

Prototype & Discovery Assistant with Claude Code

Validate with **Stakeholders, Clients & Users** at **AI Speed**

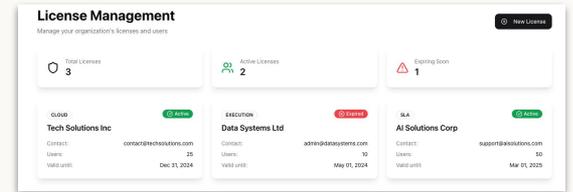
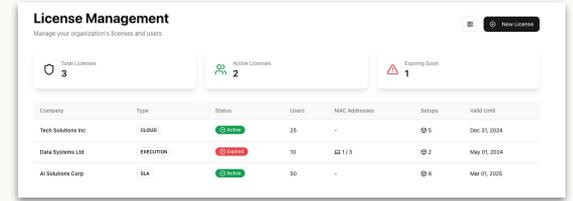
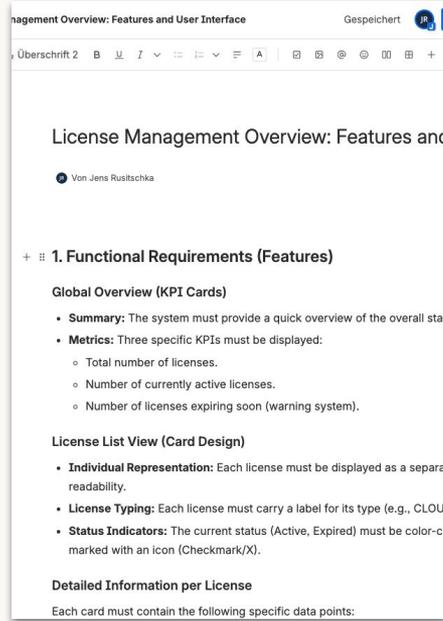
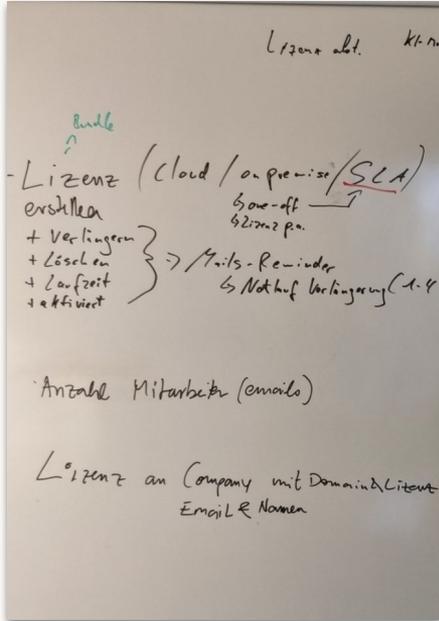
Jens Rusitschka

Claude Code Meetup Munich 26. Januar 2026

The first time I stopped building it myself.

Chatbot

AI coded



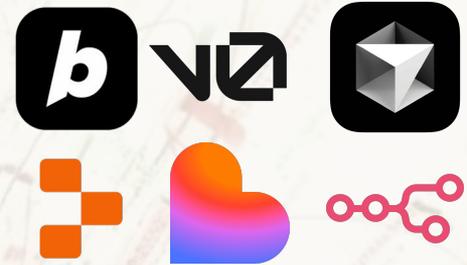
And things quickly grew beyond my control.



Chatbots



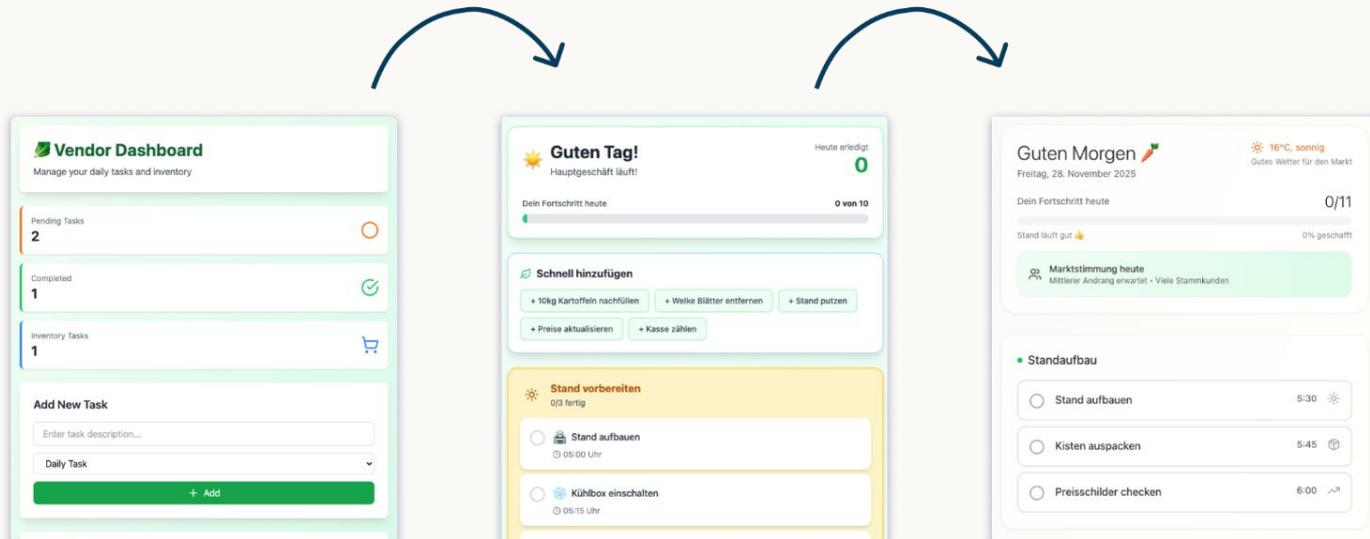
Project Data, Prompts



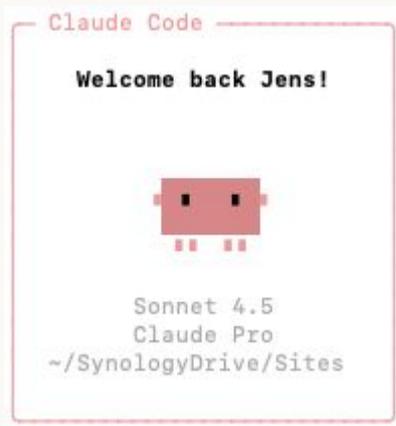
Prototypes

I needed a system....

- ...that **generates** variants with **AI**.
- ...where solutions **build on each other**.
- ...that keeps **results in one place**.



From the cloud back to the terminal.



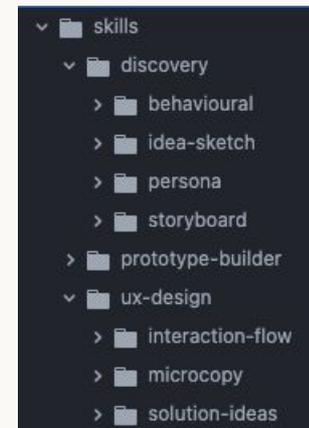
AI Agent

Terminal or IDE



Markdown

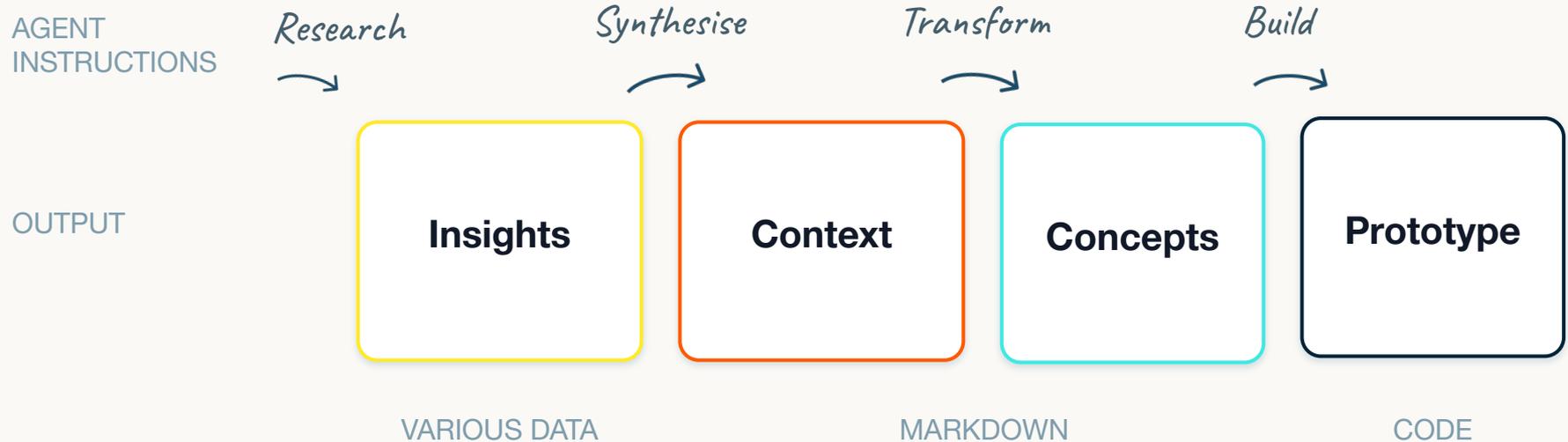
File-based data access



Skills

Prompt library

The system: From insights to prototype



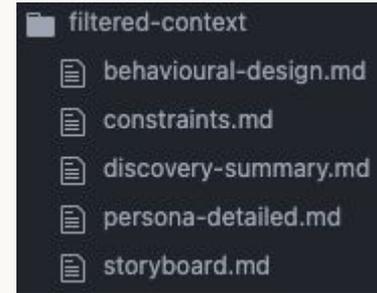
No fixed order: **You can start at any step!**

Markdown Step 1:

Create Structured Context



Synthesise

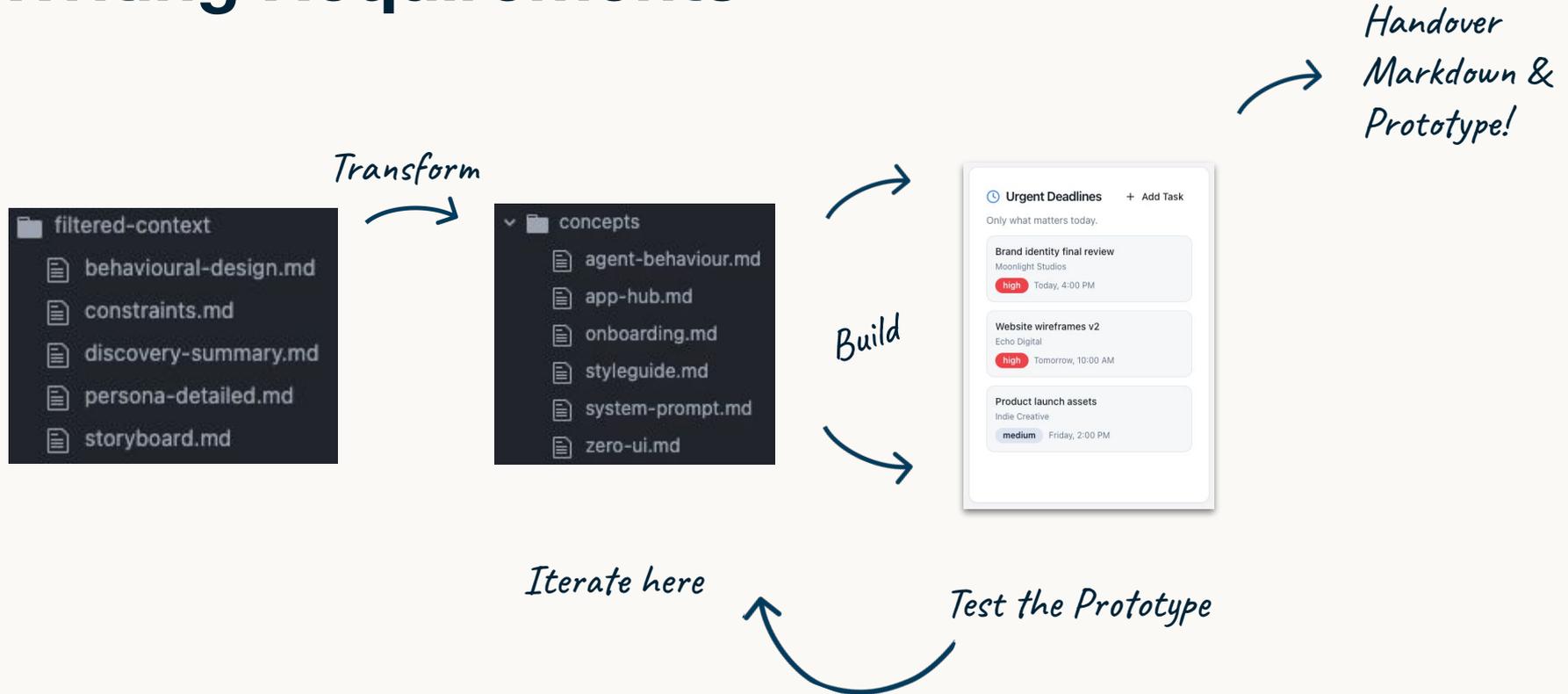


*Human in the Loop:
Time for Review*

Goal: Reduce Slop and Strengthen Focus

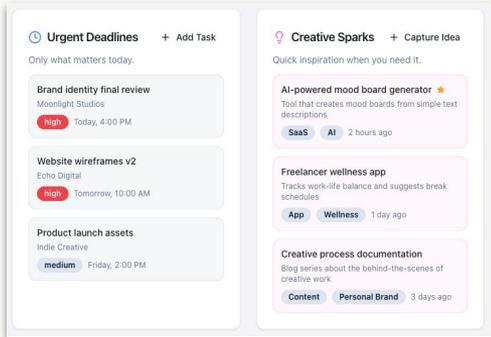
Markdown Step 2:

Writing Requirements

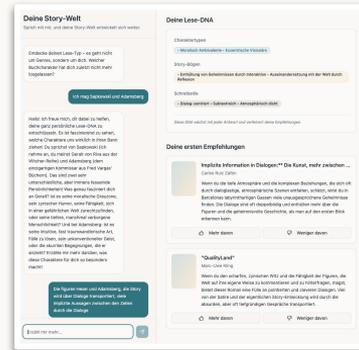


The leverage: Any interface is possible.

Structured Requirements
Markdown, JSON, YAML, ...



Graphical UI
+ AI Features



Agentic Interface

“Zero UI”



A2UI
or Voice, AR/VR, ...

Lessons Learned

- 1 **Markdown** replaces mockups
- 2 **Requirements**-based vibe coding
- 3 **Optimize prompts** in chat, then **save them as skills**
- 4 Prototype \neq production (**builder's trap**)
- 5 **Human in the loop** and reviews



**Jens
Rusitschka**



**Discovery
& AI Workflows**

