



Testmanagement und RPA: Chancen und Herausforderungen der Automatisierung

Veröffentlicht am 18. Juli 2024

Ihre Ansprechpersonen: Vithurjan Vijayakumar, Jorge Parra

Tags: Fachbeiträge, Banken und Versicherungen, Energie und Infrastruktur, Gesundheitswesen, Industrie, Technologie und Dienstleistung, Handel, Transport und Logistik, Öffentliche Verwaltung, Digitalisierung in KMU, Innovation

In der heutigen schnelllebigen Welt der Softwareentwicklung ist die Sicherstellung der Softwarequalität entscheidend für den Projekterfolg. Testmanagement ist dabei ein zentraler Bestandteil und trägt massgeblich zur Qualitätssicherung bei.

Durch systematisches Testen können Fehler frühzeitig erkannt, Anforderungen erfüllt, Risiken minimiert und Kosten gesenkt werden. Ein gut durchgeführter Testprozess führt zu stabiler, sicherer und leistungsfähiger Software, die den Bedürfnissen der Benutzerinnen und Benutzer gerecht wird und die Reputation des Unternehmens stärkt.

Testprozess

Der Testprozess umfasst mehrere Aktivitäten. Obwohl diese Aktivitäten oft in einer sequenziellen Reihenfolge dargestellt werden und logisch aufeinander aufbauen (zum Beispiel bei sequenziellen Entwicklungsmodellen wie das Wasserfallmodell), können sie in der Praxis überlappen und parallel durchgeführt werden.

In agilen Entwicklungsansätzen werden Software und Tests in kleinen Iterationen erstellt und kontinuierlich durchgeführt. Innerhalb dieses Ansatzes finden Testaktivitäten iterativ und kontinuierlich statt, um die Qualität während des gesamten Entwicklungsprozesses sicherzustellen.

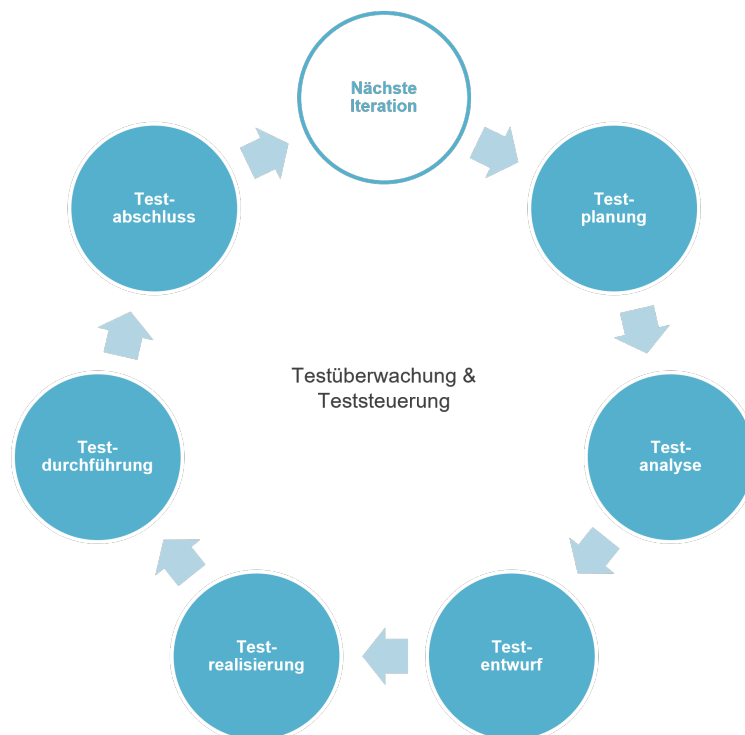


Abbildung: Iterativer Testprozess

Die Aktivitäten werden vom Testmanagement gesteuert und koordiniert, um sicherzustellen, dass die Software den Qualitätsanforderungen entspricht und erfolgreich bereitgestellt werden kann. Im Folgenden werden die einzelnen Aktivitäten des Testprozesses beschrieben.

Teststeuerung und Testüberwachung

Die Teststeuerung und Testüberwachung umfasst die fortlaufende Beobachtung des Fortschritts der Testaktivitäten. Es ist sicherzustellen, dass die Tests wie geplant durchgeführt werden und bei Bedarf die Pläne angepasst werden, um auf unerwartete Änderungen oder Probleme zu reagieren.

Testplanung

Bei der Testplanung wird die Teststrategie definiert. Auf Grundlage deren wird ein Testkonzept erstellt. Es legt die Testziele fest und beschreibt die Vorgehensweise, wie und wann die Tests durchgeführt werden sollen. Ebenfalls werden die benötigten Ressourcen, die Zeitpläne sowie die potenziellen Risiken festgehalten.

Testanalyse

Bei der Testanalyse geht es darum, zu ermitteln, was genau zu testen ist. In dieser Phase werden die Anforderungen analysiert und spezifische Testfälle sowie Testdaten identifiziert. Ziel ist es, sicherzustellen, dass alle Anforderungen durch entsprechende Tests abgedeckt sind. Dabei werden sowohl funktionale als



auch nicht-funktionale Anforderungen berücksichtigt.

Testentwurf

Beim Testentwurf geht es darum, festzulegen, wie getestet wird. Basierend auf den in der Testanalyse identifizierten Anforderungen und Kriterien werden spezifische Testfälle ausgearbeitet, die alle relevanten Aspekte der Software abdecken.

Diese Testfälle umfassen sowohl funktionale als auch nicht-funktionale Tests. Abhängig von den Anforderungen und den zu prüfenden Qualitätsmerkmalen werden unterschiedliche Testarten angewendet.

Welche Testarten es gibt, haben wir bereits in einem [Fachbeitrag](#) anhand eines Praxisbeispiels beschrieben.

Testrealisierung

Die Testrealisierung beinhaltet die Vorbereitung der Testumgebung. Dies schliesst die Einrichtung der Hardware, Software und Netzwerke ein. Zudem werden Testfälle konkretisiert. Testfälle werden in eine sinnvolle Reihenfolge gebracht und zu Testsuiten gruppiert. Basierend den entworfenen Testfällen können automatisierte Testskripte implementiert werden.

Testdurchführung

In dieser Phase werden die geplanten Tests durchgeführt. Die Testergebnisse aus den durchgeführten Testfällen bzw. den vorbereiteten Testskripten werden exakt und vollständig protokolliert. Gefundene Fehler werden dokumentiert und verfolgt, bis sie behoben sind.

Testabschluss

In der letzten Aktivität des Testprozesses wird ein Testabschlussbericht erstellt, welcher alle Testaktivitäten und -ergebnisse zusammenfasst und eine abschliessende Bewertung der durchgeführten Tests enthält. Zudem werden die Lessons Learned erfasst, um zukünftige Projekte zu verbessern.

Robotic Process Automation (RPA) als Testwerkzeug

Um eine oder mehrere der oben genannten Aktivitäten im Testprozess zu unterstützen, werden Testwerkzeuge eingesetzt. Wenn von Testwerkzeugen die Rede ist, sind oft im engeren Sinne Werkzeuge



zur Automatisierung von dynamischen Tests gemeint.

Robotic Process Automation (RPA) wird zunehmend als Testwerkzeug in der Softwareentwicklung eingesetzt, insbesondere für wiederholbare und regelbasierte Aufgaben im Testprozess. Hier sind einige spezifische Bereiche, in denen RPA im Testprozess eingesetzt werden kann:

Testanalyse

RPA-Bots können Anforderungsdokumente und User Stories analysieren, um automatisch Testfälle und Testszenarien zu erstellen. Zum Beispiel kann ein RPA-Bot eine Anforderung «Benutzer muss sich mit einer E-Mail-Adresse registrieren können» erkennen und daraufhin Testfälle wie «Registrierung mit gültiger E-Mail», «Registrierung mit ungültiger E-Mail» und «Registrierung ohne E-Mail» generieren.

Testentwurf

RPA kann genutzt werden, um Testdaten aus verschiedenen Quellen zu sammeln und diese für die Tests vorzubereiten. Beispielsweise kann ein RPA-Bot Benutzerdaten aus einer Produktionsdatenbank anonymisieren und für Testzwecke bereitstellen.

Testrealisierung

RPA-Bots können genutzt werden, um virtuelle Maschinen zu starten, Betriebssysteme und notwendige Software zu installieren und Netzwerkeinstellungen zu konfigurieren. Dies stellt sicher, dass jede Testumgebung identisch und einsatzbereit ist.

Testdurchführung

RPA ist besonders für die Durchführung von Regressionstests nützlich. Bei Regressionstests müssen dieselben Tests wiederholt ausgeführt werden, um sicherzustellen, dass neue Änderungen (z. B. bei einem neuen Software-Release) keine neuen Fehler eingeführt wurden.

Sobald ein Testfehler auftritt, kann ein RPA-Bot die Details des Fehlers erfassen, einen Fehlerbericht erstellen und diesen automatisch an das zuständige Entwicklerteam senden. RPA-Bots können den Status von gemeldeten Fehlern in einem Bug-Tracking-System überwachen und sicherstellen, dass die Fehler bis zur Behebung verfolgt werden.

Nutzen und Herausforderungen beim Einsatz von RPA



Diese Beispiele zeigen, dass RPA eine entscheidende Rolle im Testmanagement spielt. Zu den wesentlichen Nutzen zählt die **Effizienzsteigerung**. RPA-Bots können wiederholbare und zeitaufwendige Testaufgaben schnell und zuverlässig ausführen, wodurch die Effizienz erhöht wird. Die Testteams können sich dadurch auf strategisch wichtigere und komplexere Aufgaben konzentrieren.

Automatisierte Testprozesse ermöglichen es, Software schneller und mit höherer Qualität auf den Markt zu bringen (**Schnellere Markteinführung**).

Ein weiterer Vorteil sind die **Kosteneinsparungen**. Durch die Reduktion manueller Arbeit werden die Personalkosten verringert. Ausserdem sind automatisierte Tests weniger anfällig für menschliche Fehler, was zu weniger Nacharbeit und geringeren Kosten führt.

Die **Konsistenz und Zuverlässigkeit** der Testergebnisse wird ebenfalls erhöht, da RPA-Bots Tests immer auf die gleiche Art und Weise durchführen.

Trotz dieser Vorteile gibt es auch Herausforderungen beim Einsatz von RPA im Testprozess. Die **initialen Implementierungskosten und -zeit** sind hoch, da die Einrichtung und Konfiguration der RPA-Bots eine erhebliche Anfangsinvestition erfordert und zeitaufwendig sein kann.

Kontinuierliche Wartung und Aktualisierung der RPA-Skripte sind notwendig, um mit Änderungen in der Software und den Testanforderungen Schritt zu halten.

Zudem bieten RPA-Bots **begrenzte Flexibilität** und können Schwierigkeiten haben, komplexe oder unvorhergesehene Szenarien zu bewältigen. Des Weiteren bestehen **Sicherheitsrisiken und Datenschutzprobleme**, da die Automatisierung von Tests und der Zugriff auf Testdaten durch RPA-Bots besondere Vorsichtsmassnahmen erfordert.

Fazit: Automatisierung um jeden Preis?

Der Nutzen von Testwerkzeugen bzw. der Automatisierung des Testprozesses hängt stark vom jeweiligen Projektumfeld und der Reife der Entwicklungs- und Testprozesse ab. Je kürzer die Iterationszyklen gewählt werden, umso wichtiger wird die Automatisierung der Tests.

Denn mit wachsender Produktfunktionalität müssen immer mehr Testfälle abgearbeitet werden, ohne dass das kurze Iterationsintervall überschritten wird. Testautomatisierung ist ein entscheidendes Mittel, um diesem Konflikt zu begegnen.



Hingegen ist in einem chaotischen Projektumfeld, in dem auf Ad-hoc-Basis programmiert wird, Dokumentation fehlt und Tests unstrukturiert oder gar nicht durchgeführt werden, die Automatisierung der Testdurchführung wenig sinnvoll.

Ein Werkzeug kann niemals einen fehlenden Prozess ersetzen oder eine nachlässige Arbeitsweise kompensieren. So erfordert die Implementierung von RPA als Testwerkzeug eine sorgfältige Planung, kontinuierliche Wartung und die Bereitschaft, technische und organisatorische Herausforderungen zu bewältigen.

Unternehmen sollten eine ausgewogene Strategie verfolgen, um die Vorteile von RPA zu maximieren und gleichzeitig die potenziellen Risiken und Herausforderungen zu minimieren.

Quellen

Spillner, A., & Linz, T. (2019). Basiswissen Softwaretest. Bremen: dpunkt.verlag