschicht – zuverlässige Überwachung Ihrer Applikationen aus Nutzersicht
- Lückenlose Kontrolle durch Prozess- und Workflow-Automatisierung
- Die Enterprise-Subscription beinhaltet vorkonfigurierte Pakete, zusätzliche Features, Support und Wartung



Ihr Nutzen

- Universelles Testwerkzeug für jeden Use Case und jede Plattform
- Nahtlose Übermittlung der Testergebnisse an Drittsysteme wie Nagios, Icinga, DBs & Co.
- Open Core: Der Kern von Sakuli steht unter Open Source und spart damit Kosten
- Professioneller Support und Wartung in der Enterprise Version
- Stabile, zukunftssichere Plattform für Testautomatisierung dank offener Architektur und dem Einsatz von State-of-the-Art-Technologien wie Node.js

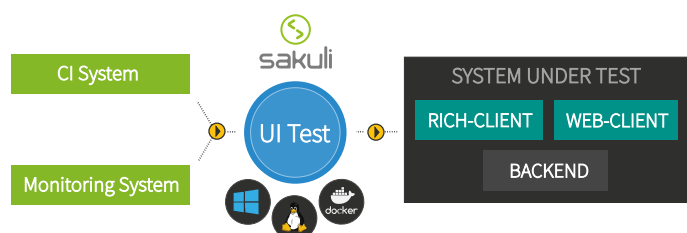
sakuli.io

Störungen und Ausfälle in IT-Anwendungen kosten Zeit und Geld – ob bei Login-Vorgängen des Webshops oder Remote-Zugriffen in der Niederlassung. Mit Sakuli testen Sie die Funktionen von Benutzeroberflächen umfassend und voll automatisiert. Das Framework kombiniert web- und bildbasierte Oberflächentests und agiert damit auch außerhalb von HTML und dem Document Object Model (DOM): Sakuli simuliert Benutzeraktionen auf grafischen und webbasierten Oberflächen, validiert die Inhalte und misst die Ausführungszeiten.











Sakuli ist die Allround-Lösung

Über eingebaute Schnittstellen können die Ergebnisse von Sakuli-Tests direkt an Monitoring-Systeme, u.a. Nagios, OMD, Icinga2 oder check_mk übermittelt werden. Diese verarbeiten die Laufzeitwerte, visualisieren sie in Graphen und alarmieren bei Performanceeinbußen oder Störungen. Durch die Kombination von E2E-Testing

und -Monitoring können Sie eingreifen, noch bevor der Benutzer funktionelle oder inhaltliche Störungen bemerkt. Zudem lassen sich Sakuli-Tests auch nahtlos in Continuous Integration-Umgebungen wie Jenkins integrieren.



Core Features (Open Source)

 Screenshot-basierte Interaktion	 DOM-basierte Interaktion	 Unterstützung der Bildschirmgrößen	 Drag and Drop	 Native Steuerung
 Verwendung der Zwischenablage	 Sicheres Passwort-Handling	 Vereinfachung von Selenium	 Node debugging	 Keine Stalelements-Fehler

Enterprise Features

Monitoring Forwarder



OMD

OMD (Gearman)
ab Version 3



Icinga 2

Icinga2 mit konfigurierter API



Check_MK

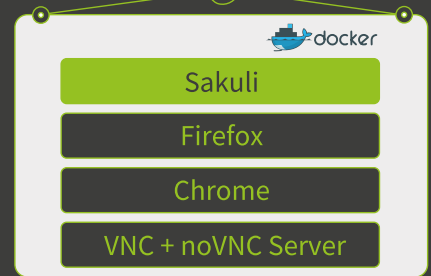
ab Version 1.5



Databases

Durch TypeORM unterstützte SQL-Datenbanken

Pre-configured Docker-Image



Sakuli in der Praxis: Einsatzbeispiele

Unternehmensweite Performance von Enterprise-Applikationen

Um die Reaktionszeiten z.B. von Citrix-Applikationen an allen Standorten zu messen, loggt sich Sakuli kontinuierlich dort ein und führt Nutzeraktionen durch. Überschreiten Checks oder deren Teilschritte die vorgegebenen Laufzeiten, so schlägt das Monitoring-System Alarm.

Funktionsgarantie im Conversion-kritischen Online-Shop

Sakuli loggt sich als Test-User im Shop ein und prüft den Bestellvorgang von der Artikelsuche über den Warenkorb bis zur Bezahlung. Probleme der Infrastruktur, aber auch von Drittsystemen, können frühzeitig erkannt und teure Ausfälle vermieden werden.

Einwandfreie SaaS-Dienstleistungen

Sakuli prüft die Verfügbarkeit und Funktion von Services externer Anbieter, bei denen es an Monitoring-Schnittstellen mangelt. Die Qualität der SaaS-Dienstleistungen wird damit messbar.