ONE TO TWO DAYS FOR ENGINEERING AND PRODUCT TEAMS

Ship Faster With AI, Without Shipping Bugs

Hands-on work on your own codebase. Your engineers learn agentic coding habits that hold up in review, and how to build AI features that stay reliable.

DURATION
One to two days
Day two covers AI features

GROUP
Engineering teams
Engineers, tech leads and product managers

FORMAT
Your own codebase
In person or live online

OUTPUT
One reviewed PR each
Plus a team playbook

YOU NEED THIS IF

- Engineers use coding agents, but review queues are growing.
- AI-written code passes tests and still breaks in production.
- Every engineer prompts differently and nobody shares what works.
- You plan AI features, but nobody owns quality, cost or evals.

WHAT THE EVIDENCE SAYS

19% slower: experienced developers on their own repos with AI tools, while they believed they were 20% faster.

46% of developers distrust the accuracy of AI tools, up from 31% a year earlier.

66% name "almost right, but not quite" answers as their top frustration with AI.

Sources: METR randomised trial, July 2025 (16 developers, 246 tasks); Stack Overflow Developer Survey 2025.

WHAT YOU LEAVE WITH

01 **Habits that hold up in review**
Spec first, small diffs, tests as the contract and a checklist for AI-written code.

02 **A rules file for your repo**
Shared agent instructions, so every engineer starts from the same context.

03 **One reviewed PR each**
Built with an agent on a real ticket from your backlog, reviewed by a trainer.

04 **A working AI feature**
A prototype with structured output, an eval harness and a cost estimate.

THE AGENDA

Agents for Your Codebase, Then AI in Your Product

Day one makes coding agents safe to use on a real codebase. Day two covers building AI features your users rely on. One-day programs run day one only.

WHEN	MODULE	YOU LEAVE WITH
DAY ONE · AGENTIC CODING		
Morning	**How coding agents work** Context windows, tools, planning, and why agents struggle on large repos.	A shared mental model
Morning	**Context engineering** Rules files, scoped tasks and the right files in context. Written for your repo.	A rules file for your repo
Afternoon	**Plan, then code** Write a spec, review the agent's plan, then execute in small diffs on a real ticket.	One reviewed PR
Afternoon	**Tests, review and security** Failing tests as targets, reviewing AI diffs, leaked secrets and invented dependencies.	A PR review checklist
DAY TWO · AI FEATURES		
Morning	**Building with model APIs** Structured output, tool calling and retrieval over your own data.	A working prototype
Midday	**Evals** Golden sets, graders and regression checks, so quality does not drift between releases.	An eval harness
Afternoon	**Cost, latency and team workflow** Model choice, caching and streaming. A PR policy for AI code and how you measure impact.	A team playbook

FIVE HABITS WE DRILL ON DAY ONE

01 — Spec before code
Write what done means before the agent starts.

02 — Small diffs
One concern per change, so review stays honest.

03 — Tests are the contract
A failing test first, then the agent makes it pass.

04 — Read every line
You own what you merge, whoever typed it.

05 — Measure, do not guess
Track cycle time and review load, not feelings.

HOW IT RUNS

Before, During and After

BEFORE

- A call with your engineering lead to pick the repo and tickets.
- Read access for trainers, or a sandbox copy of the repo.
- Agent tools installed and approved for every engineer.

DURING

- Every exercise runs on your codebase, never a toy app.
- Trainers pair with engineers on real tickets.
- Each pattern ends with a review of what the agent got wrong.

AFTER

- The rules file, checklist and playbook committed to your repo.
- A day-30 review of cycle time and review load.
- Office hours for tech leads as the tools change.

WHERE IT FITS IN THE SHARP METHOD

S Spot
Covered by other programs

H Hands-on
Pairing on real tickets in your repo.

A Apply
Rules files and PR policy go into daily work.

R Results
Cycle time and review load at day 30.

P Practice
Covered by other programs

WHAT WE COMMIT TO YOUR REPO

Agent rules file
Project context and conventions every coding agent reads first.

PR review checklist
What reviewers check on AI-written changes.

Eval harness
Golden cases and graders for your first AI feature.

Team playbook
When to use agents, how to pair and how you measure impact.

QUESTIONS TEAMS ASK

Which languages and stacks?	Yours. We prepare on your repo, so the examples match your stack and conventions.
Is our code safe?	We sign your NDA first. We can work in a sandbox copy or on your own machines only.
Will this make our team faster?	Speed comes from habits, not tools. We measure cycle time and review load at day 30.

TOOLS WE TEACH: CURSOR · CLAUDE CODE · OPENAI CODEX · GITHUB COPILOT · ANTHROPIC API · OPENAI API · GEMINI API

NEXT STEP

Book a free 15-minute discovery call.

anandwana001@gmail.com
sharpfirst.school