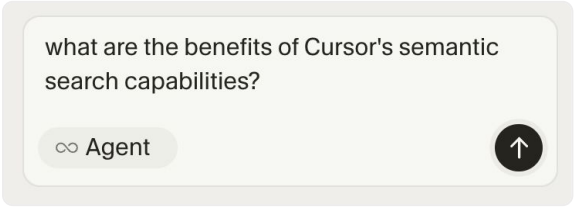

CURSOR
Product Intro &
Fundamentals of AI Coding

Key Resources

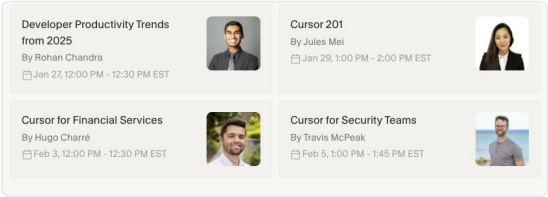
Cursor Docs

cursor.com/docs
+ AI Docs Assistant
+ “Cursor Learn” Track



Cursor Workshops

cursor.com/workshops
Live + Recorded



Cursor Learn

cursor.com/learn
Updated guidance for 2026.

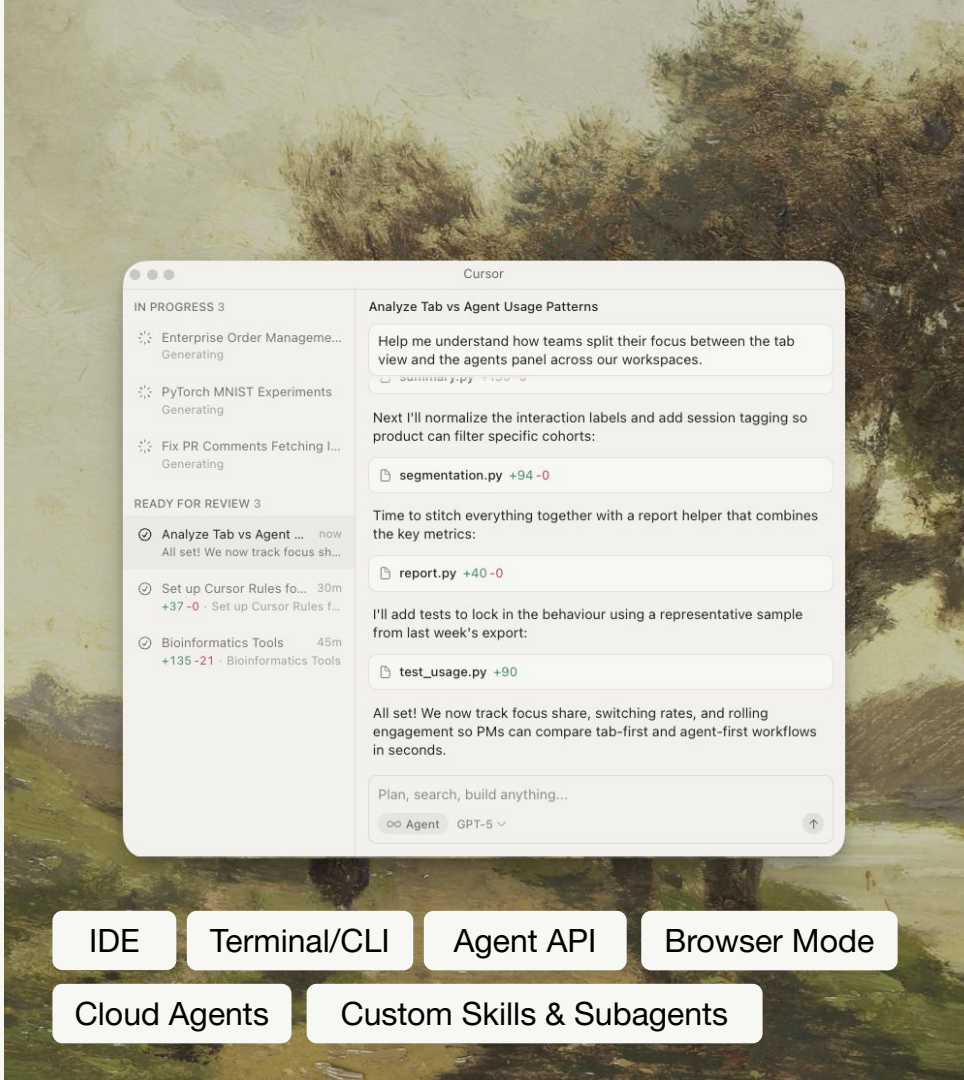
CURSOR

The Cursor Ecosystem

Cursor is the leading AI coding platform for professional software engineers.

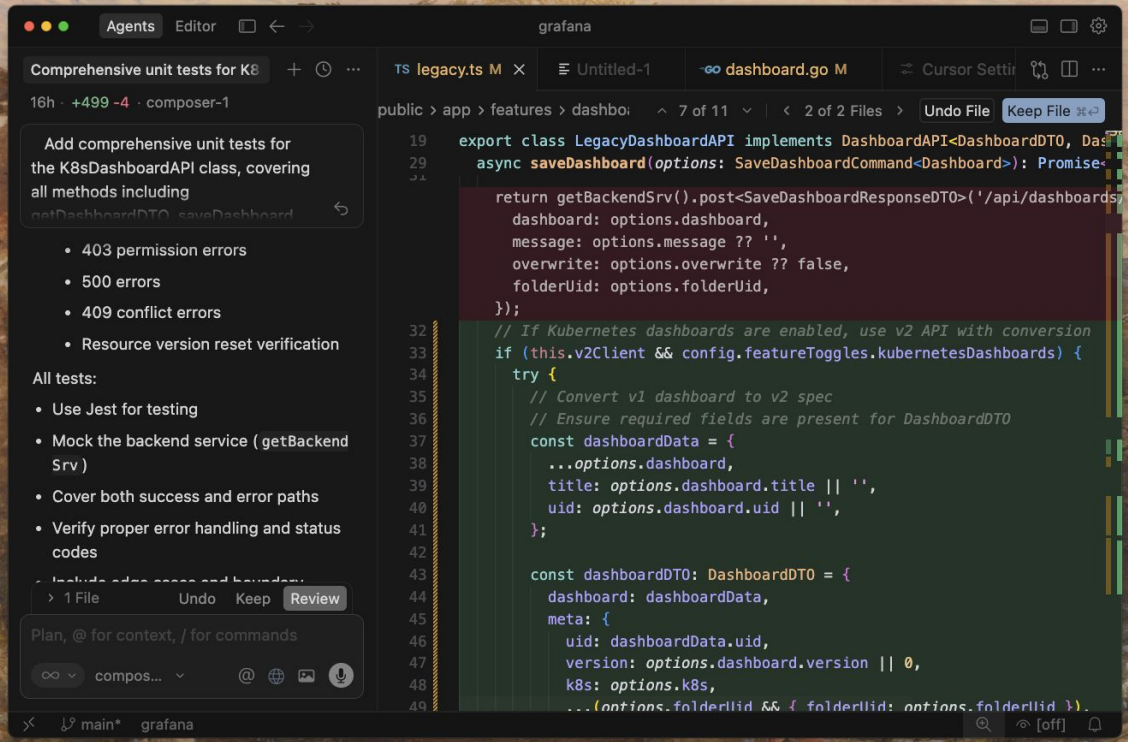
Access all frontier AI models, with built-in enhancements to make them smarter, faster, and more codebase-aware.

It's extremely effective for teams out-of-the-box, *and* allows for limitless customization on top.



(A few!)
Ways of Working

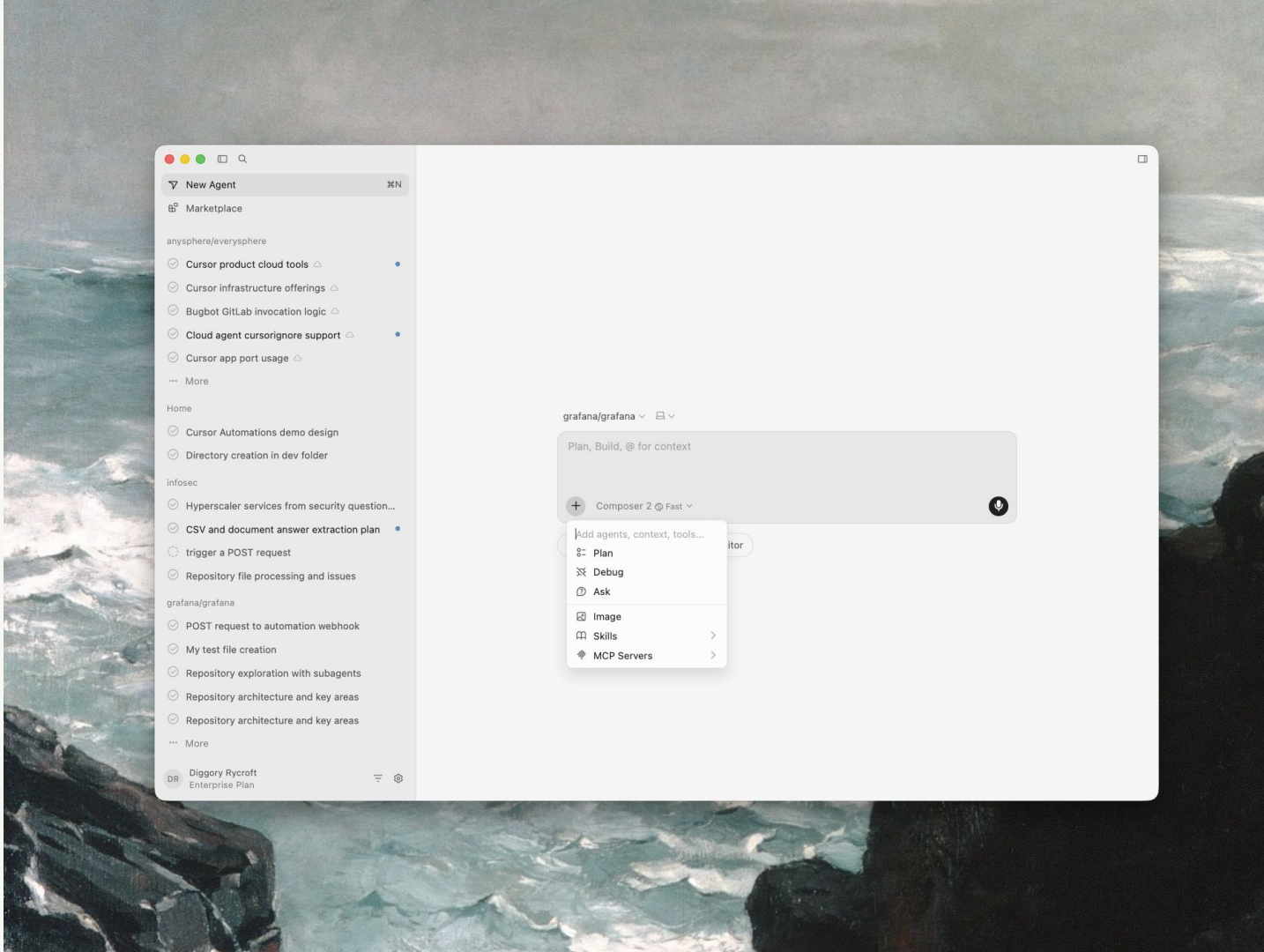
Code-Centric



(A few!)
Ways of Working

Code-Centric

Agent-First



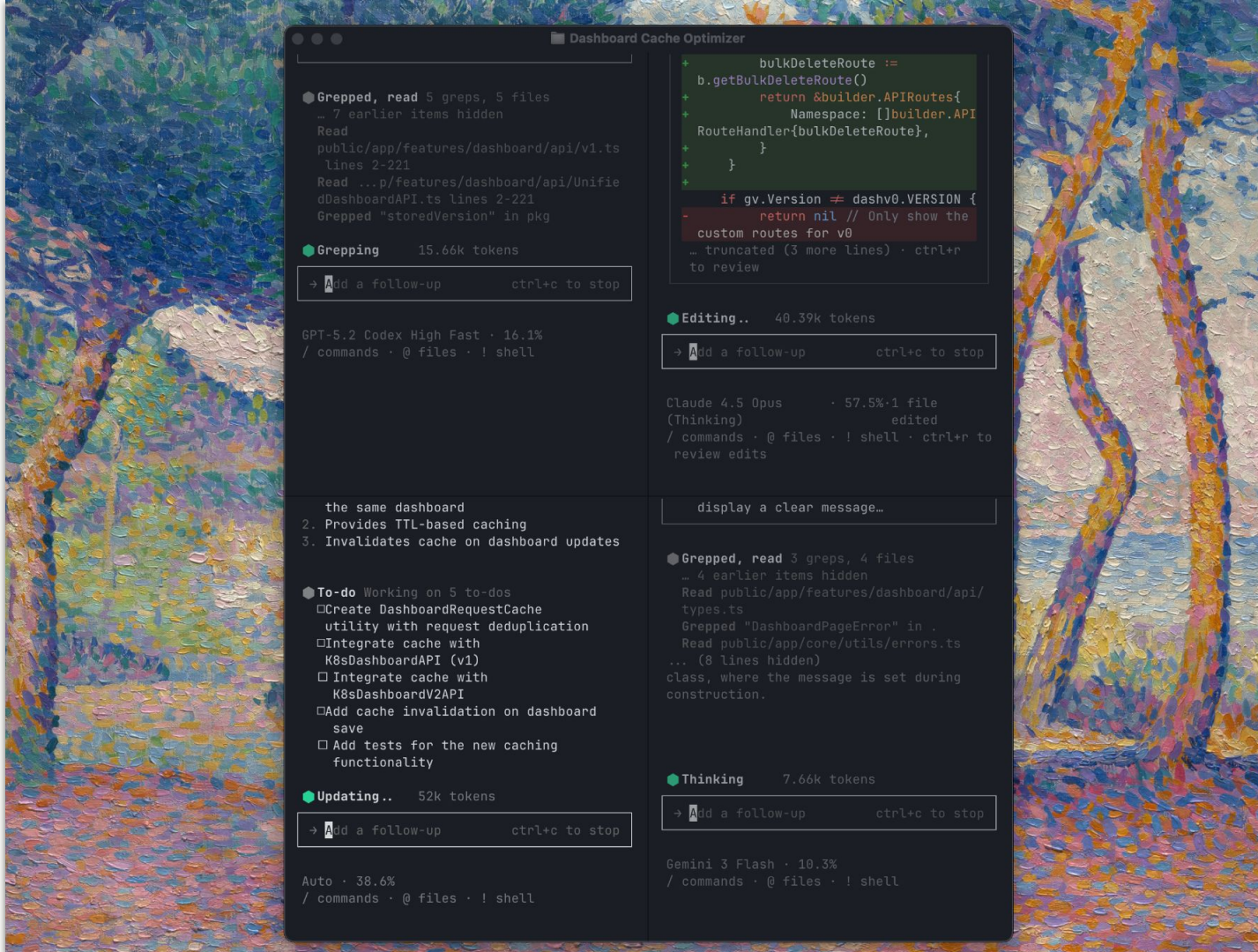
(A few!)

Ways of Working

Code-Centric

Agent-First

Multi-Agent Terminals



(A few!)

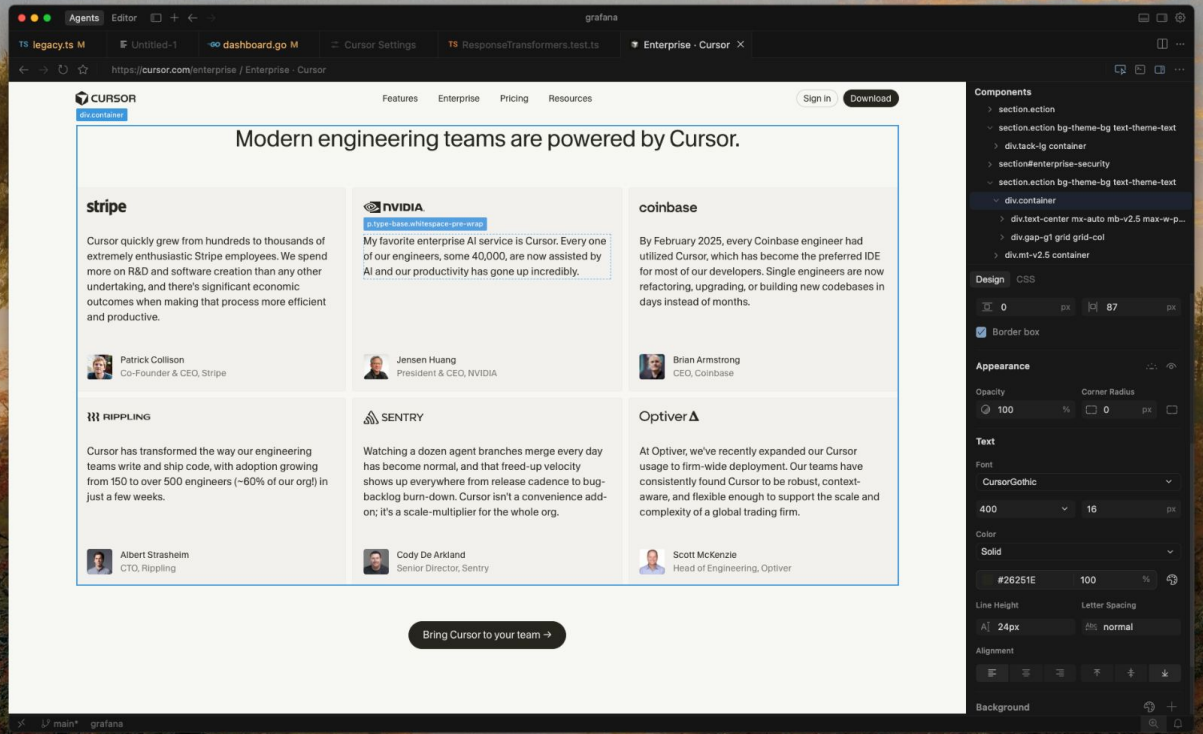
Ways of Working

Code-Centric

Agent-First

Multi-Agent Terminals

Frontend Design & Testing



(A few!)

Ways of Working

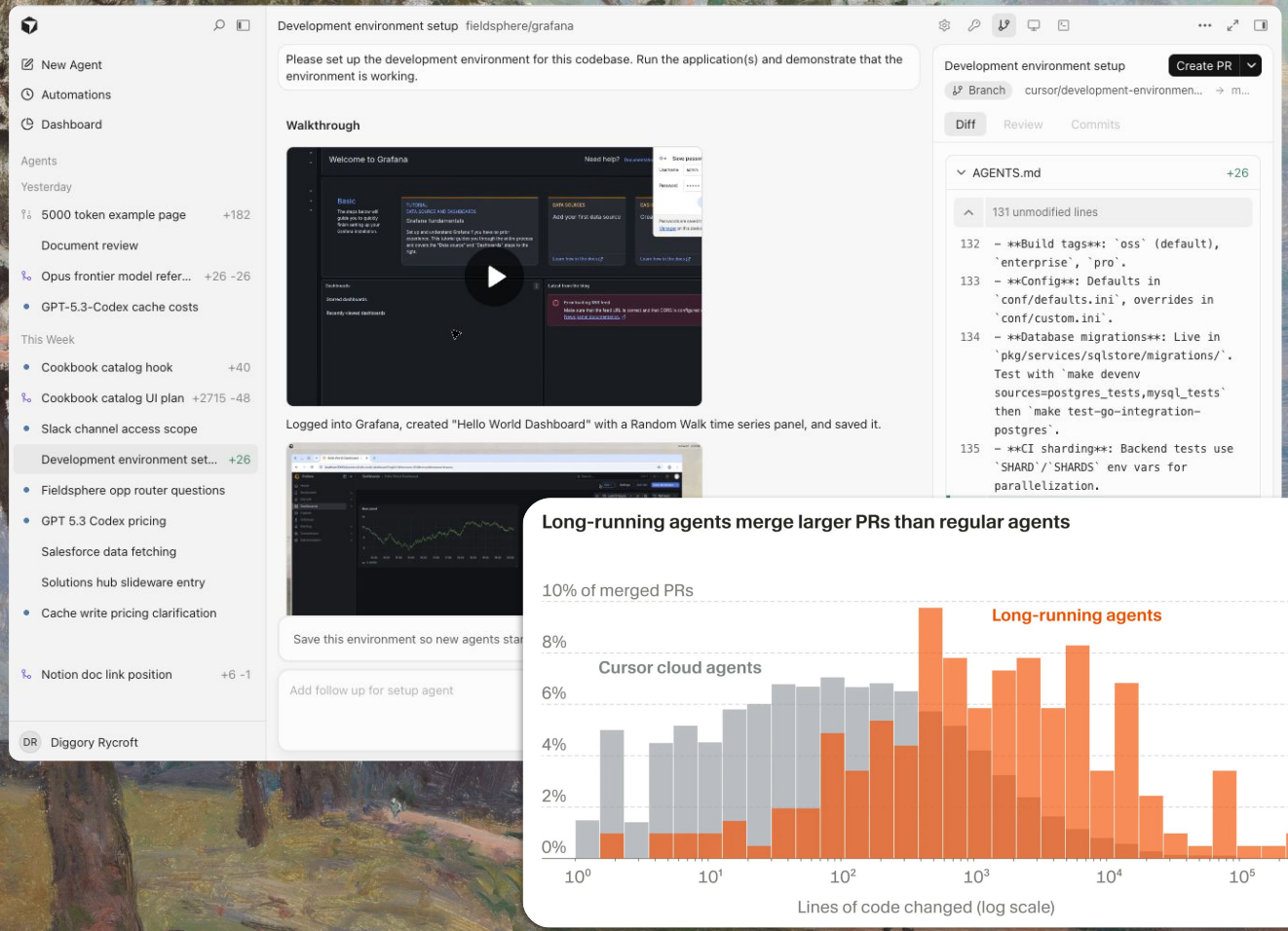
Code-Centric

Agent-First

Multi-Agent Terminals

Frontend Design & Testing

Long-Running Async Agents



(A few!)

Ways of Working

Code-Centric

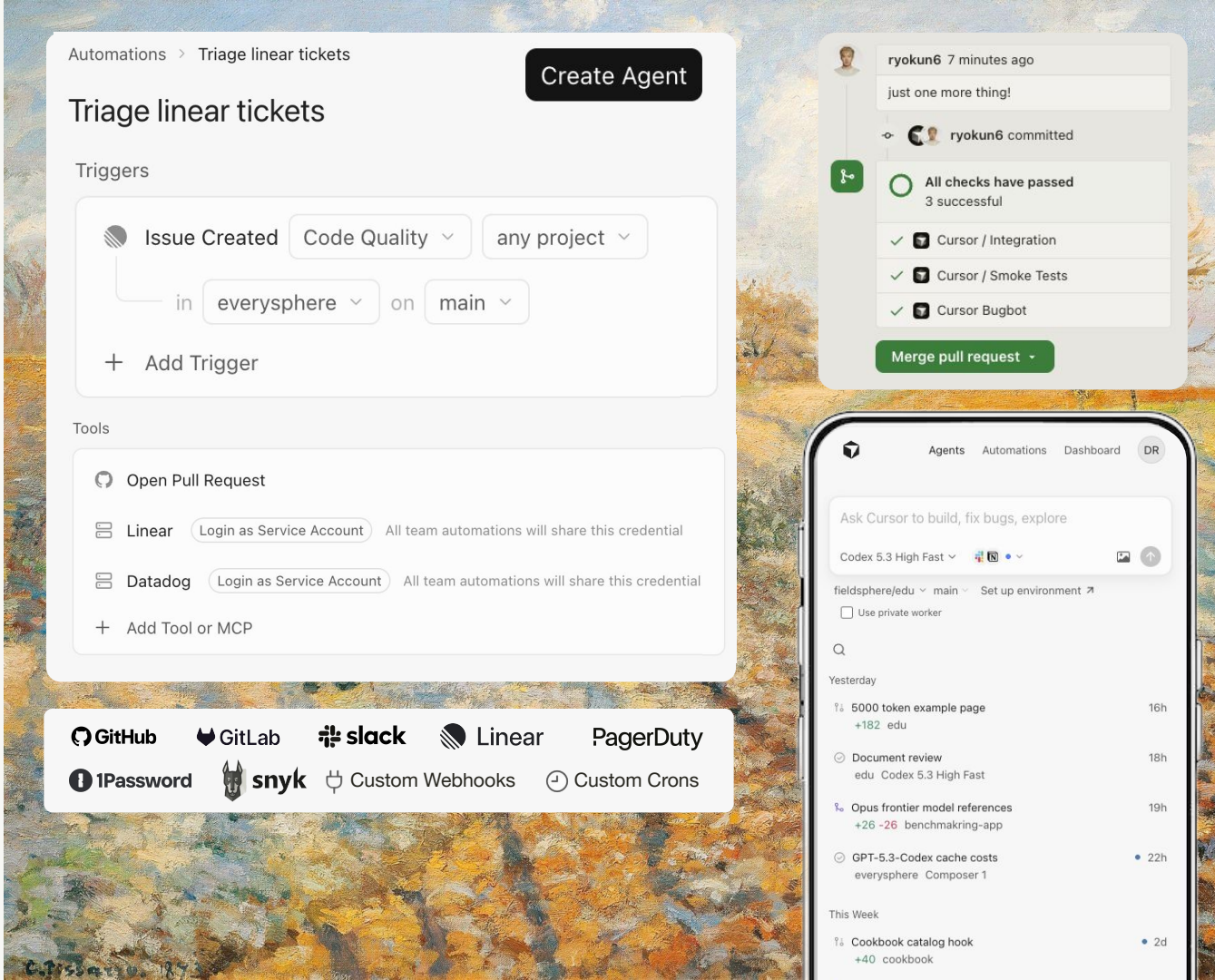
Agent-First

Multi-Agent Terminals

Frontend Design & Testing

Long-Running Async Agents

Automations across the SDLC



(A few!)

Ways of Working

Code-Centric

Agent-First

Multi-Agent Terminals

Frontend Design & Testing

Long-Running Async Agents

Automations across the SDLC

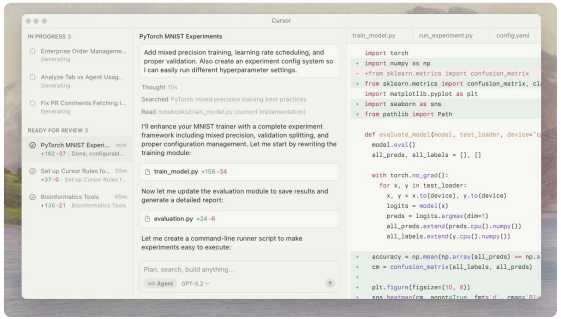
Inside your IDEs of choice

Where can I use Cursor?

Cursor Desktop

The agentic development platform made for professional SWE teams.

Adaptable to every level of agent autonomy and every style of work.



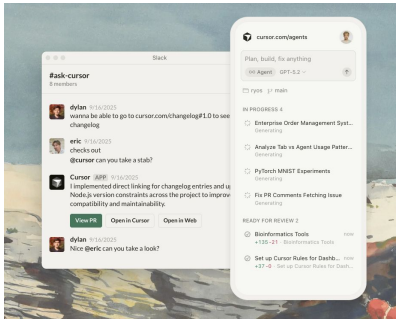
OpenVSX Extension Registry

cursor.com/download

Cloud Agent Platform

Web, Mobile, Slack, API

Trigger & manage sync + async agents and long-running tasks from anywhere.



cursor.com/agents

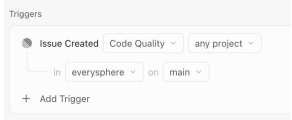
Bugbot Code Review

GitHub GitLab

Automatic bug identification, and PR autofix, at the org & repo level.

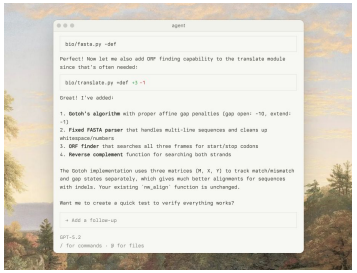
Automations

Always-on Agents augmenting the entire SDLC.



Cursor CLI

Plug Cursor into any IDE or CI/CD process.



```
curl https://cursor.com/install -fsS | bash
```

JetBrains IDEs

Available across IntelliJ, PyCharm, and the entirety of the JetBrains family. A frontier AI coding experience for devs in Java, Kotlin, Python, .NET, C#, and more.

CURSOR

10-Second Quickstart

Download Cursor Desktop: cursor.com/download

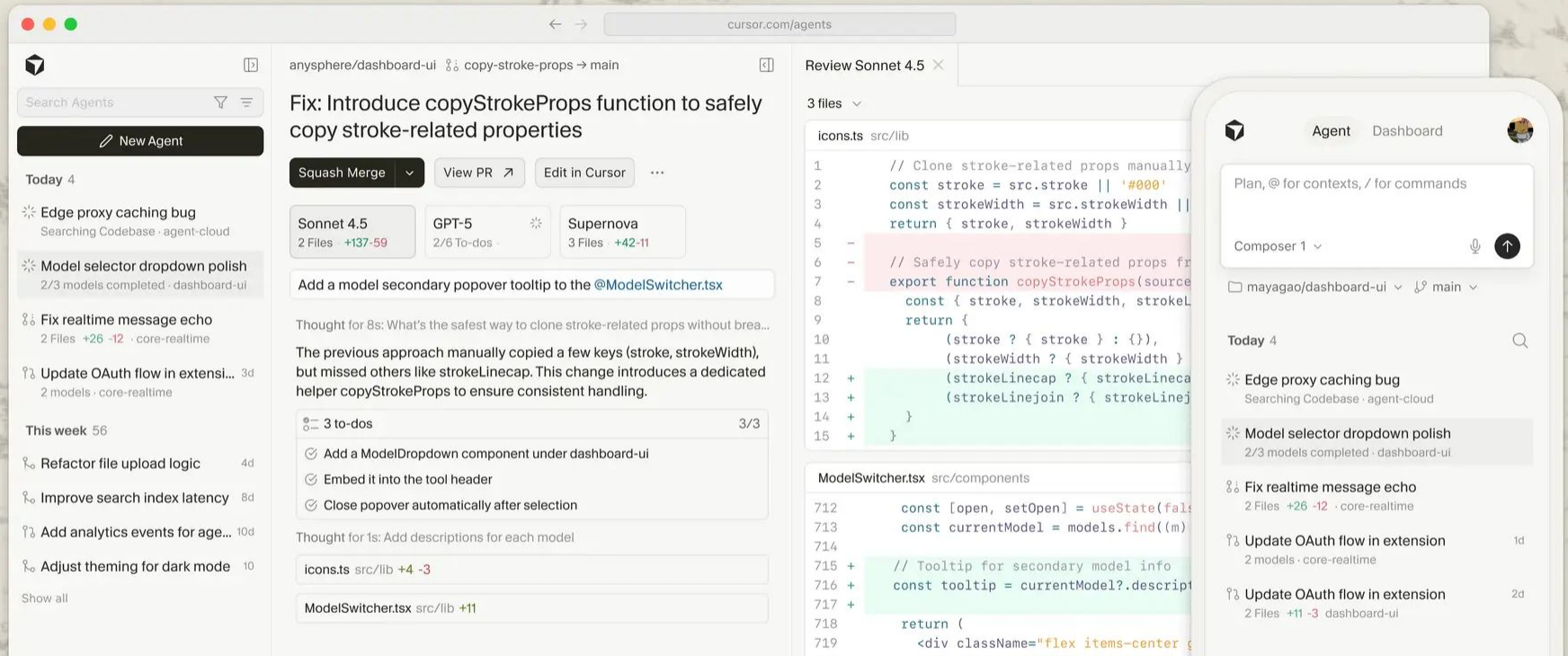
✔ VS Code settings, extensions, keybinds can be auto-imported on startup

✔ Immediately use any existing third-party Rules, Skills, & Subagents
(don't worry if this is new to you - we'll teach you about these!)

i You can access your Cursor Settings via ⌘+Shift+J / Ctrl+Shift+J
Or use the Command Palette (⌘+Shift+P) and type in what you need.

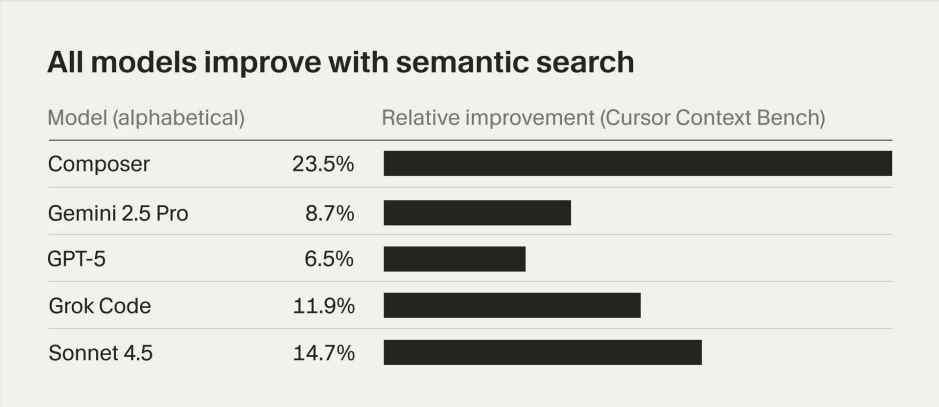
CURSOR

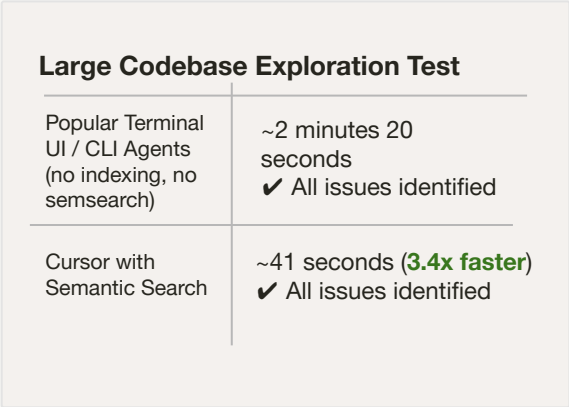
Demo: Getting Oriented



3 Need-to-Know Tips

- Models in Cursor *understand* your codebase, through **Semantic Search** & Cursor's custom embedding model. This makes models smarter, faster, and more conformant with existing patterns - especially on large codebases. Be ambitious!





https://cursor.com/blog/semsearch

CURSOR

3 Need-to-Know Tips

- Models in Cursor *understand* your codebase, through **Semantic Search** & Cursor's custom embedding model. This makes models smarter, faster, and more conformant with existing patterns - especially on large codebases. Be ambitious!

- **Plan Mode** is a superpower for more complex tasks.
  - 💡 Press shift+tab in chat to quickly transition into Plan Mode.

- Build Skills and Subagents progressively, not all upfront.
  - 💡 Use /create-skill or /create-subagent to interactively adapt work into reusable agent abilities, for you and your team.

CURSOR

Model Choice 101

Cursor is a model-neutral platform, with best-in-class harnesses for every frontier model.

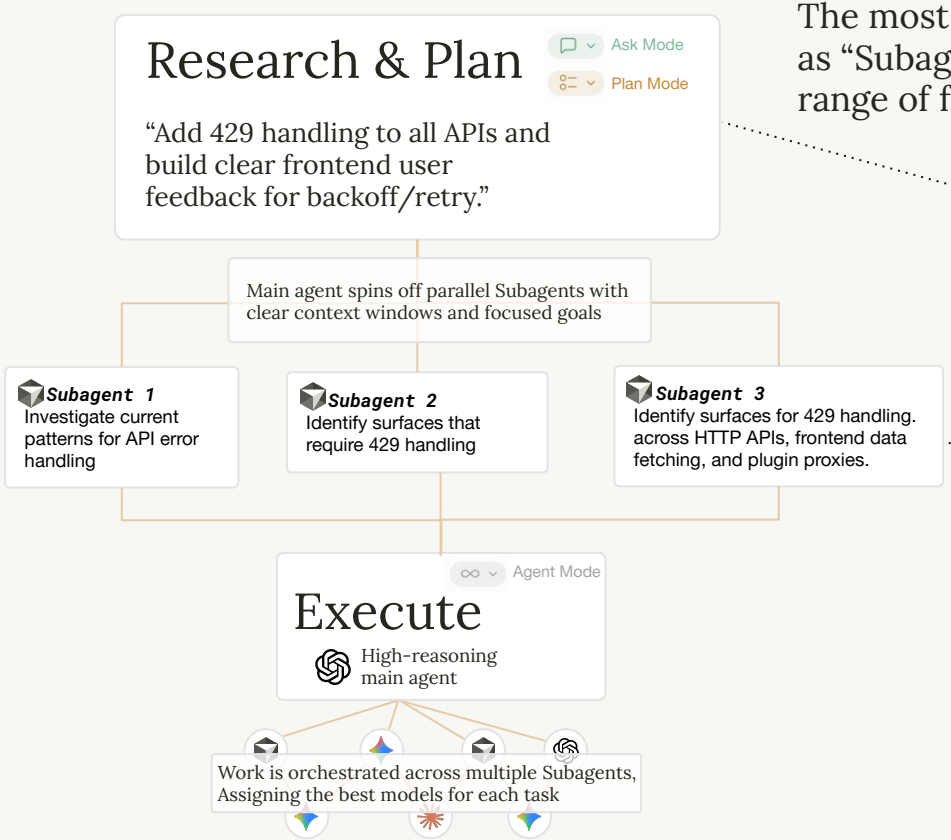
With some basic familiarity with Model Selection, you can use the best model for each task, and make efficient use of tokens.

Top Frontier Models

The most intelligent & capable frontier models.
Excellent for long-horizon tasks and asynchronous work, and suitable for Cursor's specialized ***Long-Running Agent Harness***.

Model	Spend Efficiency (input tokens per $100)	Terminal-Bench 2.0 Score
GPT-5.5 OpenAI	20 million	**82.7%**
Claude 4.7 Opus Anthropic	20 million	**69.4%**
Composer-2 Cursor	200 million	**61.7%**

Multi-Model Agent Flows



The most capable models are now able to invoke *other* models as "Subagents". Cursor lets your main agent choose from a full range of fast, intelligent, and capable models.

High-Reasoning Frontier Models
The smartest, most capable models. Great for planning, deep codebase exploration, and coordinating Subagents.

GPT-5.5 Opus 4.7 Composer 2

Fast, Cost-Effective, Focused Models

Gemini 3 Flash

Composer 2 (Dual-Purpose)

CURSOR

Token Efficiency Pointers

Rest assured, Cursor handles all *major* measures automatically.

- Caching Prompts & Context
  (Most "token usage" on your usage report just Cache Reads: ~90% more cost-effective than direct inputs)
- Dynamic Context Discovery
  (reduces total agent tokens by ~46.9%)
- Semantic Search
  (improves agent accuracy and speed in navigating large codebases)

As a user, consider these concepts for even more efficiency:

Choose the right model for the job:
Model prices and relative strengths can vary widely! Take a moment to understand relative pricing, or use Auto to keep it simple.

Use Plan Mode before Executing:
Align on an approach upfront, so that every token goes toward desirable outcomes.


Thank You
CURSOR