

Fixing Tools Not Problems

AI Helping Us Right Now Examples Part 1

Part 5 of 7: Requirements Design Coding

AI Helping Us Now

Not Sci Fi Tools Are Here Getting Better

- ⚙️ This isn't just future gazing.
- ⚙️ Tools using AI show glimpses of this future **today**.

Use Case 1 Clearer Requirements Faster

Tackling Ambiguity Early

- ⚙️ **Now** Tools transcribe/summarize meetings (Otter Fireflies MS Teams).
- ⚙️ Project tools (Jira ADO) experiment AI drafting stories suggesting tasks.
- ⚙️ AI used for drafting high quality Epics Stories **estimation hints**.

Requirements Future Vision

Active Listening Alignment Checks

- ⚙️ AI actively listens flags confusion drafts **well-formatted Stories (BDD style)**.
- ⚙️ Suggests Acceptance Criteria checks alignment with **OKRs**.
- ⚙️ Cuts down early **confusion and rework**.

Use Case 2 Faster Design Prototypes

Avoiding Complex Designs Wasted Effort

⚙️ **Now** Tools generate UI mockups from text/sketches (Uizard Miro Galileo AI).

⚙️ AI in diagramming tools helps organize ideas (Miro Lucidchart).

Design Future Vision

Conversational Architecture Instant Feedback

- ⚙️ Architects talk -> AI draws initial diagrams suggests **API structures (OpenAPI)**.
- ⚙️ Builds **quick clickable prototype** for instant feedback.
- ⚙️ Helps see problems early **avoid over-engineering**.

Use Case 3 Coding Supercharged

Automating Routine Code Reducing Toil

- ⚙️ Tired of writing boring code?
- ⚙️ **Now** GitHub Copilot suggests code sometimes whole functions.
- ⚙️ Tools use AI to analyze quality find bugs better (SonarQube).
- ⚙️ Low-Code/No-Code generates apps visually (Mendix Lovable OutSystems).

⚙️ AI Code Review platforms popular
(CodeAnt AI).

Coding Future Vision

AI as Partner Fixing Debt Explaining

- ⚙️ AI writes most routine code **unit tests (TDD style!)**.
- ⚙️ Helps fix old messy code (addressing **technical debt**).
- ⚙️ Explains complex parts.

Developer Focus Shift

From Typing Mundane Code To Guiding AI

- ⚙️ Developers focus on **tricky business logic** guiding the AI.

- ⚙️ Massively speeds up development cycle.

Impact Summary Pt 1

Speed Clarity Reduced Waste

⚙️ AI driving significant gains in early SDLC stages.

⚙️ Faster idea-to-validation reduced ambiguity/rework.

Food For Thought

Practical Benefits Today

⚙️ These tools are offering real productivity advantages now.

⚙️ Not just hype tangible improvements are possible.

Next

Testing Ops Role Changes

⚙️ Next we cover AI's impact on later SDLC stages.

⚙️ Smarter Testing Easier Operations and the evolution of our roles.

Series Index

Part 1: Remember How We Got Here

Access Part 1 PDF

Part 2: The Tool Treadmill Steps 1-3

Access Part 2 PDF

Part 3: The Tool Treadmill Steps 4-7 & Pattern

Access Part 3 PDF

Part 4: How AI Can Be Different

Access Part 4 PDF

Part 5: AI Helping Us Now Requirements Design Coding
(Current)

Access Part 5 PDF

Part 6: AI Helping Us Now Testing Ops Roles

Access Part 6 PDF

Part 7: The Big Win Getting Off the Treadmill

Access Part 7 PDF

Read the Full Article: Tired of Fixing Tools Instead of Solving Problems?...

All resources mentioned are available at **<https://agilp.org/pdf/>**

[Read the Full Article on LinkedIn](#)

Connect & Engage

LinkedIn: <https://www.linkedin.com/in/amitabhrjha/>



X (Twitter): <https://x.com/amitabhrjha>



Web: www.agilp.org



Disclaimer & Acknowledgments

The opinions expressed are my own & don't necessarily represent my employer's views. My perspective is constantly evolving, shaped by invaluable interactions with friends, colleagues, mentors, insightful authors, and industry influencers - thank you all! Much of this content, including these carousels, is co-created with AI co-pilots like ChatGPT, Gemini, and Grok. My intent is to synthesize knowledge and share it back with the community.