



Bachelor-Thesis

Titel der Abschlussarbeit

zur Erlangung des akademischen Grades

Bachelor of Science

im dualen Studiengang Informatik an der IU Internationale Hochschule

von

Maria Musterfrau
Matrikelnummer: 1234567
Musterstraße 1, 12345 Musterstadt

12. Dezember 2025

Referent: Prof. Dr. Klaus Quibeldey-Cirkel

Korreferent: Prof. Dr. Philipp Diebold

Erklärung zur Nutzung von KI-basierten Werkzeugen

Bei der Erstellung der vorliegenden Arbeit habe ich KI-basierte Anwendungen bzw. Werkzeuge als Hilfsmittel verwendet. Die verwendeten Tools (GitHub Copilot, ChatGPT 4) sowie Art und Umfang der Nutzung sind im Anhang „Verzeichnis der KI-Nutzung“ tabellarisch dokumentiert.

Ich erkläre hiermit, dass ich die von KI-Systemen generierten Vorschläge und Inhalte ausschließlich als Unterstützung verstanden und diese eigenständig geprüft, kritisch bewertet und überarbeitet habe. Die Verantwortung für die fachliche Richtigkeit, die Auswahl und Interpretation der Ergebnisse sowie die endgültige Textfassung liegt vollständig bei mir.

Darüber hinaus wurden alle weiteren Quellen und Hilfsmittel vollständig und korrekt angegeben.

Diese Erklärung folgt den Empfehlungen der Deutschen Forschungsgemeinschaft (DFG)¹ und wissenschaftlichen Fachzeitschriften².

Gender-Disclaimer

In dieser Arbeit wird aus Gründen der besseren Lesbarkeit das generische Maskulinum verwendet. Weibliche und anderweitige Geschlechteridentitäten werden dabei ausdrücklich mitgemeint, soweit es für die Aussage erforderlich ist.

1 DFG: Richtlinien für den Umgang mit generativen Modellen für Text- und Bild: <https://www.dfg.de/resource/blob/289676/89c03e7a7a8a024093602995974832f9/230921-statement-executive-committee-ki-ai-data.pdf>

2 Z.B. Theoretical Computer Science: <https://www.sciencedirect.com/journal/theoretical-computer-science/publish/guide-for-authors>

Erklärung der Selbstständigkeit

Hiermit versichere ich, die vorliegende Arbeit selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt sowie die Zitate deutlich kenntlich gemacht zu haben.

Gießen, den 12. Dezember 2025

Maria Musterfrau

Diese Abschlussarbeit beschäftigt sich mit der Optimierung von Prozessen im Bereich der Softwareentwicklung. Im Zentrum steht die Frage, wie etablierte Methoden durch moderne Technologien effizienter gestaltet werden können, ohne dabei bestehende Standards zu gefährden.

Die Motivation für diese Arbeit ergibt sich aus der wachsenden Komplexität von Softwareprojekten und dem gleichzeitigen Druck, Entwicklungszyklen zu verkürzen. Ein wichtiger Aspekt ist dabei die Sicherung der Code-Qualität trotz steigender Anforderungen.

Als Methode wurde eine Kombination aus Literaturrecherche, praktischer Implementierung eines Prototyps und evaluativen Studien durchgeführt. Die praktische Umsetzung basiert auf etablierten Frameworks und Best-Practices der Industrie.

Die wichtigsten Ergebnisse zeigen, dass durch gezielte Optimierungen eine Reduktion der Entwicklungszeit um durchschnittlich 20% erreicht werden kann, während die Codequalität gleichbleibt oder sich sogar verbessert. Diese Resultate wurden durch mehrere Experimente validiert.

Insgesamt trägt diese Arbeit zu einem besseren Verständnis der Wechselwirkungen zwischen Effizienz und Qualität in der Softwareentwicklung bei und bietet praktische Ansätze für die Industrie.

Inhaltsverzeichnis

1 Einführung	1
1.1 Motivation und Relevanz	1
1.2 Problemstellung	1
1.3 Zielsetzung	1
1.4 Abgrenzung des Themas	2
1.5 Aufbau der Arbeit	2
2 Theoretischer Hintergrund	3
2.1 Grundkonzepte	3
2.1.1 Konzept 1: Fundamentale Begriffe	3
2.1.2 Konzept 2: Etablierte Methoden	3
2.2 Verwandte Arbeiten	3
2.2.1 Ansatz A	4
2.2.2 Ansatz B	4
2.3 Vergleich und Bewertung	4
2.4 Forschungslücke	4
3 Konzept und Methodik	5
3.1 Übersicht des Lösungsansatzes	5
3.2 Architektur und Design	5
3.3 Mathematische Grundlagen	5
3.4 Methodik	6
3.5 Abgrenzung zu alternativen Ansätzen	6
4 Implementierung und Ergebnisse	7
4.1 Implementierungsdetails	7
4.1.1 Technologiestack	7
4.1.2 Architektur der Lösung	7
4.2 Experimentelles Setup	8
4.2.1 Testumgebung	8

Inhaltsverzeichnis

4.3 Ergebnisse	8
4.3.1 Leistungsmessungen	8
4.3.2 Qualitätsmetriken	8
4.4 Vergleich mit existierenden Ansätzen	9
4.5 Validierung und Verifikation	9
5 Fazit und Ausblick	11
5.1 Zusammenfassung der Ergebnisse	11
5.1.1 Beantwortung der Forschungsfragen	11
5.2 Beiträge dieser Arbeit	11
5.3 Limitierungen	12
5.4 Zukünftige Arbeiten	12
5.4.1 Kurzfristige Verbesserungen	12
5.4.2 Langfristige Perspektiven	13
5.5 Schlusswort	13
Literaturverzeichnis	15
Abkürzungsverzeichnis	15
Abbildungsverzeichnis	17
Tabellenverzeichnis	19
Listings	21
A Verzeichnis der KI-Nutzung	23
B Zusätzliche Materialien	25
B.1 Weitere Tabellen und Daten	25
B.2 Quellcode und Implementierungsdetails	25
B.3 Ergänzende Berechnungen	25

1 Einführung

1.1 Motivation und Relevanz

Die Bedeutung des Forschungsthemas ergibt sich aus aktuellen Entwicklungen in Wissenschaft und Industrie. Diese Arbeit behandelt einen Problembereich, der sowohl theoretische als auch praktische Relevanz besitzt.

1.2 Problemstellung

Bei der Bearbeitung dieses Themas stellen sich folgende zentrale Fragen:

- Wie können existierende Methoden verbessert werden?
- Welche Hindernisse sind dabei zu überwinden?
- Welche Lösungsansätze sind praktikabel?

1.3 Zielsetzung

Die Ziele dieser Arbeit sind:

1. Analyse des aktuellen Stands der Forschung
2. Identifikation von Verbesserungsmöglichkeiten

3. Entwicklung und Validierung eines Lösungsansatzes
4. Evaluation und Dokumentation der Ergebnisse

1.4 Abgrenzung des Themas

Um den Umfang zu begrenzen, konzentriert sich diese Arbeit auf spezifische Aspekte des Themas. Folgende Bereiche werden nicht behandelt:

- Zu komplexe Spezialfälle, die für diese Arbeit nicht relevant sind
- Historische Entwicklungen vor einem bestimmten Zeitpunkt
- Randbereiche, die außerhalb des Fokus liegen

1.5 Aufbau der Arbeit

Die restliche Arbeit gliedert sich wie folgt:

Kapitel 2 behandelt den theoretischen Hintergrund und gibt einen Überblick über relevante Arbeiten in der Literatur.

Kapitel 3 stellt das entwickelte Konzept vor und erläutert den gewählten Lösungsansatz.

Kapitel 4 dokumentiert die Implementierung und zeigt praktische Ergebnisse.

Kapitel 5 fasst die Erkenntnisse zusammen und gibt einen Ausblick auf zukünftige Arbeiten.

2 Theoretischer Hintergrund

2.1 Grundkonzepte

Dieses Kapitel vermittelt die notwendigen Grundlagen zum Verständnis der Arbeit. Die behandelten Konzepte bilden die theoretische Basis für die später entwickelten Lösungsansätze [Smi20].

2.1.1 Konzept 1: Fundamentale Begriffe

Zunächst werden die zentralen Begriffe erläutert und definiert, die im Kontext dieser Arbeit verwendet werden.

2.1.2 Konzept 2: Etablierte Methoden

Etablierte Methoden und Techniken werden dargestellt, um den aktuellen Stand der Technik zu verdeutlichen [DR19].

2.2 Verwandte Arbeiten

In diesem Abschnitt wird ein Überblick über relevante Forschungsarbeiten und Publikationen gegeben.

2.2.1 Ansatz A

Der erste bekannte Lösungsansatz wird kritisch analysiert mit seinen Vorteilen und Limitierungen.

2.2.2 Ansatz B

Ein alternativer Ansatz wird dargestellt und mit dem vorherigen verglichen [LB18].

2.3 Vergleich und Bewertung

Die verschiedenen Ansätze werden systematisch verglichen und bewertet. Daraus ergibt sich die Notwendigkeit eines neuen oder verbesserten Ansatzes.

2.4 Forschungslücke

Basierend auf der Analyse können folgende Lücken identifiziert werden:

- Limitation 1: Bestehende Ansätze behandeln diesen Aspekt nicht
- Limitation 2: Skalierungsprobleme in praktischen Anwendungen
- Limitation 3: Mangelnde Validierung in Echtszenarien

Diese Arbeit trägt dazu bei, diese Lücken zu schließen.

3 Konzept und Methodik

3.1 Übersicht des Lösungsansatzes

Auf Basis der Analyse in Kapitel 2 wird in diesem Kapitel ein neuer Lösungsansatz entwickelt. Der Ansatz adressiert die identifizierten Forschungslücken und nutzt die Stärken bestehender Methoden.

3.2 Architektur und Design

Das Konzept basiert auf folgenden Designprinzipien:

- Modularität: Komponenten sind unabhängig und austauschbar
- Skalierbarkeit: Das System wächst mit den Anforderungen
- Wartbarkeit: Code ist verständlich und dokumentiert
- Robustheit: Fehlertoleranz und Zuverlässigkeit

3.3 Mathematische Grundlagen

Falls erforderlich, können mathematische Modelle dargestellt werden:

$$f(x) = \sum_{i=1}^n x_i \cdot w_i$$

wobei x_i die Eingabewerte und w_i die Gewichtungen darstellen.

3.4 Methodik

Die Realisierung folgt einem systematischen Vorgehen:

1. Anforderungsanalyse: Präzise Definition der Ziele
2. Designphase: Architektur und Schnittstellen festlegen
3. Implementierungsphase: Umsetzung des Designs
4. Testphase: Validierung und Verifikation
5. Optimierungsphase: Performance und Qualität verbessern

3.5 Abgrenzung zu alternativen Ansätzen

Dieser Ansatz unterscheidet sich von den in Kapitel 2 beschriebenen Methoden durch:

- Verbesserte Effizienz durch optimierte Algorithmen
- Bessere Skalierungseigenschaften
- Praktischere Anwendbarkeit

4 Implementierung und Ergebnisse

4.1 Implementierungsdetails

In diesem Kapitel werden die praktischen Aspekte der Umsetzung dokumentiert.

4.1.1 Technologiestack

Folgende Technologien wurden für die Implementierung eingesetzt:

- Programmiersprache: Python 3.10+
- Framework: Django / FastAPI
- Datenbank: PostgreSQL
- Versionskontrolle: Git

4.1.2 Architektur der Lösung

Die Implementierung folgt dem in Kapitel 3 beschriebenen Design.

4.2 Experimentelles Setup

Die Validierung erfolgt anhand von realistischen Testszenarien.

4.2.1 Testumgebung

- Hardware: Intel i7, 16GB RAM
- Betriebssystem: Ubuntu 22.04
- Testdaten: Synthethische und reale Datensätze

4.3 Ergebnisse

Die durchgeführten Experimente zeigen folgende Ergebnisse:

4.3.1 Leistungsmessungen

Die entwickelte Lösung erreicht eine durchschnittliche Ausführungszeit von $t_{avg} = 150ms$ mit einer Standardabweichung von $\sigma = 25ms$.

4.3.2 Qualitätsmetriken

- Genauigkeit: 95%
- Verlässlichkeit: 99.2%
- Skalierbarkeit: Linear bis 10.000 Requests/min

4.4 Vergleich mit existierenden Ansätzen

Die entwickelte Lösung übertrifft bestehende Ansätze in folgenden Aspekten:

- 20% schneller als Baseline
- 15% geringerer Speicherbedarf
- Bessere Fehlertoleranz

4.5 Validierung und Verifikation

Alle kritischen Funktionen wurden durch automatisierte Tests validiert. Die Testabdeckung beträgt 92%.

5 Fazit und Ausblick

5.1 Zusammenfassung der Ergebnisse

Diese Arbeit hat gezeigt, dass die in Kapitel 3 und 4 entwickelte Lösung die identifizierten Probleme effektiv adressiert. Die durchgeführten Experimente validieren den gewählten Ansatz.

5.1.1 Beantwortung der Forschungsfragen

Forschungsfrage 1: Wie können existierende Methoden verbessert werden?

Die Arbeit zeigt, dass durch gezielte Optimierungen der Architektur und Algorithmen signifikante Verbesserungen erreicht werden können.

Forschungsfrage 2: Welche praktischen Auswirkungen hat der neue Ansatz?

In realen Szenarien zeigen sich Leistungssteigerungen von durchschnittlich 20% sowie verbesserte Zuverlässigkeit.

5.2 Beiträge dieser Arbeit

Diese Arbeit leistet folgende Beiträge zur Forschung:

1. Entwicklung eines neuen Lösungsansatzes basierend auf aktuellen Erkenntnissen

2. Praktische Validierung durch umfangreiche Experimente
3. Identifikation von Verbesserungspotentialen in bestehenden Methoden
4. Publikation von Erkenntnissen für die wissenschaftliche Community

5.3 Limitierungen

Trotz der positiven Ergebnisse gibt es folgende Limitierungen:

- Die Experimente wurden in kontrollierter Umgebung durchgeführt
- Skalierungstests waren auf 10.000 Requests/min begrenzt
- Die Validierung konzentrierte sich auf spezifische Datensätze

5.4 Zukünftige Arbeiten

Auf Basis dieser Arbeit ergeben sich mehrere Richtungen für zukünftige Forschung:

5.4.1 Kurzfristige Verbesserungen

- Optimierung der Speichernutzung für Echtzeit-Anwendungen
- Erweiterung der Testabdeckung auf weitere Datensätze
- Integration mit bestehenden Systemen

5.4.2 Langfristige Perspektiven

- Erweiterung des Ansatzes auf verwandte Problemdomänen
- Untersuchung von Hybrid-Methoden
- Machine-Learning basierte Optimierungen

5.5 Schlusswort

Diese Arbeit trägt zu einem besseren Verständnis der untersuchten Problematik bei und bietet praktische Lösungen, die in der Industrie angewendet werden können. Die entwickelten Methoden bilden eine solide Grundlage für zukünftige Forschung und praktische Anwendungen.

Literaturverzeichnis

- [Aut21] Example Author. *Example Website*. <https://www.example.com>. Accessed: 2021-12-13. 2021.
- [DR19] Jane Doe und John Roe. „An In-Depth Research on Examples“. In: *Journal of Examples* 42.3 (2019), S. 123–456. DOI: 10.1234/example.2019.042.
- [LB18] Alice Lee und Bob Brown. „Example Conference Paper“. In: *Proceedings of the 10th International Conference on Examples*. Hrsg. von Carol White und Dave Green. Example City: Example Publisher, Juli 2018, S. 789–798.
- [Mil17] Eve Miller. „A Chapter on Examples“. In: *Handbook of Examples*. Hrsg. von Frank Black. Example City: Example Publisher, 2017. Kap. 5, S. 99–120.
- [Smi20] John Smith. *An Example Book*. 2nd. Example City: Example Publisher, 2020.

Abbildungsverzeichnis

Tabellenverzeichnis

Listings

A Verzeichnis der KI-Nutzung

Die folgende Tabelle dokumentiert den Einsatz KI-basierter Werkzeuge bei der Erstellung dieser Arbeit. Jede Nutzung wurde eigenständig geprüft, kritisch bewertet und überarbeitet.

Hinweis: Alle generierten Vorschläge wurden eigenständig geprüft, fachlich bewertet und gegebenenfalls angepasst oder verworfen. Die Verantwortung für die finale Fassung liegt vollständig beim Autor/bei der Autorin.

KI-Tool	Zweck	Kontext/Eingangstext	Generierte Ausgabe	Kapitel
Anthropic Claude Opus 4.5	Konzeptualisierung	Strukturierung des methodischen Ansatzes	Vorschlag zur Gliederung der Methodik	Kap. 3.1
Anthropic Claude Opus 4.5	Code-Refactoring	Optimierung bestehender Algorithmen	Vorschläge zur Verbesserung der Effizienz	Kap. 4.3
Cohere Command A	RAG/Agentik	Entwurf eines Retrieval-Flows	Vorschlag für Tool-Aufrufe und Prompt-Struktur	Kap. 3.2
DeepL Agent	Übersetzung	Deutsche Formulierung für Abstract	Professionelle englische Übersetzung	Abstract
GitHub Copilot	Code-Vervollständigung	Python-Funktion für Datenverarbeitung	Vorschläge für Funktionsimplementierung	Kap. 4.2
Google Gemini 3 Pro	Literaturrecherche	Zusammenfassung aktueller Forschungsarbeiten	Überblick über Stand der Technik	Kap. 2.1
Mistral Large 3	Technische Zusammenfassung	Auswertung von API-/RFC-Dokumenten	Kompakte Zusammenfassung der Kernaussagen	Kap. 2.4
OpenAI GPT-5.2	Formulierungsverbesserung	Überarbeitung der Einleitung	Alternative Formulierungen für Problemstellung	Kap. 1.2, S. 5
xAI Grok 4.1	Agentische Aufgaben	Recherche zu aktuellen Statistiken	Konsolidierte Stichpunkt-Zusammenfassung mit Quellenhinweisen	Kap. 2.3

B Zusätzliche Materialien

In diesem Anhang können weitere zusätzliche Materialien eingefügt werden, die für das Verständnis der Hauptarbeit hilfreich sind.

B.1 Weitere Tabellen und Daten

Tabellarische Daten, die zu umfangreich für die Hauptkapitel sind.

B.2 Quellcode und Implementierungsdetails

Längere Quellcode-Listings oder detaillierte Implementierungen.

B.3 Ergänzende Berechnungen

Mathematische Herleitungen oder detaillierte Berechnungen, die nicht im Haupttext nötig sind.