



Die Top 3 der Fehler bei automatisiertem End-to-End-Testing Zu komplex und aufwendig?

Von Christian Frenk, Objektkultur Software GmbH

End-to-End-Tests gelten oft als schwierig in der Durchführung und zeitaufwendig in der Wartung. Dem gegenüber steht jedoch der Nutzen, den die Testperspektive des Users mit sich bringt. Diese ermöglicht die Behebung von Problemen, die oftmals erst bei Livegang ersichtlich werden. In Wirklichkeit gibt es immer wiederkehrende Fehlerquellen, die eine erfolgreiche Durchführung eines End-to-End-Tests verhindern.

Fangen wir von vorne an: Wie notwendig ist End-to-End-Testing überhaupt?

Haben Sie sich schon einmal gefragt, ob die Bestellungen Ihrer Kunden nach einem Update Ihres Online-Shops noch korrekt an das ERP-System weitergeleitet werden? Oder ob die neue CRM-Konfiguration möglicherweise zu Problemen bei der Registrierung von App-Nutzern führt? Durch die Integration vieler verschiedener Komponenten und Services entstehen immer komplexere Systemlandschaften, die auf ihre Funktionalität hin geprüft werden müssen. Um den reibungslosen Ablauf der Geschäftsprozesse sicherzustellen, reichen Unit- und Integrationstests alleine nicht aus. Diese betrachten nur einzelne Komponenten und deren Interaktion miteinander. Viel interessanter ist die Betrachtung einer gesamten Umgebung. An dieser Stelle setzen End-to-End-Tests an. Diese simulieren in einer Testumgebung realitätsnahe Benutzerabläufe und stellen kritische Prozesse automatisiert sicher. Ein Beispiel: Bei einem klassischen Bestellprozess auf einer E-Commerce-Plattform würde das Absenden einer Bestellung sowie die nachfolgenden Prozessschritte vom Versand der Bestellbestätigung bis hin zur Rechnungsstellung kontrolliert werden.

Die 3 häufigsten Fehler bei der Umsetzung:

1. Zu frühe Entwicklung

End-to-End-Tests benötigen für ihre Implementierung einen fortgeschrittenen Entwicklungsstand und eine integrierte Systemlandschaft. Werden sie in einem zu frühen Stadium eingeführt, in dem noch viele Änderungen im Frontend zu erwarten sind, entsteht ein unnötig hoher Wartungsaufwand und es kann zu Verzögerungen bei der eigentlichen Anwendungsentwicklung kommen.

2. Falscher Fokus in der Teststrategie

Der richtige Fokus in der Teststrategie ist für eine erfolgreiche Integration der Tests elementar. Entsprechend der Testpyramide von Mike Cohn sollte die Kategorie der End-to-End-Tests den geringsten Anteil an der Gesamtheit der automatischen Tests in einem Softwareprojekt einnehmen. Daher gilt es, den Fokus auf wenige kritische Geschäftsprozesse zu legen und ihren „Happy Flow“ mit End-to-End-Tests abzudecken.

3. Instabile „flaky“ Tests

Durch die Vielzahl an Abhängigkeiten zwischen unterschiedlichen Systemen gibt es ein hohes Potential für Systemausfälle oder Verzögerungen bei der Verarbeitung der simulierten Benutzerinteraktionen. Ebenso spielt die Komplexität der Anwendung bzw. der Benutzeroberfläche eine Rolle. Interaktive Elemente oder Benachrichtigungen/Popups können die Ausführung des geplanten Testszenarios stören. Diese Situationen führen zu Timeouts und einem ungewünschten Fehlschlagen des End-to-End-Tests und schließlich zu einer unnötigen Fehleranalyse.

So wird die Implementierung erfolgreich:

- ✓ Zur Vermeidung von unnötigen Entwicklungs- und Wartungskosten gilt es, mit dem End-to-End-Testing erst bei einem bestimmten Reifegrad des Systems zu starten.
- ✓ Für eine erfolgreiche Umsetzung von End-to-End-Testing müssen bereits im Vorfeld relevante konzeptionelle Entscheidungen hinsichtlich der Testausführung, Testdaten sowie der gewünschten Testmodalitäten getroffen werden. Der Scope des Tests sollte dabei auf besonders kritischen Geschäftsprozessen und Kernfunktionalitäten der Systeme liegen.
- ✓ Eine Robustheit des Tests sollte gewährleistet werden:
 - / Testbarkeit des Codes bzw. Frontend bereits während der Anwendungsentwicklung berücksichtigen. Dazu gehört die konsequente Vergabe von ID's für testrelevante Schaltflächen und Elemente.
 - / Entwicklung einer Strategie für das Management von Timeouts (z.B. automatische Waits, Retries).
- ✓ Eine nachvollziehbare Auflistung der Benutzeraktionen und Visualisierung der Anwendungen bei Fehlereintritt erleichtern die Fehleridentifikation.

Wenn Sie die Geschwindigkeit Ihrer Produktentwicklung beibehalten und sich auf die kundenorientierte Systemgestaltung fokussieren möchten, bietet sich externe Expertenunterstützung an. So können Sie sich das notwendige Know-how einholen und die Softwarequalität sicherstellen.



Haben Sie Fragen? Dann kontaktieren Sie uns.

Christian Frenk / Consultant / +49 (0) 721 83 04 05 00 / cfr@objektkultur.de / www.objektkultur.de