

Optimierung und Dokumentation der Entwicklungs-, Test- und Release- Prozesse eines Kleinunternehmens

PROJEKTARBEIT

für die Prüfung zum
Bachelor of Science

des Studiengangs Angewandte Informatik

an der Dualen Hochschule Baden-Württemberg Karlsruhe

von

Mael Dossoh

Agabedatum 16.07.2024

| | |
|-------------------------------|---------------------------|
| Matrikelnummer: | 3167941 |
| Kurs: | 22B5 |
| Ausbildungsfirma: | ProSystems GmbH, Sinsheim |
| Betreuer im Unternehmen: | M.Sc. Benno Schweikert |
| Gutachter der Studienakademie | Prof. Dr. Marcus Strand |

Erklärung

„(gemäß §5(3) der „Studien- und Prüfungsordnung DHBW Technik“ vom 29. 9. 2017)
Ich versichere hiermit, dass ich meine Projektarbeit mit dem Thema: „**Optimierung und Dokumentation der Entwicklungs-, Test- und Release-Prozesse eines Kleinunternehmens**“, „selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe. Ich versichere zudem, dass die eingereichte elektronische Fassung mit der gedruckten Fassung übereinstimmt.“

Sinsheim, 16.07.2024

Ort,Datum

Unterschrift

Sperrvermerk

Der Inhalt dieser Arbeit darf weder als Ganzes noch in Auszügen Personen außerhalb des Prüfungsprozesses und des Evaluationsverfahrens zugänglich gemacht werden, sofern keine anders lautende Genehmigung der Ausbildungsstätte vorliegt.

Inhaltsverzeichnis

| | |
|-------------------------------------------------------------------------------------------------------------------------------------------------|-----|
| Inhaltsverzeichnis | I |
| Abbildungsverzeichnis | III |
| Tabellenverzeichnis | IV |
| Listingzeichnis | V |
| Abkürzungsverzeichnis | VI |
| 1. Einleitung | 1 |
| 1.1. Motivation und Ziele der Arbeit | 1 |
| 1.2. Relevanz des Themas für kleine Unternehmen | 2 |
| 1.3. Aufbau der Arbeit (Geplantes Vorgehen) | 2 |
| 2. Grundlagen und Begriffsdefinitionen | 3 |
| 2.1. Definition von Prozessen | 3 |
| 2.2. Relevante Normen und Standards (z.B. ISO, IEEE) | 3 |
| 2.3. Definition und Bedeutung von Entwicklungs-, Test- und Release-Prozessen .. | 3 |
| 2.4. Effizienz und Effektivität in der Prozessoptimierung | 3 |
| 2.5. Methoden zur Identifikation von Schwachstellen | 3 |
| 2.6. Dokumentationspflicht gegenüber Auftraggebern | 3 |
| 2.7. Überblick gängiger Tools in der Softwareentwicklung | 3 |
| 3. Analyse der bestehenden Entwicklungs-, Test- und Release-Prozesse | 4 |
| 3.1. Durchführung einer Mitarbeiterumfrage (z.B. mittels Microsoft Forms) | 4 |
| 3.2. Auswertung des Mitarbeiterfeedbacks | 4 |
| 3.3. Beschreibung der aktuellen Prozesse | 4 |
| 3.4. Visuelle Darstellung der Prozesse (Diagramme und Flussdiagramme) | 4 |
| 3.5. Analyse der Effizienz und Effektivität der bestehenden Prozesse | 4 |
| 4. Einarbeitung von Mitarbeitern | 5 |
| 4.1. Bedeutung der Einarbeitung neuer Mitarbeiter | 5 |
| 4.2. Bestehende Einarbeitungsprozesse | 5 |
| 4.3. Dokumentation und Schulungsmaterialien | 5 |
| 5. Vorschläge zur Prozessoptimierung (Basierend auf den Ergebnissen der Mitarbeiterumfrage, der Analyse, sowie auf Industriestandards) | 6 |
| 5.1. Anpassung dieser Normen auf die Bedürfnisse eines Kleinunternehmens | 6 |
| 5.2. Identifikation und Priorisierung von Verbesserungspotentialen | 6 |
| 5.3. Implementierung und Anpassung von gängigen Tools der Softwareentwicklung (z.B. Jira, Git, CI/CD-Pipelines) | 6 |
| 5.4. (Vielleicht?) Kosten-Nutzen-Analyse der vorgeschlagenen Maßnahmen | 6 |
| 5.5. (Vielleicht?) Vorschläge zur Optimierung der Einarbeitung | 6 |

| | |
|-------------------------------------------------------------------------------------------|-----|
| 6. Schlussfolgerung und Ausblick | 7 |
| 6.1. Zusammenfassung der wichtigsten Erkenntnisse | 7 |
| 6.2. Bewertung der vorgeschlagenen Maßnahmen | 7 |
| 6.3. Ausblick auf zukünftige Entwicklungen und weitere Optimierungsmöglichkeiten | 7 |
| 6.4. Acronyms | 7 |
| 6.5. Lists | 7 |
| 6.6. Figures and Tables | 8 |
| 6.6.1. Figures | 8 |
| 6.6.2. Tables | 8 |
| 6.7. Code Snippets | 8 |
| Literaturverzeichnis | VII |

Abbildungsverzeichnis

| | |
|----------------------------------|---|
| Abbildung 1: Image Example | 8 |
| Abbildung 2: Image Example | 8 |

Tabellenverzeichnis

| | |
|--------------------------------|---|
| Tabelle 1: Table Example | 8 |
|--------------------------------|---|

Listingzeichnis

| | |
|------------------------------------|---|
| Listing 1: Codeblock Example | 9 |
|------------------------------------|---|

Abkürzungsverzeichnis

| | |
|-------------|-----------------------------------|
| API | Application Programming Interface |
| HTTP | Hypertext Transfer Protocol |
| QM | Qualitätsmanagement |
| REST | Representational State Transfer |
| URL | Uniform Resource Locator |

1. Einleitung

1.1. Motivation und Ziele der Arbeit

Die Disziplin des Qualitätsmanagements (QM) hat sich über die Jahre erheblich weiterentwickelt und umfasst heute eine breite Palette von Prinzipien, Methoden und Techniken, die darauf abzielen, die Qualität von Produkten und Dienstleistungen sicherzustellen und zu verbessern. Vor dem 20. Jahrhundert erfolge die Sicherung von Qualität handwerklicher Produkte in erster Linie durch den jeweiligen Meister, der für die Endqualität der hergestellten Waren verantwortlich war. Diese Vorgehensweise wurde jedoch mit der industriellen Revolution und der damit einhergehenden Massenproduktion zunehmend als unzureichend erachtet, sodass erste Ansätze zur systematischeren Qualitätskontrolle entwickelt wurden [Tob14]. Insbesondere in der Automobilindustrie, aber auch in anderen Branchen, wurden Qualitätsmanagement-Systeme eingeführt, um die Qualität der Produkte zu sichern und zu verbessern. Heute sind Qualitätsmanagement-Systeme in vielen Branchen Standard und werden von Unternehmen genutzt, um die Kundenzufriedenheit zu steigern und die Wettbewerbsfähigkeit zu erhöhen.

Die vorliegende Arbeit verfolgt das Ziel die Softwareentwicklungs-Prozesse eines Kleinunternehmens zu analysieren und potentielle Verbesserungspotentiale aufzudecken. Dadurch soll Einarbeitung neuer Mitarbeiter der Softwareentwicklungsabteilung eines Kleinunternehmens zu optimieren und dabei Dokumentationspflichten gegenüber Auftraggebern zu erfüllen. Dazu ist es erforderlich, die bestehenden Entwicklungs-, Test- und Release-Prozesse zu identifizieren und gemäß den einschlägigen Normen, wie beispielsweise der ISO 9000 und der ISO 25000, zu dokumentieren. Die genannten Normen legen insbesondere Wert auf einen prozessorientierten Ansatz, welcher zur systematischen Verbesserung und Kontrolle von Geschäftsprozessen beiträgt. Wie Deming bereits feststellte, sind Prozesse und Qualitätsmanagement von entscheidender Bedeutung, um die Effizienz und Effektivität von Abläufen zu steigern. Die Definition und Dokumentation klarer Prozesse führt zu einer Verbesserung der Konsistenz und Zuverlässigkeit der Arbeitsergebnisse. Zudem wird die Transparenz und Nachvollzieh-

barkeit für alle Beteiligten erhöht. Wie Juran hervorhob, bedeutet Qualität, das Richtige richtig zu tun, wenn niemand zusieht. Die Einhaltung von Qualitätsstandards, wie beispielsweise ISO 9000 und ISO 25000, kann Unternehmen dabei unterstützen, die Qualität ihrer Produkte und Dienstleistungen zu sichern und kontinuierlich zu optimieren.

Die Implementierung eines formellen Qualitätsmanagementsystems stellt für kleine Unternehmen oft eine Herausforderung dar. Oftmals fehlt es an einer klaren Organisation und Struktur, da diese Unternehmen vermeintlich zu klein sind, um umfassende Systeme effektiv durchzuführen. Dennoch kann die Einführung vereinfachter Versionen dieser Normen erheblich zur Verbesserung der internen Abläufe beitragen. Wie Crosby feststellte, müssen kleine Unternehmen zwar flexibel und innovativ sein, dies bedeutet jedoch nicht, dass sie auf strukturierte Prozesse verzichten können. Ein pragmatischer Ansatz, der die spezifischen Bedürfnisse und Ressourcen kleiner Unternehmen berücksichtigt, kann diesen helfen, die Vorteile des Qualitätsmanagements zu nutzen, ohne dabei übermäßig belastet zu werden.

1.2. Relevanz des Themas für kleine Unternehmen

1.3. Aufbau der Arbeit (Geplantes Vorgehen)

- Grundlagen und Begriffsdefinitionen
- Analyse der bestehenden Entwicklungs-, Test- und Release-Prozesse
- Einarbeitung von Mitarbeitern
- Vorschläge zur Prozessoptimierung
- Schlussfolgerung und Ausblick

2. Grundlagen und Begriffsdefinitionen

2.1. Definition von Prozessen

2.2. Relevante Normen und Standards (z.B. ISO, IEEE)

2.3. Definition und Bedeutung von Entwicklungs-, Test- und Release-Prozessen

2.4. Effizienz und Effektivität in der Prozessoptimierung

2.5. Methoden zur Identifikation von Schwachstellen

2.6. Dokumentationspflicht gegenüber Auftraggebern

2.7. Überblick gängiger Tools in der Softwareentwicklung

3. Analyse der bestehenden Entwicklungs-, Test- und Release-Prozesse

3.1. Durchführung einer Mitarbeiterumfrage (z.B. mittels Microsoft Forms)

3.2. Auswertung des Mitarbeiterfeedbacks

3.3. Beschreibung der aktuellen Prozesse

3.4. Visuelle Darstellung der Prozesse (Diagramme und Flussdiagramme)

3.5. Analyse der Effizienz und Effektivität der bestehenden Prozesse

4. Einarbeitung von Mitarbeitern

4.1. Bedeutung der Einarbeitung neuer Mitarbeiter

4.2. Bestehende Einarbeitungsprozesse

4.3. Dokumentation und Schulungsmaterialien

5. Vorschläge zur Prozessoptimierung (Basierend auf den Ergebnissen der Mitarbeiterumfrage, der Analyse, sowie auf Industriestandards)

5.1. Anpassung dieser Normen auf die Bedürfnisse eines Kleinunternehmens

5.2. Identifikation und Priorisierung von Verbesserungspotentialen

5.3. Implementierung und Anpassung von gängigen Tools der Softwareentwicklung (z.B. Jira, Git, CI/CD-Pipelines)

5.4. (Vielleicht?) Kosten-Nutzen-Analyse der vorgeschlagenen Maßnahmen

5.5. (Vielleicht?) Vorschläge zur Optimierung der Einarbeitung

6. Schlussfolgerung und Ausblick

6.1. Zusammenfassung der wichtigsten Erkenntnisse

6.2. Bewertung der vorgeschlagenen Maßnahmen

6.3. Ausblick auf zukünftige Entwicklungen und weitere Optimierungsmöglichkeiten

6.4. Acronyms

Use the `acr` function to insert acronyms, which looks like this Hypertext Transfer Protocol (HTTP).

Application Programming Interfaces are used to define the interaction between different software systems.

REST is an architectural style for networked applications.

URL is a reference to a web resource.

6.5. Lists

Create bullet lists or numbered lists.

- These bullet
 - points
 - are colored
-
1. It also
 2. works with
 3. numbered lists!

6.6. Figures and Tables

Create figures or tables like this:

6.6.1. Figures



Abbildung 1 — Image Example



Abbildung 2 — Image Example

6.6.2. Tables

| Names | Area | Parameters |
|-----------------|-----------------------------|----------------------------------------------------------|
| cylinder.svg | $\pi h \frac{D^2 - d^2}{4}$ | h : height D : outer radius d : inner radius |
| tetrahedron.svg | $\frac{\sqrt{2}}{12} a^3$ | a : edge length |

Tabelle 1 — Table Example

6.7. Code Snippets

Insert code snippets like this:


```
1  #show "ArtosFlow": name => box[
2    #box(image(
3      "logo.svg",
4      height: 0.7em,
5    ))
6    #name
7  ]
8
9  This report is embedded in the
10 ArtosFlow project. ArtosFlow is a
11 project of the Artos Institute.
```

Listing 1 — Codeblock Example

Literaturverzeichnis

- [Tob14] Tobias, Werner: *Handwerk in der postindustriellen Gesellschaft: Handlung und Struktur in einem handwerklich orientierten Traditions Gewerbe am Beispiel von Druckerwerkstätten in der ehemaligen "Buchstadt" Leipzig.*, Leipzig: Leibniz-Institut für Länderkunde e.V., 2014