

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 15.07.2024

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, 15.07.2024

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

5.2. Identifikation und Priorisierung von Verbesserungspotentialen	5
5.3. Implementierung und Anpassung von gängigen Tools der Softwareentwicklung (z.B. Jira, Git, CI/CD-Pipelines)	5
5.4. (Vielleicht?) Kosten-Nutzen-Analyse der vorgeschlagenen Maßnahmen	5
5.5. (Vielleicht?) Vorschläge zur Optimierung der Einarbeitung	5
6. Schlussfolgerung und Ausblick	6
6.1. Zusammenfassung der wichtigsten Erkenntnisse	6
6.2. Bewertung der vorgeschlagenen Maßnahmen	6
6.3. Ausblick auf zukünftige Entwicklungen und weitere Optimierungsmöglichkeiten	6
6.4. Acronyms	6
6.5. Lists	6
6.6. Figures and Tables	7
6.6.1. Figures	7
6.6.2. Tables	7
6.7. Code Snippets	7
6.8. References	8
Literaturverzeichnis	I

Abbildungsverzeichnis

Abbildung 1: Image Example	7
Abbildung 2: Image Example	7

Tabellenverzeichnis

Tabelle 1: Table Example	7
---------------------------------------	----------

Listingzeichnis

Listing 1: Codeblock Example	8
---	----------

Abkürzungsverzeichnis

API	Application Programming Interface
HTTP	Hypertext Transfer Protocol
REST	Representational State Transfer
URL	Uniform Resource Locator

1. Einleitung

1.1. Motivation und Ziele der Arbeit

Um die Einarbeitung neuer Mitarbeiter zu erleichtern und die Dokumentationspflicht gegenüber Auftraggebern zu erfüllen, sollen die bestehenden Entwicklungs-, Test- und Release-Prozesse identifiziert und gemäß den Normen, wie beispielsweise ISO 9000 und ISO 25000, dokumentiert werden. Zusätzlich sollen mögliche Verbesserungspotentiale aufgedeckt und dargestellt werden, um die Effizienz und Effektivität der Prozesse zu steigern.

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 International Organization for Standardization [1]

2. Grundlagen und Begriffsdefinitionen

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

2.2. Definition von Prozessen

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

6.8. References

Cite like this International Organization for Standardization [1]. Or like this [1]. You can also reference by adding <ref> with the desired name after figures or headings.

Or like this I. O. for Standardization (ISO) [2]

Literaturverzeichnis

- [1] International Organization for Standardization, „ISO/IEC 18004: Information technology – Automatic identification and data capture techniques – QR code bar code symbology specification“, in *ISO/IEC 18004:2000*, 2000.
- [2] I. O. for Standardization (ISO), „ISO/IEC 25000:2005, Software Engineering - Software Product Quality Requirements and Evaluation (SQuaRE)“. 2005.