

Forschungsprojekt
Auswirkungen des KI-Einsatzes innerhalb der Aufgaben
des Software-Engineerings

Impuls im Rahmen des Doktorandentags am 24. Juni 2025

Prof. Dr.-Ing. habil. Andreas Schmietendorf

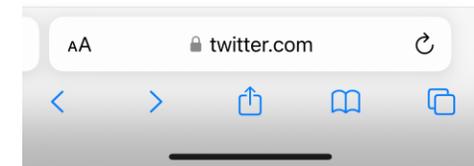
HWR Berlin/FB2 sowie Otto-von-Guericke Universität Magdeburg/FIN

Motivation und Einordnung

Quo Vadis – Programmiersprachen?

“The hottest new programming language is English”

Zitat von Andrej Karpathy (einer der Gründer von OpenAI – Anbieter von ChatGPT) auf ehemals Twitter, Januar 2023



Quelle der rechten Abb.: Karpathy, A. on X (ehm. Twitter) – siehe auch <https://karpathy.ai>, Jan. 2023

Ziele des Projekts

- Erfassung des wissenschaftlichen Sachstands zu den mit KI einhergehenden Veränderungen beim Software-Engineering.
- Empirische Analysen zu den Erwartungshaltungen innerhalb der Software Engineerings Community.
- Prototypische Tests zur Gewinnung eigener Erfahrungen (Aufgaben, Werkzeuge und APIs – u.a. Software Reengineering).
- Herausforderungen für die Ausbildung (Schwerpunkte der Veränderungen).



rechte Abb.: · KI-basiert erzeugt mit Hilfe von Microsoft Copilot, Prompt: Erzeuge ein Bild für Projektziele, 19.05.2025

Fokus - Effizienz der Softwareentwicklung

- KI-basiert erzeugbare Artefakte:
 - Use-Cases & User-Stories
 - Architektur- und Design-Pattern
 - Quellcode „Snippets“
 - Quellcode-Transformationen
 - Semantische Quellcode-Suche
 - Testfälle und Testdaten
 - Quellcode-Analyse (Reviews)
 - ...
- Herausforderungen:
 - Testen und Bewerten der Ergebnisse
 - Gewährleistung Compliance
 - Risikomanagement und Vertrauen
 - ...

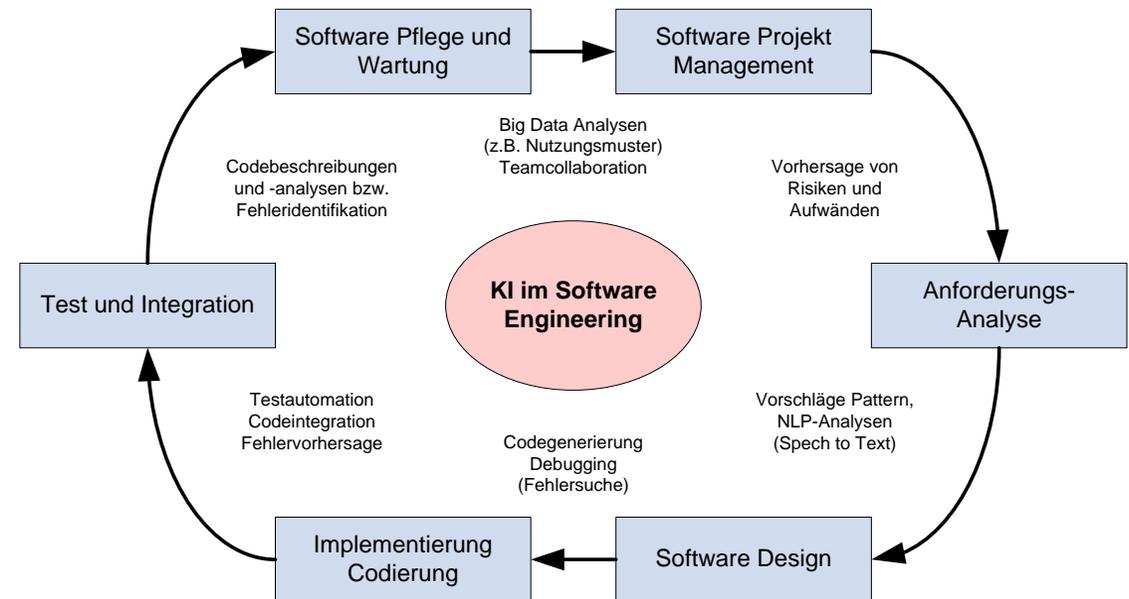


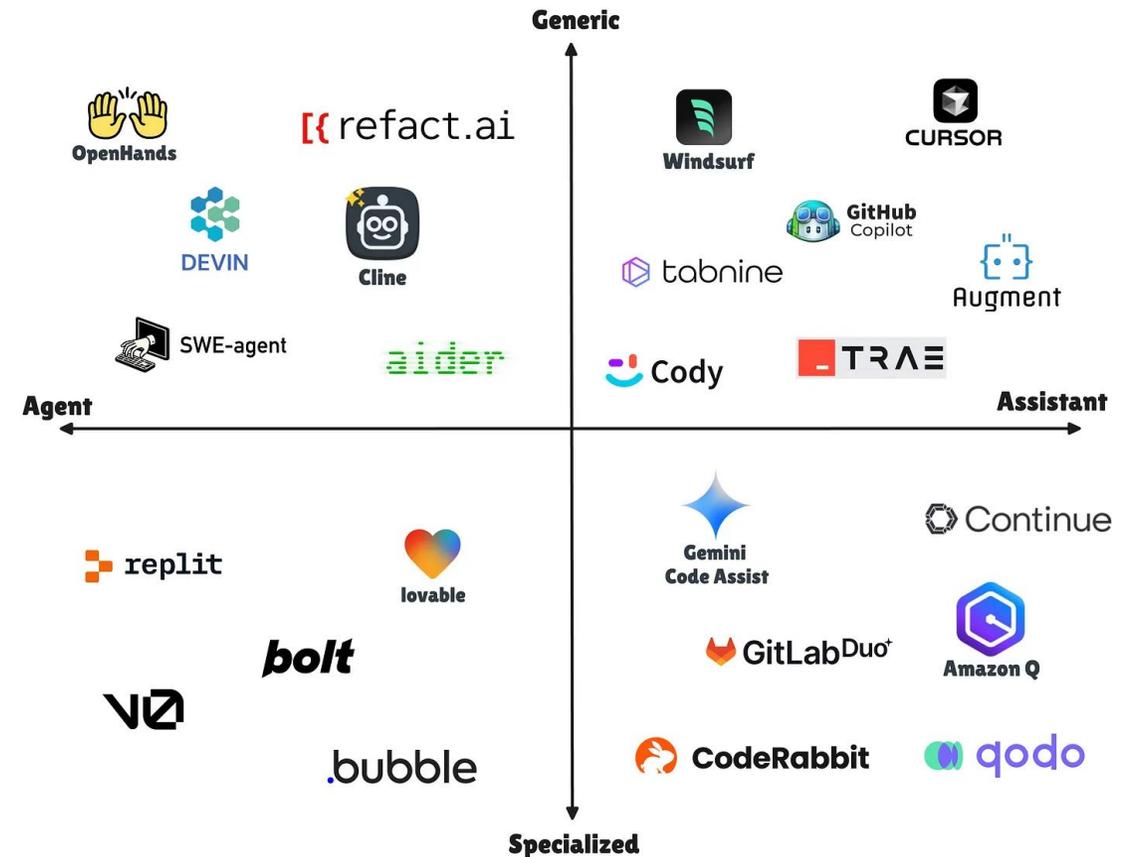
Abb. In Anlehnung an.: Barenkamp, M.; Rebstadt, J.; Thomas, O. (2020): Applications of AI in classical software engineering. In: AI Perspect 2 (1), S. 1–15. DOI: 10.1186/s42467-020-00005-4, letzter Abruf Feb. 2023

Werkzeuge in der Codierung

- Generisch vs. Spezialisiert
- Agentenbasiert vs. Assistenz



<https://notebooklm.google.com/notebook/b5e474fd-7d0e-42ce-940d-f9744a5a7607/audio>



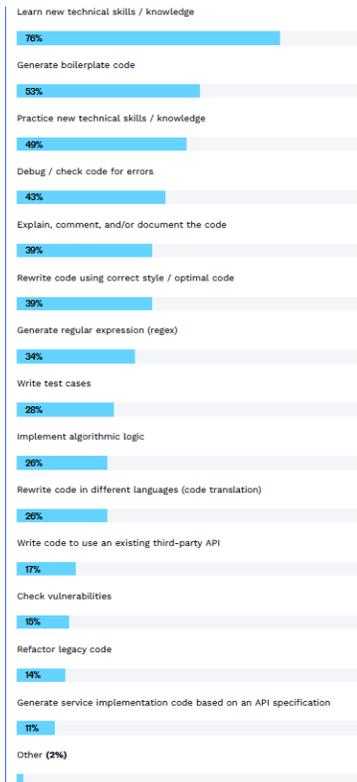
Quelle der rechten Grafik: <https://generativeprogrammer.com/p/ai-coding-assistants-landscape>

Empirische Analysen zum Nutzungsverhalten

How developers use AI coding assistants

81% of developers surveyed say they use AI-powered coding assistants. That's nearly twice the percentage of developers who said they used AI tools in Stack Overflow's 2023 Developer Survey, conducted 6 months earlier.

Here's how they use them:



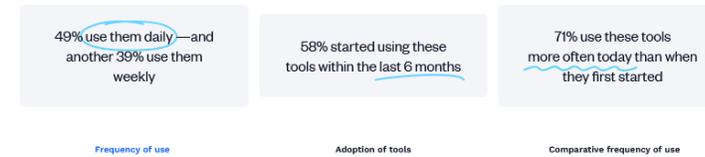
Key takeaways: The most common use of AI coding assistant tools isn't actually for writing, commenting, or debugging code: it's for learning. Over three-quarters of developers who use AI-powered coding assistants use them to learn new skills or knowledge. This stands in contrast to early studies of these tools from Stack Overflow, GitHub, and Google, which focus on the benefits for developers' productivity.



Link zu den Ergebnissen der Umfrage

Adoption and frequency of AI coding assistant use

Among developers who say they use an AI coding assistant:



Question: How often do you use these tools?



What makes developers nervous about AI

We also wanted to know what reservations developers have about AI uses in software engineering.

Here's what they said:



Key takeaways: The biggest concern of developers surveyed was the quality of AI-generated code. This aligns with Stack Overflow's finding, where only 2.85% of developers surveyed "highly trusted" the accuracy of AI output.

Quelle der Umfrage bzw. Abbildungen: <https://codesignal.com/report-developers-and-ai-coding-assistant-trends> (Umfrage unter 1000 Entwicklern)

Sicherheit und Vertrauen

Chancen – Generative KI-Modelle (genKI_M)

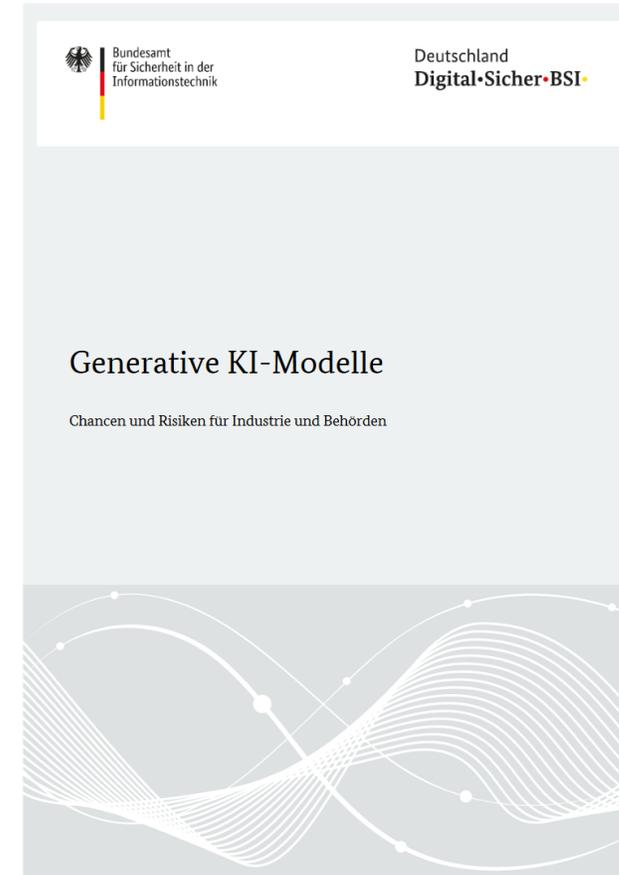
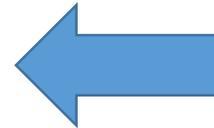
- Massive Effizienzvorteile
- ...

Risiken - genKI_M

- Risiken bei der ordnungsgemäßen Nutzung von generativen KI-Modellen (R1 – R9),
- Risiken durch eine missbräuchliche Nutzung von generativen KI-Modellen (R10 – R16),
- Risiken infolge von Angriffen auf generative KI-Modelle (R17 - R28)

Gegenmaßnahmen - genKI_M

- Auswahl des Modells und Betreiber
- Erklärbarkeit XAI
- **Steigerung der Robustheit**
- ...



Quelle: Generative KI-Modelle, Version 2, BSI, 2025
https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/KI/Generative_KI-Modelle.pdf?blob=publicationFile&v=7

Robustheit – Stabilität des KI-Modells bei veränderten Rahmenbedingungen (KI-Vertrauen)

- Ungenaue, unvollständige oder inkonsistente Eingabedaten.
- Veränderte Umgebungsbedingungen (z.B. Jahreszeit/Lichtverhältnisse).
- Angemessener Umgang mit vielfältigen Benutzerkontexten.
- Umgang mit Störungen bzw. expliziten Angriffen.
- Robustheit der KI mit Hilfe von Fehlerinjektion untersuchen.
 - Schwachstellen identifizieren
 - Beispiel: Forschung am Fraunhofer IESE:
 - Siebert, J.: Robuste KI-Systeme entwickeln mit dem Badgers Bad Data Generator, Workshop in Anlehnung an die Themen des IFAF-Forschungsprojekts TAHAI
 - <https://www.iese.fraunhofer.de/blog/badgers-open-source-python-bibliothek>
- Test der Robustheit bei Quellcodetransformationen
 - Masterarbeit: Martin Weiss (Uni MD)
 - Framework und Testergebnisse
 - Partner: Robert Bosch GmbH

Weiterführende Information

Öffentlicher Workshop am Fraunhofer IESE

Organisation:

Dr. Andreas Jedlitschka, Fraunhofer IESE

Prof. Dr. Jens Heidrich, HS Mainz

Prof. Dr. Andreas Schmietendorf, HWR Berlin

Unterstützt durch:

GI-Fachgruppe Measurement & Data Science

Verband der Central Europe Measurement Group

Ort:

Fraunhofer IESE Kaiserlautern

1-tägiger Workshop:

05. November 2025

Potentielle Themen:

KI-getriebenen Veränderungen im Software Engineering

- Ergebnisse empirischer Analysen – Z.B. Erwartungen
- Fallstudien zum Einsatz von KI-Tools
- Durch KI unterstützte Entwicklungsaufgaben
- Mögliche Risiken (z.B. beim Vibe Coding)
- KI-Demokratisierung vs. Vendor „Lock In“
- Test und Bewertung von „KI-Ergebnissen“
- KI im DevOps-Diskurs
- Umgang mit Compliance-Anforderungen
- Veränderte Skill-Bedürfnisse
- ...

Hochschule für
Wirtschaft und Recht Berlin
Berlin School of Economics and Law
Berliner Schriften zu modernen Integrationsarchitekturen
Hrsg.: Prof. Dr.-Ing. habil. Andreas Schmietendorf

Hochschule für Wirtschaft und Recht Berlin
Fachbereich II
Wirtschaftsinformatik – Systementwicklung

Andreas Schmietendorf, Erik Rodner,
Ralf Schnieders (Hrsg.)

Herausforderungen Low-Code orientierter KI-Ansätze
Ergebnisse öffentlicher Expertenworkshops am Fraunhofer
IESE und der HTW Berlin unter Berücksichtigung
der TAHAI-Projektresultate

SHAKER
PUBLISHING

Aktuelle Buchpublikation

- Agile & fachgetriebene Entwicklung
- Generischer Funktionsumfang
- Auswahlkriterien für LCPs
- Analyse realer LCPs
 - Outsystems
 - KNIME
 - ...
- KI und Low-Code

NEUERSCHEINUNG



Seit Jahrzehnten wird versucht, den Einsatz von Quellcodes zu reduzieren. Vor zirka 10 Jahren wurde hierfür der Low-Code-Begriff geprägt. Im Zusammenhang mit den Erfordernissen einer allgegenwärtigen Digitalisierung versprechen Low-Code basierte Entwicklungen eine agilere-, kompositorische-, visuell orientierte- und vor allem fachgetriebene Softwareentwicklung. Auch der Bedarf an klassischen Softwareentwicklern sollte damit reduziert werden können.

In der vorliegenden Monografie setzen sich Andreas Schmietendorf und Michael Knuth mit existierenden Arbeiten im Diskurs eines Low-Code orientierten Software Engineerings, mit der Auswahl zur Entwicklung benötigter Low-Code-Plattformen, aber auch mit den funktionalen Eigenschaften exemplarisch analysierter Low-Code Produkte auseinander. Darüber hinaus werden die aktuellen Möglichkeiten generativer KI-Ansätze im Diskurs von Low-Code basierten Softwareentwicklungen aufgezeigt.

Aspekte des Software Engineerings im Diskurs einer Low-Code orientierten Softwareentwicklung

Andreas Schmietendorf, Michael Knuth

2024, 124 Seiten
ISBN 978-3-8325-5733-1
Preis: 36.00 €



→ Bestellung über den Buchhandel oder unsere Webseite
www.logos-verlag.de