Legacy AI-Coding

request slides

AI-Coding.pro

www.AGILE.COACH

Dr. Timon Fiddike
Email: timon@agile.coach
Web: agile.coach, ai-coding.pro
Connect on Linkedin: Timon

People

Timon Fiddike

Technology

Processes

People

ICF International Coach Federation

AGILE.COACH

berlin

Timon Fiddike

Scrum Alliance CST TRAINER

Technology

https://ai-coding.pro/

Processes

Cost?

Cost-Effectivenes

1. Make it effective
2. Save tokens == € (while preserving effect)

- Choose a good model (not cheap, not “auto”)
- Think and check your prompt before sending
- At least at the beginning:
    - Parallel thinking / preparation is o.k., but:
    - Finish one thing before starting the next session
- Finish what you start in 1 h (cache hits are 1/10 price, see Claude API pricing)
- Clear context / start new session unless you have a reason to keep it (good for cost and adherence)
- Create tool wrappers to keep output short in case of success (easy way to save tokens)
- Stop if things get weird! Go back one or two steps!
- Carefully use prompting levels (behaviour from user perspective, technical plan, code, results)

Search... ⌘K

Models
Models overview
Choosing a model
What's new in Claude 4.6
Migration guide
Model deprecations
Model cards
System prompts

Pricing
Pricing

Console

Pricing

Pricing

Copy page

Learn about Anthropic's pricing structure for models and features

This page provides detailed pricing information for Anthropic's models and features. All prices are in USD.

For the most current pricing information, please visit claude.com/pricing.

Model pricing

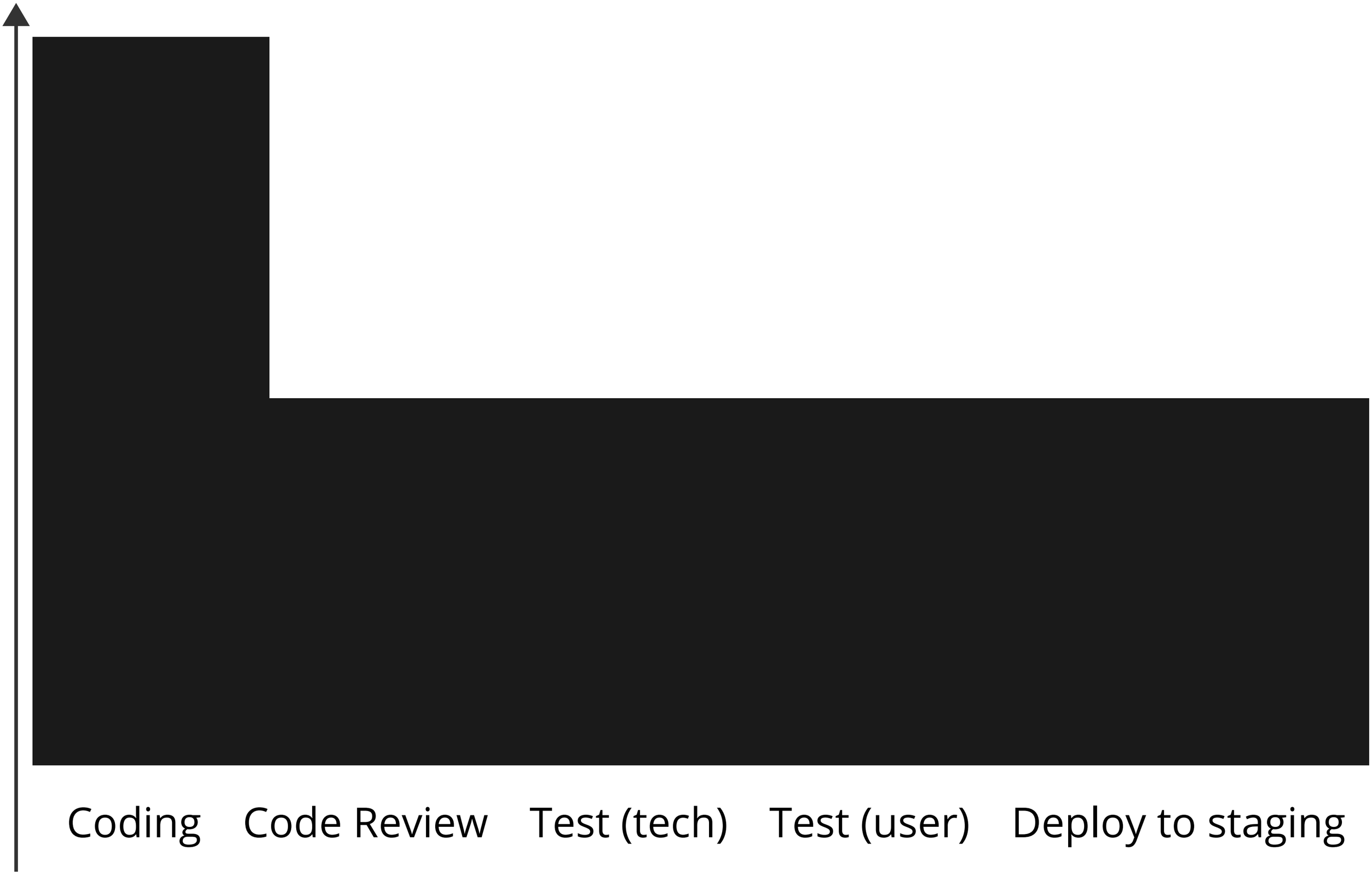
The following table shows pricing for all Claude models across different usage tiers:

Model	Base Input Tokens	5m Cache Writes	1h Cache Writes	Cache Hits & Refreshes	Output Tokens
Claude Opus 4.6	$5 / MTok	$6.25 / MTok	$10 / MTok	$0.50 / MTok	$25 / MTok
Claude Opus 4.5	$5 / MTok	$6.25 / MTok	$10 / MTok	$0.50 / MTok	$25 / MTok
Claude Opus 4.1	$15 / MTok	$18.75 / MTok	$30 / MTok	$1.50 / MTok	$75 / MTok
Claude Opus 4	$15 / MTok	$18.75 / MTok	$30 / MTok	$1.50 / MTok	$75 / MTok
Claude Sonnet 4.6	$3 / MTok	$3.75 / MTok	$6 / MTok	$0.30 / MTok	$15 / MTok
Claude Sonnet 4.5	$3 / MTok	$3.75 / MTok	$6 / MTok	$0.30 / MTok	$15 / MTok
Claude Sonnet 4	$3 / MTok	$3.75 / MTok	$6 / MTok	$0.30 / MTok	$15 / MTok

Capacity

Coding Code Review Test (tech) Test (user) Deploy to staging

Capacity

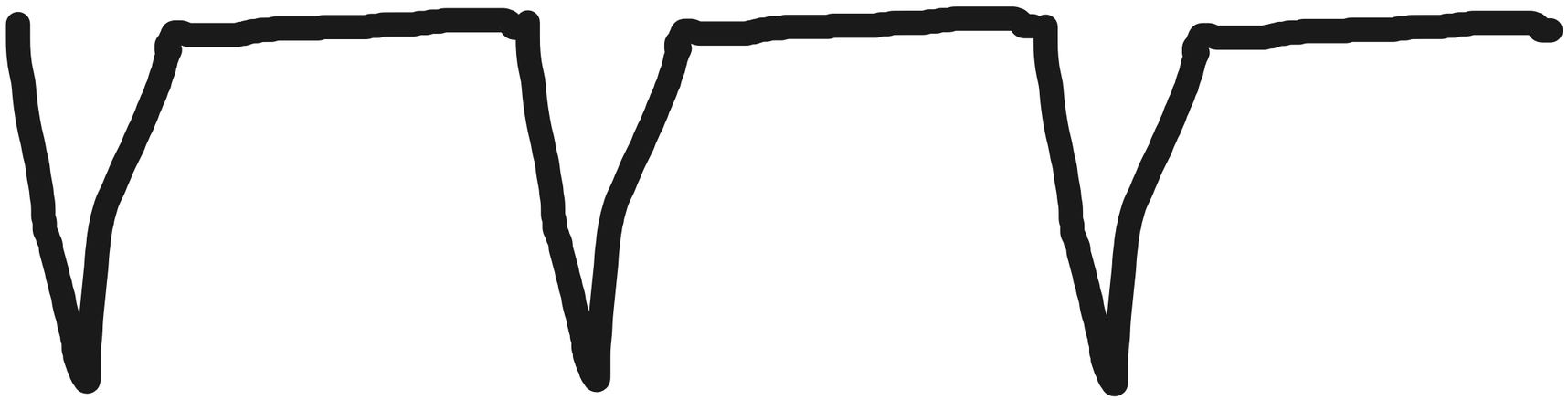


Coding Code Review Test (tech) Test (user) Deploy to staging

Capacity

Coding Code Review Test (tech) Test (user) Deploy to staging

Human

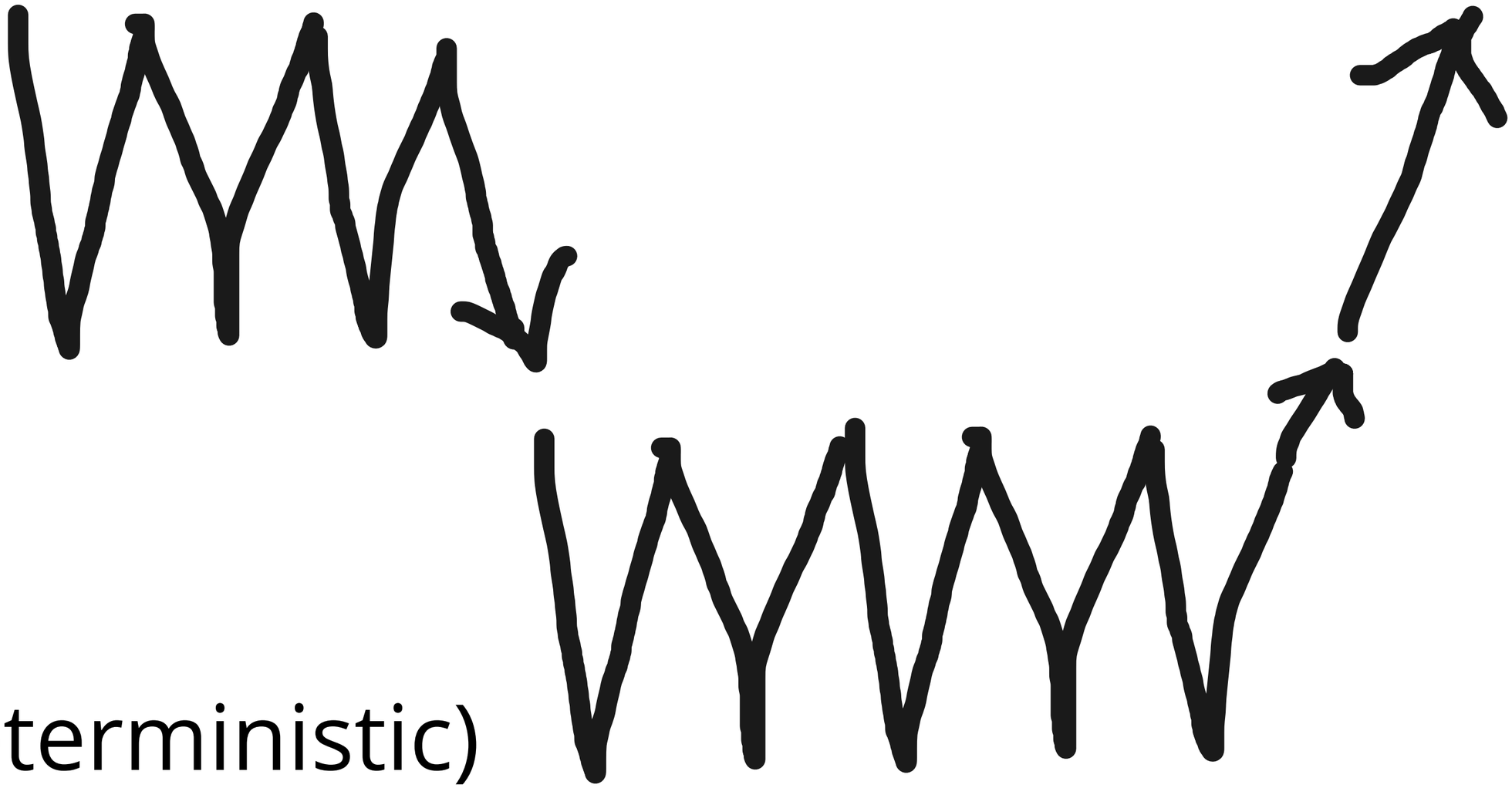


Agent

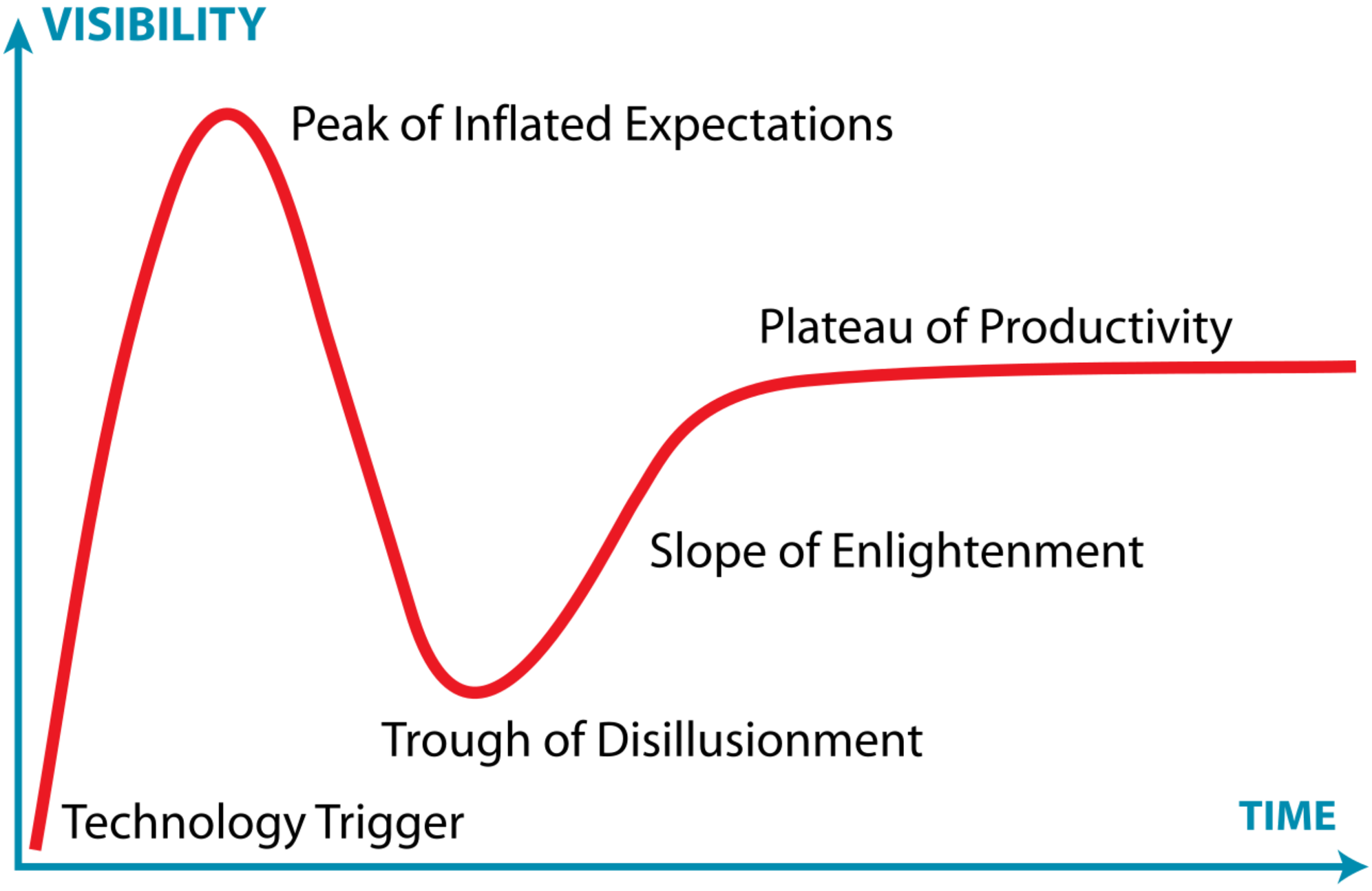
Human

Agent

Tools (deterministic)



Plan (what) Plan (how) Coding Code Review Test (tech) Test (user)


VISIBILITY
Peak of Inflated Expectations
Plateau of Productivity
Slope of Enlightenment
Trough of Disillusionment
Technology Trigger
TIME

Q&A and Discussion

Questions?

Your Experience?

We can also go deeper into my process with many more sides :-)

Who has done AI Coding via ...

1: Copy & Paste with ChatGPT

2: AI Code Completion

3: IDE with AI Chat und Agent
(plugin or native)

4: AI Agent with Auto-Run

Did anyone read the articles?

AI-Coding Lernen: Ein Weg in 10 Schritten: Ein Erfahrungsbericht

https://ai-coding.pro/de/blog/ai-coding-lernen-ein-weg-in-10-schritten-ein-erfahrungsbericht/

Haltung zu KI-Coding: Von Vermeidung zu Realismus: Ein Erfahrungsbericht

https://ai-coding.pro/de/blog/haltung-ai-coding-vermeidung-realismus-erfahrungsbericht/

...

Very Many Options

Show: Cursor: Models

Good: No dependency on one service. But:

FOMO

Today: Cursor + Claude Sonnet 4.6

Suchtrends ansehen

Löschen | Suchbegriffe vorschlagen

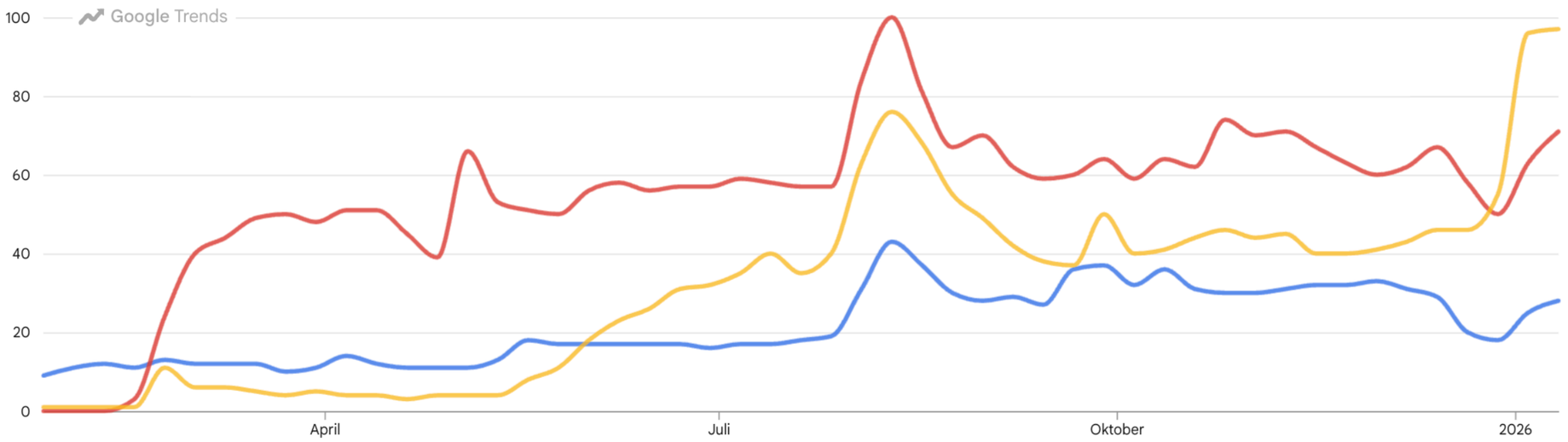
- Github Copilot — Software
- Cursor-Editor — Software
- Claude Code — Suchbegriff

Weltweit | 01.02.2025 - 15.01.2026 | Websuche

Interesse im zeitlichen Verlauf

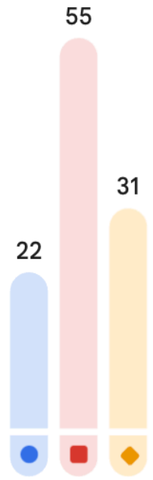
Weltweit · 01.02.2025 - 15.01.2026

Sowohl Suchbegriffe als auch Suchthemen sind enthalten, werden aber unterschiedlich gemessen. Weitere Informationen



Durchschnittliches Interesse

Weltweit · 01.02.2025 - 15.01.2026



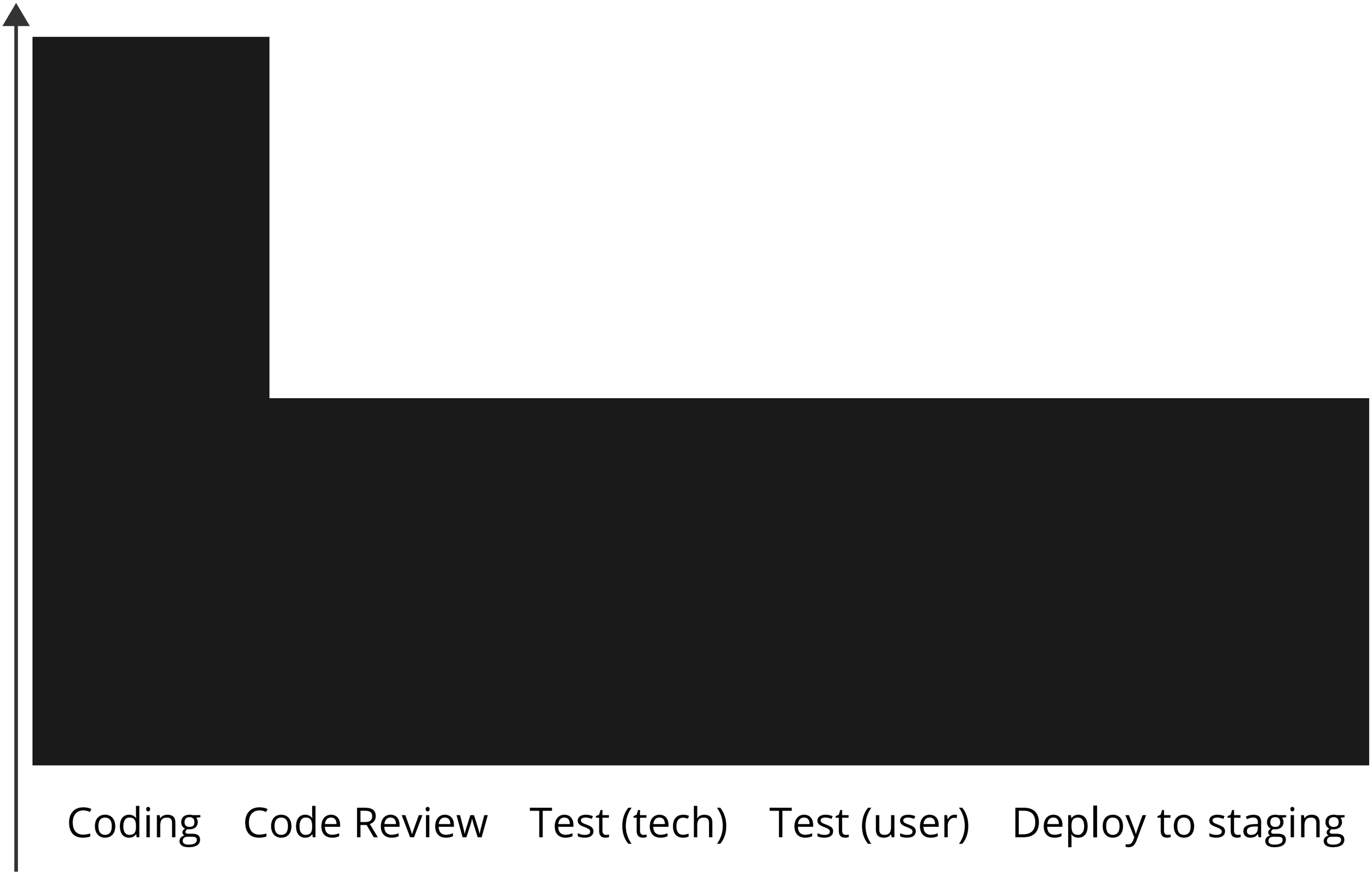
Very Many Options FOMO

The most important part is something else:

Capacity

Coding Code Review Test (tech) Test (user) Deploy to staging

Capacity

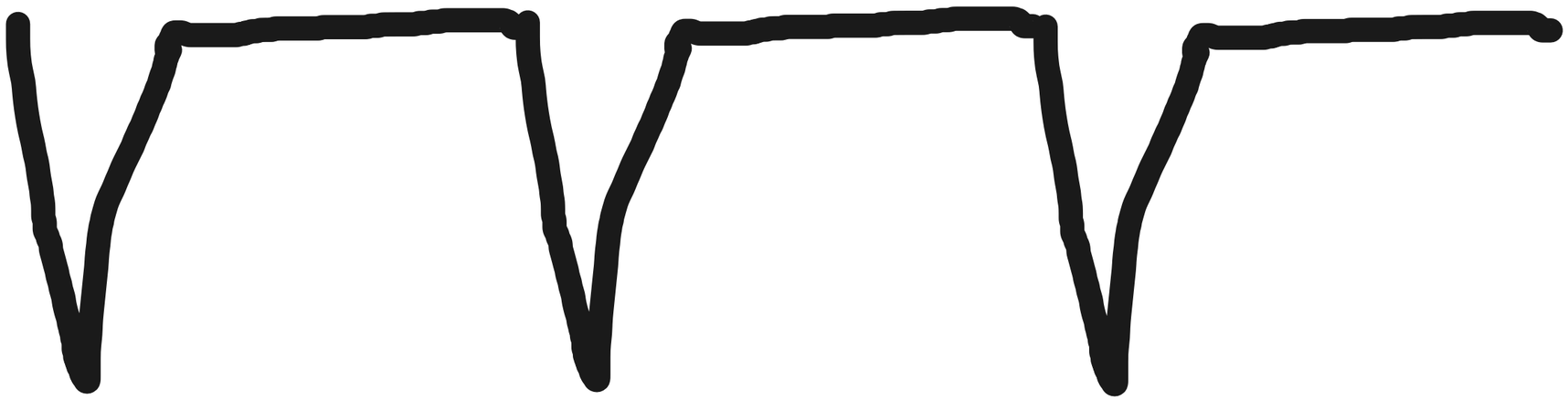


Coding Code Review Test (tech) Test (user) Deploy to staging

Capacity

Coding Code Review Test (tech) Test (user) Deploy to staging

Human

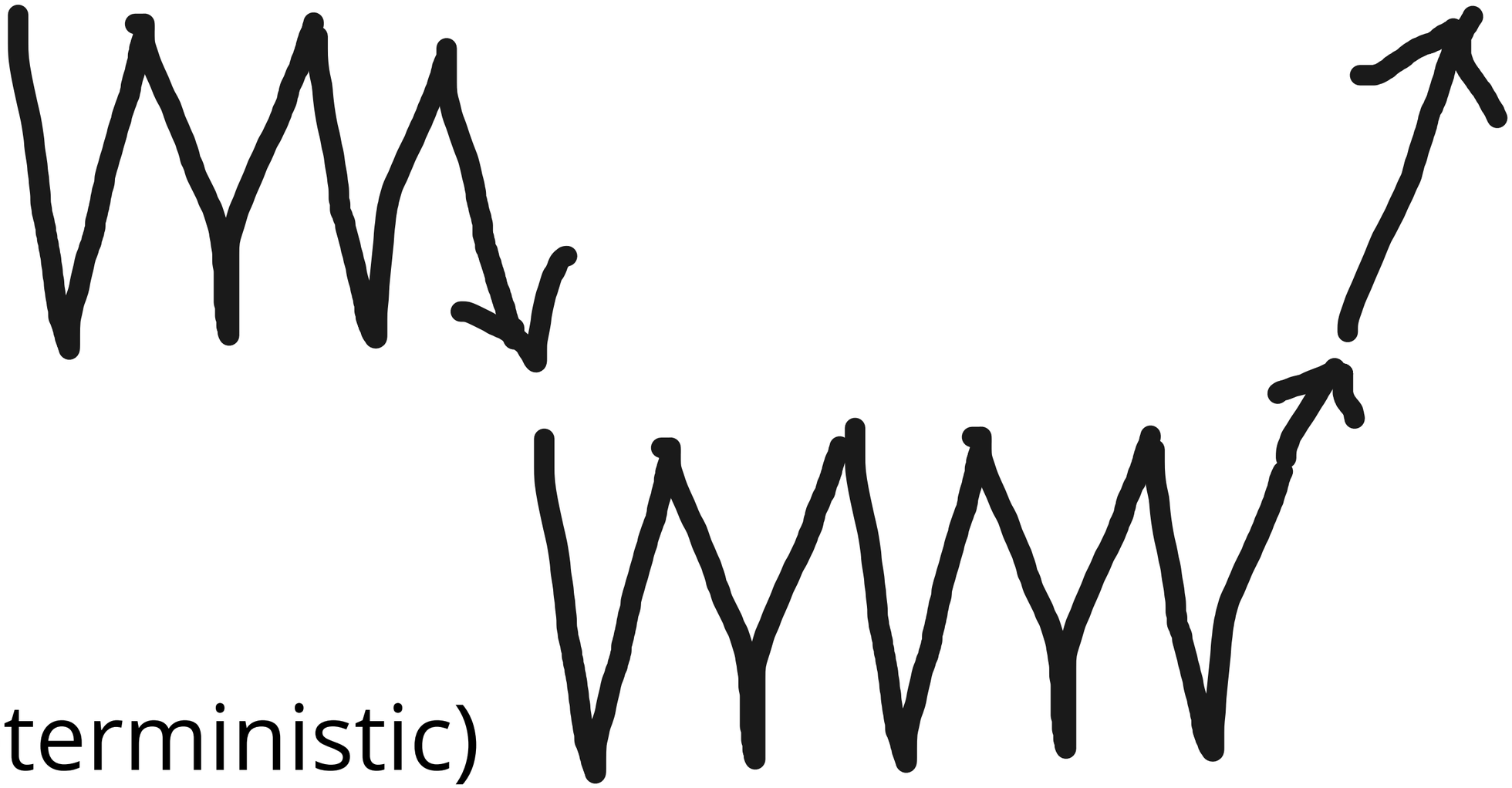


Agent

Human

Agent

Tools (deterministic)



Plan (what) Plan (how) Coding Code Review Test (tech) Test (user)

Adding a field

Imagine you need to add a field to an entity in your app:
What do you need to take care of?

Please write sticky notes!

Adding a field

Demo

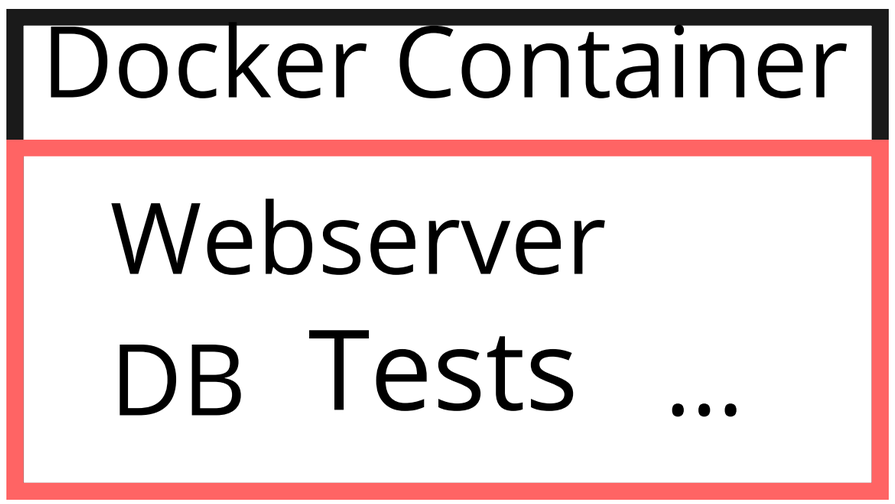
(Focus on feedback)

10 Steps to Enable Fundamental How

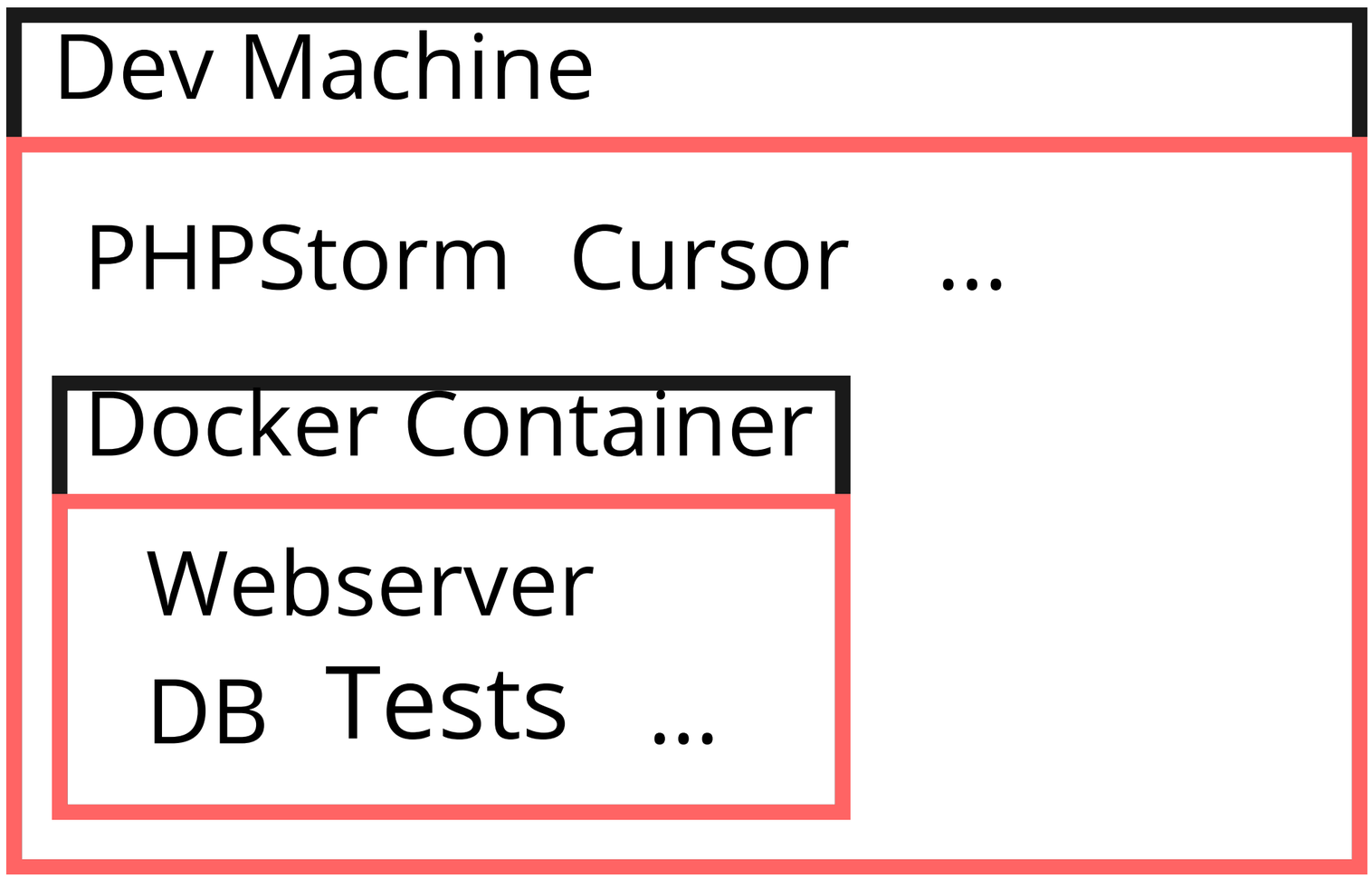
AI-Coding Lernen: Ein Weg in 10 Schritten: Ein Erfahrungsbericht
https://ai-coding.pro/de/blog/ai-coding-lernen-ein-weg-in-10-schritten-ein-erfahrungsbericht/

Copy & Paste with ChatGPT
Tab Code Completion
IDE with AI Chat and Agent
Agent with Auto-Run Mode in VM
More Technical Context
Rules for the Agent
Planning in Multiple Levels
Faster Feedback Loops
Learning from Mistakes
Relying More on Agents

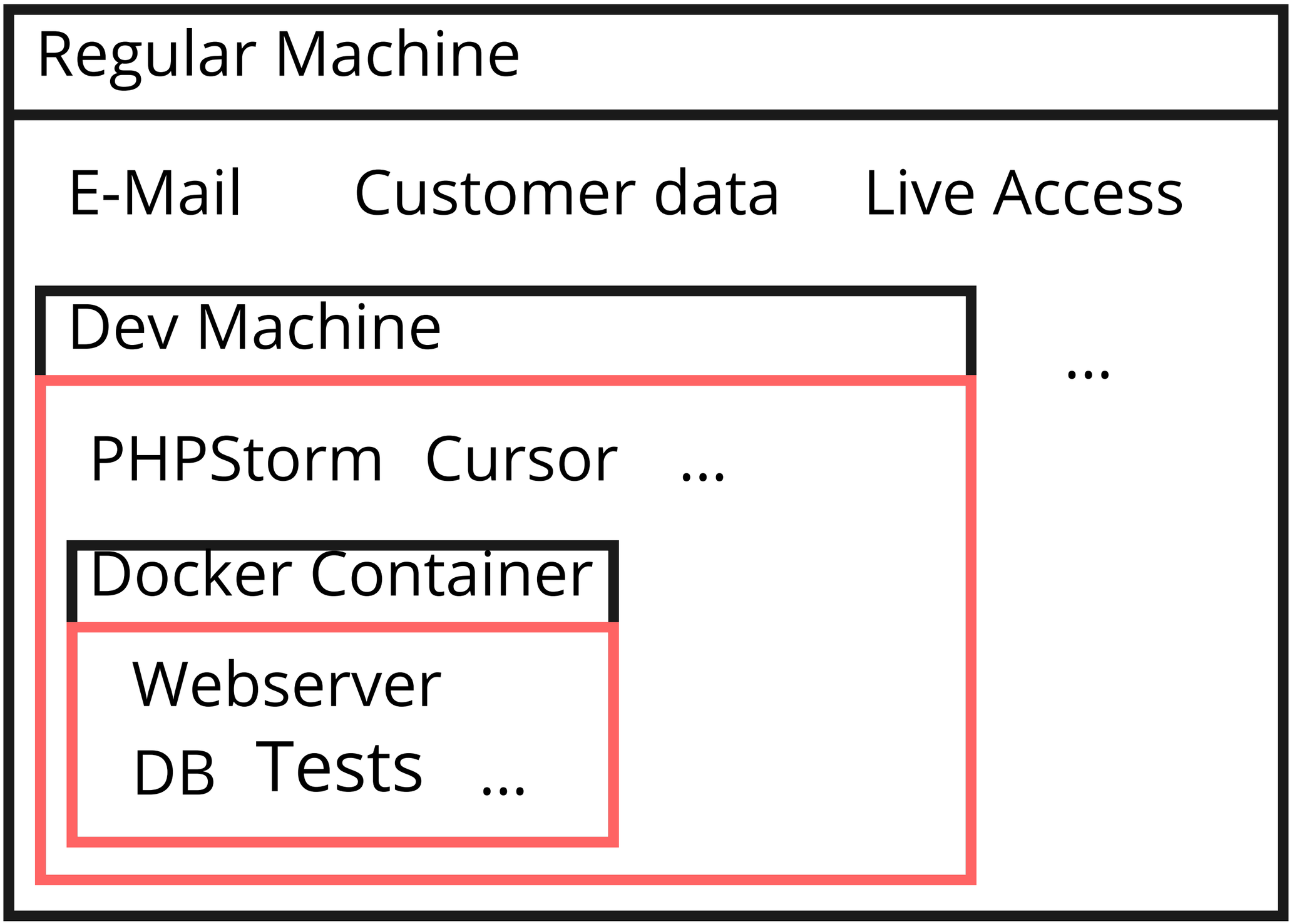
Autorun: Freedom and Limits



Autorun: Freedom and Limits



Autorun: Freedom and Limits



Autorun on Dev Machine?!

Regular Machine

E-Mail Customer data Live Access

Dev Machine

PHPStorm Cursor ...

Docker Container

Webserver

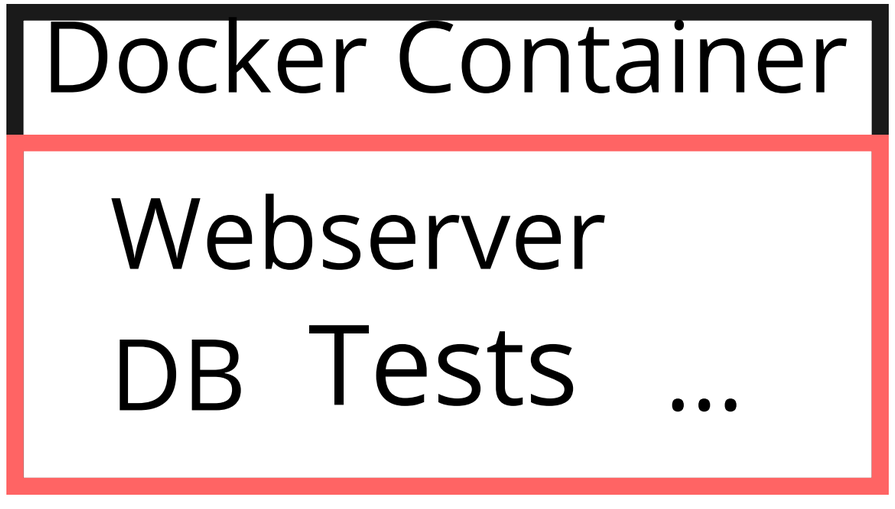
DB Tests ...

...

Rebuild Docker Container

Imagine you need to rebuild your docker container:
What do you need to take care of?

Please write sticky notes!



Rebuild Docker Container

Demo

- dash/bin/dev/docker-dev-setup.sh
- dash/bin/dev/docker-dev-start.sh

10 Steps to Enable Fundamental How

AI-Coding Lernen: Ein Weg in 10 Schritten: Ein Erfahrungsbericht
https://ai-coding.pro/de/blog/ai-coding-lernen-ein-weg-in-10-schritten-ein-erfahrungsbericht/

Copy & Paste with ChatGPT
Tab Code Completion
IDE with AI Chat and Agent
Agent with Auto-Run Mode in VM
More Technical Context
Rules for the Agent
Planning in Multiple Levels
Faster Feedback Loops
Learning from Mistakes
Relying More on Agents

More Technical Context

1. Standard Framework
2. Documentation inside Repo
3. Automated Tests
4. MDC Rules
5. ...

Without Standard Framework?

- Need more of everything else.

10 Steps to Enable Fundamental How

AI-Coding Lernen: Ein Weg in 10 Schritten: Ein Erfahrungsbericht
https://ai-coding.pro/de/blog/ai-coding-lernen-ein-weg-in-10-schritten-ein-erfahrungsbericht/

Copy & Paste with ChatGPT
Tab Code Completion
IDE with AI Chat and Agent
Agent with Auto-Run Mode in VM
More Technical Context
Rules for the Agent
Planning in Multiple Levels
Faster Feedback Loops
Learning from Mistakes
Relying More on Agents

10 Steps to Enable Fundamental How

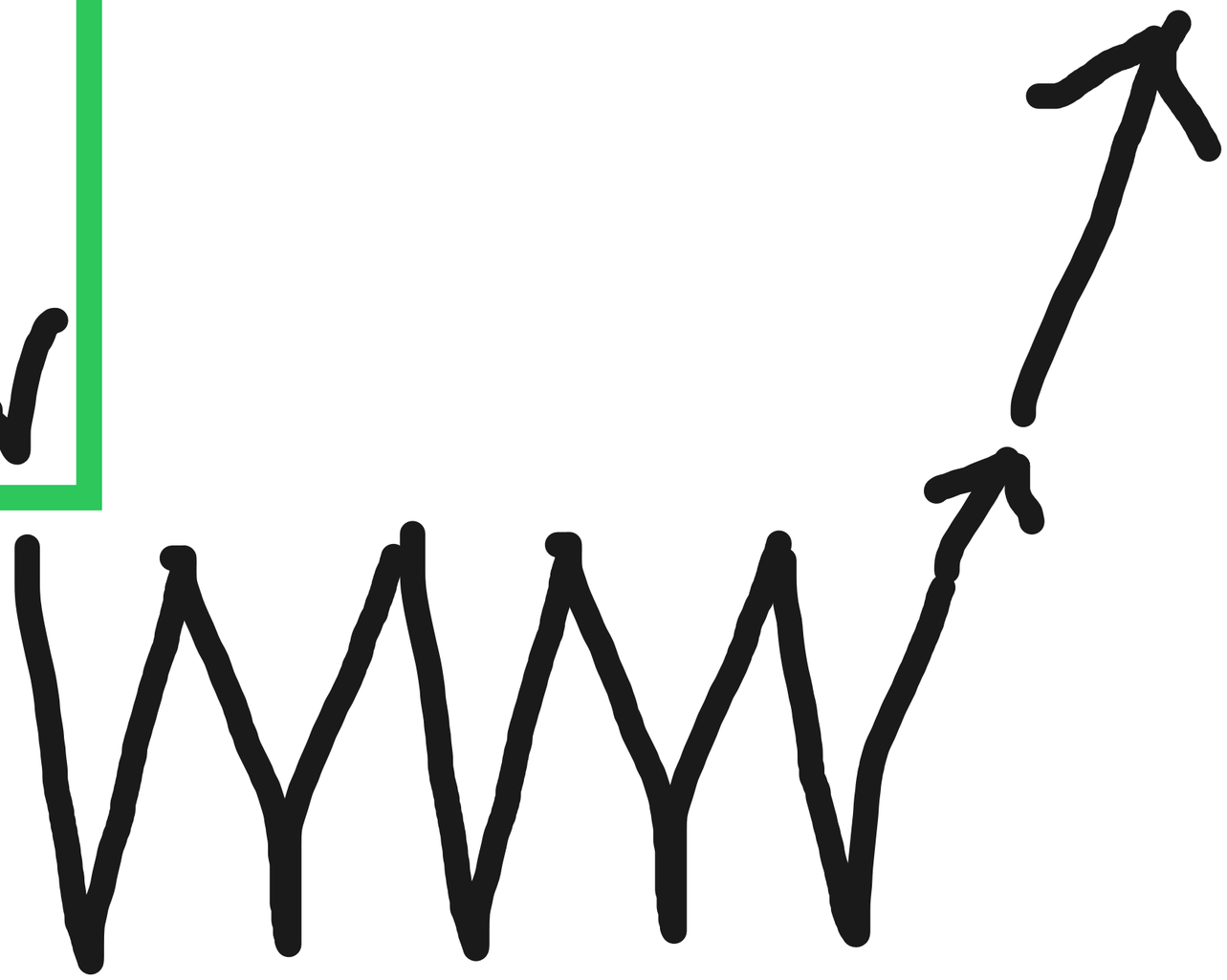
AI-Coding Lernen: Ein Weg in 10 Schritten: Ein Erfahrungsbericht
https://ai-coding.pro/de/blog/ai-coding-lernen-ein-weg-in-10-schritten-ein-erfahrungsbericht/

Copy & Paste with ChatGPT
Tab Code Completion
IDE with AI Chat and Agent
Agent with Auto-Run Mode in VM
More Technical Context
Rules for the Agent
Planning in Multiple Levels
Faster Feedback Loops
Learning from Mistakes
Relying More on Agents

Human

Agent

Tools (deterministic)



Plan (what) Plan (how) Coding Code Review Test (tech) Test (user)

Planning in Multiple Levels

1. Clarify Target State, including User, Goal, Feature(s), Interaction.
2. Detailed Technical Plan, including Architecture, Migrations etc.

Clarify Target State (Cursor "Ask")

I want to enable a <User Type>
to achieve <Goal>,
probably using <Feature>, <Interaction>.
Help me clarify how the feature
should work from a user perspective.
Ask me questions, one at a time, consider
each answer before asking the next q.

Things get weird?
Too many questions?
Late questions brings up valid concern about early decisions?

Things get weird?

Too many questions?

Late questions brings up valid concern about early decisions?

Learn!

Try to understand, make notes of what you learned, draft a better starting prompt, clear context(!) and: Start over.

Detailed Technical Plan (Cursor "Plan")

Same session as before:
Create a detailed impmenetation plan.
Ask me questions, one at a time, consider each answer before asking the next q.

Optional:
Extra hint for faster feedback loop in "go".

Things get weird?

Too many questions?

Late questions brings up valid concern about early decisions?

Learn!

Try to understand, make notes of what you learned, draft a better starting prompt, clear context(!) and: Start over.

10 Steps to Enable Fundamental How

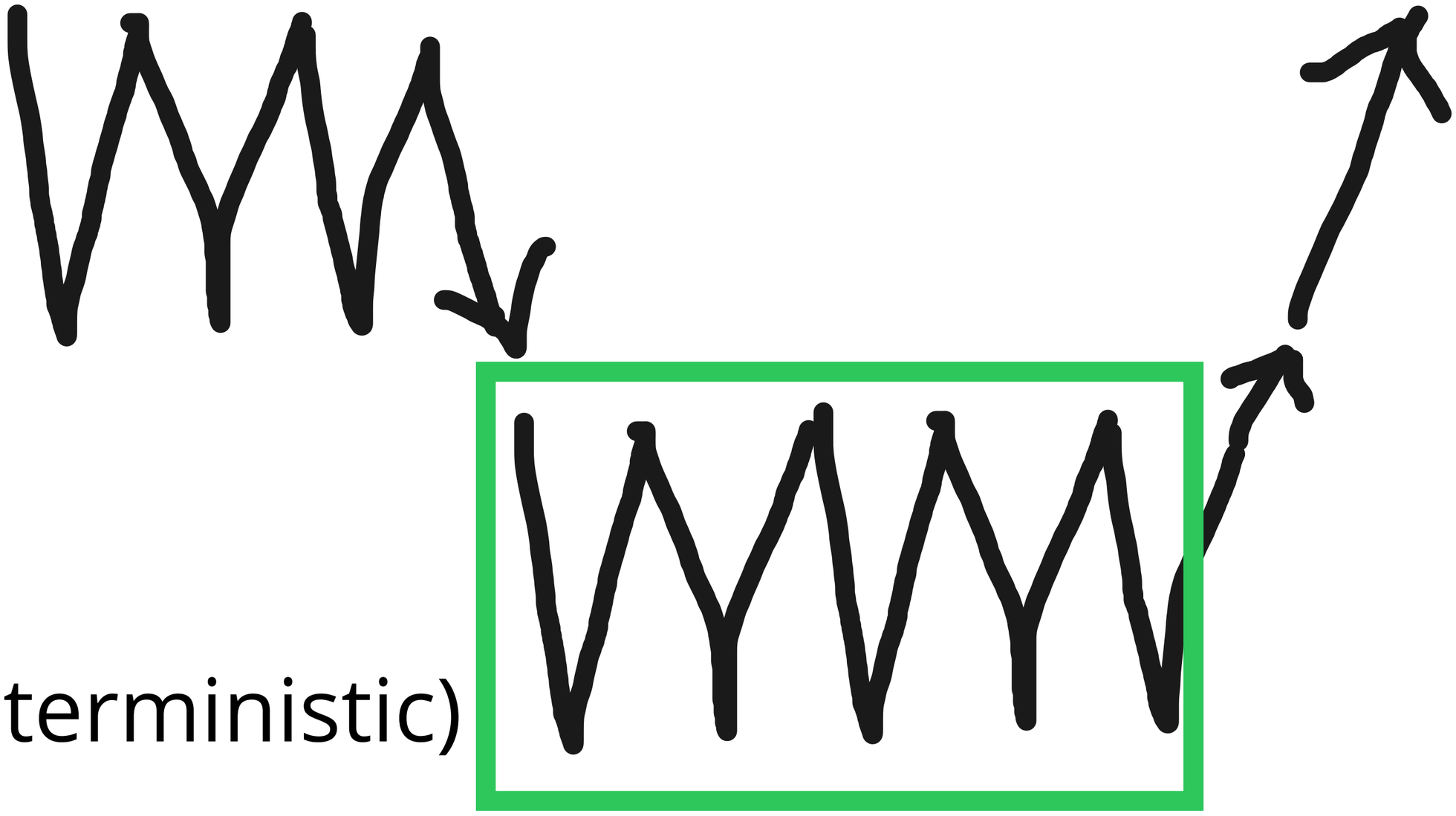
AI-Coding Lernen: Ein Weg in 10 Schritten: Ein Erfahrungsbericht
https://ai-coding.pro/de/blog/ai-coding-lernen-ein-weg-in-10-schritten-ein-erfahrungsbericht/

Copy & Paste with ChatGPT
Tab Code Completion
IDE with AI Chat and Agent
Agent with Auto-Run Mode in VM
More Technical Context
Rules for the Agent
Planning in Multiple Levels
Faster Feedback Loops
Learning from Mistakes
Relying More on Agents

Human

Agent

Tools (deterministic)



Plan (what) Plan (how) Coding Code Review Test (tech) Test (user)

Faster Feedback Loops

Quality Script (always, via MDC)

- PHPStan,
- Linter
  (PHP, YAML),
- Formatter,
- Automated Tests
  (Unit, Integration, HTTP)
- Automated E2E
  (Playwright, Chrome CLI)

Case by Case (prompt)

- Browse Tab (Cursor)
- curl CLI
- sql CLI
- chrome CLI
- MCP

Triggers for Feedback Loops 1

- Quality script (always, via MDC)
    - Show examples in Dash

Triggers for Feedback Loops 2

- Quality script (always, via MDC)
    - Show examples in Dash
- Case by case (prompt)

Prompt Example:

Implement, then click through the new workflow using Chrome via browser_navigate. Fix any errors and re-validate up to 5 times. Report how many cycles of fix+re-validate were needed and why they were needed.

10 Steps to Enable Fundamental How

AI-Coding Lernen: Ein Weg in 10 Schritten: Ein Erfahrungsbericht
https://ai-coding.pro/de/blog/ai-coding-lernen-ein-weg-in-10-schritten-ein-erfahrungsbericht/

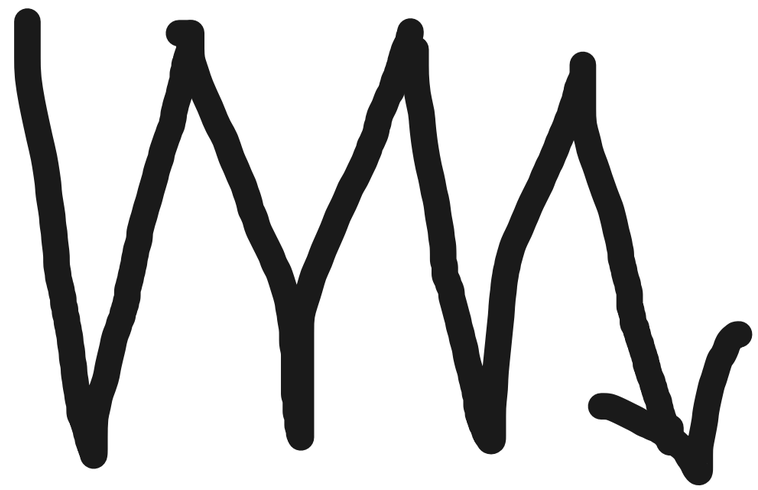
Copy & Paste with ChatGPT
Tab Code Completion
IDE with AI Chat and Agent
Agent with Auto-Run Mode in VM
More Technical Context
Rules for the Agent
Planning in Multiple Levels
Faster Feedback Loops
Learning from Mistakes
Relying More on Agents

Learning from Mistakes: Goal

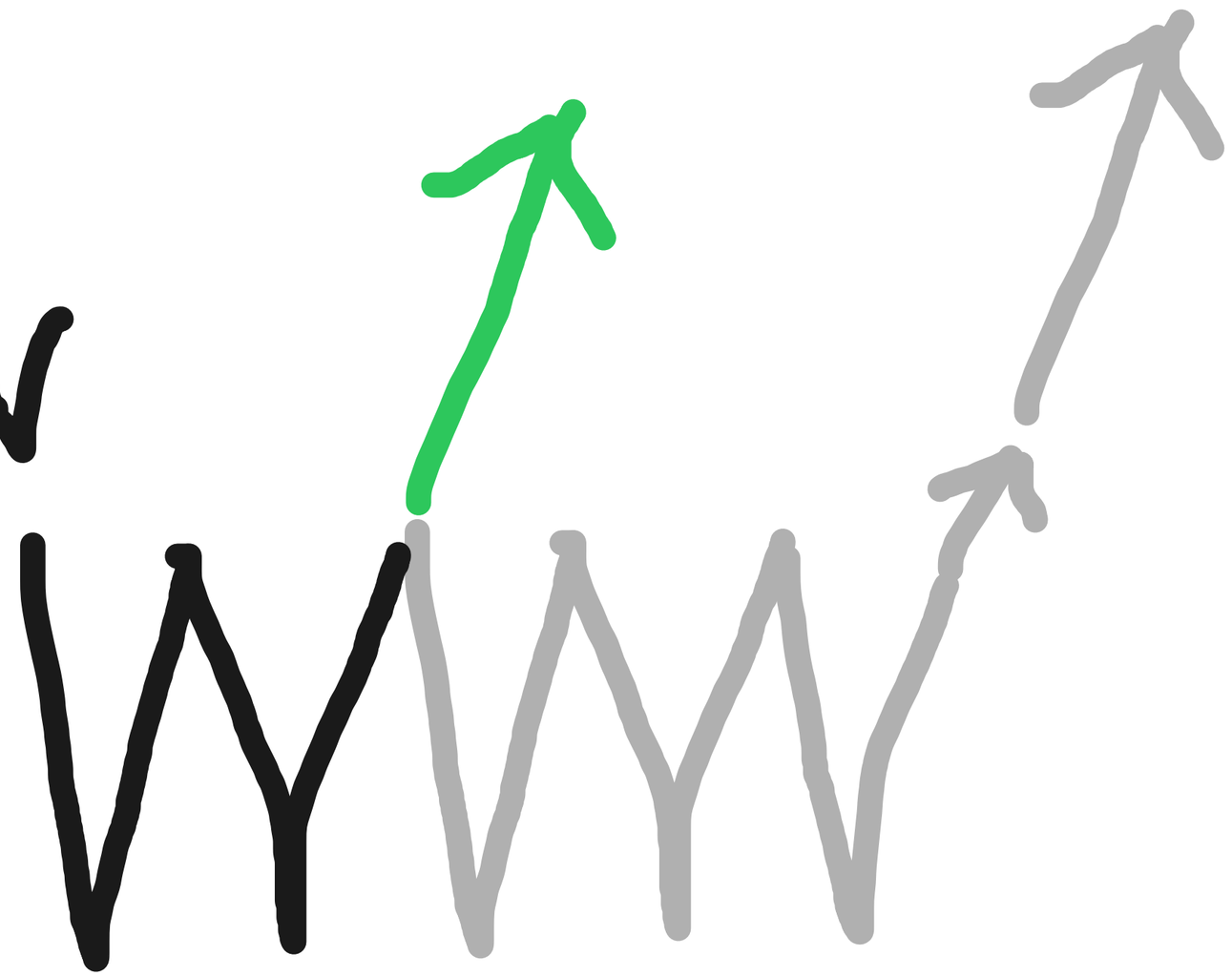
Goal: Achieve high quality results

- more quickly,
- using fewer tokens

Human



Agent



Tools (deterministic)

Learning from Mistakes: How

If the implementation prompt was:
Implement, then click through the new workflow using Chrome via browser_navigate. Fix any errors and re-validate up to 5 times. Report how many cycles of fix+re-validate were needed and why they were needed.

... agent working ... reports thing that went wrong

Learning prompt example:
Create an MDC rule to prevent thing that went wrong.

Normally:

Hello World

Hello World

Skip, and try things on your own codebase: small, medium, large

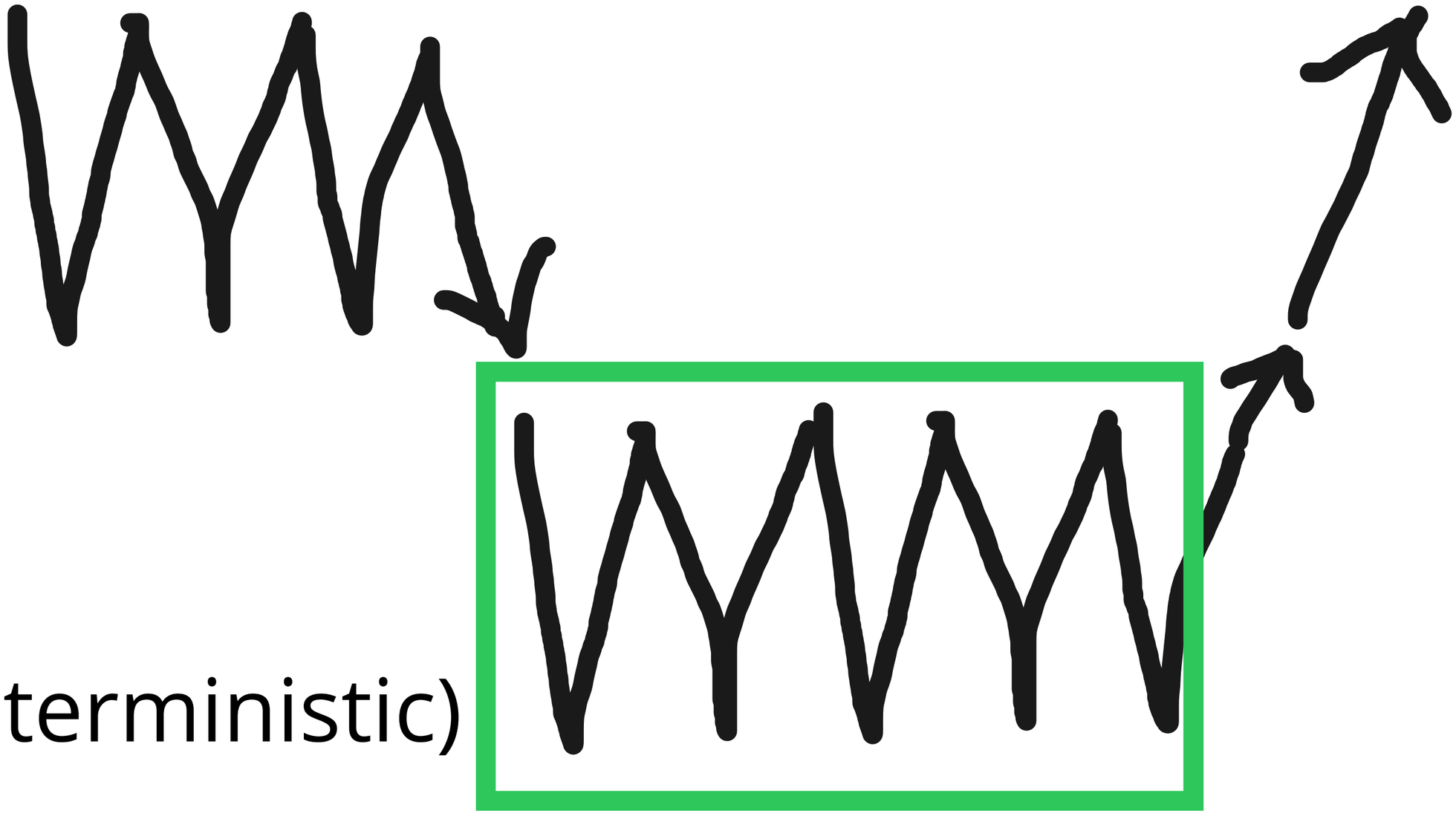
Keep in mind as plan b.

Prepare Legacy Coding

Human

Agent

Tools (deterministic)



Plan (what) Plan (how) Coding Code Review Test (tech) Test (user)

Prepare Legacy Coding

Which steps do we need?

- Tests
    - Very high coverage
    - Increase testability (if needed)
- Quality Tools: Everything? Or pre change flow?
    - PHPStan
    - Linter
    - Formatter
- MDC (when / how to execute tools, rule management)
- Container setup from Repo
    - Synthetic / anonymised Data?
- Deployment strategy?

Legacy Coding

Where do we start?

Architecture Change?

Directly build new things?
(and document current system super well?)

Architecture Change?

Architecture Vision?

Step by Step Executable Transition Strategy?

Both clear enough? Then prompt for transition plan.

Prompt together

Diverge Merge Pattern:

- Quietly and quickly think and write prompt ideas in parallel (create frames as we go)
- Looks at all ideas:
    - Merge OR
    - identify variants to try separately

Prompts: Log + Learn

- check prompts together
- log every prompt we want to try
- execute
- log success / failure
- try to understand

Accept Excitement / Stress

- deeply not deterministic
  (even at hardware level)
- surprises are normal
- try to automate control
  as much as possible, but:
- accept experimentation