

Exposé

Analyse und Bewertung von Software-Code-Qualität von Erweiterungen in Content-Management-Systemen anhand von TYPO3

Bachelor-Arbeit

zur Erlangung des Grades Bachelor of Science in Medieninformatik
des Fachbereichs Informatik und Medien an der
Berliner Hochschule für Technik

vorgelegt von:
Renate Ubartas

Betreuer extern: Sebastian Kreideweiß
Gutachter: Prof. Dr. Dragan Macos

Ubartas, Renate Bachelor of Science Studiengang Medieninformatik	S82371@bht-berlin.de Geplanter Beginn: 01.06.2023 Geplante Abgabe: 31.08.2023
--	---

Berlin, den 03. April 2023

Inhaltsverzeichnis

1. Problemstellung	3
2. Erkenntnisinteresse.....	4
3. Fragestellung	4
4. Ziele und Hypothesen	5
a. Ziele.....	5
b. Hypothesen	6
5. Forschungsstand.....	7
6. Methode	7
Literaturverzeichnis	8

1. Problemstellung

Content-Management-Systeme (CMS) haben in den vergangenen Jahren enorm an Bedeutung gewonnen, da sie es Betreiber:innen von Webseiten ermöglicht, ohne Programmierkenntnisse Inhalte zu erstellen und zu verwalten. Ein sehr bekanntes und gängiges CMS ist TYPO3, welches viele Funktionen und Programm-Erweiterungen (im weiteren Verlauf Extensions genannt) bietet, mit denen Webseiten individuell erstellt, gestaltet und gehandhabt werden. Die Codequalität der TYPO3 Extensions kann jedoch stark variieren, da diese von unterschiedlichen Programmierer:innen entwickelt werden, die möglicherweise unterschiedliche Erfahrungen oder Programmierfähigkeiten besitzen. Das kann zu Problemen, wie schlechter Leistung, mangelnder Sicherheit oder schlechter Wartbar- und Erweiterbarkeit der Extensions führen.

Nicht nur die Möglichkeiten von TYPO3 Extensions sind mit den Jahren gestiegen, sondern auch der Umfang des dazu gehörenden Programmcodes. Es kann vorkommen, dass an der Entwicklung mehrere Teams und viele Programmierer:innen beteiligt sind. Dies ist einer von vielen Aspekten, warum die Entwicklung von lesbarem und verständlichem Code so wichtig ist. (Ljung & Gonzalez-Huerta, 2022) Die Definition von guter Codequalität ist jedoch nicht eindeutig, da es von unterschiedlichen Faktoren abhängig ist, wie z. B. welche Personengruppe die Qualität beurteilt. Codequalität kann somit in vielerlei Hinsicht betrachtet werden, wie z. B. die strukturelle Qualität. (Chapell) Die strukturelle Qualität von Software widmet sich der eigentlichen Code-Erstellung durch Entwickler:innen und soll in dieser Bachelorarbeit betrachtet werden. Chappell führt hierfür mehrere Punkte auf, wie beispielsweise Code-Wartung, -Verständlichkeit oder -Sicherheit. Zusammengefasst soll betrachtet werden, wie der Code organisiert ist, wie einfach der Code erweiterbar und verständlich ist oder wie gut erstellter Code zur Sicherheit der Anwendung beitragen kann.

Es ergibt sich daraus das Problem, dass Software-Code einem gewissen Standard entsprechen sollte, um als „guter“ Code zu gelten und Entwickler:innen viel Arbeit investieren müssen, um diesem Standard zu entsprechen. Ein zusätzliches Problem, das sich in diesem Zusammenhang ergibt, ist die Tatsache, dass die Beurteilung von guter Codequalität häufig eine subjektive Angelegenheit darstellt und somit nicht einheitlich bewertet werden kann. Entwickler:innen können unterschiedliche Auffassungen darüber haben, welche Merkmale besonders wichtig für gute Codequalität sind.

2. Erkenntnisinteresse

Unleserlicher, schwer zu pflegender und zu erweiternder Software-Code kann eine Vielzahl von langfristigen Problemen verursachen, die die Produktivität des Entwicklungsteams verringern können. Ein Beispiel ist die Abnahme der Produktivität aufgrund der Anhäufung von "schlechtem" Code im Laufe der Zeit. Zur Bekämpfung dieser Abnahme könnten weitere Entwickler:innen hinzugezogen werden, die jedoch Schwierigkeiten haben, sich in den "schlechten" Code einzuarbeiten. Dies erhöht den Zeitdruck auf das Entwicklungsteam, was wiederum dazu führen kann, dass noch mehr "schlechter" Code hinzugefügt wird. Somit würden zwar mehr Entwickler:innen an dem Projekt arbeiten, die Codequalität würde jedoch dennoch sinken (Martin, 2009).

Gute Codequalität kann somit einen positiven Einfluss auf Entwickler:innen nehmen und sie dazu animieren ebenfalls guten Code zu schreiben. Es gibt jedoch auch viele weitere Gründe, warum Codequalität einen sehr hohen Stellenwert hat. Es können dadurch Ressourcen und somit Kosten eingespart werden und durch die effizientere Programmierung hat dies ebenfalls einen positiven Einfluss auf die Nachhaltigkeit.

Durch die Anwendung etablierter Praktiken und Normen, wie beispielsweise Code Conventions, Best Practices, Test-Driven Development und Continuous Integration, lässt sich bereits ein hohes Niveau an Codequalität erreichen. Dennoch ist die Programmierung von Code von Natur aus ein subjektiver Prozess, bei dem individuelle Interpretationen von Bereichen, die nicht eindeutig vorgegeben sind, zum Tragen kommen können. Infolgedessen kann es trotz der Verwendung bewährter Praktiken und Standards immer noch zu einer schlechten Programmierung von Code kommen.

Das Erkenntnisinteresse dieser Bachelorarbeit besteht darin, herauszufinden, welche Maßnahmen ergriffen werden können, um dem entgegenzuwirken oder wie die Qualität von bestehendem Code verbessert werden kann.

3. Fragestellung

Es gibt mehrere Möglichkeiten und Ansätze um Software-Codequalität zu verbessern, wie die Verwendung von Best Practices und Standards, Code-Reviews, automatische Tests aber auch Schulungen der Entwickler:innen, die das Bewusstsein für die Bedeutung von Code Qualität zu schärfen und ihnen Fähigkeiten und Kenntnisse vermitteln qualitativ hochwertigen Code zu schreiben. Ein weiterer Aspekt ist der

Einsatz von Software-Tools zur statistischen Code-Analyse. Diese Tools können eingesetzt werden, um Probleme in Bezug auf Codequalität, Sicherheit und Performance frühzeitig zu erkennen und den Entwickler:innen eine gute Unterstützung bieten, von Anfang an „guten“ Code zu schreiben. In dieser Arbeit wird die Umsetzung von „gutem“ Software-Code mit Unterstützung solcher Software Tools betrachtet.

Die Forschungsfrage der Bachelor Arbeit lautet dementsprechend:

„Welche Merkmale und Kriterien sind wichtig für die Bewertung der Codequalität von TYPO3 Extensions und wie können statische Software-Analyse Tools wie SonarQube die Entwickler und Entwicklerinnen dabei unterstützen guten Code anhand dieser Kriterien zu schreiben?“

Es soll betrachtet werden, wie Entwickler:innen von Analysetools profitieren können, um besseren Code zu schreiben und damit die Codequalität von TYPO3 Extensions verbessern.

Die Bachelorarbeit befasst sich zwar mit dem allgemeinen Thema der Software-Qualität, dieses Thema wird jedoch auf eine spezifische Untergruppe eingegrenzt. Dabei wird die Qualität von Extensions für CMS untersucht, am Beispiel von TYPO3-Extensions.

Das zu untersuchende Material soll aus verschiedenen Quellen stammen, einschließlich wissenschaftlicher Artikel, Fachliteratur zum Thema Code-Qualität wie z.B. das Handbuch "Clean Code: A Handbook of Agile Software Craftsmanship" und Online-Materialien. Neben der Durchführung einer umfassenden Literaturrecherche soll auch eine Expertenbefragung von mehreren Entwicklern durchgeführt werden, um herauszufinden, welche Kriterien und Qualitätsmerkmale von Bedeutung sind und zu einer guten Software-Qualität beitragen. Ziel ist es, eine Zusammenstellung detaillierter Informationen und Meinungen zu erlangen, die möglicherweise nicht in anderen schriftlichen Quellen verfügbar sind.

4. Ziele und Hypothesen

a. Ziele

Das Ziel dieser Bachelorarbeit ist es, die Bedeutung von Codequalität aufzuzeigen und zu verdeutlichen. Eine hohe Codequalität ermöglicht ein nachhaltiges Management von TYPO3-Extensions, da weniger Ressourcen gebraucht werden, um eine Extension zu erstellen oder zu erweitern, wenn der Code leicht verständlich,

nachvollziehbar und einfach erweiterbar ist. Es gibt jedoch viele Kriterien und Qualitätsmerkmale, anhand derer guter Code bewertet werden kann.

Um die Codequalität von TYPO3-Extensions zu bewerten, werden in dieser Bachelorarbeit Experteninterviews mit Fachleuten, die Extensions erstellen oder vorhandene Extensions erweitern, durchgeführt. Anhand dieser Interviews sollen die wichtigsten Metriken für gute Codequalität identifiziert werden. Hierbei wird untersucht, welche Metriken für die unterschiedlichen Experten besonders wichtig sind. Es soll untersucht werden, ob es gemeinsame Ansichten gibt und ob es Merkmale von guter Codequalität gibt, die besonders häufig genannt werden. Die subjektiven Meinungen der Fachexperten sollen im Rahmen dieser Untersuchung als Grundlage dienen, um daraus objektive Anhaltspunkte für gute Codequalität abzuleiten. In den Experteninterviews soll auch ermittelt werden, welche Bedeutung Fachexperten der guten Codequalität beimessen.

Darüber hinaus sollen in dieser Arbeit Tools wie SonarQube und andere statische Code-Analyse-Tools untersucht werden, um zu prüfen, wie sie dabei unterstützen können, die Codequalität zu verbessern und Entwickler:innen dabei helfen, guten Code zu schreiben. Hierfür soll betrachtet werden, ob Analysetools dabei unterstützen können, die Merkmale von gutem Code, die von den Fachexperten im Interview abgeleitet wurden, umzusetzen.

Durch die Untersuchung der Codequalität sollen die wichtigsten Kriterien und Merkmale von Codequalität identifiziert werden und es soll betrachtet werden, wie diese anschließend umgesetzt werden kann. Mit den Ergebnissen dieser Arbeit können Empfehlungen für die Entwicklung von TYPO3-Extensions gegeben werden, die zu einer höheren Codequalität und somit zu einer nachhaltigen Entwicklung beitragen können.

b. Hypothesen

- Es wird erwartet, dass die Analyse von Tools, wie SonarQube, dazu beitragen wird, die Codequalität von TYPO3-Extensions zu verbessern, indem sie Entwickler:innen dabei helfen, die von den Fachexperten identifizierten Merkmale von gutem Code umzusetzen. Diese Hypothese basiert auf der Annahme, dass die Tools dazu beitragen können, die Codequalität durch automatisierte Analyse und Rückmeldungen zu erhöhen.
- Es wird erwartet, dass die Ergebnisse dieser Arbeit Empfehlungen für die Entwicklung von TYPO3-Extensions liefern werden, die zu einer höheren

Codequalität und somit zu einer nachhaltigen Extension-Entwicklung beitragen können.

- Je höher die Codequalität von TYPO3-Extensions ist, desto nachhaltiger ist ihr Management, da weniger Ressourcen für die Erstellung und Erweiterung benötigt werden. Diese Hypothese basiert auf der Annahme, dass guter Code leichter verständlich, nachvollziehbar und einfacher erweiterbar ist und somit weniger Zeit aufgewendet werden muss diese zu erstellen oder zu bearbeiten.
- Je höher die Codequalität von bestehenden TYPO3-Extensions ist, desto wahrscheinlicher ist es, dass Entwickler:innen, die an dieser Extension programmieren, ebenfalls gute Codequalität einhalten. Diese Hypothese basiert auf der Annahme, dass Entwickler:innen in einem Software-Projekt unter Zeitdruck arbeiten, dabei kann eine hohe Codequalität der vorhandenen Extension dazu beitragen, dass die Entwickler:innen eher geneigt sind, die vorhandene Codebasis zu verbessern, anstatt den schlechten Code einfach zu erweitern.

5. Forschungsstand

Es gibt bereits einige statische Analyse Tools, wie beispielsweise PHPStan, SonarQube, Psalm, Scrutinizer, PHPCheckstyle, Phan, PHPStorm Code Inspections, PHP_CodeSniffer. **TODO**

6. Methode

Zur Erhebung der Daten sollen Befragungen durchgeführt werden. Als Interviewpartner sollen Personen ausgewählt werden, die im Bereich der TYPO3-Extensions tätig sind und über Fachwissen in der Entwicklung von Software-Code verfügen, sei es bei der Erstellung neuer Extensions oder bei der Erweiterung und Anpassung bestehender Extensions. Die Anzahl der Interviewten wird auf eine begrenzte Anzahl von 8 bis 10 Fachexperten beschränkt.

Die Befragung soll als leitfadengestütztes Interview durchgeführt werden. **TODO**

Literaturverzeichnis

Martin, R. C. (2009). *Clean Code - A Handbook of Agile Software Craftmanship*.
Boston: Pearson Education, Inc.

Scheufele, B., & Engelmann, I. (2009). *Empirische Kommunikationsforschung*.
Konstanz: UVK Verlagsgesellschaft mbH.