

Testing

Teil 3 – Techniken der Softwareprüfung

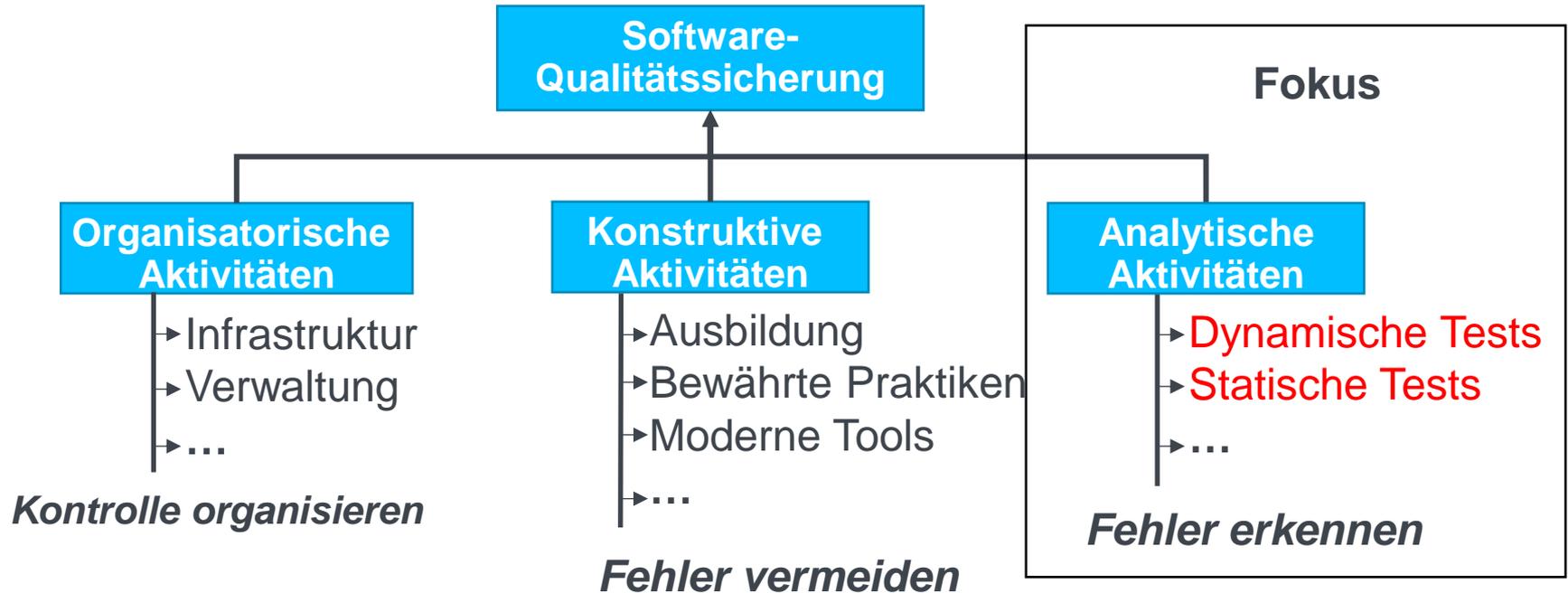
Qualitätssicherung

Bitte beachten Sie, dass der Begriff "Qualitätssicherung" in zweierlei Hinsicht interpretiert werden kann:

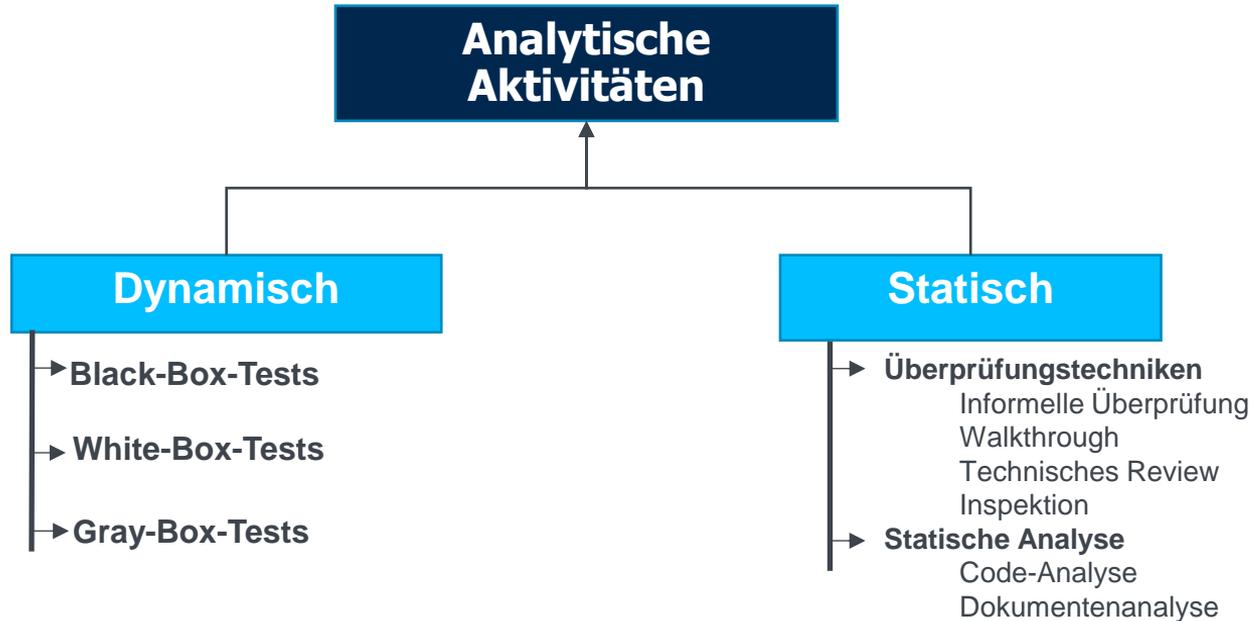
Im Rahmen der ISO 9000-Normen ist der Sammelbegriff für alle Aktivitäten zur Verbesserung der Qualität "Qualitätsmanagement". Der Begriff "Qualitätssicherung" bezeichnet in diesem Zusammenhang insbesondere den Prozess der Erfassung, welche Aufgaben des Qualitätsmanagements durchgeführt wurden.

In anderen Szenarien, zu denen auch der Inhalt dieses Vortrags gehört, umfasst der Begriff "Qualitätssicherung" jedoch üblicherweise alle Aktivitäten, die mit Qualität zu tun haben, was gleichbedeutend ist mit dem, was ISO 9000 als "Qualitätsmanagement" bezeichnet.

Klassifizierung der QA-Aktivitäten



Analytische Aktivitäten



Statische und dynamische Analyse

- Statische und dynamische Analyse sind zwei verschiedene Ansätze, die in der Qualitätssicherung verwendet werden, jeder mit seinen Methoden und Zielen.
- Beide spielen eine entscheidende Rolle bei der Gewährleistung der Qualität und Zuverlässigkeit von Softwareprodukten, und ergänzen sich gegenseitig

Statische Analyse

- **Statische Analyse** ist eine Methode zum Testen von Software, ohne den Code auszuführen. Dabei wird der Quellcode, die Dokumentation und verwandte Dokumente überprüft, um Fehler zu finden. Sie werden in der Regel in den früheren Phasen des Entwicklungslebenszyklus durchgeführt.
- Das Hauptziel der statischen Analyse besteht darin, Fehler frühzeitig zu finden, bevor sie sich zu ernsthaften Problemen entwickeln.

Statische Analyse

Techniken:

Zur statischen Analyse werden Code Reviews, Walkthroughs und Inspektionen eingesetzt. Es gibt auch Tools für die statische Analyse, um potenzielle Probleme im Code automatisch zu erkennen.

Beispiel: Ein Entwickler schreibt eine Funktion zur Berechnung des Durchschnitts einer Liste von Zahlen. Während eines Code Reviews (also einer statische Analyse) könnte ein Kollege feststellen, dass die Funktion nicht prüft, ob die Liste leer ist. In diesem Fall würde eine Division durch 0 zu einem Fehler führen. Dieses Problem kann erkannt und korrigiert werden, bevor der Code ausgeführt wird.

Dynamische Analyse

- Bei der dynamischen Analyse wird die Software durch Ausführen des Codes getestet.
- Das Verhalten und die Ergebnisse der Software werden unter verschiedenen Bedingungen beobachtet.
- In erster Linie soll bestätigt werden, dass das Softwareprodukt gemäß seinen Spezifikationen funktioniert und die Bedürfnisse der Benutzer erfüllt.

Dynamische Analyse

Techniken:

Zur statischen Analyse werden Unit-Tests, Integrationstests, Systemtests und Abnahmetests eingesetzt. Diese Tests werden auf verschiedenen Integrationssebenen der Software durchgeführt und prüfen die funktionalen und nicht-funktionalen Anforderungen.

Beispiel: Die zuvor entwickelte Funktion zur Berechnung des Durchschnitts einer Liste von Zahlen wird mit verschiedenen Listen ausgeführt. Danach wird das Ergebnis mit dem tatsächlichen Durchschnitt verglichen, um sicherzustellen, dass die Berechnung korrekt erfolgt. Wenn die Funktion für einige Beispiellisten korrekt funktioniert, kann vermutet werden, dass sie für alle Listen funktioniert. Es tritt trotz der Tests jedoch vor, dass Sonderfälle zu Fehlern führen, wie etwa die angesprochene Division durch 0 bei einer leeren Liste. Deshalb sollte man beim Anlegen der Tests solche Sonderfälle beachten.

Statische Analysetechniken

- Statische Analysetechniken sind Methoden zur Bewertung von Software-Artefakten, ohne den Code auszuführen.
- Diese Techniken helfen, Fehler frühzeitig zu entdecken, was die Kosten und den Zeitaufwand für die Behebung von Fehlern reduziert, wenn diese sonst erst später im Entwicklungszyklus gefunden werden.
- Statische Analysetechniken können auf verschiedene Dokumente angewandt werden, die während des Softwareentwicklungszyklus erstellt werden, darunter Anforderungen, Entwurfsdokumente, Quellcode und Testfälle.

Statische Analysetechniken

- **Analysetechniken**
 - Informelle Überprüfung
 - Walkthrough
 - Technisches Review
 - Inspektion
- **Statische Analyse**
 - Automatische Code-Analyse
 - Dokumentenanalyse

Statische Analysetechniken: Analysetechniken

Informelle Überprüfung: Informelle Überprüfungen sind Überprüfungen der Entwicklungsartefakte ohne einen formalen Prozess. Dazu gehört die Überprüfung von Entwurfsdokumenten oder Code durch den Entwickler selbst oder einen Kollegen.

- **Beispiel:** Ein Entwickler liest kritisch über das Entwurfsdokument der Benutzeroberfläche der digitalen Bibliothek, um sicherzustellen, dass es mit den festgelegten Anforderungen übereinstimmt, bevor er mit der Implementierung beginnt.

Statische Analysetechniken: Analysetechniken

Walkthrough: Ein Walkthrough ist ein formaler Prozess, bei dem der Autor eines Software-Artefakts dieses einer Gruppe von Kollegen vorstellt. Er ist weniger formal als die Inspektion und erfordert keine gründliche Vorbereitung.

- **Beispiel:** Der Ersteller des Datenbankschemas der digitalen Bibliothek erklärt den Kollegen das Datenbankschema. Ein Kollege stellt Rückfragen, weil er sich unsicher ist, ob das Datenbankschema die Suche nach Büchern effizient verarbeiten kann.

Statische Analysetechniken: Analysetechniken

Technisches Review: Technische Reviews sind formeller als Walkthroughs und beinhalten eine detaillierte Untersuchung der Software-Artefakte durch ein Expertenteam, um Fehler, Verstöße gegen Entwicklungsstandards und andere Probleme zu erkennen.

- **Beispiel:** Bei einer technischen Überprüfung des Anforderungsdokuments für das digitale Bibliothekssystem wird überprüft, ob alle Benutzerrollen (z.B. Bibliothekar, Mitglied, Gast) mit den entsprechenden Berechtigungen korrekt definiert sind.

Statische Analysetechniken: Analysetechniken

Inspektion: Die Inspektion ist die formellste Überprüfungsart und beinhaltet eine gründliche und detaillierte Untersuchung des Software-Artefakts durch Experten, um Fehler zu identifizieren. Der Prozess ist strukturiert und umfasst vordefinierte Phasen wie Planung, Überblick, Vorbereitung, Inspektionssitzung und Nachbereitung.

- **Beispiel:** Das Authentifizierungsmodul der digitalen Bibliothek muss hohe Sicherheitsanforderungen erfüllen. Deshalb wird eine Inspektion des Quellcodes mit Sicherheitsexperten durchgeführt.

Statische Analysetechniken: Statische Analyse

Code-Analyse: Bei der Code-Analyse wird der Quellcode mithilfe automatisierter Tools auf potenzielle Fehler, Code-Smells, Sicherheitslücken und die Einhaltung von Codierungsstandards untersucht.

- **Beispiel:** Eines statisches Code-Analyse-Tool prüft den Backend-Code der digitalen Bibliothek und findet Hinweise auf eine mögliche SQL-Injection-Schwachstelle. Diese wird daraufhin von einem Entwickler manuell geprüft.

Statische Analysetechniken: Statische Analyse

Dokumentenanalyse: Hierbei wird die Dokumentation (Anforderungen, Entwurfsdokumente, Benutzerhandbücher) untersucht, um sicherzustellen, dass sie vollständig, genau, konsistent und eindeutig ist.

- **Beispiel:** Eine technische Überprüfung des Benutzerhandbuchs wird durchgeführt, um sicherzustellen, dass die Anweisungen für die Verwendung des Filmverleih-Managementsystems klar sind, einschließlich der Frage, wie Kunden nach Filmen suchen, DVDs ausleihen und Feedback für das Empfehlungssystem geben können.

Tobias Eisenreich

Universität Stuttgart
Institut für Software Engineering
Empirisches Software Engineering

Umm-e-Habiba

Universität Stuttgart
Institut für Software Engineering
Empirisches Software Engineering



Universität Stuttgart

Institut für Maschinelle Sprachverarbeitung
Institut für Software Engineering



Industrie- und Handelskammer
Reutlingen

Reutlingen | Tübingen | Zollernalb



Region Stuttgart



Industrie- und Handelskammer
Karlsruhe



LUDWIG-
MAXIMILIANS-
UNIVERSITÄT
MÜNCHEN

Lizenzbestimmungen

„Techniken der Softwareprüfung“ von Umm-e-Habiba und Tobias Eisenreich, KIB3 / Uni Stuttgart

Das Werk - mit Ausnahme der folgenden Elemente:

- Logos der Verbundpartner und des Förderprogramms
- im Quellenverzeichnis aufgeführte Medien

ist lizenziert unter:

 [CC BY 4.0 \(https://creativecommons.org/licenses/by/4.0/deed.de\)](https://creativecommons.org/licenses/by/4.0/deed.de)

(Namensnennung 4.0 International)

Quellenverzeichnis

Titelfoto: <https://unsplash.com/de/fotos/Hzp-1ua8DVE>