

(K)einen Mehrwert durch Agentic Coding?

Wo Agentic AI im Software Development Lifecycle wirklich hilft (und wo nicht)

– *Julia Kordick*



**Who here is actively using
agentic AI tools in their day-to-
day work?**

**Who is responsible for deciding
which AI tools your team uses?**

**Who is deeply sceptical of the
agentic AI hype?**

**Who has been burned by
agentic AI doing something
unexpected?**

Who has tried writing detailed specs or work orders before handing tasks to an AI agent?

hey ho, let's go into vs code :)

Agentic AI Foundation

- Shared, written engineering standards
- Spec and documentation quality (prerequisite, not afterthought)
- Guardrails: deterministic vs. non-deterministic tasks; documented boundaries
- Tool selection and enablement: tool literacy + judgment
- Testing and code review: same standards for AI and human code
- Security and compliance by design
- Observability and auditability
- Ownership and accountability model

~~Agentic AI~~ DevEx + Platform Engineering Foundation

- Platform engineering and DevEx as central enablers, not support functions
- Human validation gates before AI output reaches production
- Define, measure, and communicate success metrics (including costs)
- Feedback loop: revisit strategy, tools, and guardrails as you learn
- Psychological safety to experiment without fear
- Technical debt management: AI accumulates it faster

Open questions

- The senior bottleneck: more output, same review capacity
- The junior engineer path: does the learning model still hold?
- The communication gap: leadership decisions, engineering consequences



Julia Kordick ☁️ ✓

👤 sr software engineer | cloud solution architect | tech yuppie | feminist - sr sw gbb @ microsoft

Cologne, North Rhine-Westphalia, Germany · [Contact info](#)

5,355 followers · 500+ connections



Microsoft



university of applied sciences and arts hanover

Open to

Add section

Add custom button



Thanks!