

Bachelor Thesis

Die Zukunft der Software-Entwicklung unter dem Einfluss künstlicher
Intelligenz: Chancen, Herausforderungen und praxisorientierte
Anwendungen

zur Erlangung des akademischen Grades

Bachelor of Science

eingereicht im Fachbereich Mathematik, Naturwissenschaften und Informatik an der
Technischen Hochschule Mittelhessen

von

Jan Ole Schmidt

25. Juni 2025

Referent: Prof. Dr. Tim Mustermann

Korreferent: Prof. Dr. Max Häuser

Erklärung zur Verwendung von generativer KI

In Übereinstimmung mit den Empfehlungen der Deutschen Forschungsgemeinschaft (DFG)¹ und denen der Zeitschrift Theoretical Computer Science² erkläre ich (der Autor/die Autorin) hiermit den Einsatz von generativer KI.

Bei der Vorbereitung dieser Arbeit habe ich ChatGPT 4 verwendet, um ausschließlich die Lesbarkeit und Sprache zu verbessern. Nach der Verwendung von ChatGPT 4 habe ich den Inhalt überprüft und nach Bedarf bearbeitet und übernehme die volle Verantwortung für den Inhalt dieser Arbeit.

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 25. Juni 2025

Jan Ole Schmidt

1 DFG formuliert Richtlinien für den Umgang mit generativen Modellen für Text- und Bild: <https://www.dfg.de/en/news/news-topics/announcements-proposals/2023/info-wissenschaft-23-72>

2 Erklärung zur Verwendung von generativer KI in wissenschaftlichen Arbeiten: <https://www.sciencedirect.com/journal/theoretical-computer-science/publish/guide-for-authors>

Inhaltsverzeichnis

1	Einführung	1
1.1	Motivation	1
1.2	Zielsetzung und Fragestellungen	2
1.3	Methodik	3
1.4	Aufbau der Arbeit	4
1.5	Abgrenzung	5
2	Theoretische Grundlagen	7
2.1	Künstliche Intelligenz: Definitionen und Technologien	7
2.1.1	Beispiele für generative KI-Tools in der Praxis	7
2.1.2	Wichtige Algorithmen und Modelle in der Softwareentwicklung	7
2.2	Generative KI-Tools: Funktion und Anwendung	7
3	Praktische Demonstration	9
3.1	Zielsetzung und Vorgehen	9
3.1.1	Motivation für ein praktisches Beispiel	9
3.1.2	Eingesetzte KI-Tools	9
3.1.3	Technischer Aufbau	9
3.2	Vorstellung der App „Locals“	10
3.2.1	Architektur und Aufbau	10
3.2.2	Bestehende Funktionalitäten	11
3.3	Implementierung der interaktiven Kartenansicht mit KI-Unterstützung	11
3.3.1	Integration der KI-Tools	11
3.3.2	Entwicklungsprozess mit KI-Unterstützung	12
3.4	Erste Evaluierung	12
3.4.1	Erfolge und Herausforderungen	12
3.4.2	Qualitative Bewertung	12
3.5	Zwischenfazit	12
4	Chancen	13
4.1	Effizienzsteigerung und Automatisierung	13
4.2	Neue Werkzeuge und Methoden	13

5	Herausforderungen durch KI in der Softwareentwicklung	15
5.1	Sicherheits- und Datenschutzaspekte	15
5.1.1	Sicherheitsrisiken durch generative Modelle	15
5.2	Ethische und soziale Implikationen	15
5.2.1	Ethische Konflikte und Bias in KI-Systemen	15
5.2.2	Langfristige Auswirkungen auf Entwickler:innen-Rollen	15
5.2.3	Technische und organisatorische Hürden bei der Einführung von KI	15
6	Wirtschaftliche und gesellschaftliche Auswirkungen	17
6.1	Veränderungen in Softwareunternehmen	17
6.2	Auswirkungen auf den Arbeitsmarkt und Entwickler:innen-Rollen	17
6.3	Zukunftsperspektiven und strategische Empfehlungen	17
6.4	Kosten-Nutzen-Analyse von KI-gestützter Softwareentwicklung	17
7	Fazit und Ausblick	19
7.1	Erwartete Erkenntnisse	19
7.2	Zusammenfassung der Erkenntnisse	19
7.3	Handlungsempfehlungen und Zukunftsperspektiven	19
	Literaturverzeichnis	21
	Abkürzungsverzeichnis	24
	Abbildungsverzeichnis	25
	Tabellenverzeichnis	27
	Listings	29
A	Anhang 1	31
B	Anhang 2	33

1 Einführung

Künstliche Intelligenz (KI) hat in den vergangenen Jahren einen rasanten Aufschwung erlebt und beeinflusst bereits vielfältige Branchen von der Medizin bis zur Finanzwelt. Auch die Softwareentwicklung bleibt nicht verschont: Dort eröffnen KI-gestützte Verfahren ein breites Spektrum neuer Einsatzfelder. So kann KI nicht nur das Schreiben von Code und das Durchführen automatisierter Tests erleichtern, sondern auch innovative Methoden für Fehlersuche und Qualitätssicherung bereitstellen.

Diese Potenziale gehen jedoch mit weitreichenden Fragestellungen einher. Neben technischen Aspekten wie Sicherheit und Code-Qualität spielt auch die gesellschaftliche Dimension eine Rolle, etwa die Frage nach ethischen Standards oder Veränderungen im Berufsbild „Softwareentwickler“. Insbesondere generative KI, zum Beispiel in Form von Large Language Models (LLMs), wirft Fragen zu Datenschutz, Verantwortung und methodischer Einbindung in agile Prozesse auf.

Vor diesem Hintergrund setzt die vorliegende Arbeit an: Sie soll beleuchten, wie KI-gestützte Technologien den Softwareentwicklungsprozess langfristig prägen und welche Herausforderungen sich dabei ergeben. Dies betrifft sowohl die konkrete Arbeitssituation von Entwickler:innen als auch die strategischen Überlegungen von Unternehmen.

1.1 Motivation

Die Relevanz des Themas ergibt sich aus den gegenwärtigen Entwicklungen in Forschung und Praxis: Immer mehr Unternehmen erforschen aktiv den Einsatz von KI-Technologien, um sich Effizienzvorteile und Innovationsschübe zu sichern. Gleichzeitig zeigt sich in vielen Studien ein Spannungsverhältnis zwischen den Versprechen generativer KI – zum Beispiel automatisierte Code-Generierung und intelligente Projektsteuerung – und den Risiken, etwa unzureichender Transparenz, Sicherheitslücken oder ethischen Verzerrungen.

Nach aktuellen Schätzungen (Stand: 2025) erreicht der Softwaremarkt in Deutschland ein Volumen von rund 31 Milliarden US-Dollar und Entwicklerinnen und Entwickler verbringen laut Erhebungen durchschnittlich bis zu 17 Stunden pro Woche mit

Wartungsaufgaben – dies zeigt, wie dringend Automatisierungsansätze und Qualitätsverbesserungen in der Praxis benötigt werden. Gerade hier setzt die generative KI an: Sie kann beispielsweise durch das automatisierte Erstellen von Boilerplate-Code oder durch Code-Assistenten wie GitHub Copilot nicht nur die Effizienz im Entwicklungsprozess deutlich steigern, sondern auch das Fachkräftethema ein Stück weit abfedern. Allerdings lassen sich aus diesem Trend auch kontroverse Fragen ableiten, etwa inwiefern die starke Abhängigkeit von KI-Modellen die Rollen und Kompetenzen von Softwareentwickler:innen und -entwicklern langfristig verändert – oder wie Unternehmen sicherstellen können, dass durch KI-gestützte Automatisierung weiterhin qualitativ hochwertige, wartbare und sichere Software entsteht [Sie24].

Hinzu kommt, dass Softwareentwicklung durch agile Methoden wie Scrum oder Kanban bereits stark dynamisiert ist: Teams agieren flexibel, stehen aber auch unter stetigem Veränderungsdruck. Wenn dann zusätzlich KI als Tool oder „Co-Entwickler“ eingebunden wird, steigen die Anforderungen an Prozessgestaltung, Rollenverteilung und Qualitätsmanagement weiter. Genau hier setzt diese Arbeit an: Sie möchte klären, wie Entwickler und Entscheider KI sinnvoll in den Softwarelebenszyklus integrieren können, wo praxisnahe Chancen liegen und welche neuen Stolpersteine zu beachten sind.

Dabei deuten aktuelle Marktanalysen darauf hin, dass KI-Assistenten die Arbeitsweise von Softwareentwicklern erheblich beschleunigen können. Laut einer von Deloitte zitierten Studie ist es beispielsweise möglich, dass KI-basierte Coding-Tools – zumindest bei Routinetätigkeiten – die dafür benötigte Entwicklerzeit um bis zu 50% reduzieren (vgl. [S.24]). Gleichzeitig fließt in diesem Bereich laut Branchenberichten inzwischen weltweit massiv Kapital, was auf die steigende Bedeutung hinweist. Daraus ergibt sich die Notwendigkeit, Chancen und Herausforderungen systematisch zu analysieren und klare Handlungsempfehlungen aufzustellen, damit die Integration von KI in Entwicklungsprozessen nicht nur technisch, sondern auch ethisch und organisatorisch gut gelingt.

1.2 Zielsetzung und Fragestellungen

Das Ziel dieser Arbeit ist es, die Auswirkungen von KI auf die Softwareentwicklung zu analysieren und praxisnahe Handlungsempfehlungen für Unternehmen und Entwickler abzuleiten. Dabei werden insbesondere folgende Forschungsfragen untersucht:

FF-1 Wie verändert generative KI traditionelle Entwicklungspraktiken in der Softwareentwicklung?

- FF-2 Welche spezifischen Herausforderungen entstehen durch KI-gestützte Softwareentwicklung hinsichtlich Sicherheit, Ethik und Code-Qualität?
- FF-3 Wie kann Generative KI Softwareentwickler in einem agilen Entwicklungsprozess unterstützen?
- FF-4 Wie lassen sich bestehende generative KI-Tools (Cursor, GitHub Copilot, v0 etc.) in den Entwicklungsprozess einer React-Native-App integrieren, und welchen Einfluss hat das auf Entwicklungszeit und Code-Qualität?

Darüber hinaus wird ein praktisches Beispiel in Form einer React Native-App vorgestellt, um zu untersuchen, wie bestehende KI-Tools in einen realen Entwicklungsprozess integriert werden können und wie generative KI den Entwicklungsprozess in einem praxisnahen Projekt unterstützt.

1.3 Methodik

Diese Arbeit verfolgt eine theoretische und literaturbasierte Herangehensweise, um ein umfassendes Verständnis der aktuellen Forschungslage zu generativer KI in der Softwareentwicklung zu erhalten. Die Methodik umfasst folgende Schritte:

1. **Literaturrecherche:** Analyse wissenschaftlicher Publikationen aus IEEE Xplore, arXiv, SpringerLink und weiteren relevanten Fachquellen mit Fokus auf aktuelle Studien zur KI-gestützten Softwareentwicklung.
2. **Kategorisierung der Forschungsthemen:** Identifikation und Gruppierung zentraler Themenfelder wie Automatisierung, Produktivität, Sicherheitsrisiken und ethische Fragestellungen.
3. **Vergleichende Analyse:** Gegenüberstellung der identifizierten Chancen und Herausforderungen auf Basis aktueller Studien und Fachbeiträge.
4. **Synthese und Ableitung von Schlussfolgerungen:** Entwicklung praxisorientierter Handlungsempfehlungen für den Einsatz von KI in der Softwareentwicklung.
5. **Praktische Demonstration:** Im Rahmen der Arbeit wird exemplarisch ein Map-Screen (interaktive Kartenansicht) in der React Native-App „Locals“ entwickelt. Dabei werden ausgewählte generative KI-Tools (z.B. Cursor, v0 oder GitHub Copilot) eingesetzt, um Code-Generierung, Tests und Qualitätsverbesserungen zu demonstrieren. Die Erfahrungen aus diesem praktischen Teil werden dokumentiert und anschließend mit den theoretischen Erkenntnissen abgeglichen, um aufzuzeigen,

inwieweit KI die Effizienz und Qualität im Entwicklungsprozess tatsächlich steigern kann.

Diese Methodik erlaubt es, die bestehende Forschung systematisch zu strukturieren und relevante Erkenntnisse für die Praxis abzuleiten.

1.4 Aufbau der Arbeit

Die Arbeit gliedert sich in folgende Kapitel:

- **Kapitel 1: Einleitung**
 - Darstellung von Hintergrund, Motivation, Zielsetzung, Forschungsfragen, methodischer Vorgehensweise und Abgrenzung.
- **Kapitel 2: Theoretische Grundlagen**
 - Definition und Funktionsweise von generativer KI in der Softwareentwicklung
 - Übersicht relevanter KI-Modelle, Algorithmen und Beispiele für KI-gestützte Entwicklungswerkzeuge
- **Kapitel 3: Praktische Demonstration**
 - Vorstellung des Projekts "Localsünd" dessen Architektur
 - Implementierung einer interaktiven Kartenansicht in der React Native-Anwendung mithilfe generativer KI-Technologien
 - Darstellung der Implementierungsschritte, Code-Beispiele und erste Evaluationsergebnisse
- **Kapitel 4: Chancen durch KI**
 - Effizienzsteigerung und Automatisierung
 - Neue Werkzeuge und Methoden
 - Verbesserte Code-Qualität und Fehlerminimierung
 - Einfluss von KI auf agile Entwicklungsmethoden
 - Bezugnahme auf die Erkenntnisse aus Kapitel 3
- **Kapitel 5: Herausforderungen durch KI**

- Sicherheits- und Datenschutzaspekte
 - Ethische Implikationen und Bias in KI-Modellen
 - Langfristige Auswirkungen auf Entwickler:innen-Rollen
 - Organisatorische und technologische Hürden
 - Risiken durch Abhängigkeit von KI-generiertem Code
 - Analyse der in Kapitel 3 möglicherweise aufgetretenen Herausforderungen und Problematiken
- **Kapitel 6: Wirtschaftliche und gesellschaftliche Auswirkungen**
 - Veränderungen in Softwareunternehmen
 - Auswirkungen auf den Arbeitsmarkt und Entwickler:innen-Rollen
 - Zukunftsperspektiven und strategische Empfehlungen
 - Kosten-Nutzen-Analyse von KI-gestützter Softwareentwicklung
 - **Kapitel 7: Fazit und Ausblick**
 - Zusammenfassung der theoretischen und praktischen Erkenntnisse
 - Diskussion offener Forschungsfragen
 - Ableitung von Handlungsempfehlungen und Ausblick auf zukünftige Entwicklungen

1.5 Abgrenzung

Die Arbeit konzentriert sich auf die theoretische Analyse der Chancen und Herausforderungen von KI in der Softwareentwicklung. Folgende Aspekte werden bewusst ausgeklammert:

- **Technische Implementierungen:** Es werden keine neuen KI-Modelle oder Algorithmen entwickelt.
- **Empirische Studien:** Die Arbeit basiert auf einer literaturgestützten Analyse und führt keine Befragungen oder Experimente durch.

- **Rechtliche Rahmenbedingungen:** Eine detaillierte Untersuchung rechtlicher oder regulatorischer Aspekte wird nicht vorgenommen.

Obwohl ein begrenzter praktischer Teil in Form einer Funktionsimplementierung gezeigt wird, dient dieser in erster Linie als Proof of Concept. Eine umfassende empirische Evaluierung oder die Entwicklung eigener KI-Modelle findet nicht statt.

2 Theoretische Grundlagen

2.1 Künstliche Intelligenz: Definitionen und Technologien

- Überblick über generative KI-Modelle
- Relevante Algorithmen und Methoden in der Softwareentwicklung

2.1.1 Beispiele für generative KI-Tools in der Praxis

2.1.2 Wichtige Algorithmen und Modelle in der Softwareentwicklung

2.2 Generative KI-Tools: Funktion und Anwendung

- Beispiele für generative KI-Tools in der Praxis
- Einsatzmöglichkeiten und Optimierungspotenziale

3 Praktische Demonstration

3.1 Zielsetzung und Vorgehen

3.1.1 Motivation für ein praktisches Beispiel

In diesem Kapitel wird die praktische Anwendung von KI-Tools in der Softwareentwicklung anhand der Implementierung eines Map-Screens bzw. einer interaktiven Kartenansicht für die App „Locals“ demonstriert. Die Wahl fiel auf dieses Beispiel, da es...

3.1.2 Eingesetzte KI-Tools

Die Implementierung erfolgt unter Verwendung verschiedener KI-gestützter Entwicklungstools:

- Cursor, GitHub Copilot, v0, ...

3.1.3 Technischer Aufbau

Der technische Stack basiert auf:

- React Native für die mobile Entwicklung
- Firebase als Backend-as-a-Service
- Expo für die Entwicklung von React Native-Anwendungen
- ...

3.2 Vorstellung der App „Locals“

3.2.1 Architektur und Aufbau

Die App „Locals“ ist eine mobile Anwendung, die Nutzer:innen dabei unterstützt, lokale Events zu entdecken, zu erstellen und zu verwalten. Sie richtet sich an ein junges, urbanes Publikum und zielt darauf ab, soziale Interaktionen rund um Veranstaltungen zu fördern.

Technologischer Stack und Architektur

- **Frontend:** Entwicklung mit React Native, TypeScript und Expo zur plattformübergreifenden Bereitstellung für iOS und Android.
- **Backend:** Nutzung von Firebase für Authentifizierung, Datenhaltung und Synchronisation.
- **Navigation:** Einsatz von `@react-navigation/native` und `expo-router` für ein modernes, tab-basiertes Navigationskonzept.
- **State-Management:** Eigene Context-Provider (`AuthProvider`, `EventsProvider`) für Authentifizierungs- und Eventdaten.
- **UI-Komponenten:** Verwendung von `@expo/vector-icons` und `lucide-react-native` zur Gestaltung eines ansprechenden User Interface.
- **Maps & Location:** Die Integration von `react-native-maps` und `expo-location` ermöglicht eine interaktive Kartenansicht, die sowohl den Nutzerstandort als auch Events auf einer Karte anzeigt.

Die App ist modular aufgebaut und umfasst drei zentrale Bereiche:

- **Explore-Screen:** Event-Feed nach Interessen und Standort
- **Map-Screen:** Interaktive Kartenansicht mit Event-Markern und Filterfunktion
- **Profil-Screen:** Persönliche Eventübersicht und Verwaltung

Während der Explore- und Profil-Screen bereits Grundfunktionen aufweisen, wird der Map-Screen im Rahmen dieser Arbeit als prototypisches Demonstrationsbeispiel gezielt entwickelt und evaluiert. Die praktische Realisierung dieses Features erfolgt unter Einsatz generativer KI-Tools und steht im Mittelpunkt der anschließenden Kapitel.

Struktur der Haupteinstiegskomponente (RootLayout)

Die zentrale Einstiegskomponente der App ist das `RootLayout`. Sie übernimmt das Laden benutzerdefinierter Fonts, die Einbindung von Authentifizierungs- und Events-Kontexten sowie die nahtlose Nutzerführung nach dem Login. Die Navigation wird dabei strikt vom Authentifizierungsstatus gesteuert, sodass nicht eingeloggte Nutzer:innen automatisch zur Login-Ansicht weitergeleitet werden. Dies wird mit Hilfe der Hooks `useAuth`, `useSegments` und `useRouter` umgesetzt.

3.2.2 Bestehende Funktionalitäten

Zu den bereits implementierten Basisfunktionen von Locals gehören:

- **Benutzerauthentifizierung:** Sichere Registrierung und Anmeldung über Firebase Authentication.
- **Profilverwaltung:** Verwaltung persönlicher Daten sowie Übersicht über besuchte und selbst erstellte Events.
- **Eventverwaltung:** Anlegen, Bearbeiten und Löschen von Events.
- **Tab-Navigation:** Ermöglicht den nahtlosen Wechsel zwischen den drei Hauptbereichen „Explore“, „Map“ und „Profil“.
- **Responsives Design:** Durch Nutzung von `react-native-safe-area-context` und Expo UI-Komponenten wird eine konsistente Darstellung auf unterschiedlichen Geräten gewährleistet.

Der Map-Screen ist als zentrales, innovatives Feature der App konzipiert. Die Implementierung und Weiterentwicklung dieses Moduls wird im weiteren Verlauf dieser Arbeit als praktisches Beispiel für den Einsatz generativer KI in der Softwareentwicklung demonstriert und analysiert.

3.3 Implementierung der interaktiven Kartenansicht mit KI-Unterstützung

3.3.1 Integration der KI-Tools

Die Einrichtung der Entwicklungsumgebung umfasst...

3.3.2 Entwicklungsprozess mit KI-Unterstützung

[Code-Beispiele und Gegenüberstellungen...]

3.4 Erste Evaluierung

3.4.1 Erfolge und Herausforderungen

Die praktische Implementierung hat gezeigt...

3.4.2 Qualitative Bewertung

Die Analyse der Entwicklung fokussiert sich auf:

- Zeiteffizienz
- Code-Qualität
- Wartbarkeit

3.5 Zwischenfazit

Die praktische Demonstration hat wichtige Erkenntnisse für die weitere Analyse geliefert:

- Kernerkenntnisse...
- Überleitung zu folgenden Kapiteln...

4 Chancen

Der Einsatz von KI in der Softwareentwicklung bietet eine Vielzahl an Vorteilen. Besonders hervorzuheben sind Effizienzsteigerungen durch Automatisierung, die Entwickler:innen von repetitiven Aufgaben entlasten und ihnen mehr Zeit für kreative und konzeptionelle Arbeit geben. Dieses Kapitel untersucht die wichtigsten Potenziale, die KI-Technologien für den Softwareentwicklungsprozess mit sich bringen.

Im praktischen Teil dieser Arbeit wird zudem untersucht, wie sich durch KI-Tools Entwicklungsaufgaben für eine interaktive Kartenansicht automatisieren lassen und welche konkreten Effizienzsteigerungen hier auftreten können.

4.1 Effizienzsteigerung und Automatisierung

- Automatisierung von Entwicklungsprozessen
- Optimierung der Kollaboration durch KI
- Verbesserung der Codequalität

4.2 Neue Werkzeuge und Methoden

- Innovative Ansätze für die Softwareentwicklung

5 Herausforderungen durch KI in der Softwareentwicklung

Trotz der vielversprechenden Möglichkeiten von KI-gestützten Entwicklungsmethoden existieren Herausforderungen, die nicht vernachlässigt werden dürfen. Besonders Datenschutz- und Sicherheitsaspekte spielen eine zentrale Rolle, ebenso wie ethische Fragen zur Fairness und Transparenz von KI-Modellen. In diesem Kapitel werden die wesentlichen Problembereiche diskutiert, die mit der zunehmenden Integration von KI in die Softwareentwicklung verbunden sind.

5.1 Sicherheits- und Datenschutzaspekte

5.1.1 Sicherheitsrisiken durch generative Modelle

5.2 Ethische und soziale Implikationen

5.2.1 Ethische Konflikte und Bias in KI-Systemen

5.2.2 Langfristige Auswirkungen auf Entwickler:innen-Rollen

5.2.3 Technische und organisatorische Hürden bei der Einführung von KI

6 Wirtschaftliche und gesellschaftliche Auswirkungen

6.1 Veränderungen in Softwareunternehmen

- Auswirkungen auf Geschäftsmodelle und Prozesse
- Veränderungen in der Softwareentwicklung und im Projektmanagement
- Rolle von KI bei der Automatisierung von Softwareentwicklungsaufgaben

6.2 Auswirkungen auf den Arbeitsmarkt und Entwickler:innen-Rollen

- Verschiebung der gefragten Kompetenzen und Qualifikationen
- Neue Berufsbilder und veränderte Karrierewege
- Auswirkungen auf die Arbeitsplatzsicherheit und die Notwendigkeit der Weiterbildung

6.3 Zukunftsperspektiven und strategische Empfehlungen

- Notwendige Anpassungen für Unternehmen und Entwickler
- Möglichkeiten der Integration von KI in bestehende Entwicklungsprozesse
- Regulatorische und ethische Implikationen für eine nachhaltige KI-Nutzung

6.4 Kosten-Nutzen-Analyse von KI-gestützter Softwareentwicklung

- Analyse der wirtschaftlichen Effizienz und Kostenersparnis

- Vergleich der Investitionskosten und erwarteten Produktivitätsgewinne
- Langfristige wirtschaftliche Auswirkungen für Unternehmen und die Softwarebranche

7 Fazit und Ausblick

7.1 Erwartete Erkenntnisse

Es wird erwartet, dass KI-gestützte Softwareentwicklung nicht nur Effizienzsteigerungen ermöglicht, sondern auch die Arbeitsweise von Entwicklern nachhaltig verändert. Besonders relevant ist die Frage, inwieweit generative KI langfristig klassische Programmieraufgaben übernimmt oder ergänzt. Darüber hinaus soll analysiert werden, welche Herausforderungen in Bezug auf Sicherheit, Ethik und wirtschaftliche Auswirkungen entstehen.

7.2 Zusammenfassung der Erkenntnisse

7.3 Handlungsempfehlungen und Zukunftsperspektiven

Literaturverzeichnis

- [A.24] A., Downie und M., Finio: KI in der Softwareentwicklung (2024), URL <https://www.ibm.com/de-de/think/topics/ai-in-software-development>, iBM analysiert den Einfluss von KI auf die Softwareentwicklung und hebt hervor, wie KI-Technologien die Produktivität, Genauigkeit und Innovation steigern können. Der Artikel betont die Rolle von generativer KI und großen Sprachmodellen bei der Optimierung des Entwicklungszyklus.[Zugegriffen: 16. Januar 2025]
- [Bra24] BRAUN, A. M.: KI in der Softwareentwicklung: Wie effektiv codet KI wirklich? (2024), URL <https://www.computerwoche.de/article/2832991/wie-effektiv-codet-ki-wirklich.html>, dieser Artikel untersucht die Effizienz von KI in der Softwareentwicklung und diskutiert, inwieweit KI-gestützte Tools den Programmierprozess unterstützen und verbessern können.[Zugegriffen: 18. Januar 2025]
- [Cou24] COUTINHO, Mariana; MARQUES, Lorena; SANTOS, Anderson; DAHIA, Marcio; FRANCA, Cesar und DE SOUZA SANTOS, Ronnie: The Role of Generative AI in Software Development Productivity: A Pilot Case Study (2024), URL <https://arxiv.org/abs/2406.00560>, diese Fallstudie untersucht den Einfluss generativer KI auf die Produktivität in der Softwareentwicklung und liefert erste empirische Erkenntnisse.
- [Cui24] CUI, Zheyuan (Kevin); DEMIRER, Mert; JAFFE, Sonia; MUSOLFF, Leon; PENG, Sida und SALZ, Tobias: The effects of Generative AI on high skilled work: Evidence from three field experiments with software developers (2024), URL https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4945566, die Autoren präsentieren Ergebnisse aus drei Feldexperimenten, die den Einfluss generativer KI auf die Arbeit von hochqualifizierten Softwareentwicklern untersuchen.
- [Don24] DONVIR, Anujkumarsinh; PANYAM, Sriram; PALIWAL, Gunjan und GUJAR, Praveen: The Role of Generative AI Tools in Application Development: A Comprehensive Review of Current Technologies and Practices, in: *2024 International Conference on Engineering Management of Communication and Technology (EMCTECH)*, URL <https://ieeexplore.ieee.org/document/10741797>, dieses Review bietet einen umfassenden Überblick über aktuelle generative KI-Tools in der Anwendungsentwicklung und

diskutiert deren Technologien und Praktiken.

- [J23] J, Bhuvana; RANJAN, Vivek und BHADUARIYA, Nirmednra: Integration of AI in the Realm of Software Development, in: *2023 International Conference on Advances in Computation, Communication and Information Technology (ICAICCIT)*, URL <https://ieeexplore.ieee.org/document/10465893>, die Studie analysiert die Integration von KI in die Softwareentwicklung und identifiziert Vorteile, Herausforderungen und Best Practices für Entwickler und Organisationen.
- [Mar24] MARTINOVIĆ, Boris und ROZIĆ, Robert: Impact of AI Tools on Software Development Code Quality, in: Tomislav Volarić; Boris Crnokić und Daniel Vasić (Herausgeber) *Digital Transformation in Education and Artificial Intelligence Application*, Springer Nature Switzerland, Cham, URL https://link.springer.com/chapter/10.1007/978-3-031-62058-4_15, die Autoren analysieren die Auswirkungen von KI-Tools auf die Codequalität in der Softwareentwicklung und identifizieren sowohl Vorteile als auch potenzielle Risiken.
- [Mat21] MATSUMOTO, Kenichi: Conceptual Framework for Next-Generation Software Ecosystems, in: *2021 IEEE/ACIS 22nd International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD)*, URL <https://ieeexplore.ieee.org/document/9705010>, dieses Papier präsentiert ein konzeptionelles Framework für zukünftige Software-Ökosysteme und untersucht, wie KI-Technologien in diese integriert werden können, um Effizienz und Innovation zu fördern.
- [ND23] NGUYEN-DUC, Anh; CABRERO-DANIEL, Beatriz; PRZYBYLEK, Adam; ARORA, Chetan; KHANNA, Dron; HERDA, Tomas; RAFIQ, Usman; MELEGATI, Jorge; GUERRA, Eduardo; KEMELL, Kai-Kristian; SAARI, Mika; ZHANG, Zheyang; LE, Huy; QUAN, Tho und ABRAHAMSSON, Pekka: Generative Artificial Intelligence for Software Engineering – A Research Agenda (2023), URL <https://arxiv.org/abs/2310.18648>, dieses Papier skizziert eine Forschungsagenda für den Einsatz generativer KI im Software Engineering und identifiziert offene Forschungsfragen in verschiedenen Bereichen.
- [S.24] S., Sulabh: The Future of Coding is Here – How AI is Reshaping Software Development (2024), URL <https://www.deloitte.com/uk/en/Industries/technology/blogs/2024/the-future-of-coding-is-here-how-ai-is-reshaping-software-development.html>, deloitte untersucht, wie KI die Softwareentwicklung transformiert, und diskutiert die Auswirkungen auf Entwickler, Prozesse und die gesamte Branche. Der Artikel betont die Bedeutung von KI für die zukünftige Gestaltung der Softwareentwicklung.[Zugegriffen: 15. Januar 2025]
- [Sch24] SCHMITT, Anuschka; GAJOS, Krzysztof Z. und MOKRYN, Osnat: Generative AI in the Software Engineering Domain: Tensions of Occupational Identity

- and Patterns of Identity Protection (2024), URL <https://arxiv.org/abs/2410.03571>, die Studie untersucht, wie generative KI die berufliche Identität von Softwareingenieuren beeinflusst und welche Strategien zum Schutz dieser Identität angewendet werden.
- [Shi23] SHI, Yong; SAKIB, Nazmus; SHAHRIAR, Hossain; LO, Dan; CHI, Hongmei und QIAN, Kai: AI-Assisted Security: A Step towards Reimagining Software Development for a Safer Future, in: *2023 IEEE 47th Annual Computers, Software, and Applications Conference (COMPSAC)*, URL <https://ieeexplore.ieee.org/document/10196879>, die Autoren diskutieren, wie KI-gestützte Sicherheitslösungen die Softwareentwicklung sicherer gestalten können, indem sie Bedrohungen frühzeitig erkennen und proaktive Maßnahmen ermöglichen.
- [Sie24] SIEBERT, Dr. J. und JEDLITSCHKA, Dr. Andreas: Generative KI im Software Engineering: Szenarien und künftige Entwicklungen (2024), URL <https://www.iese.fraunhofer.de/blog/generative-ki-softwareentwicklung>, das Fraunhofer IESE beleuchtet verschiedene Anwendungsszenarien für generative KI im Software Engineering und diskutiert zukünftige Entwicklungen sowie potenzielle Herausforderungen in diesem Bereich.[Zugegriffen: 15. Januar 2025]
- [Wan18] WANGOO, Divanshi Priyadarshni: Artificial Intelligence Techniques in Software Engineering for Automated Software Reuse and Design, in: *2018 4th International Conference on Computing Communication and Automation (ICCCA)*, URL <https://ieeexplore.ieee.org/document/8777584>, die Autoren erforschen, wie KI-Techniken die automatisierte Wiederverwendung und das Design von Software unterstützen können, um Entwicklungszeiten zu verkürzen und die Qualität zu verbessern.
- [Wei24] WEISZ, Justin D.; HE, Jessica; MULLER, Michael; HOEFER, Gabriela; MILES, Rachel und GEYER, Werner: Design Principles for Generative AI Applications, in: *Proceedings of the CHI Conference on Human Factors in Computing Systems*, CHI '24, ACM, URL <http://dx.doi.org/10.1145/3613904.3642466>, in diesem Papier werden sechs Designprinzipien für generative KI-Anwendungen vorgestellt, die einzigartige Herausforderungen adressieren und Empfehlungen für die Gestaltung effektiver und sicherer Nutzererfahrungen bieten.

Abbildungsverzeichnis

Tabellenverzeichnis

Listings

.

A Anhang 1

B Anhang 2