Fakultät Informatik

Evaluation von Contract Testing in Continuous Integration Pipelines zur Sicherstellung der Interoperabilität in einer Microservice-Architektur

Bachelorarbeit im Studiengang Informatik

vorgelegt von

Jonas Lang

Matrikelnummer 363 0314

Erstgutachter:	Prof. Dr.-Ing. Matthias Meitner
Zweitgutachter:	Prof. Dr. Ronald Petrlic
Betreuer:	Sebastian Foersch
Unternehmen:	msg systems ag

Inhaltsverzeichnis

Kapitel 1

Problemstellung

Microservice-Architekturen werden zunehmend für moderne Softwareanwendungen eingesetzt und bieten eine flexible Alternative zu monolithischen Systemen. Sie bieten zahlreiche Vorteile, darunter eine höhere Skalierbarkeit, Flexibilität und eine verbesserte Modularität, indem sie einzelne Services unabhängig voneinander bereitstellen und aktualisieren lassen. Trotz dieser Vorteile ergeben sich jedoch einige Herausforderungen in Bezug auf die Qualitätssicherung, insbesondere im Hinblick auf die Interoperabilität zwischen den verschiedenen Services.

Eine zentrale Problemstellung liegt darin, dass Änderungen an einzelnen Microservices potenziell unerwartete Seiteneffekte auf abhängige Services haben können. Klassische Testmethoden wie End-to-End-Tests und Integrationstests sind zwar in der Lage, solche Probleme aufzudecken, jedoch sind sie oft mit hohen Kosten, langen Ausführungszeiten und einer erhöhten Komplexität in der Wartung verbunden. Dies macht sie insbesondere in Continuous Integration (CI)-Pipelines schwer handhabbar und ineffizient.

Contract Testing hat sich als eine potenziell leistungsfähige Alternative herausgestellt. Es ermöglicht eine gezielte Validierung von Service-Schnittstellen, indem Verträge (Contracts) zwischen Consumer- und Provider-Services definiert und automatisch überprüft werden. Allerdings gibt es bisher nur begrenzte wissenschaftliche Untersuchungen zur praktischen Umsetzung und Wirksamkeit dieser Methodik in CI/CD-Umgebungen.

Daher ist es essenziell, eine detaillierte Untersuchung zur Anwendung von Contract Testing in Continuous Integration Pipelines durchzuführen, um herauszufinden, ob und inwieweit diese Testmethode eine nachhaltige Verbesserung der Interoperabilität in einer Microservice-Architektur gewährleisten kann.

Kapitel 2

Zielsetzung

Das Hauptziel dieser Arbeit ist die systematische Untersuchung und Evaluation von Contract Testing in Continuous Integration Pipelines, um dessen Potenzial zur Sicherstellung der Interoperabilität in einer Microservice-Architektur zu bewerten. Dabei werden insbesondere folgende Aspekte betrachtet:

- Analyse verschiedener Implementierungsmöglichkeiten von Contract Testing in Continuous Integration Pipelines.
- Untersuchung der Effektivität von Contract Testing bei der frühzeitigen Fehlererkennung und deren Auswirkungen auf die Qualitätssicherung.
- Bewertung des Nutzens von Contract Testing hinsichtlich der langfristigen Wartbarkeit, Skalierbarkeit und Effizienz von Microservice-Architekturen.

Um diese Ziele zu erreichen, wird eine experimentelle Studie durchgeführt, bei der Contract Testing in einer realen Microservice-Anwendung integriert und evaluiert wird. Durch den Vergleich mit traditionellen Testmethoden sollen Vor- und Nachteile von Contract Testing systematisch herausgearbeitet werden. Abschließend werden konkrete Handlungsempfehlungen für den praktischen Einsatz in Continuous Integration Pipelines entwickelt und dokumentiert.

Kapitel 3

Vorgehensweise

Die Arbeit folgt einer methodischen Vorgehensweise, die sowohl eine theoretische Analyse als auch eine praktische Evaluation umfasst. Zunächst werden die theoretischen Grundlagen untersucht, um ein fundiertes Verständnis für Microservice-Architekturen, Continuous Integration Pipelines und Teststrategien zu schaffen. Hierbei wird ein besonderer Fokus auf Contract Testing gelegt werden, indem dessen Mechanismen und Konzepte detailliert erläutert und gängige Frameworks betrachtet werden.

Anschließend erfolgt eine umfassende Analyse der Implementierungsmöglichkeiten von Contract Testing in Continuous Integration Pipelines. Dabei werden verschiedene Ansätze untersucht und mit etablierten Testmethoden verglichen. Die Bewertung erfolgt anhand von Kriterien wie Skalierbarkeit, Wartungsaufwand und Fehlererkennungsrate. Zudem werden Best Practices für die Implementierung von Contract Testing identifiziert und Herausforderungen beim Einsatz dieser Methode in agilen Entwicklungsteams herausgearbeitet.

Im praktischen Teil der Arbeit wird eine experimentelle Untersuchung durchgeführt, in der Contract Testing in eine bestehende Continuous Integration Pipeline der realen Microservice-Anwendung „msg.Sensorik“ integriert wird. Dazu werden geeignete Frameworks ausgewählt und implementiert. Anschließend werden gezielte Testszenarien durchgeführt, um die Auswirkungen auf den Entwicklungs- und Deployment-Prozess zu analysieren. Durch den Vergleich der Testergebnisse mit bestehenden Teststrategien soll die Effektivität von Contract Testing bewertet werden. Besondere Aufmerksamkeit gilt hierbei der Frage, inwiefern Contract Testing zur frühzeitigen Identifikation von Schnittstellenfehlern beiträgt, und ob es eine effizientere Alternative zu traditionellen Testmethoden darstellt.

Abschließend werden die gewonnenen Erkenntnisse analysiert und konkrete Handlungsempfehlungen zur Integration von Contract Testing in Continuous Integration Pipelines abgeleitet. Hierbei wird sowohl der potenzielle Nutzen als auch die Grenzen und Herausforderungen dieser Testmethode diskutiert. Die Ergebnisse der Arbeit sollen dazu beitragen, Entwicklern und Software-Architekten eine fundierte Entscheidungsgrundlage für die Implementierung von Contract Testing in ihren Softwareentwicklungsprozessen zu liefern.