

Frontend in the age of AI

Erik Hanchett · Frontend Nation



The Shift to AI Coding

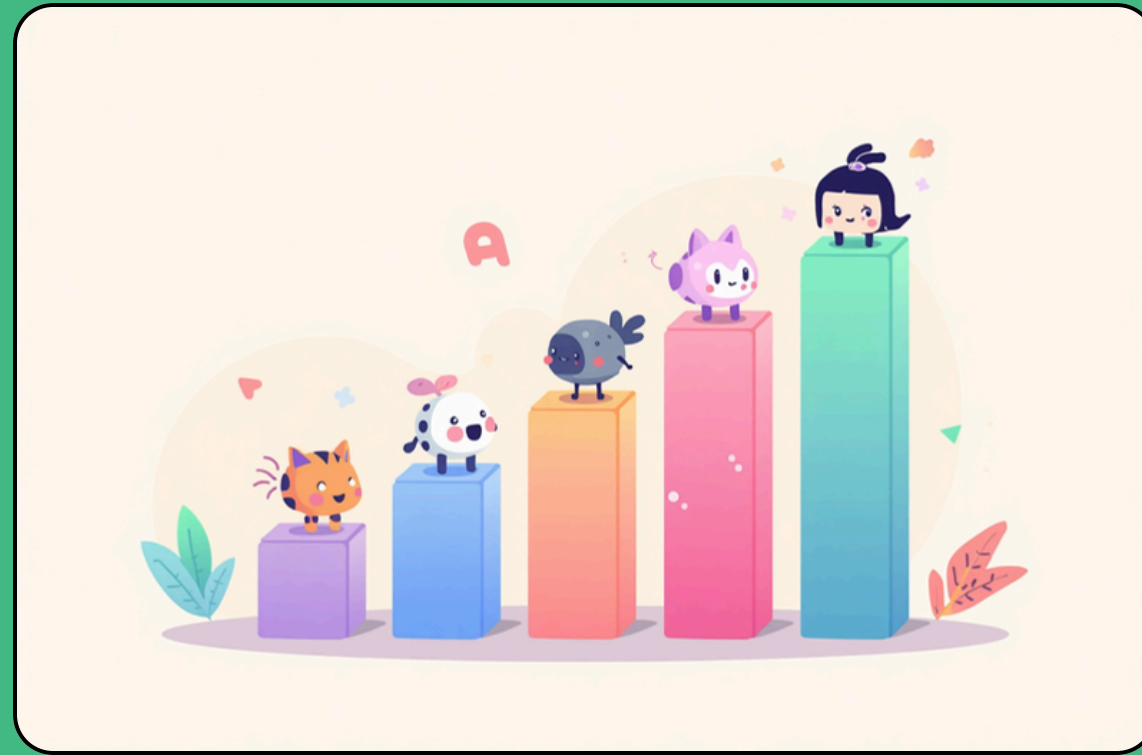


The Shift Is Real: AI-Assisted Coding is Mainstream



Discovery

Devs are trying AI tools individually, no team standards yet.



Adoption

Teams use AI intentionally with rules and feedback loops.



Integration

AI is becoming a **core part** of coding workflows.



I haven't written code
by hand in over 6
months



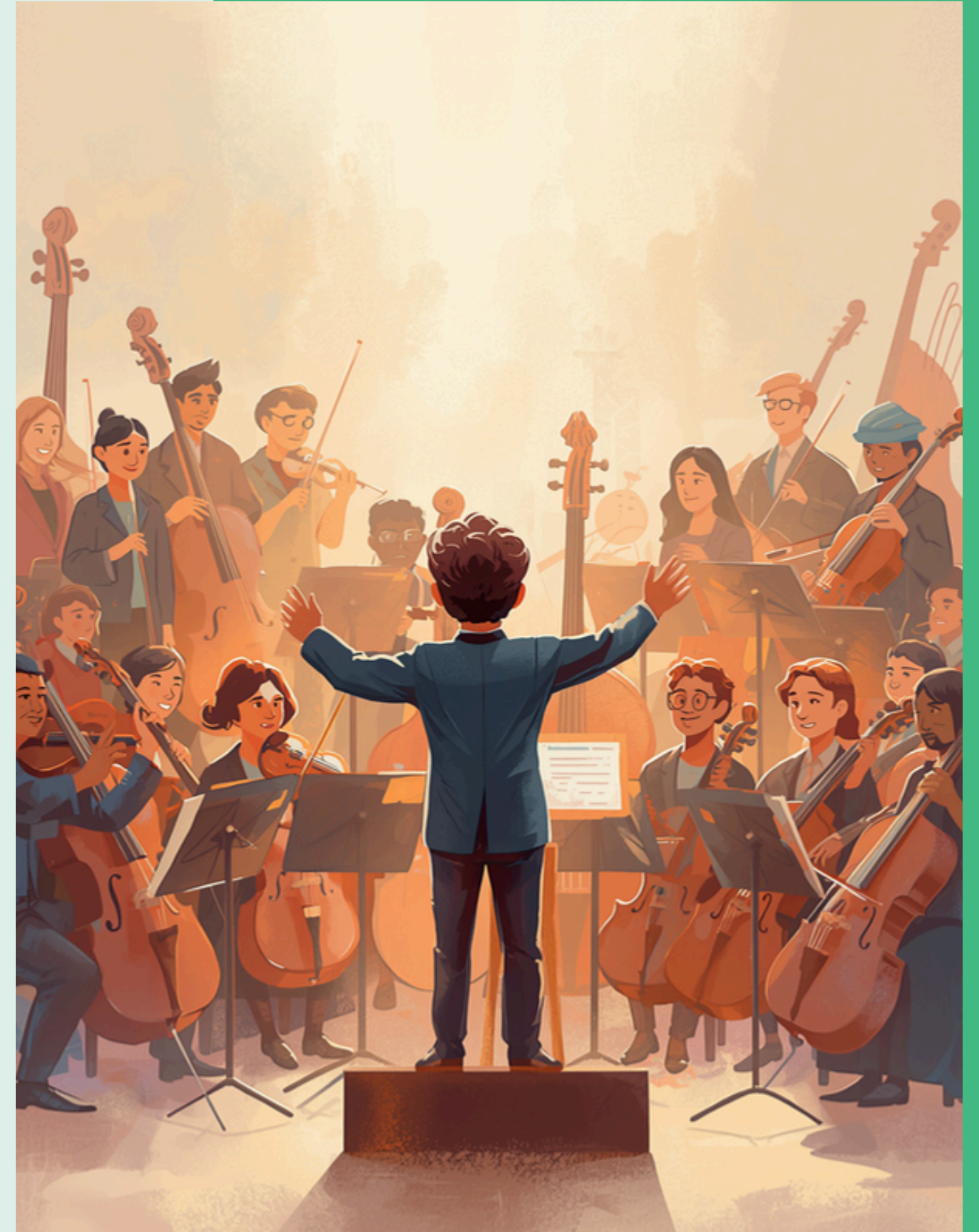
From Code Writer to Conductor

Embracing the Age of AI Tools

Transitioning from writing every line of code to guiding **intelligent agents** has transformed my role. Now, I focus on **orchestration** and creativity more than typing alone.

Expertise Matters

“Your expertise
and taste is
MORE
valuable, then
ever right now”

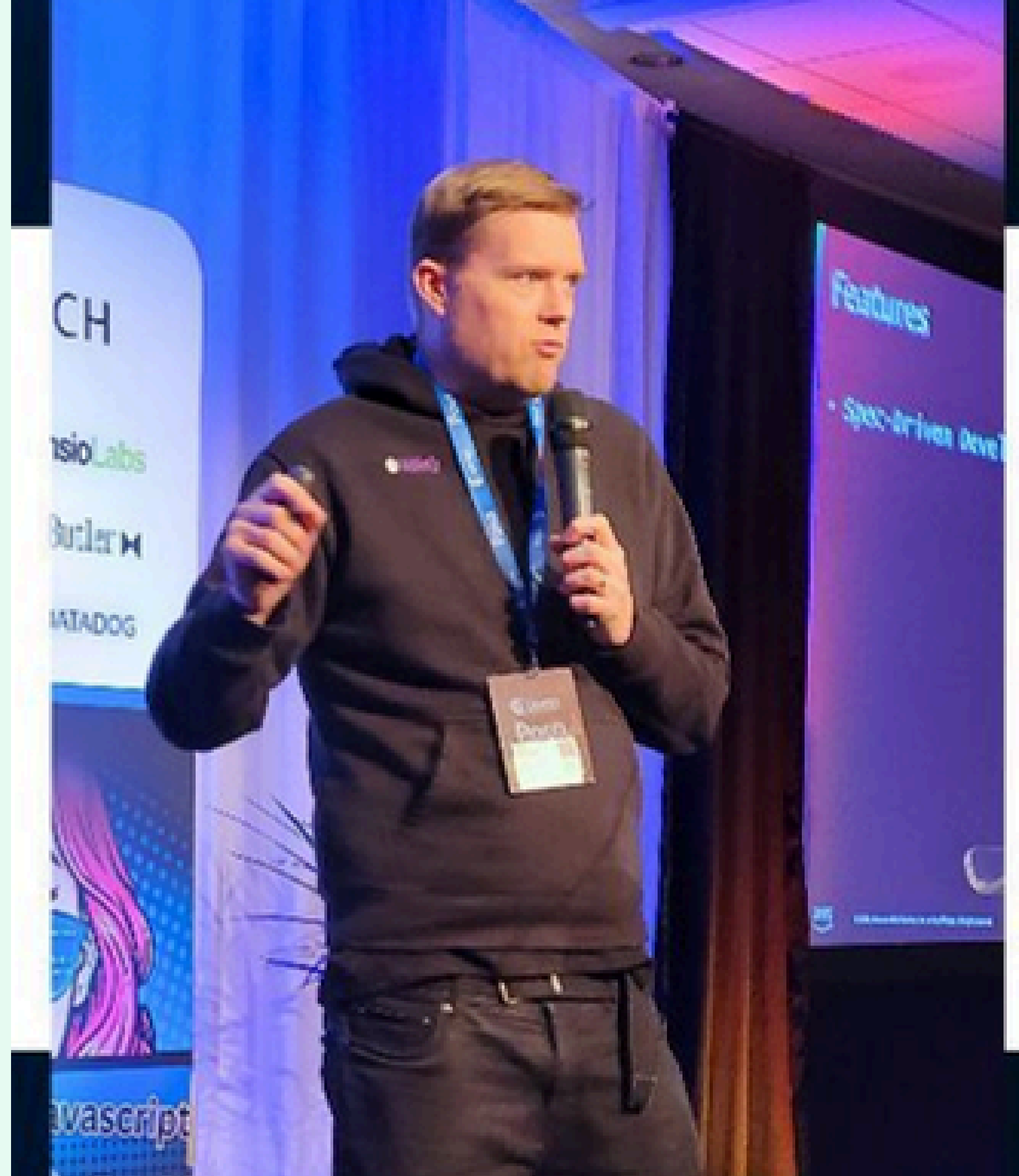


Meet Erik Hanchett

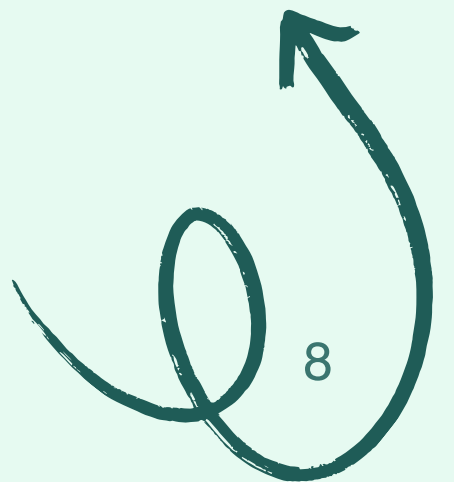
ErikCH

www.programwitherik.com

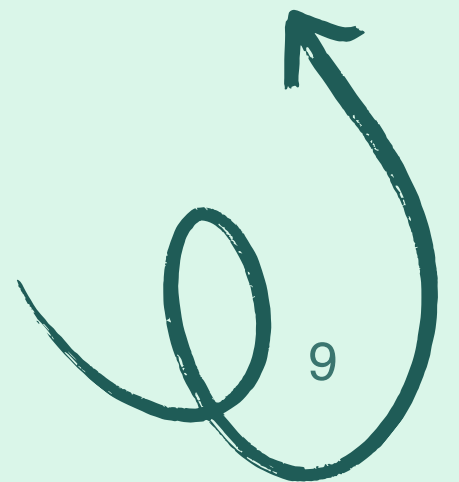
Erik Hanchett is a **Senior Developer Advocate** at AWS with over 15 years of experience in software development. He is the creator of Kiro content and the "Program with Erik" series, focusing on modern development practices.



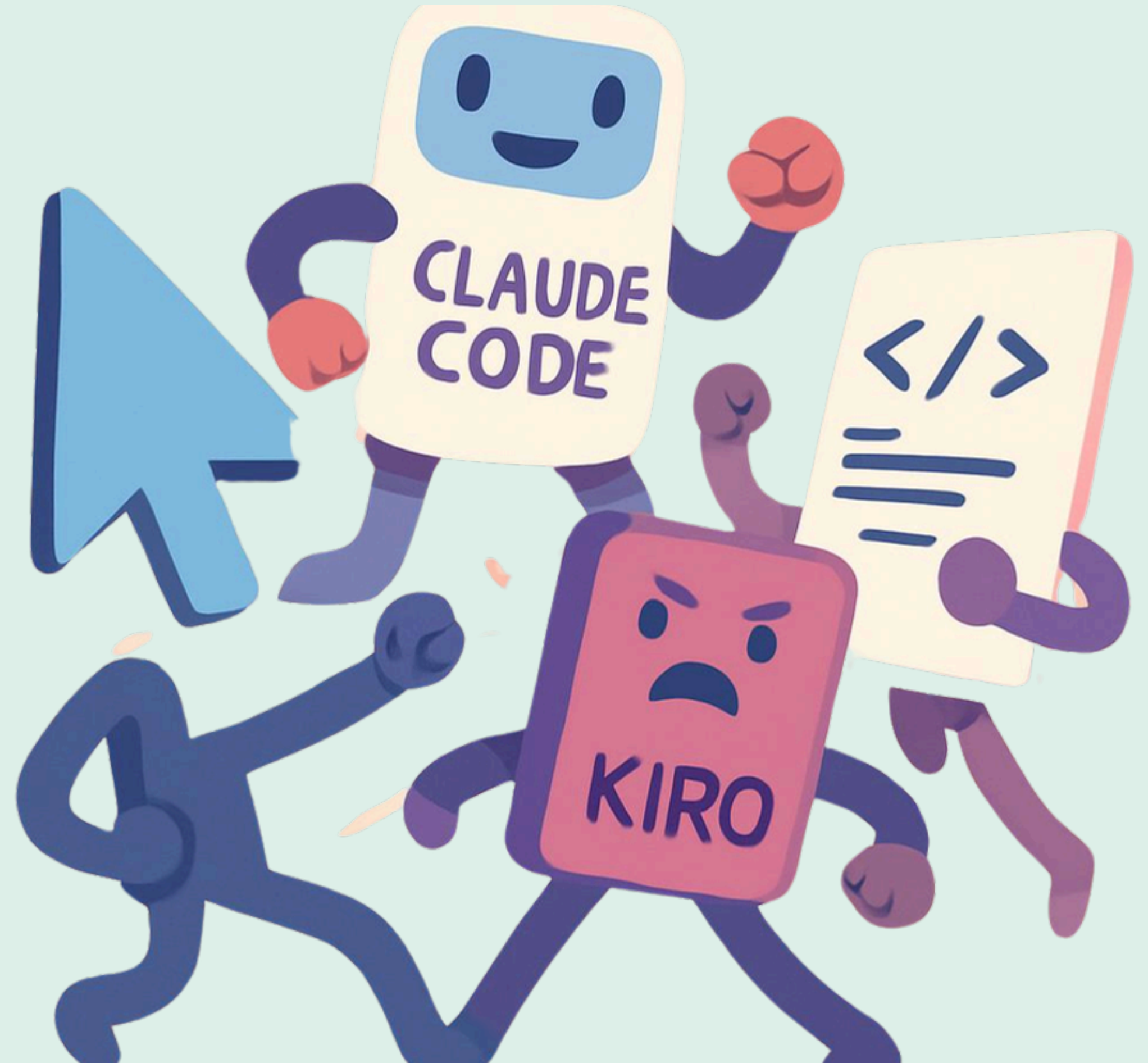
Where to Begin?



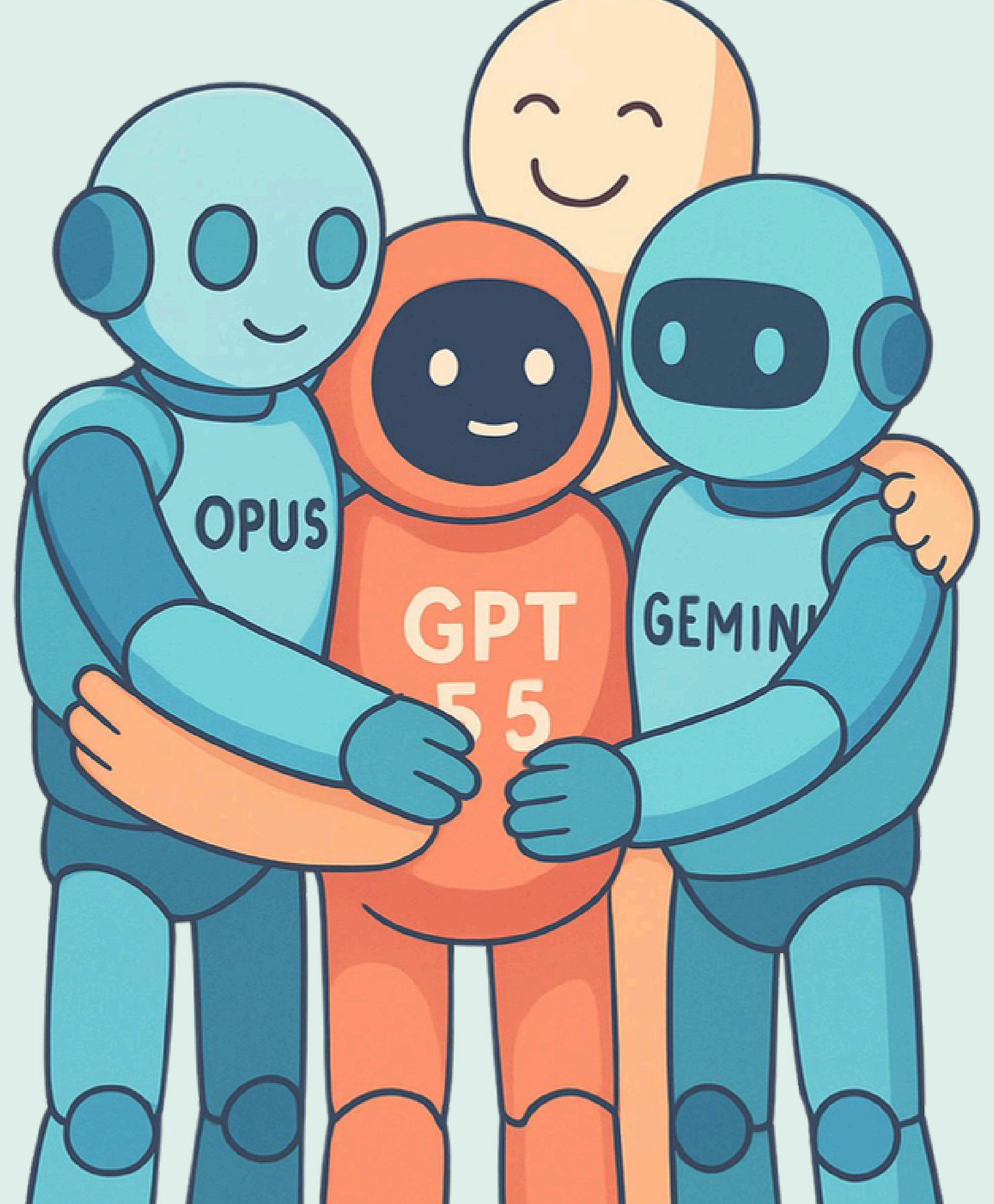
Pick a Harness and Model



Kiro
Claude Code
Codex
Cursor



**Pick a
frontier
model**



CLI > IDE

- Use your IDE as your viewer , the CLI as your agent



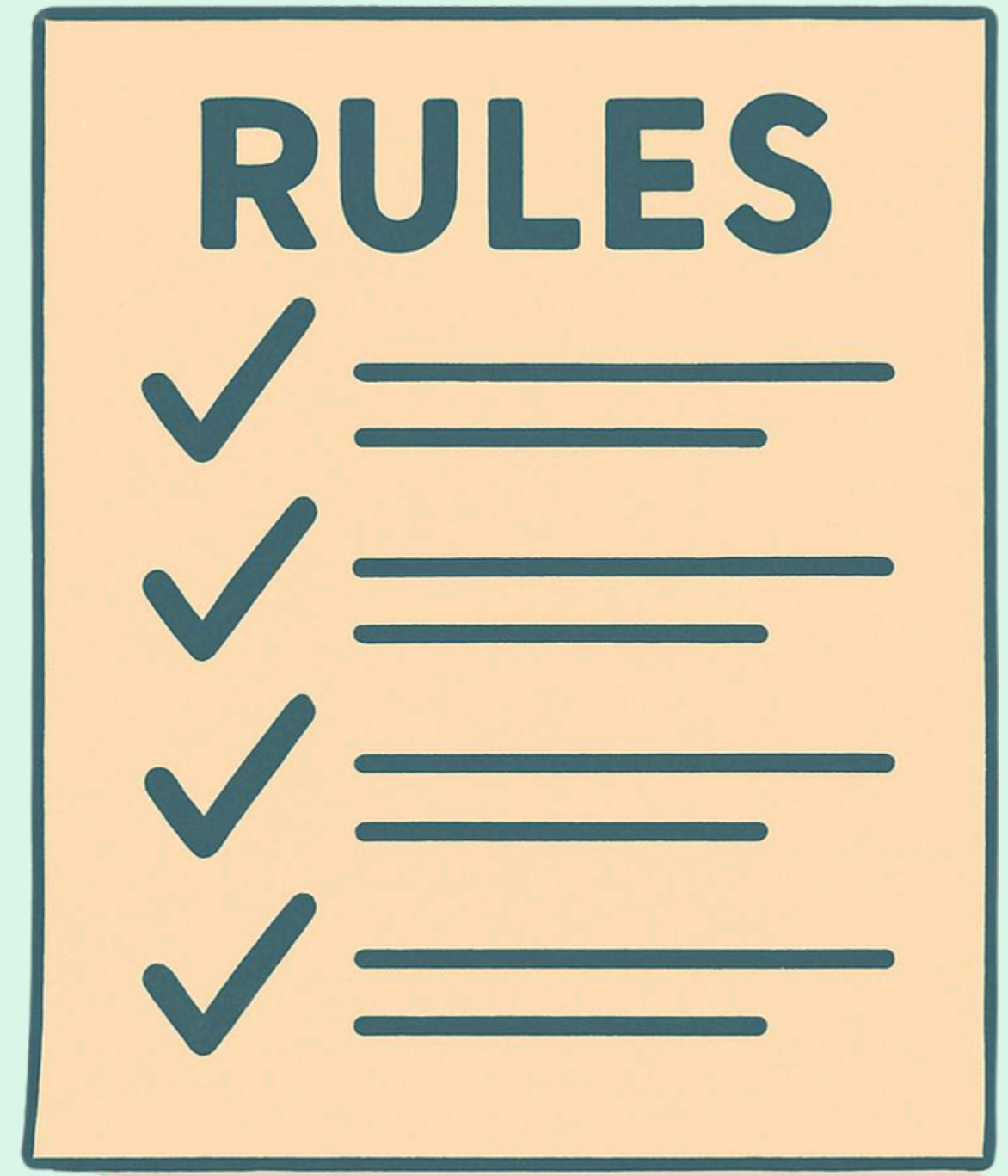
Type > Talk

- **Less vocal strain**



Create rules (and skills) for the AI

LLMs operate statelessly, requiring you to communicate your standards anew each session. Use `AGENTS.md`, `CLAUDE.md`, or `.cursorrules` to maintain clarity and consistency. Skills are reusable, domain-specific modules, or recipes.



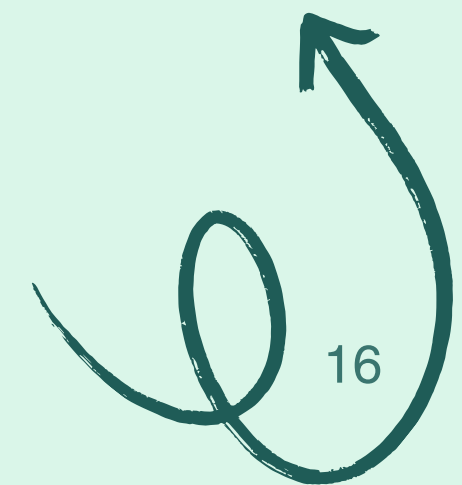
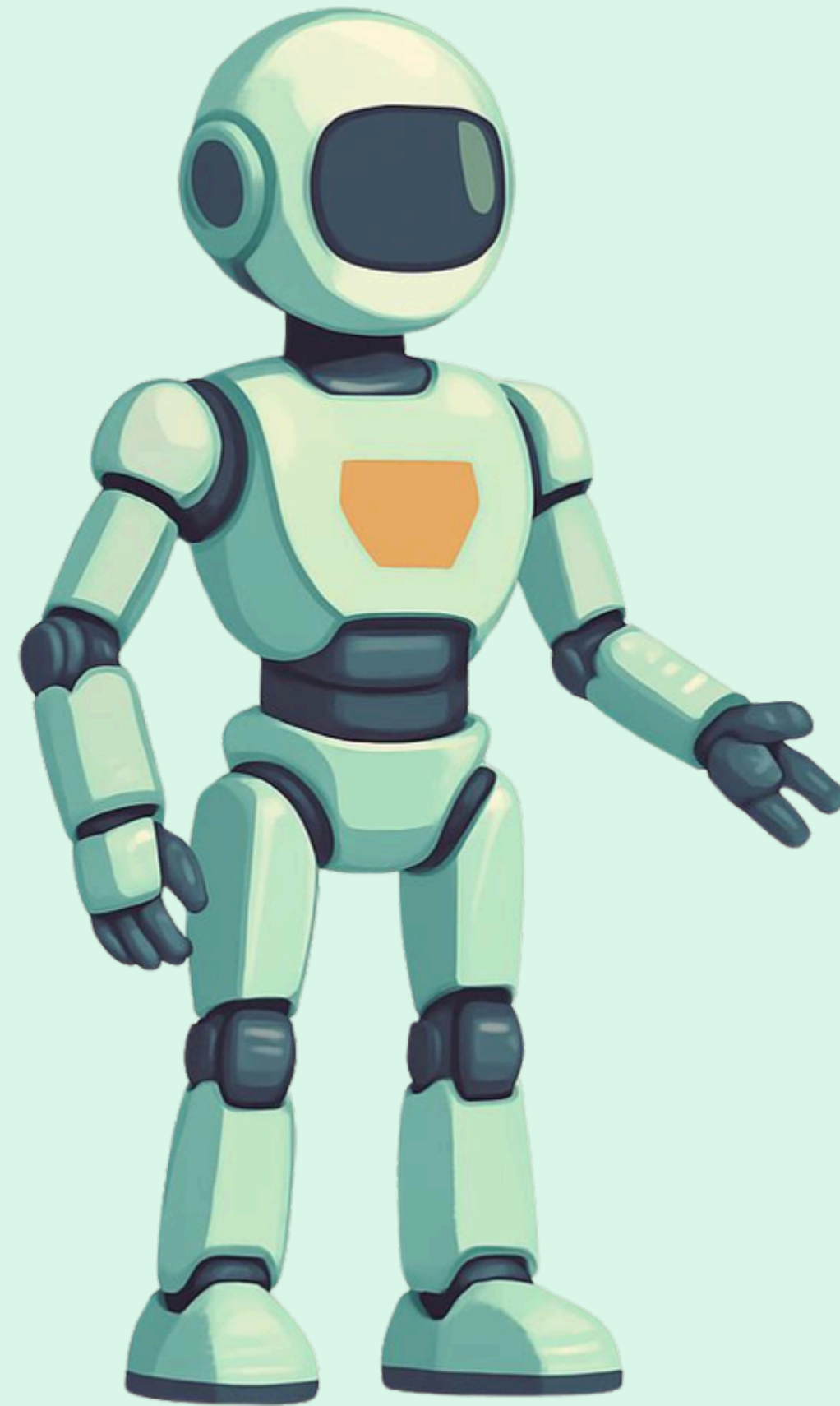
Create rules (AGENTS.md)

- **Project's purpose**
- **Architecture**
- **Coding standards**
- **Keep it brief**



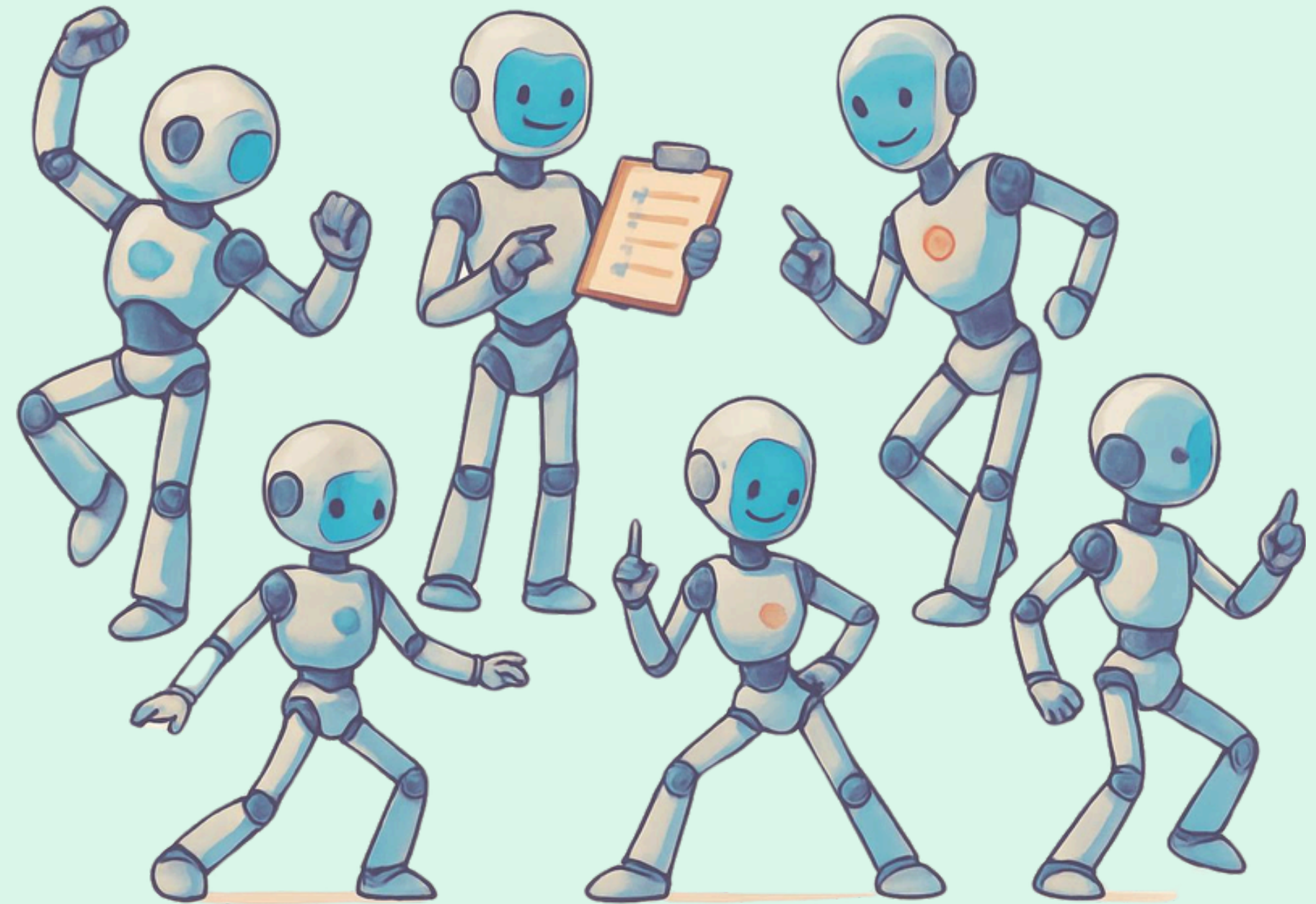
Create agents

- **/agents or /plugin**



So what can they do?

- Context
- Skill(s)
- Hooks
- Subagents
- Allowed Tools
- Autonomous



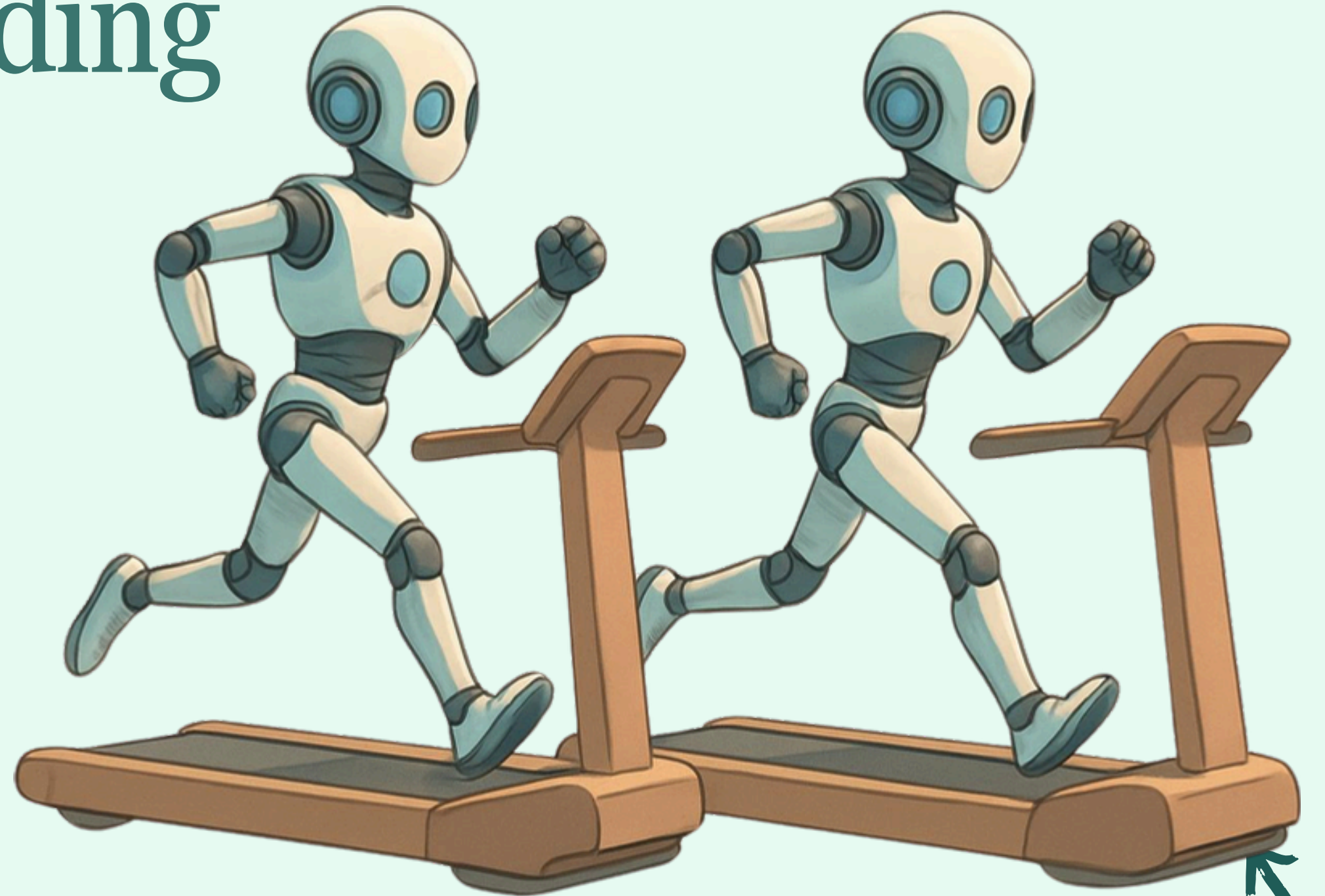
Working with Coding agents/tools

Two Coding Assistants

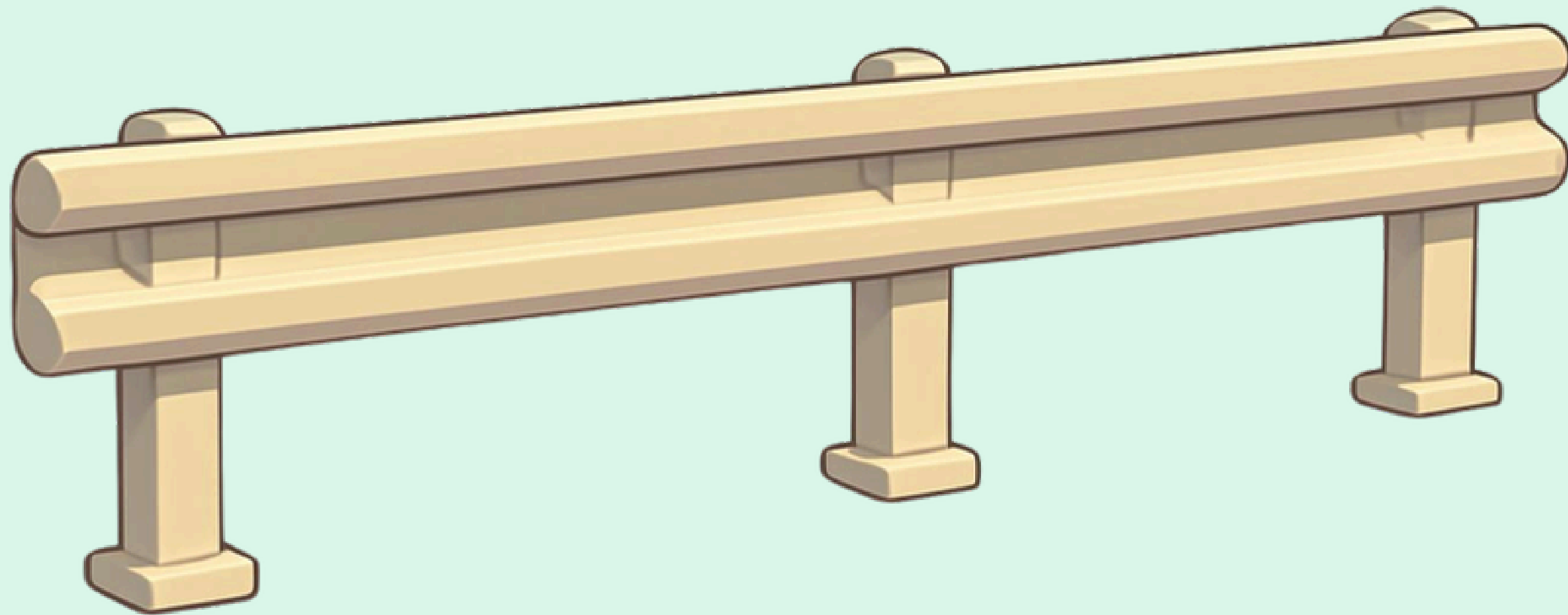
You can effectively run **two simultaneously** to increase productivity. This allows for parallel coding tasks.

File Management

Keeping tasks **small and scoped** is essential. This approach helps prevent conflicts.



Setup Guardrails



Guardrails

- Tests / Vitest
- Linting
- CI Checks
- Code Coverage
- Error Tracking
- Pre-commit hooks

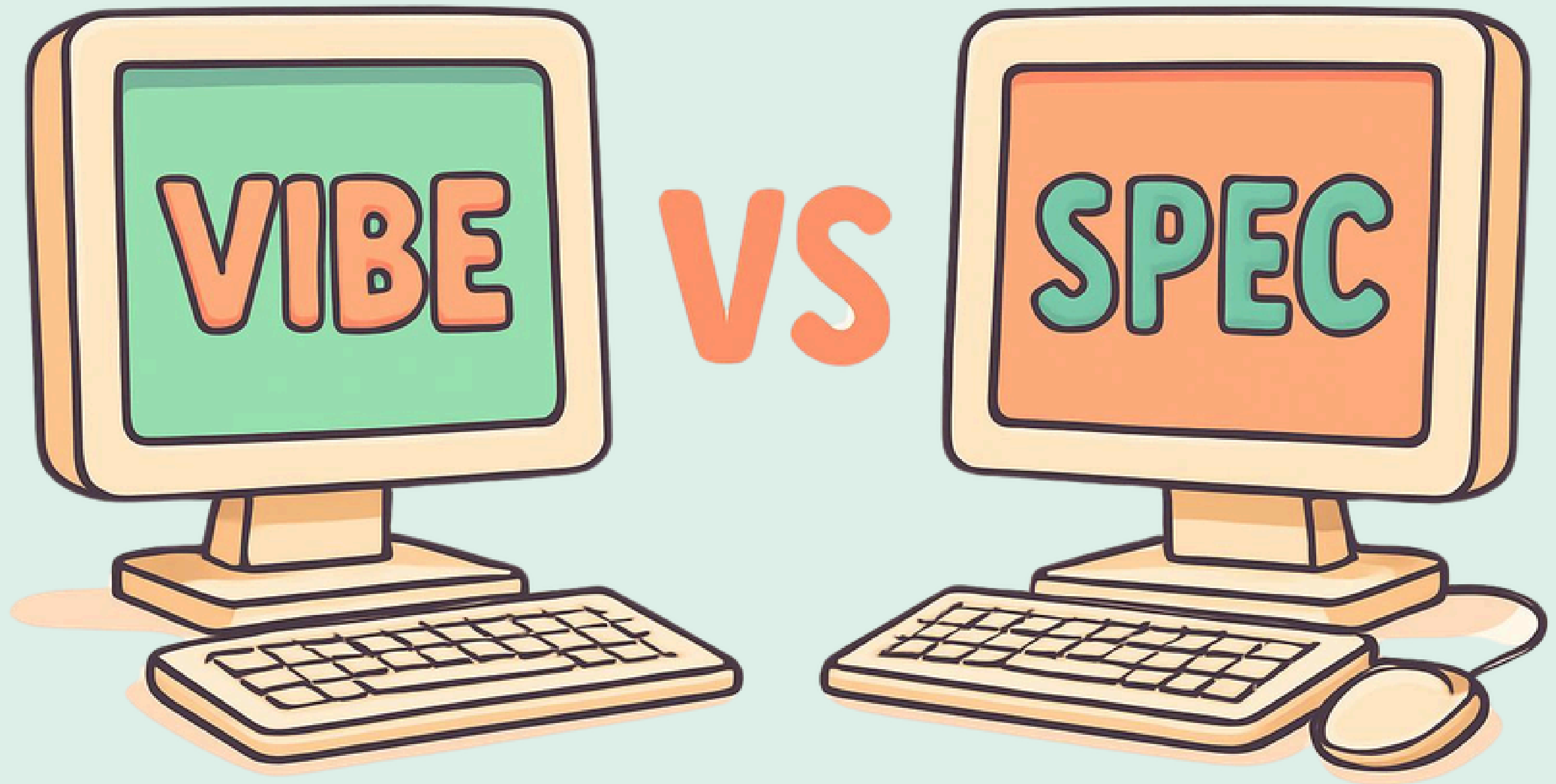




Code Reviews Are Bottlenecks

AI creates faster than we can review





Vibe Coding

- Easy to start with
- Fast prototyping
- Fixing bugs
- Small scoped tasks



Spec Driven

- Good for feature development
- Scope is large or unclear
- Need to parallelize
- Scales well for complex features



Embracing Your New Role



Thank You!

Connect with me on LinkedIn

Erik Hanchett

www.programwitherik.com

