

Architecture for the Automated Testing of the TV as a Service platform

Studiengang: MAS Information Technology

Das automatisierte Testing ist von grundlegender Bedeutung um die Qualität einer Software Lösung bei häufigen Releases sicher zu stellen. Um die TV as a Service Plattform der Netstream AG in einer höheren Frequenz releasen zu können, soll daher eine Architektur für das automatisierte Testing der Plattform erarbeitet werden. Diese soll den manuellen Test-Aufwand verringern und sowohl das Testing der Backend- wie auch der Frontend-Komponenten beinhalten.

Problemstellung

Die TV as a Service Plattform soll es den Kunden der Netstream AG ermöglichen, ihren Endkunden eine TV-Lösung anzubieten, ohne dass sie die dafür nötige Infrastruktur selber betreiben müssen. Die Plattform umfasst neben der Bereitstellung der Medien-Streams auch kundenspezifische Endkunden-Applikationen für Set-Top-Boxen, Browser und mobile Geräte. Automatisierte Tests für die Plattform wurden bisher nur sehr punktuell umgesetzt und die offiziellen System-Tests werden vom Quality Assurance Team grösstenteils manuell mit den Frontend-Applikationen durchgeführt. Sie verursachen entsprechend viel Aufwand und benötigen sehr viel Zeit. Test-Umgebungen wurden bisher manuell aufgesetzt und stehen darum nur in beschränkter Anzahl zur Verfügung. Dies erlaubt uns zurzeit nur vier Releases der Plattform pro Jahr und auch kleine Änderungen führen zu erheblichen manuellen Aufwänden für das Quality Assurance Team. Das Ziel dieser Arbeit soll es daher sein, eine Architektur für das automatisierte Testing der TV as a Service Plattform zu entwerfen. Diese soll die Qualität der Plattform bei regelmässigen, automatisierten Deployments (idealerweise täglich) sicherstellen und das Vertrauen in die Qualität der Plattform erhöhen. Weiter soll auch der manuelle Quality Assurance Aufwand stark reduziert werden.

Vorgehen

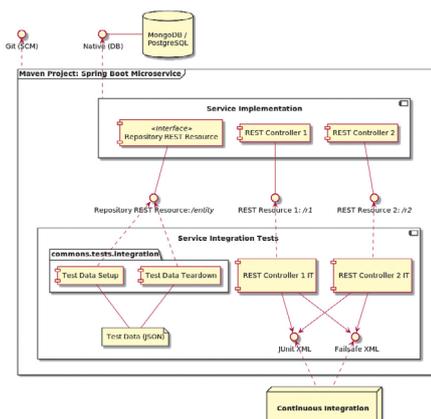
Automatisiertes Testing war schon länger ein Thema in unserer Entwicklungsabteilung. Um die Erwartungen und Anforderung der betroffenen Interessengruppen auf einen Nenner zu bringen und diese zu sammeln, wurde daher als erstes ein Workshop abgehalten. In dem Workshop wurden die Anforderungen und Architekturziele diskutiert und festgehalten. Weiter wurde auch diskutiert, welche Metriken hilfreich für die Messung der Qualität unsere Qualitätssicherungs-Anstrengungen sein könnten. Auf dieser Basis wurden dann verschiedenen Lösungsansätze für die Teilproblematiken evaluiert und daraus ein Architekturvorschlag definiert. Der Architekturvorschlag wurde dann in einem Software Architektur Dokument dokumentiert.



Julian Hanhart

Lösung

Eine grundlegende Anforderung war, dass das automatisierte Testing vom bestehenden Team implementiert und gepflegt werden kann. Daher wurde die Architektur darauf ausgelegt, dass das Testing in die existierenden Prozesse integriert und mit bekannten Werkzeugen implementiert werden kann. Die automatisierten Tests, sowohl für die Backend wie auch für die Frontend-Komponenten, sollen daher auch auf der bestehenden (oder einer kompatiblen) Continuous Integration Plattform ausgeführt werden können. Die Tests sollen auch alle Ebenen der Test-Pyramide erfassen: Von Unit Tests, über Integration Tests bis zu Smoke Tests und Regression Tests. Um das Erstellen von Test-Umgebungen vereinfachen zu können, wurden auch Prozesse für das Provisionieren der Umgebungen auf einem Kubernetes-Cluster und das Generieren und Initialisieren der Test-Daten darauf entworfen und mit Prototypen verifiziert. Die erarbeitete Architektur soll nun zur Umsetzung vorgeschlagen werden und als Grundlage für den Aufbau und Ausbau des automatisierten Testing der TV as a Service Plattform dienen.



Bausteinsicht der Integration Tests