MADRE

madre.run/en/manual

Learn to use MADRE in 10 minutes.

You'll install it, connect the AI you already use and ask it your first question. If you've never used the Terminal, don't worry: I'll tell you what to copy and where to paste it.

Manual · MADRE 1.1.3

---

10 min
to your first answer

10 steps
the page remembers which ones you finished

No coding
you just copy and paste

STEP 1 · 30 S

What MADRE is and what you need.

MADRE is a chat that opens in your browser. From there you give work to the AIs that code: Codex, Claude Code, Gemini CLI, OpenCode and Cursor. You can talk to several of them in the same conversation. It runs on your computer, it's free and it doesn't ask you to create an account.

Tick what you already have

- ☐ **A Mac, a Linux computer or a Windows computer.** On Windows, the MADRE installer sets up everything it needs. You'll see it in step 2.
- ☐ **Node 22.5 or newer (on Mac and Linux).** Node is a program MADRE needs to run. If you don't have it, step 2 shows you how to install it. On Windows, the installer installs it.
- ☐ **An account with some AI.** Your ChatGPT or Claude account, or an API key from another company (step 4 explains what that is). MADRE uses yours and charges nothing extra.

IF YOU'RE NEW

Four words you'll see often

Terminal
A window where you type commands. Here you only use it to open MADRE.

Agent
What we call each AI that codes. It reads the files in your project, answers you and, if you give it permission, changes them.

Project
A folder on your computer. Each folder gets its own chat in MADRE.

Mode
The permission you give the AI in each message, from #0 to #4. If you don't pick one, the AI only reads.

⚑ **You're good if** your computer is ready and you know which AI account you'll sign in with.

STEP 2 · 2 MIN

Install MADRE.

Pick your system. When you see a command, hit Copy, paste it in the Terminal and press Enter.

MACOS

1 Open the Terminal

Press `⌘` + `Space`, type **Terminal** and press `Enter`.

2 Install Node

If you have Homebrew, paste this:

```
brew install node
```

If you don't know what Homebrew is, download the Mac installer from nodejs.org and open it like any other program.

3 Open MADRE

Paste this line. The first time it asks «Ok to proceed?», which means «Can I download MADRE?». Type `y` and press `Enter`.

```
npx @jossuealcala/madre
```

LINUX

1 Open the Terminal

On many systems it opens with `Ctrl` + `Alt` + `T`. If not, look for **Terminal** in your apps.

2 Install Node

Install it with your system's package manager or from nodejs.org. Then check which version you got:

```
node --version
```

It should say v22.5 or a higher number. If it shows a lower one, install the one from nodejs.org.

3

Open MADRE

Paste this line. The first time it asks «Ok to proceed?», which means «Can I download MADRE?». Type `y` and press `Enter`.

```
npx @jossuealcala/madre
```

MADRE opens by itself in your browser, at `http://127.0.0.1:4317`. That address belongs to your computer: nobody else can open it from the internet. If it's already taken, MADRE uses the next one.

Don't close the Terminal while you use MADRE, because that's where it's running. To shut it down, use the power button at the end of MADRE's bar or press `Ctrl` + `C` in the Terminal. To open it another day, paste the same command. It always downloads the newest version.

IF YOU'RE NEW

Want to use MADRE on a server?

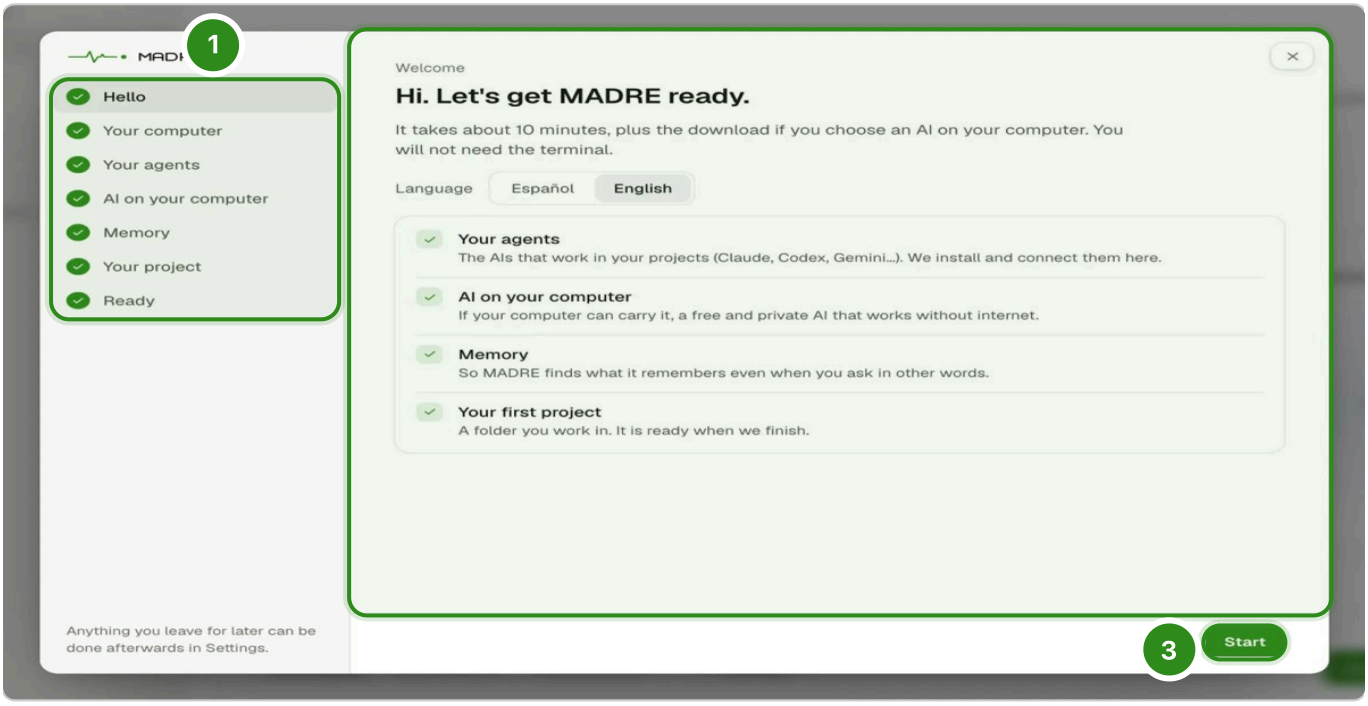
MADRE has no password. That's why it only accepts connections from the same machine (`127.0.0.1`). Don't open its ports to the internet. To use it on a server, connect from your computer with an SSH tunnel: `ssh -L 4317:127.0.0.1:4317 -L 4318:127.0.0.1:4318 -L 4319:127.0.0.1:4319 -L 4320:127.0.0.1:4320 user@server`. On the server, run `npx @jossuealcala/madre --no-open` and open `http://127.0.0.1:4317/salas` in your computer's browser. Home uses port 4317 and each project the next free one. If you open more projects, add more `-L`. Sign in with your own user, not root.

You're good if you see MADRE open in your browser.

STEP 3 · 2 MIN

The first time, MADRE guides you.

When you open it for the first time, MADRE shows you a seven-step welcome. That's where you connect your AIs and pick your first project, all with buttons. If you don't have a project, MADRE makes a test one for you. When you finish, it takes you to write your first message.



1. The seven steps are on the left. You can go to any of them; the ones you finish get a tick.
2. Each step tells you what it is and gives you the button to do it: connect an agent, download the AI or choose your folder.
3. «Continue» takes you to the next one. Whatever you leave for later can be done in Settings.

Closed it? You can open it again from Home, in **Settings › Projects › See the welcome again**. It picks up from the step where you left off.

You're good if you finished the welcome or you know where to open it again.

STEP 4 · 1½ MIN

Connect your AIs.

You connect each AI only once and it's ready for all your projects. One is enough to get started.

Start with the account you already have

Codex OpenAI

You sign in with your ChatGPT account. MADRE opens your browser so you can sign in. An OpenAI API key works too.

Claude Code Anthropic

Pick **Claude account** if you pay for a Claude subscription, or **Anthropic Console** if you have API credit. The page gives you a code and you paste it in MADRE.

Gemini CLI Google

You need an API key from Google AI Studio. Google no longer lets Gemini CLI sign in with a free personal account.

OpenCode Many providers

It doesn't bring an AI of its own. You connect the AI company you already use by pasting its API key.

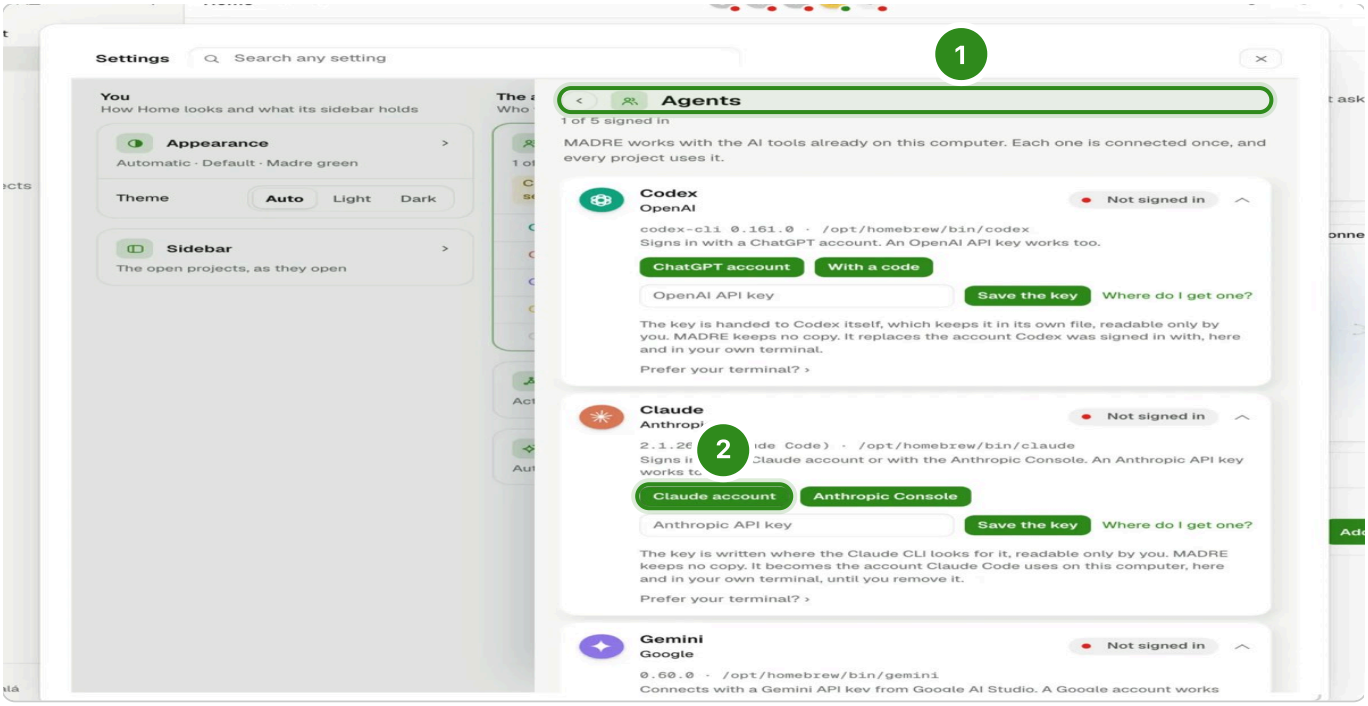
Cursor Cursor

If you already have Cursor and its program for the Terminal, MADRE finds it. You sign in with your Cursor account from its button.

1 **Says «Not installed»?**
Click **Install** and MADRE installs it.

2 **Sign in**
Each AI has its own button to sign in. When it says «Connected», it's ready.

3 **Click Continue**
You can connect the others later, in Settings › Agents.



1. In Home, open Settings (the gear, bottom left) › Agents. The first time, the welcome takes you there.
2. Open an agent that says «Not signed in» and press its button, such as «Claude account». You sign in with your account on the provider's page. If you do not have it, you will see Install.
3. Once it says «Connected», you can talk to it.

IF YOU'RE NEW

What's an API key?

It's a long password the AI company gives you so another program can use your account. You paste it only once. MADRE keeps it where that AI looks for it, protected under your user, and doesn't keep a copy.

Rather use the Terminal? These are the commands to sign in

Codex

```
codex login
```

Claude Code

```
claude auth login
```

Gemini CLI

```
gemini
```

Then type `/auth` and choose how to sign in.

OpenCode

```
opencode auth login
```

Cursor

```
cursor-agent login
```

You're good if at least one AI says «Connected».

STEP 5 · 1 MIN

Pick your project's folder.

In MADRE, a project is a folder on your computer. It can be your app, your website or an empty folder to start something new.

1 **Add your projects**

MADRE suggests the folders that look like projects. Pick the ones you want and click **Add and continue**.

2 **Open one**

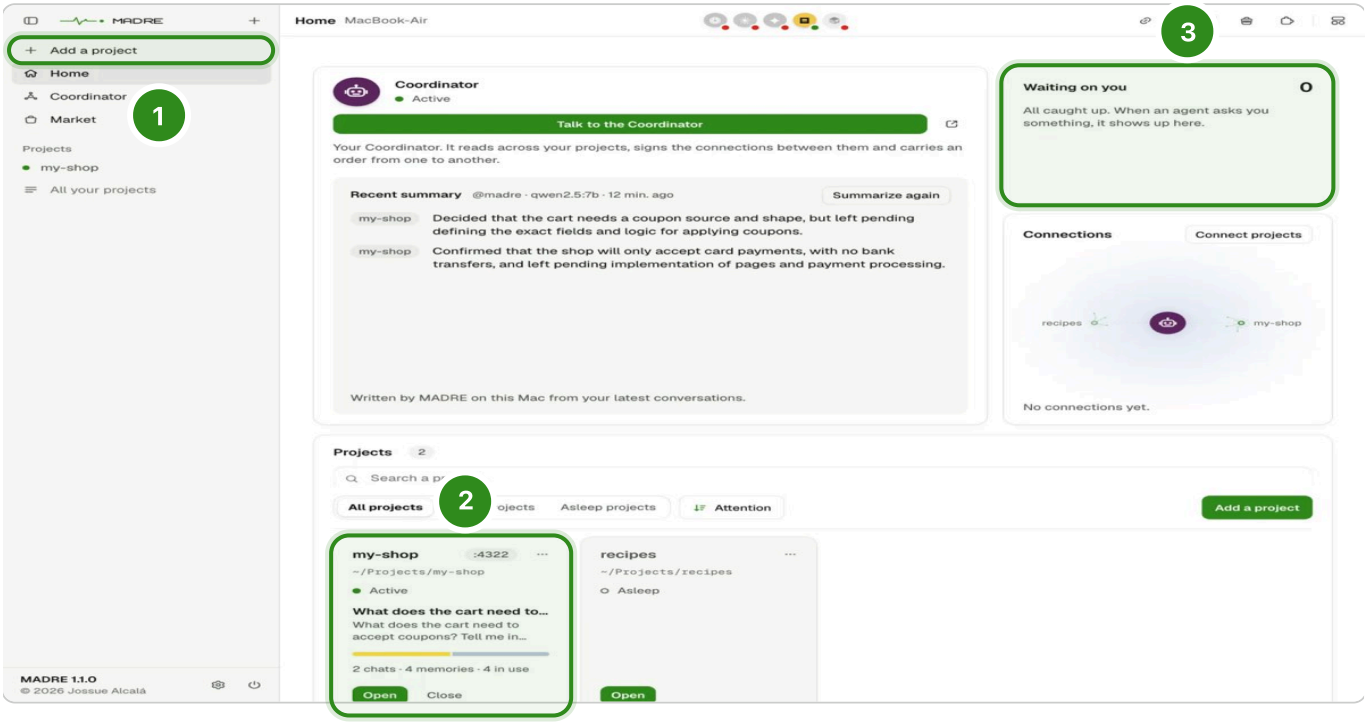
Pick the one you'll work on today and click **Done**. The others stay paused until you open them.

IF YOU'RE NEW

Don't have a project?

Create an empty folder, for example «my-first-app», and add it. In Ideas to build there are sample projects with the messages you can write to the AIs.

Then you land on **Home**. There you see all your projects, what happened in each one and which one is waiting for your answer. From there you open each project's chat. To come back another day, use the same command as in step 2.



1. In the left column, press «Add a project» and choose the folder.
2. It shows up under Projects. Press Open on its card to go in.
3. «Waiting on you» gathers what was left open: a question from an agent, a permission or a reminder.

You're good if your project shows up in Home and, when you open it, you see its chat.

STEP 6 · 1½ MIN

Write to an AI.

Below the chat there's a text box. In your message you put which AI you're talking to, what you let it do and what you want it to do.

THIS IS HOW YOU WRITE IT IN MADRE

```
@claude #1 Explain what this project does, no jargon.
```

Who answers

Claude Code, with your Claude account or your Anthropic Console.

With which permission

#1 Chat only

Does it ask for your key?

No.

What happens

It reads your project and answers. It changes nothing.

Four symbols you can use in the box

@ Who

`@claude` writes to Claude. Type `@` and the list shows up.

 # What you allow

From `#0` to `#4`. If you don't put one, it uses `#1`, which only chats and changes nothing.

! Which file

`!src/app.js:12-20` shows it only those lines of that file.

/ A command

Opens the command menu. For example, `/stopall` stops everything.

`Enter` sends the message. To start a new line, use `⌥` + `Enter` (`Alt` + `Enter` on Linux).

On each answer you see which AI replied, with which permission, which model it used and how many tokens it spent. If one takes too long, click the little square next to its timer and only that

one stops. **Stop all**, in the bar, stops them all.

IF YOU'RE NEW

What's a token?

It's the unit used to measure an AI's work. A token is roughly a piece of a word. Your plan or credit gets used up in tokens, and MADRE tells you how many each answer used.

You're good if an AI already answered something about your project.

STEP 7 · 1 MIN

Use several AIs in the same chat.

All the AIs read the same conversation. That's why you can ask one to continue what another one did, without explaining anything again.

What having several is good for

One does, another reviews

Claude makes a change and you ask Codex to review it. Codex already saw everything that happened in the chat.

```
@codex review what Claude
just did and tell me if it
missed anything.
```

The same question to several

Put several names at the start of the message. You see your question once and each AI answers separately.

```
@opencode @claude what is the
cart missing to take coupons?
```

A round table

Up to four AIs answer, one after another, and @madre sums up where they agree and where they don't. You need your local AI (step 9).

```
@madre, ask the team: where
should I start improving this
project?
```

If one runs out of plan

If Claude hits its usage limit, you write to another one and it carries on where it left off.

```
@gemini keep going with what
Claude was doing.
```

To spend less

Give the heavy work to the AI you pay for. Ask your local AI, which is free, about what was already discussed.

```
@madre what did we decide
last time about the design?
```

Claude, OpenCode and Codex can ask other copies of themselves for help, called subagents. They do it with the same permission you gave them, from `#1` to `#3`. While they work you see a small box for each one. At the end, the answer says how many helped, for example «Helped by 2 subagents», and there you can see what each one did.

Want to see several AIs work together? In Ideas to build there are two examples I tested. In one, Claude makes a plan from a single message and the others carry it out. In the other, they get together to share ideas and end up making an app.

How they remember what was said

01

Before answering

Each AI gets the latest messages of the chat. In the conversation you see it as a line that says HANDOFF and how many messages it got.

02

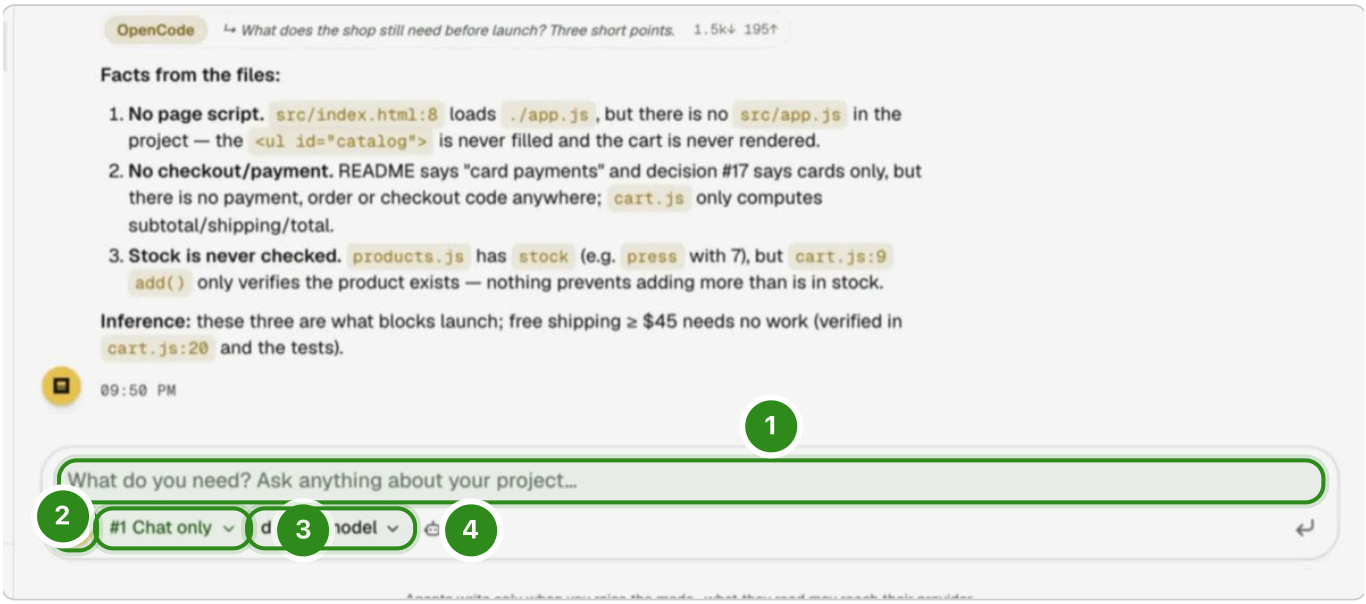
Every 10 messages

Or after 10 minutes with no activity, MADRE writes down what matters in short notes: what was decided, facts about the project, how you like things and what's still pending.

03

Days later

When they're needed, those notes are passed to whichever AI is answering, even if another one made the decision.



1. Type in the box and press ↵ to send.
2. The circle on the left is the agent that will answer. Tap it to choose another.
3. The tag with # is the permission. With #1 the agent reads and answers you, but changes nothing.
4. Beside it you choose the model and, for some agents, how much it thinks. If you are not sure, leave it.

Your local AI writes those notes, at no cost. If you don't have one, the cheapest AI you have connected writes them. You can see them and delete them in the **Memory map**.

Do you also use an editor like Cursor or VS Code? The **Continuity** module saves where the work stands inside your project, in the `.ahp/` folder. That way your editor knows too. From MADRE you check it with `/ahp status`.

You're good if you know how to ask one AI to continue what another one did.

STEP 8 · 1 MIN

Give them permission to change files.

If you don’t pick a permission, the AIs only read. For them to create or change files, you raise the permission. No AI can raise it by itself.

● TRY EACH PERMISSION, RISK FREE

These examples only read or work in a `practice/` folder, so your project doesn’t change. If you don’t have that AI, swap the `@name` for one you do have.

`#0 Incognito` For asking something without it being saved in the chat or the memory.

```
@claude #0 What would you call an app for splitting bills with friends? Give me 5.
```

What happens: It answers and, when you reload, it’s gone.

```
@gemini #0 Explain what an API is like I’m 12.
```

What happens: None of this goes into the project’s memory.

`#1 Chat only` For understanding your project or asking for opinions. Nothing changes.

```
@claude #1 Explain how this project is laid out: what’s in each folder.
```

What happens: It reads and answers.

```
@codex #1 What three things would you improve in this project? Don’t change anything yet.
```

What happens: You get ideas you can later ask for in #2 or #3.

```
@claude #1 What’s missing before I can publish this project?
```

What happens: A to-do list, without touching anything.

`#2 Create` For creating new files. It doesn’t touch the ones you have.

```
@claude #2 Create a practice/ folder with a hello.html file that says
«Hello, MADRE» in big letters.
```

What happens: practice/hello.html shows up. Double-click it.

```
@gemini #2 In practice/, create ideas.md with 10 feature ideas for this
project.
```

What happens: A new file full of ideas. Your project stays the same.

#3 Edit For changing files. It asks for your key and first saves a point so you can undo.

```
@claude #3 In practice/hello.html, make the background blue and center
the text.
```

What happens: It changes the file you made with #2. Don't like it? Undo.

```
@codex #3 Add a button to practice/hello.html that changes the
background color on every click.
```

What happens: When it's done you see which files it touched.

#4 Full access For also running commands, like installing, testing or using git. It asks for your key twice. Use it only when you need to.

```
@claude #4 Inside practice/, init git and commit what you created.
Don't push.
```

What happens: It runs commands on your computer. That's why you tell it clearly what it shouldn't do.

```
@codex #4 Run the project's tests and tell me which ones fail. Don't
change any file.
```

What happens: It runs the tests and explains the result.

1 **Pick #3 Edit**

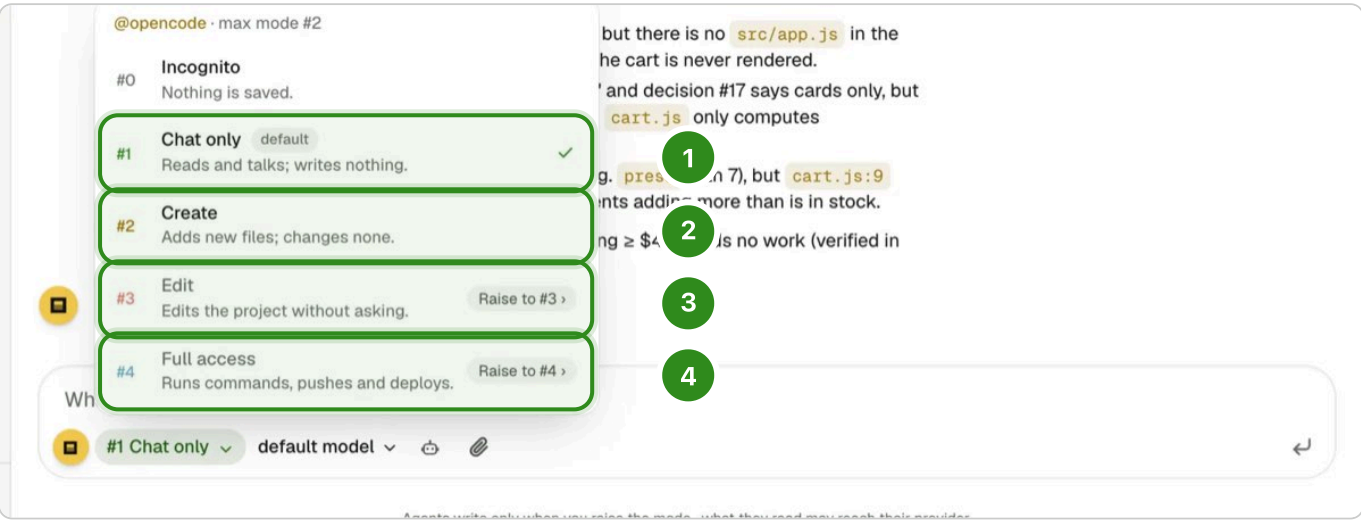
In the permission row, below the box, or type `#3` in your message.

2 **Confirm with your key**

Your key is a word that confirms it was you. At first it's the name of the project's folder. You can change it to a phrase of your own in Settings › Permissions.

3 **Check and, if you want, undo**

When it's done you see which files changed. **Undo** leaves the project as it was before that answer.



1. #1 Chat only: reads your project and answers you.
2. #2 Create: adds new files.
3. #3 Edit: changes your files. It asks you to confirm, and MADRE keeps a point to undo.
4. #4 Full access: runs commands and may publish. It asks you to confirm twice.

With `#4` Full access, the AI can run commands and publish things, for example upload your code with `git push`. What leaves your computer can't be undone. Use it only with an AI you'd lend your computer to.

You're good if you tried a permission in the practice/ folder and you know where Undo is.

STEP 9 · 1 MIN

Add a free AI that runs on your computer.

This step is optional. @madre is an AI that runs on your computer with Ollama, a free program. It doesn't use up your plan and what you ask it doesn't leave your computer.

What it does

- ✓ Answers with what the project remembers and tells you where it got it.
- ✓ Runs the round table and gives you the summary.
- ✓ Writes the memory notes, at no cost.

What it doesn't

- ✕ It doesn't create or change files.
- ✕ It doesn't run commands.
- ✕ It doesn't make things up. If something was never discussed, it tells you.

MACOS

1 Install Ollama

In Home, open **Settings › Local AI** and click **Install Ollama**. If you have Homebrew, MADRE installs it. If not, **Get Ollama** takes you to ollama.com to download it. From the Terminal it would be:

```
brew install ollama
```

LINUX

1 Install Ollama

In Home, open **Settings › Local AI** and click **Install Ollama**. MADRE uses the official installer. From the Terminal it would be:

```
curl -fsSL https://ollama.com/install.sh | sh
```

2 Turn it on

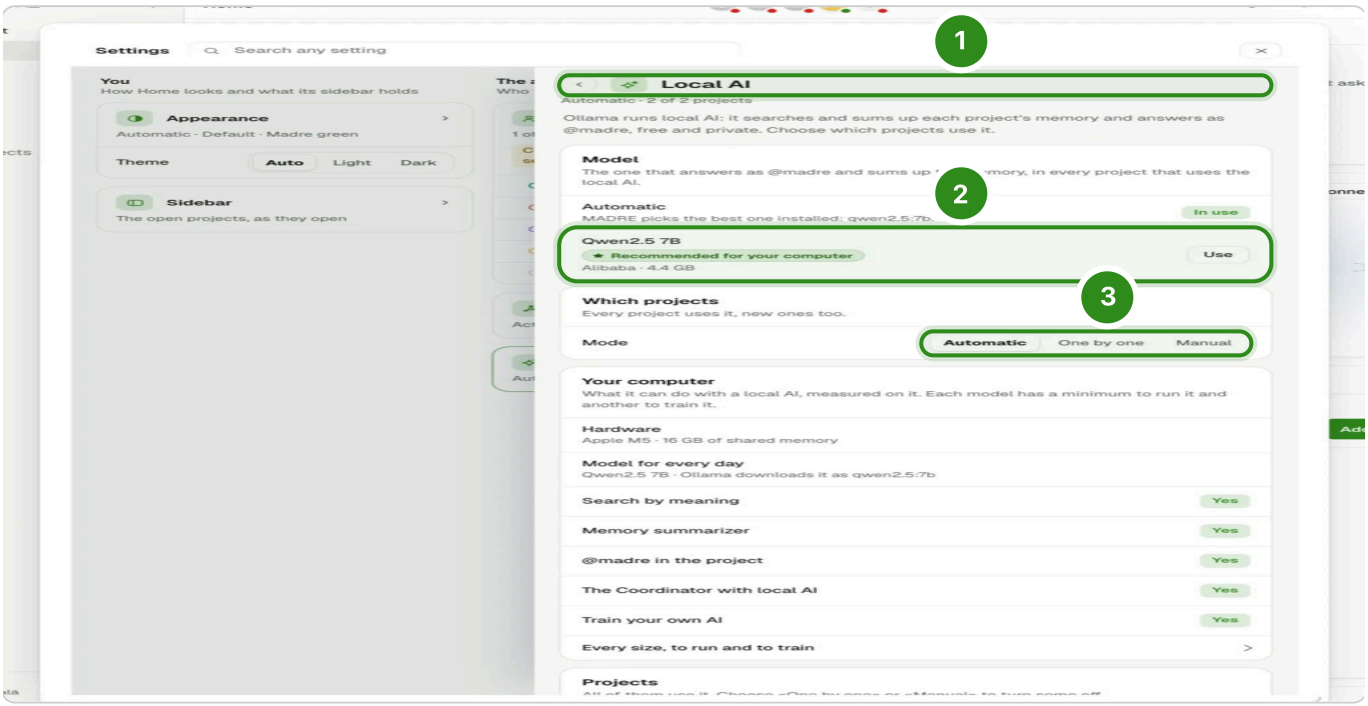
If it says «Stopped», click **Start Ollama**.

3 **Download the model**

The model is the AI you're going to download. Under **Model**, the one that says **Recommended for your computer** is the right one for your machine. Click **Download**, or **Use** if you already have it. It takes a few minutes, depending on your internet.

4 **Try it**

Type `@madre what do you know about this project?` and it answers with what the project remembers.



1. In Home, open Settings › Local AI.
2. Download the model marked «Recommended for your computer»: it is already sized for your Mac.
3. Choose which projects use it.

IF YOU'RE NEW

What's Ollama? And a model?

Ollama is the program that makes the AI run on your computer. The model is the AI itself. `qwen2.5:3b` is a small model that works on regular computers. If you later install a bigger one, @madre uses the best one it finds.

Your local AI turns on in all your projects. If you want to choose which ones, use **Which projects**, on that same screen.

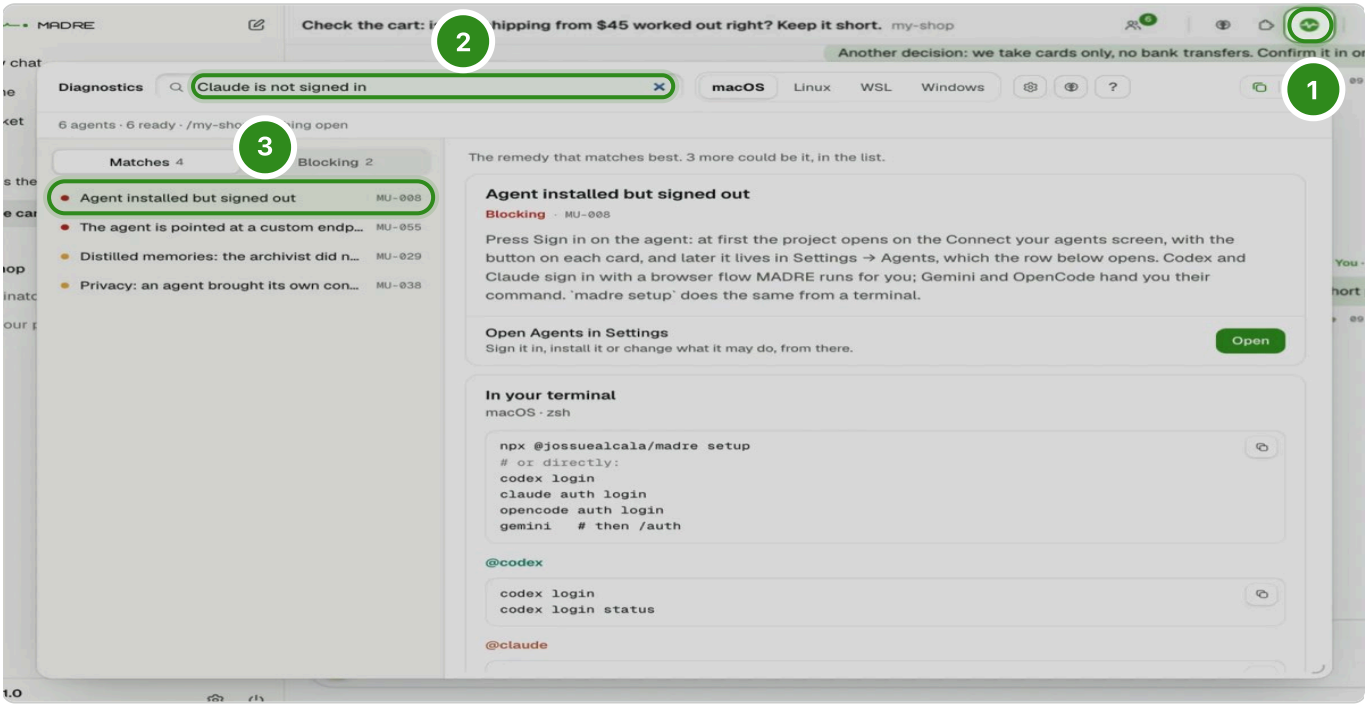
Want an AI that knows your project? In Settings › What it remembers › **Train your own AI** you can train a model with what your project learned. I explain it in Train your AI.

You're good if @madre shows up among your AIs and answers you.

STEP 10 · 30 S

Something not working? Use Diagnostics.

Diagnostics is in your project's bar. Write what's happening in your own words, for example «gemini won't sign in», and it tells you how to fix it. Often it comes with a button that fixes it for you.



1. Press Diagnostics in the bar.
2. Type what you see, in your own words.
3. Open the case that looks like yours and follow what it says.

The most common problems when you start

The Terminal says «command not found: npx».

Node is missing or the Terminal can't find it yet. Install it (step 2), close the Terminal, open it again and paste the command once more.

The browser didn't open.

Open the address that shows up in the Terminal yourself. It's almost always `http://127.0.0.1:4317`.

An AI says «Not signed in».

Go to Settings › Agents and use its button to sign in. To see which AIs MADRE found without opening it, paste this in the Terminal:

```
npx @jossuealcala/madre doctor
```

An AI changed something I didn't want.

Click **Undo** on the changes card. The project goes back to how it was before that answer.

@madre doesn't show up.

Ollama is off or has no model. In Home › Settings › Local AI you turn it on and download the model.

On Windows, Ollama won't install.

Ubuntu is missing `zstd`. In the Ubuntu window paste `sudo apt-get install -y zstd` and click **Install Ollama** again.

Gemini says Google rejected its credentials.

Google no longer accepts Gemini CLI with free personal accounts. Get an API key at Google AI Studio and paste it in Settings › Agents › Gemini.

I want the new version.

The usual command downloads the newest version every time you open MADRE. If MADRE tells you there's a new one, you can also update it with one click.

Got an error message? Look it up in Known errors. Still not fixed? Tell me what happened. You don't need to create an account and you'll see my reply right there.

You're good if you know where to ask for help, inside and outside MADRE.

ONCE YOU USE IT

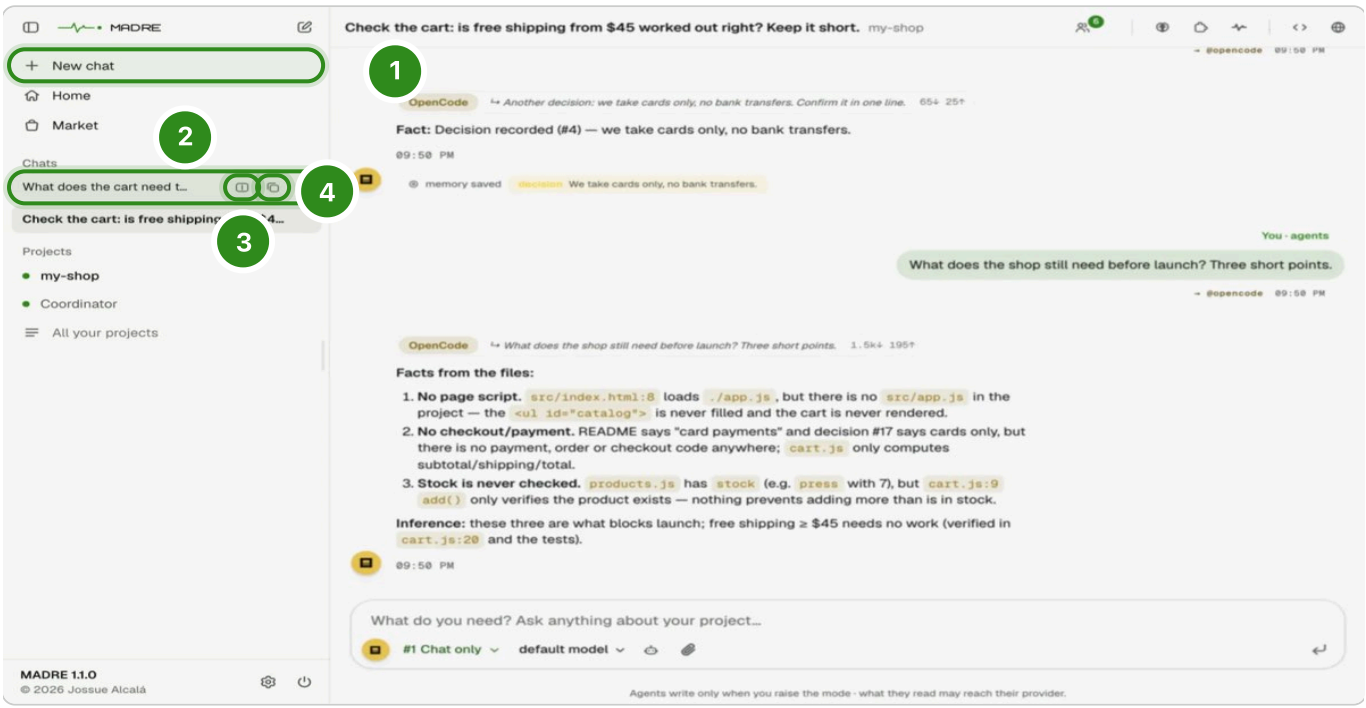
Other things you can do.

You don’t need these to get started. Read them when you need them. Each one takes a minute and comes with the MADRE screen and a number on what you have to press.

ONCE YOU USE IT · 45 S

Your chats, on the left.

Each project can have several chats. They’re in the left column, like in any messaging app. They all share the project’s memory: what was decided in one, the AI knows in the others.



1. «New chat» starts a conversation.
2. Your chats stay below it, the latest on top. Click one to go back to it.
3. Hover over a chat: the first button opens it beside, in the other half of the window.
4. The second opens it in its own window, which you can move. The ✕ deletes it.

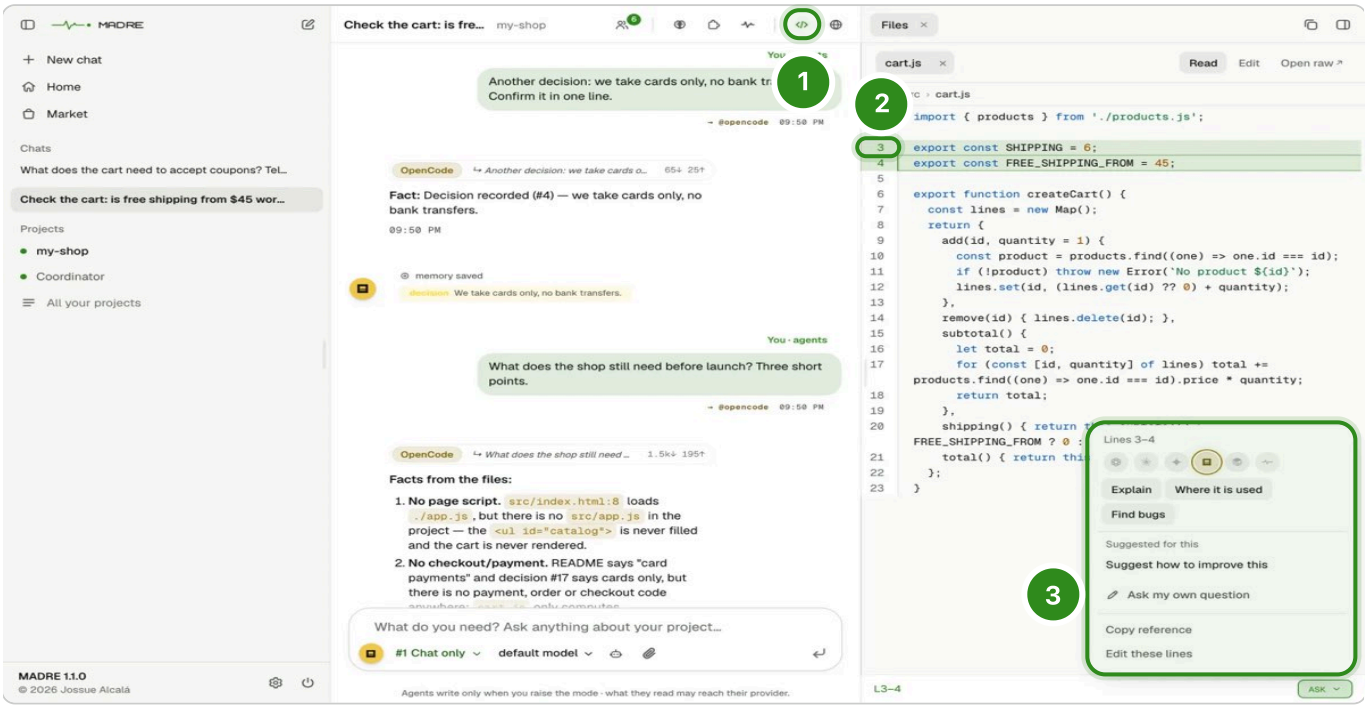
The button at the top left hides the column so you have more room. In a narrow window, the column opens over the conversation.

You’re good if you opened a new chat and went back to the previous one.

ONCE YOU USE IT · 1 MIN

Your files, next to the chat.

Code opens your project's files to the right of the chat. What each AI changed is marked in its colour, so you know who changed what.



1. Press Files (</>) in the bar at the top and open a file from the list.
2. Click the number of a line. With ⇧ you choose several.
3. Press ASK: choose the agent and what you want, or copy the reference to quote it in the chat.

1 Open Code

It's the **Code** button on the bar. With `⌘ P` you find a file by its name.

2 Mark the lines

Click a line number. With `⇧` (Shift) you mark several.

3 Ask

Click **ASK**, pick the AI and what you want: to explain it, improve it, write tests or fix it.

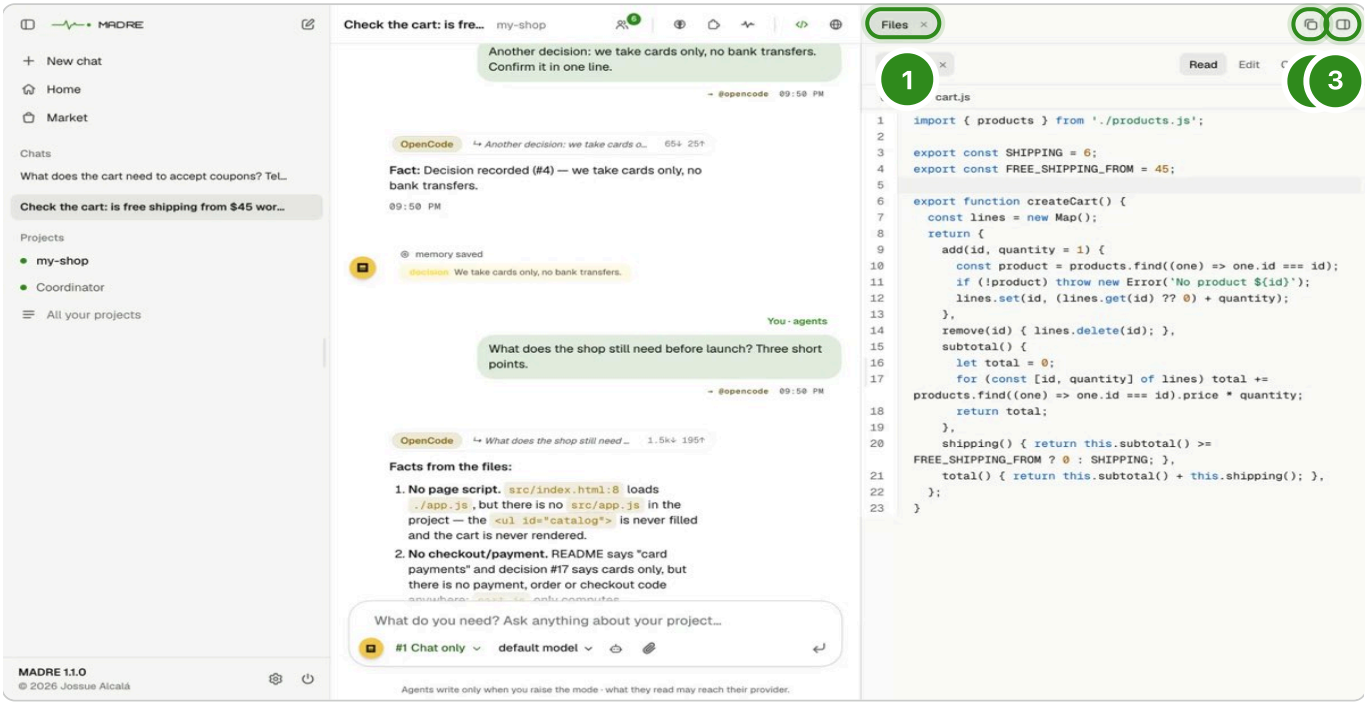
Want to change it yourself? Switch to **EDIT**. `⌘ S` saves, `⌘ F` finds and `⌘ /` turns the line into a comment.

You're good if an AI explained a few lines of your project to you.

ONCE YOU USE IT · 45 S

The tools on the right.

Code, Your servers and the other tools open to the right of the chat, each in its own tab. You can take any of them out to a separate window and move it wherever you like.



1. Each open tool is a tab. The tab's ✕ closes it.
2. The window button takes it out of the column, to put it wherever you like.
3. The last button closes the whole column.

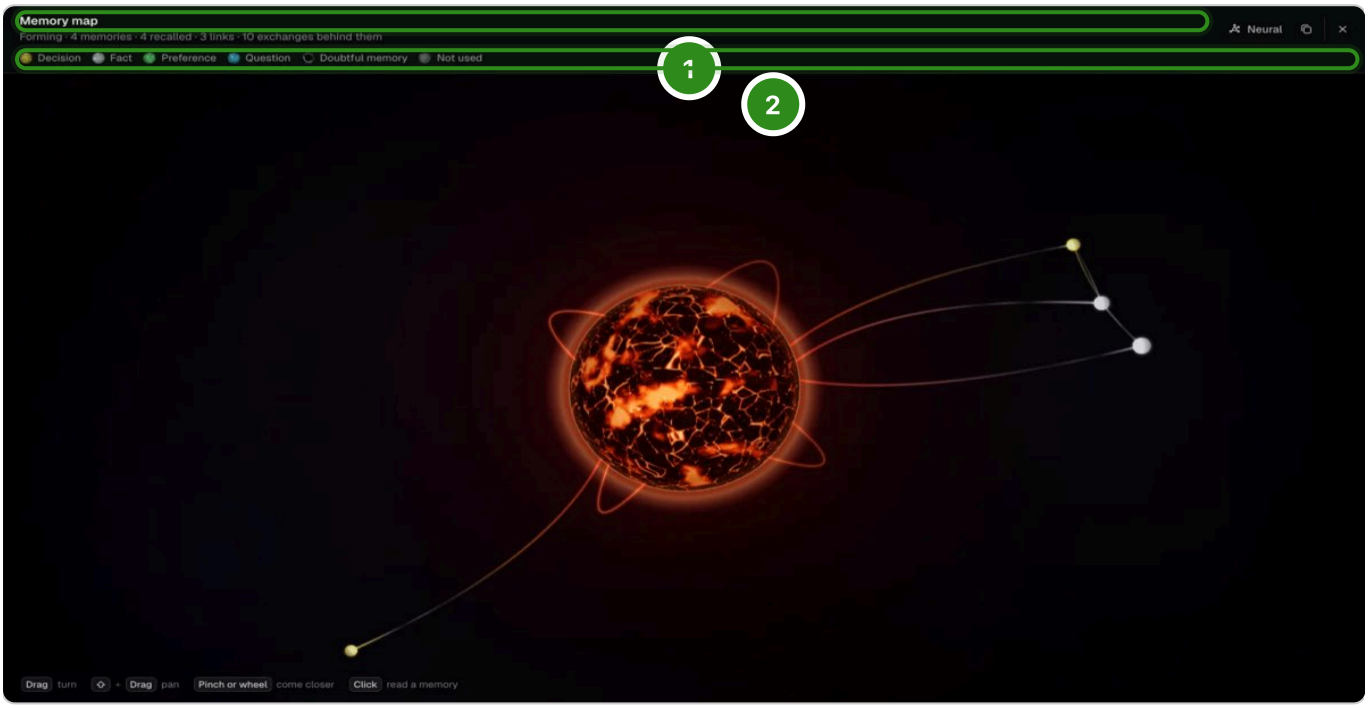
If you make your browser window narrow, whatever no longer fits beside the chat moves to a separate window by itself.

You're good if you took a tool out to its own window and put it back.

ONCE YOU USE IT · 45 S

The project's memory.

MADRE keeps what gets decided in your project: names, decisions and how you like things. When you ask an AI something, MADRE passes it what it needs to know, even if another AI said it weeks ago.



1. Press Memory map in the bar. Type the project's name and it opens full screen.
2. Each dot is something MADRE remembers. Its color says whether it is a decision, a fact or a preference.
3. Click one to read it. If it is no longer true, you forget it from there.

1 **Ask it to remember something**

For example, write: `remember that prices always include tax`.

2 **Ask another AI**

Ask a different AI how the prices work. It already knows.

3 **Check it on the map**

Click **Memory map**, on the bar. Click a star to read it. If it's no longer true, you can forget it right there.

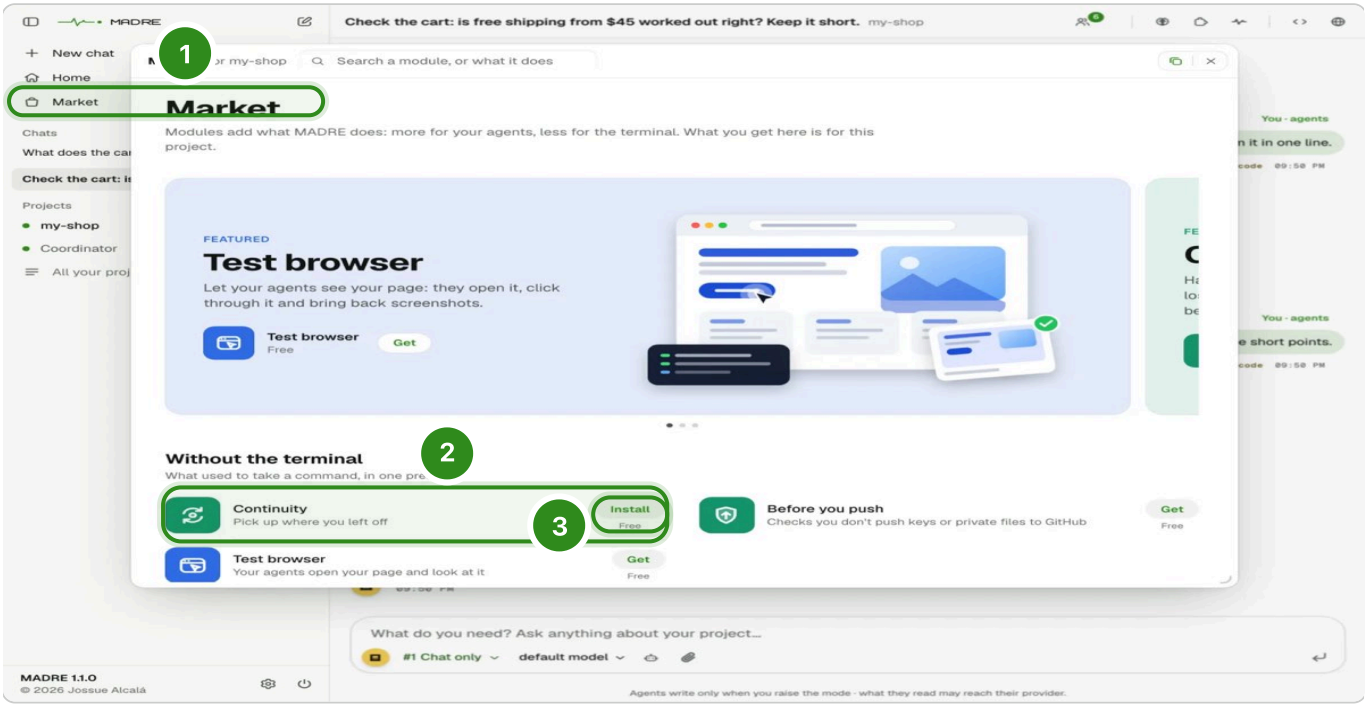
With `#0` Incognito nothing is saved to the memory.

You're good if another AI knew something you told the first one.

ONCE YOU USE IT · 1 MIN

Modules: features you turn on.

Modules are extra features you turn on when you need them. For example: notes, reminders, shorter answers or seeing your page inside MADRE. While they're off, they do nothing.



1. Press Market in the left column.
2. Tap a module to read what it does, what it touches in your project and who made it.
3. Press Get or Install. Afterwards you turn it on or off in Modules, in the project's bar.

1 **Open Modules**

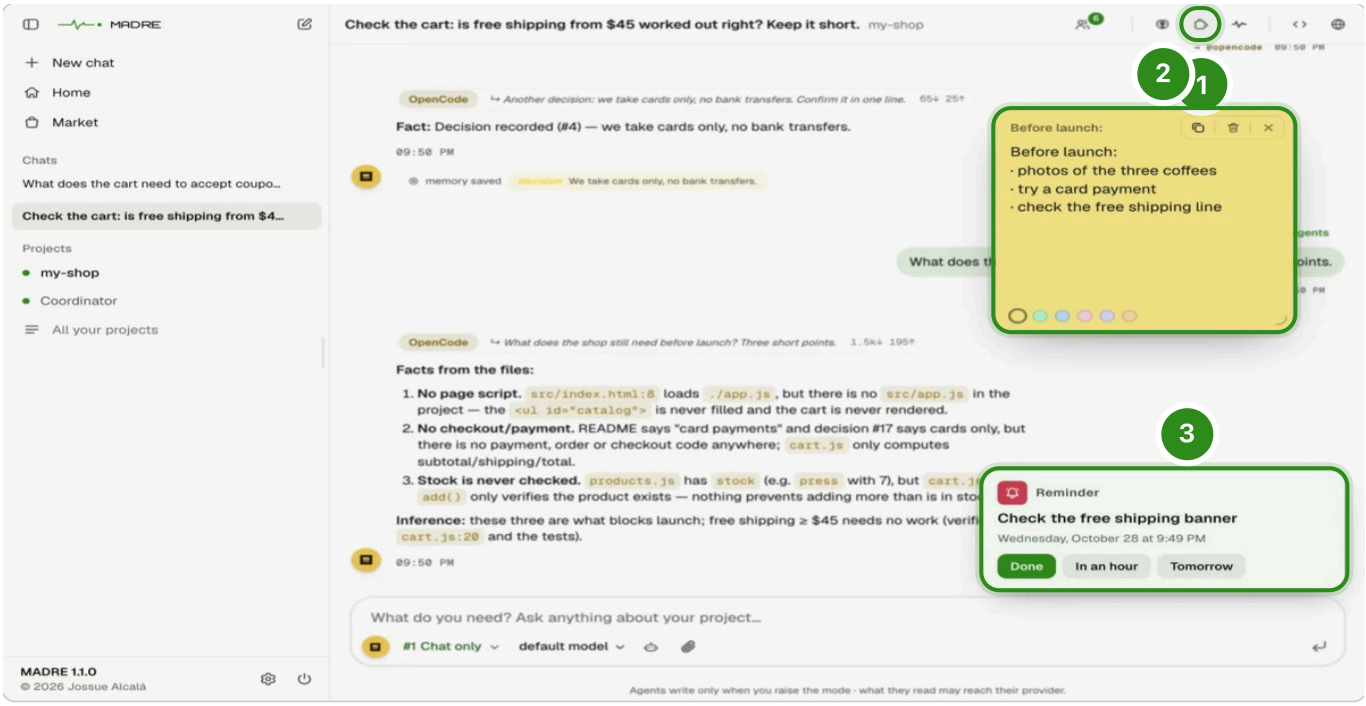
It's on your project's bar.

2 **Read before you turn it on**

Click a tile. It tells you what it does, what it changes in your project and what it needs.

3 **Turn it on**

With its switch. Try **Token saver**: answers come out shorter and spend less.



1. Turn on Notes and Reminders in Modules, in the bar at the top.
2. Each note opens its window: write, change its color or throw it away.
3. When a reminder's time comes, MADRE shows it in the project and in «Waiting on you», in Home: «Done», «In an hour» or «Tomorrow».

Notes and Reminders are two modules. A note is like a coloured slip of paper stuck to your project. A reminder lets you know on the day and at the time you say, and you can ask any AI for one: `remind me tomorrow at 10 to check the banner`.

IF YOU'RE NEW

And the Market?

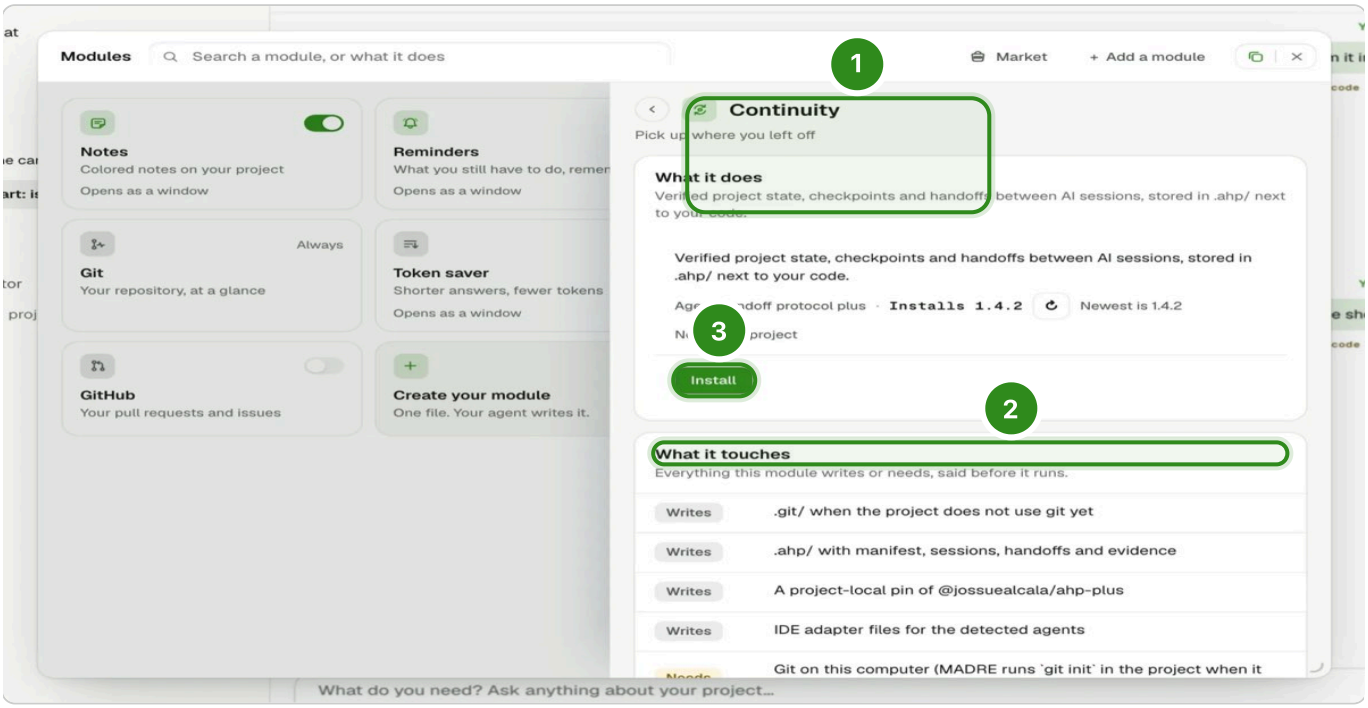
It's the module store. You open it with the **Market** button in Home, or in Modules › Market inside your project. Each module has its page, with what it does and pictures. There you choose whether you want it in all your projects or just one. New modules show up on their own, without updating MADRE.

You're good if you turned a module on and saw what it does.

ONCE YOU USE IT · 45 S

Continuity: another AI picks up where you left off.

Continuity saves, inside your project, where the work stands: what was decided, what's left and points to undo. If you switch AI, editor or computer, the next one starts from there.



1. In Modules, tap Continuity.
2. «What it touches» shows what it writes: a .ahp/ folder inside your project, in plain text.
3. Press Install. Your project needs to use git, with at least one commit.

1 **Open it**

In **Modules**, click **Continuity**.

2 **Read what it changes**

Under **What it touches** you see that it creates a `.ahp/` folder in your project, with text files.

3 **Install it**

Click **Install**. Your project needs to use git and have at least one saved change (a commit).

IF YOU'RE NEW

And outside MADRE?

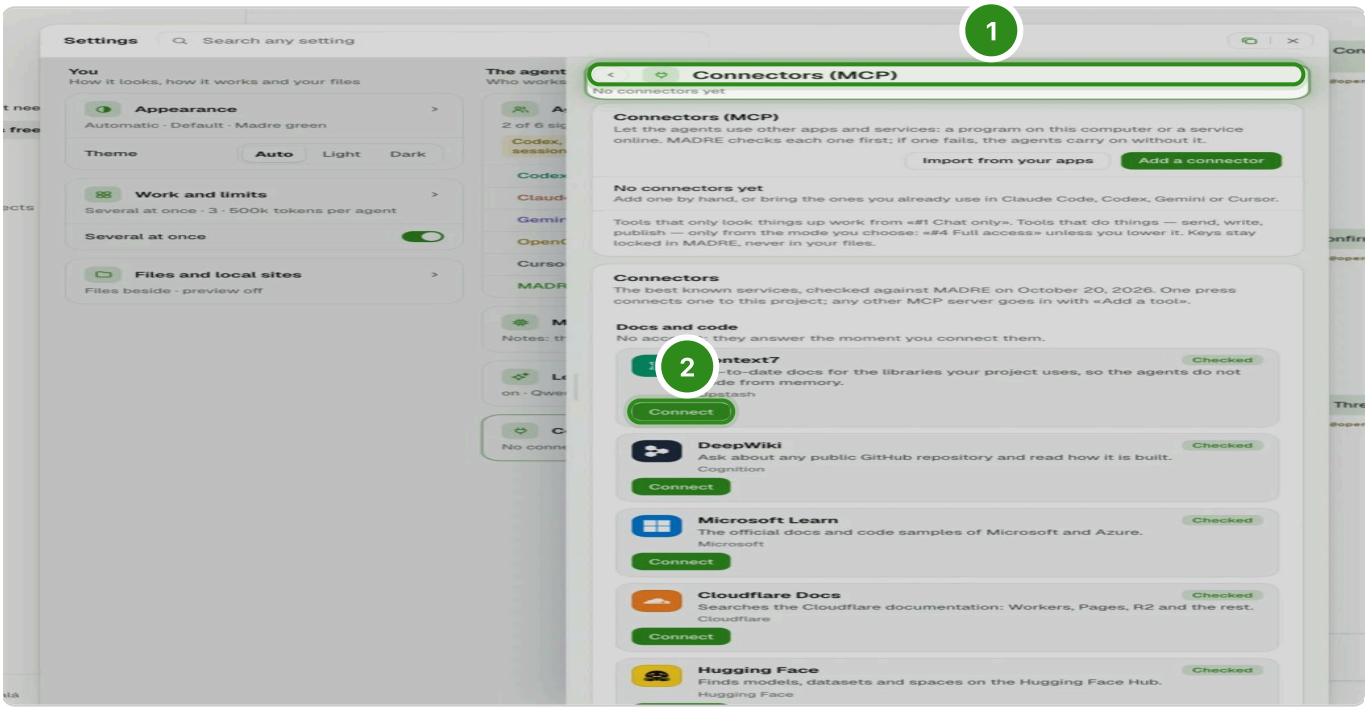
Outside MADRE it's called AHP+. Claude Code, Codex and OpenCode read it too, and the folder is saved in git along with your project.

You're good if your project has the .ahp/ folder.

ONCE YOU USE IT · 45 S

Connectors: let your AIs use other apps.

A connector lets your AIs use other apps, like GitHub, Notion or your calendar. They're also called MCP. You connect them once in each project.



1. Open Settings › Connectors (MCP).
2. Choose one from the catalog and press Connect. Some open their page so you can sign in.
3. Press «Check the connection» to confirm it answers.

1 **Open Connectors**

In **Settings › Connectors (MCP)**.

2 **Connect one**

Pick one from the list and click **Connect**. Some open their page so you can sign in with your account.

3 **Check that it works**

Check the connection tells you whether it answers.

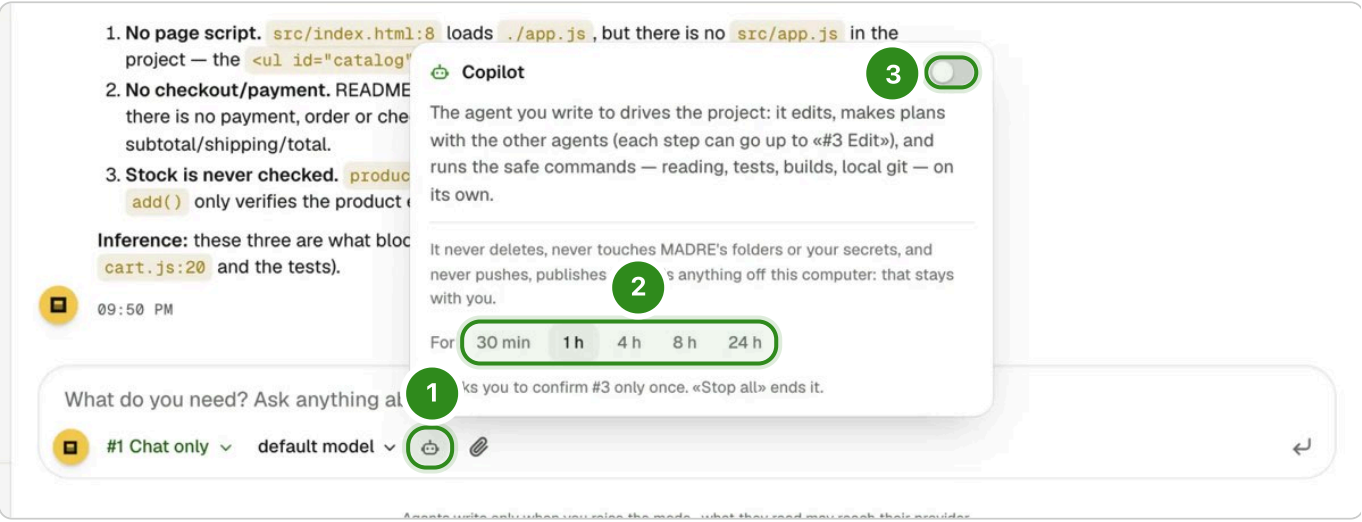
Already had connectors in Claude Code, Codex, Gemini or Cursor? With «Import from your apps» you bring them into MADRE.

You're good if a connector answers.

ONCE YOU USE IT · 30 S

Copilot: let the AI work on its own for a while.

For a long job, Copilot lets the AI move forward without asking your permission at every step. It can edit files and run commands that only read. If it needs to run your tests or build, it waits for your click. It never deletes or publishes.



1. Press the robot beside the box.
2. Choose for how long: from 30 minutes to 24 hours.
3. Turn it on and write the task. It asks you to confirm only once.

1 **Click the robot**

It's beside the text box.

2 **Pick how long**

From 30 minutes to 24 hours.

3 **Turn it on and give it the task**

It asks you to confirm only once. It turns off by itself when the time is up, or with STOP ALL.

You're good if the AI moved forward several steps without asking you.

ONCE YOU USE IT · 30 S

Running now: what's working right now.

Running now shows you what MADRE is doing in all your open projects: which AI is answering, which plan is moving forward and which server is on. You can also stop it from there.

1 **Find the beating line**

It's on the bar at the top, in every project and in Home. The number says how many things are working, and the line moves while something works.

2 **Open it**

Under **Now** you see each AI that's answering: what you asked it, how much of its limit it has used and whether it has helpers. You also see the plans with their steps, the servers and what's being installed. Below is what finished a moment ago and the upcoming reminders.

3 **Go there or stop it**

Open a row. **Go to the chat** takes you to that message. **Stop** stops only that.

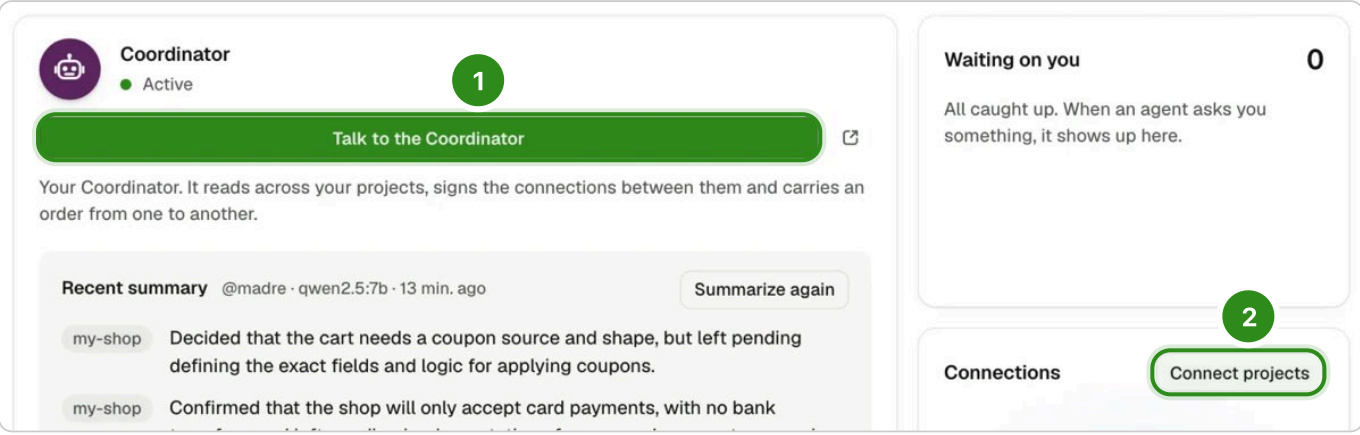
Stop everything stops everything in all your projects, and asks you to confirm. In a narrow window, the panel comes up from the bottom.

You're good if you watched an AI work from Running now.

ONCE YOU USE IT · 30 S

The Coordinator: an AI that sees all your projects.

The Coordinator is in Home. You can ask it about any of your projects. If you connect two projects, one can look up what the other learned.



1. In Home, press «Talk to the Coordinator».
2. Press «Connect projects» and choose two. You decide what it may look up and until when.

1 Talk to it

In Home, click **Talk to the Coordinator**. Ask it, for example, which project you were on yesterday.

2 Connect two projects

With **Connect projects**. You choose what it can look up and until when. Your files and your chats never get mixed.

You're good if the Coordinator answered you about one of your projects.

ONCE YOU USE IT · 1½ MIN

Train your AI on what your project knows.

This is the most advanced part of the manual. The first time, set aside an hour. MADRE doesn't train anything on its own or send anything online. You take out what your project learned, train a small model on your computer and @madre starts answering with it.

IF YOU'RE NEW

What does training a model mean?

You take a small model that already knows how to talk and teach it with examples from your project: your questions and the answers that helped you. You don't start from scratch, you just adjust it. At the end you have a file that Ollama can use like any other model.

What goes in the training file

What goes in

Your questions in the project with the answer you got, and the memory's notes turned into question and answer.

What stays out

Nothing from `#0 Incognito`, no answers you marked as bad, none from @madre and none that are a single line.

What gets deleted

Before the file is saved, paths, names, emails and keys are deleted.

Where it goes

On your computer, in `~/.pulse/rooms/<project>/dataset/`: `train.jsonl` to learn from, `valid.jsonl` to check against and `manifest.json` with the summary.

How many examples you need

Under 200, it learns the tone and little else. From about 1000 it's worth it. Under 50, the training doesn't start.

What computer you need

A Mac with an Apple chip

It's the tested option. All of the Mac's memory counts: with 16 GB you can train a model of up to 3B, one of the small ones.

Linux with an NVIDIA card

It's being tested. First the card is checked. If it doesn't answer, you can't go on.

Neither

The model works, though slower, but it can't be trained: it would take days.

1 **Export**

In your project, open **Settings › What it remembers** and, under **Train your own AI**, click **Export**. It tells you how many examples came out. From the Terminal, in the project's folder, it would be:

```
npx @jossuealcala/madre dataset
```

2 **Pick the size**

Open **Step by step**, in that same section. MADRE checks your computer and marks the biggest model it can train and use. You can pick another one.

3 **Set up Python (only once)**

It installs in a separate space so nothing else on your computer changes. On Linux, **Step by step** gives you the version for your card.

```
python3 -m venv ~/.madre-train && source ~/.madre-
train/bin/activate && pip install mlx-lm
```

4 **Train**

Copy the command from **Step by step**. It already has your folders and the size. It takes 10 to 40 minutes with small models, and an hour or more with big ones.

5 **Hand it to Ollama**

Three more commands, also in **Step by step**. They join what it learned with the model, convert it to Ollama's format and save it with the name `madre-<project>`.

6 **Use it**

In **Settings › Local AI**, click **Check again**. @madre switches to your model, and **What it remembers** shows it.

Before you trust it, ask @madre ten things that were decided in your project and five that never came up. It should get the first ones right and tell you the others never came up. If it makes things up, delete it with `ollama rm madre-<project>` and train it again with more examples.

You're good if your project has its training file.

ONCE YOU USE IT · 30 S

How to uninstall MADRE.

You uninstall it from Home. Before deleting anything, it tells you what will be deleted and what stays.

1 **Open it**

In Home, go to **Settings › Version** and click **Uninstall...**.

2 **Read the list**

The conversations, the memory, the settings, the `.pulse` folder in each project and the points to undo get deleted. Your projects and your AIs aren't touched.

3 **Confirm**

Click **Uninstall**. MADRE closes everything and then deletes. If a project can't be closed, it deletes nothing and tells you which one.

What the AIs created in `.pulse/out` stays, unless you tick the box. If you installed MADRE with `npm -g`, at the end it gives you the command to remove the program: `npm uninstall -g @jossuealcala/madre`.

You're good if you know how to remove MADRE without losing your projects.

ONCE YOU USE IT · 30 S

The license.

Since version 1.1, MADRE uses the Business Source License 1.1. Its code is public: you can read it, change it and share it.

1. **Free for you and your team**
   For individuals and for organizations of up to 10 people. Also if you provide services to others, as a freelancer or as an agency.

2. **Paid for larger organizations**
   An organization of more than 10 people, or anyone who wants to offer MADRE to others as a service, needs a commercial license. Write to me through the form, under “Something else”.

3. **Free software after four years**
   Each version moves to the free GNU AGPL v3 license four years after it’s published.

Versions 1.0.2 to 1.0.5 stay under the GNU AGPL v3. This is a summary. What counts is the full text of the license.

You’re good if you know whether MADRE is free for you.

IDEAS TO BUILD

Not sure what to do first? Pick one.

I tested all of them in MADRE, in an empty folder, with these exact messages (in Spanish). Create a folder, add it in Home and copy the messages in order.

A spaceship game by six AIs

✓ I tested it with Cursor, @madre, Codex, Gemini, Claude and OpenCode · 6 messages · about 13 min

THE MESSAGES, IN ORDER

1 ● @cursor #1 Chat only

```
@cursor #1 We're making an arcade spaceship game in a single
file, ship.html. Design the mechanics, the enemies, the
power-ups that drop and the visual style in a short list.
Don't write code yet.
```

Cursor designed the mechanics, five enemies, five power-ups and the style. Before wrapping up, it asked @madre on its own whether anything had already been decided in the project's memory.

HANDOFFS @cursor → @madre → @cursor · UP TO 4 MESSAGES

2 ● @codex #2 Create

```
@codex #2 Build ship.html with the design Cursor proposed:
full-screen canvas, arrows to move and space to shoot.
```

Codex read Cursor's design in the conversation and built it in one file. You didn't have to paste anything.

HANDOFF @cursor → @codex · 6 MESSAGES

3 @gemini #1 Chat only

```
@gemini #1 Read ship.html and compare it with Cursor's
design: what got left out, and which three things would make
it more fun?
```

Gemini compared the game with what Cursor asked for: the boss missiles and the ring explosions were missing. It suggested screen shake, sound and a dash to dodge.

HANDOFF @codex → @gemini · 8 MESSAGES

4 @claude #3 Edit

```
@claude #3 Add what Gemini found missing and its three ideas
to ship.html, without breaking what Codex did.
```

Claude added all five, then on its own asked Codex to test the game with 900 simulated frames. Everything passed.

HANDOFFS @gemini → @claude → @codex → @claude
· UP TO 14 MESSAGES

5 @opencode #3 Edit

```
@opencode #3 Add a credits screen to ship.html that opens
with the C key and tells what each AI contributed in this
conversation.
```

OpenCode wrote the credits by reading in the conversation what each one did. Nobody told it who did what.

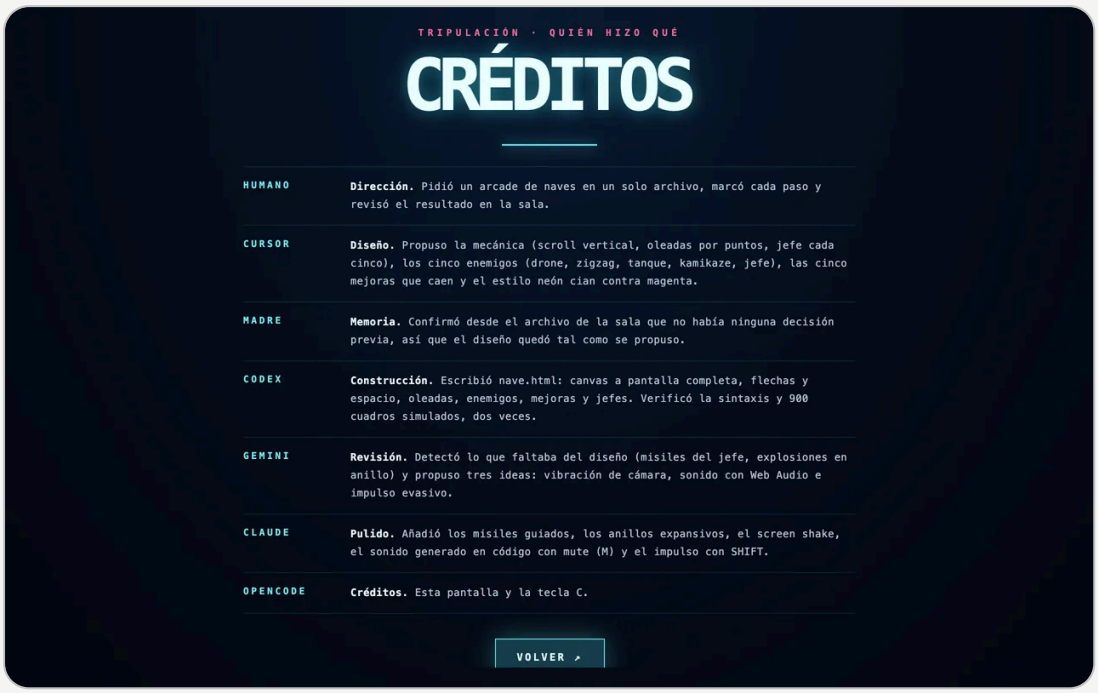
HANDOFF @claude → @opencode · 16 MESSAGES

6 @opencode #3 Edit

```
@opencode #3 Almost there. If I open the credits mid-game,
the PAUSE screen shows through behind the text. Make the
credits cover everything behind them.
```

OpenCode saw the credits background was half transparent and made it solid.

What you'll end up with: The game file. Double-click and play: arrows to move, space to shoot, Shift to dash and C to see who did what.



This is how it came out for me, untouched. Yours will come out different, because each AI does it its own way.

Open it and try it ↗ It's the file that came out of the test, untouched and in Spanish.

Your personal site

✓ I tested it with Cursor, Claude, Codex, Gemini and OpenCode · 6 messages · about 9 min

THE MESSAGES, IN ORDER

1 ● @cursor #1 Chat only

```
@cursor #1 I want a one-page personal website, just HTML and
CSS. My name is Ana Torres and I'm a product designer in
Guadalajara. Suggest sections, palette, typefaces and a
visual detail that makes it memorable, with no images from
the internet.
```

Cursor suggested four sections, a palette of paper, ink, cobalt, jade and sun, two typefaces and an AT monogram that draws itself.

2 @claude #2 Create

```
@claude #2 Build it in index.html and styles.css following Cursor's proposal. Make up three projects, each with a cover drawn in CSS, and use hola@example.com as the email.
```

Claude built it following Cursor's proposal to the letter, with three made-up projects and covers drawn only with CSS.

HANDOFF @cursor → @claude · 2 MESSAGES

3 @codex #1 Chat only

```
@codex #1 Review index.html and styles.css: how does it look on a phone and how accessible is it? Give me a short list of fixes.
```

Codex reviewed it without touching it and found things to fix on very narrow screens, in link sizes and in contrast.

HANDOFF @claude → @codex · 4 MESSAGES

4 @gemini #3 Edit

```
@gemini #3 Apply the fixes Codex found and add automatic dark mode.
```

Gemini applied Codex's fixes one by one and added dark mode.

HANDOFF @codex → @gemini · 6 MESSAGES

5 @opencode #3 Edit

```
@opencode #3 Add a discreet line to the footer of index.html saying what each AI did in this conversation.
```

OpenCode put in the footer what each AI did, taken from the conversation, at the minimum text size Gemini had set.

HANDOFF @gemini → @opencode · 8 MESSAGES

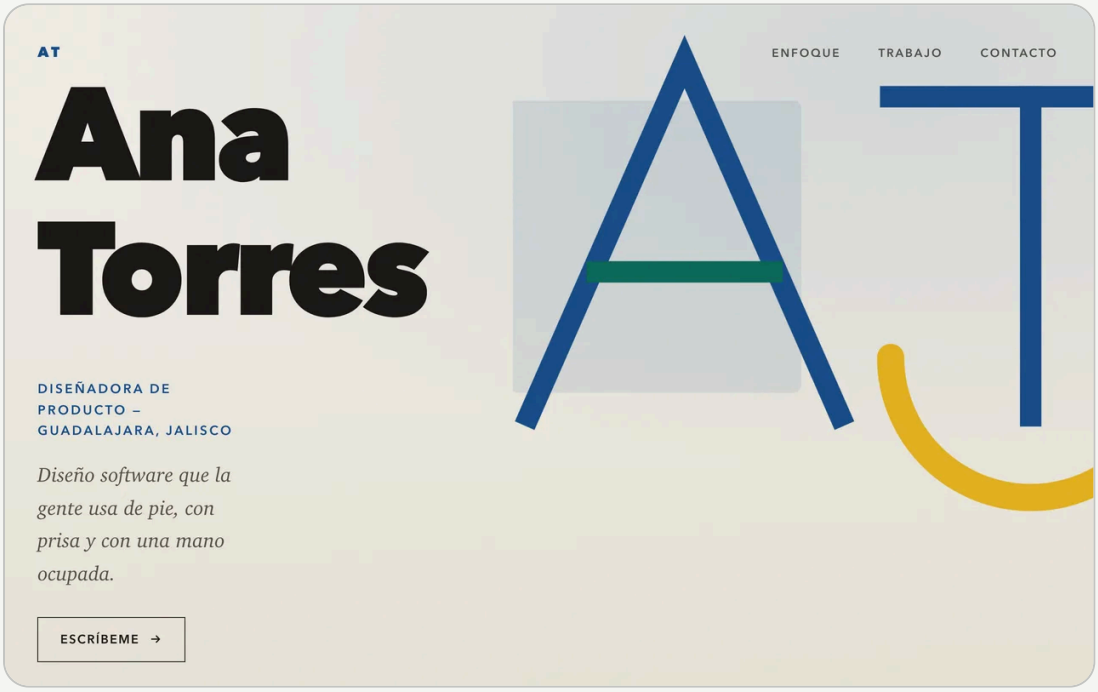
6 @claude #3 Edit

```
@claude #3 Almost there. The page pulls its fonts from Google
Fonts and I want it with nothing from the internet. Swap them
for system fonts with the same character, keep the name just
as strong, and update the credits line in the footer.
```

Claude dropped Google Fonts and picked heavy system fonts so the name still hits hard. It also updated the credits.

HANDOFF @opencode → @claude · 10 MESSAGES

What you'll end up with: Your site in index.html and styles.css. Double-click to see it, or turn on Preview in Modules to see it inside MADRE.



This is how it came out for me, untouched. Yours will come out different, because each AI does it its own way.

Open it and try it ↗

It's the file that came out of the test, in Spanish. I only removed the LinkedIn link, which could lead to a real person's profile.

A Trello-style board

✓ I tested it with Gemini, Cursor, OpenCode, Codex and Claude · 6 messages · about 8 min

THE MESSAGES, IN ORDER

1 @gemini #2 Create

```
@gemini #2 Make a Trello-style board in a single file,
board.html: three columns (To do, Doing, Done), add cards,
drag them between columns, edit them with a double click and
save everything in the browser. Make it look modern and open
with sample cards.
```

Gemini made it in one file, with sample cards, drag and drop and saving in the browser.

2 @cursor #1 Chat only

```
@cursor #1 Review board.html as if you were going to use it
every day: what's missing? Short list.
```

Cursor reviewed it as a daily user would: you couldn't reorder within a column, deleting didn't ask first and there was no backup or undo.

HANDOFF @gemini → @cursor · 2 MESSAGES

3 @opencode #3 Edit

```
@opencode #3 Add the most useful of Cursor's suggestions to
board.html and tell me what you left out and why.
```

OpenCode added the most useful of Cursor's list (reordering, undo with Ctrl+Z, export and import) and explained what it left out.

HANDOFF @cursor → @opencode · 4 MESSAGES

4 ● @codex #1 Chat only

```
@codex #1 Review board.html: what happens with empty cards,
super long text or if I drop a card outside a column? Short
list.
```

Codex reviewed it without touching it and found that an empty card could slip in on import and that text had no limit.

HANDOFF @opencode → @codex · 6 MESSAGES

5 ● @claude #3 Edit

```
@claude #3 Fix what Codex found and add a card in Done that
tells what each AI did in this conversation.
```

Claude fixed what Codex found and added a card in Done that tells what each AI did. Nobody told it who did what.

HANDOFF @codex → @claude · 8 MESSAGES

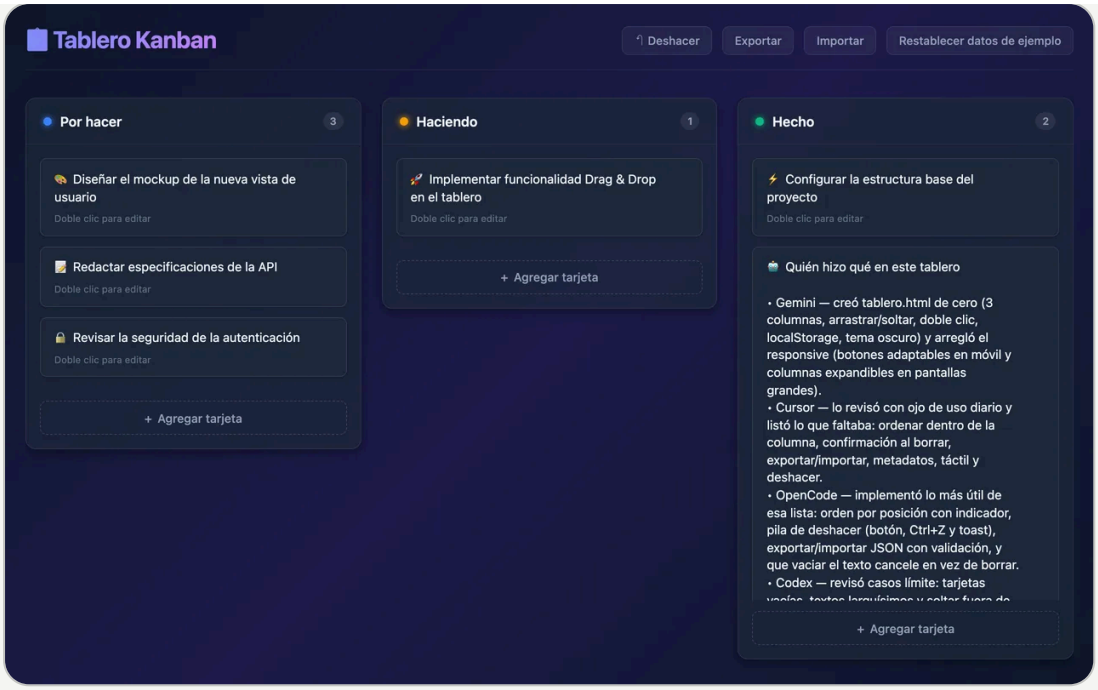
6 ● @gemini #3 Edit

```
@gemini #3 Almost there. On a phone the buttons at the top
run off the screen, and on a big screen the columns don't
fill the width. Fix it and add your fix to the who-did-what
card.
```

Gemini fixed the buttons for phones, made the columns fill the screen and added itself to the who-did-what card.

HANDOFF @claude → @gemini · 10 MESSAGES

What you'll end up with: board.html, ready to use. It opens with sample cards and the who-did-what card.



This is how it came out for me, untouched. Yours will come out different, because each AI does it its own way.

Open it and try it ↗ It's the file that came out of the test, untouched and in Spanish.

Your spending tracker

✓ I tested it with @madre, Codex, Claude, Gemini, OpenCode and Cursor · 5 messages · about 8 min

THE MESSAGES, IN ORDER

1 ● @madre Round table

```
@madre, ask the team: I want a page to log my monthly
spending and see the total per category with a chart, no
server and no libraries from the internet, in a single file.
How would you build it?
```

Codex, Claude, Gemini and OpenCode said how they'd build it, each one reading the previous ones («I agree with @codex on the essentials…»). @madre summed up where they agreed and where they didn't. `Needs your local AI`

HANDOFFS @madre → @codex → @claude → @gemini → @opencode → @madre · UP TO 9 MESSAGES

2 ● @claude #2 Create

```
@claude #2 Create spending.html with the best of the team's proposals, make it look like a modern finance app with dark mode, and tell me who you took what from.
```

Claude built it from the best of the table and tells you who it took what from, like Codex's skeleton and Gemini's SVG chart.

HANDOFF @madre → @claude · 12 MESSAGES

3 ● @opencode #3 Edit

```
@opencode #3 Add a month of sample data to spending.html so the chart shows from the start, and a button to delete it.
```

OpenCode added 23 sample expenses, with a notice at the top and a button to remove them that leaves yours alone.

HANDOFF @claude → @opencode · 14 MESSAGES

4 ● @cursor #1 Chat only

```
@cursor #1 Review spending.html: does it do what the team agreed on at the round table? What's missing?
```

Cursor checked it against what the table agreed: it covers the essentials, but the sample data wasn't saved until your first change.

HANDOFF @opencode → @cursor · 16 MESSAGES

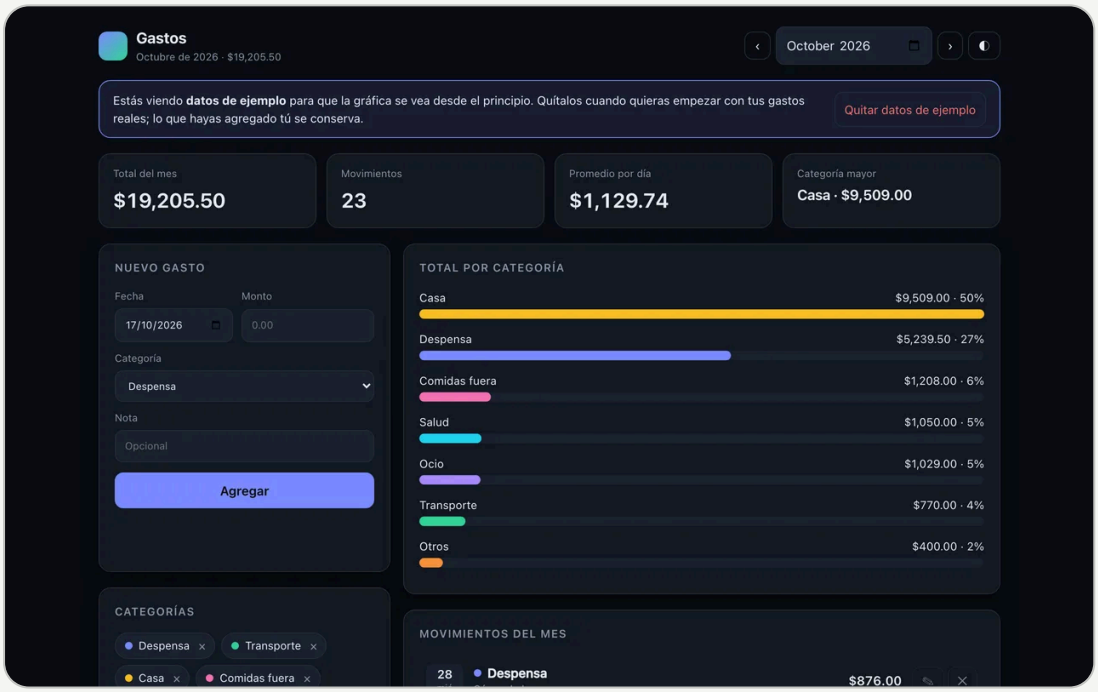
5 @gemini #3 Edit

```
@gemini #3 Add what Cursor found missing to spending.html,
and put at the bottom who contributed what in this
conversation.
```

Gemini fixed what Cursor saw and put at the bottom what each AI contributed, starting with the round table.

`HANDOFF @cursor → @gemini · 4 MESSAGES`

What you'll end up with: The spending file: you log expenses, see the month's total, the daily average and the chart per category, and back up your data as JSON.



This is how it came out for me, untouched. Yours will come out different, because each AI does it its own way.

Open it and try it ↗ It's the file that came out of the test, untouched and in Spanish.

A game made as a team

You give Claude a single instruction. Claude writes a plan and MADRE follows it step by step: Cursor designs, Codex builds it, Gemini reviews it and Claude ends with a summary. You just watch.

✓ I tested it with Claude, Cursor, Codex, Gemini and OpenCode · 3 messages · about 9 min

THE MESSAGES, IN ORDER

1 @claude #2 Create

```
@claude #2 Make a plan with the team to build a memory game
with emoji cards from scratch in memory.html: @cursor
proposes the design without writing code, @codex writes the
whole game following Cursor, @gemini reviews it and suggests
upgrades, and you wrap up summarizing what came out and how
to play.
```

Claude wrote a four-step plan and MADRE ran it on its own: Cursor designed the game, Codex wrote it to the letter, Gemini reviewed it and suggested five upgrades, and Claude wrapped up explaining how to play.

HANDOFFS @claude → @cursor → @codex → @gemini → @claude · UP TO 8 MESSAGES

2 @opencode #3 Edit

```
@opencode #3 Apply Gemini's upgrades to memory.html and add a
«Who made it» button that tells what each AI contributed in
this conversation.
```

OpenCode applied Gemini's five upgrades and added «Who made it», with what each AI contributed, down to the emojis Cursor picked.

HANDOFF @claude → @opencode · 10 MESSAGES

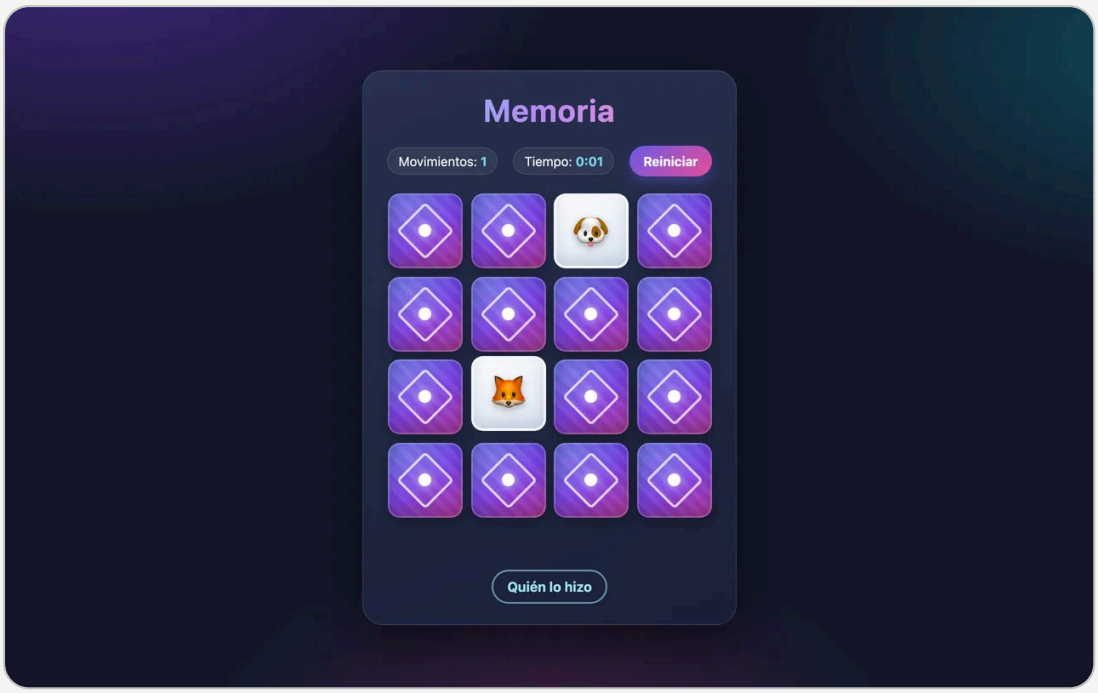
3 @claude #3 Edit

```
@claude #3 Almost there. The game works, but it looks too
plain. Give it personality: cards that flip in 3D with a
drawn back, lively colors and a celebration when you win.
Don't change how it plays and add your part to «Who made it».
```

Claude gave it purple cards with a drawn back that flip in 3D, lively colors and confetti when you win, and added itself to «Who made it».

HANDOFF @opencode → @claude · 4 MESSAGES

What you'll end up with: The game file: eight emoji pairs, moves, time, restart and the «Who made it» button.



This is how it came out for me, untouched. Yours will come out different, because each AI does it its own way.

Open it and try it ↗

It's the file that came out of the test, untouched and in Spanish.

Brainstorm that ends in an app

A brainstorm between AIs that ends in an app. Each one reads what the others said before chiming in, and the one that builds it explains why it chose that idea.

✓ I tested it with @madre, Codex, Claude, Gemini, OpenCode and Cursor · 5 messages · about 8 min

THE MESSAGES, IN ORDER

1 ● @madre Round table

```
@madre, ask the team: I have an empty folder and a free
afternoon. What small but surprising app can we make in a
single HTML file, offline? One idea each.
```

Codex pitched a constellation that sings, Claude a garden of living patterns, Gemini a musical pendulum and OpenCode a notebook that saves itself. @madre summed it up.

Needs your local AI

HANDOFFS @madre → @codex → @claude → @gemini → @opencode → @madre · UP TO 9 MESSAGES

2 ● @claude #2 Create

```
@claude #2 From the team's ideas, pick the most surprising
one and build it in app.html, with a design that feels
handmade. Say why you picked it.
```

Claude didn't pick its own idea: it picked OpenCode's, because it surprises by what it is, not by how it looks. Saving downloads the very same file with your notes inside.

HANDOFF @madre → @claude · 12 MESSAGES

3 ● @cursor #1 Chat only

```
@cursor #1 Try app.html as if you were the one who proposed
the idea: what's missing for it to really surprise?
```

Cursor tried it as OpenCode would and found a problem: the page explained the trick before you got to live it.

HANDOFF @claude → @cursor · 14 MESSAGES

4 @gemini #3 Edit

```
@gemini #3 Apply what Cursor said to app.html. Nothing should get cut off on a phone.
```

Gemini removed the spoiler, added a generation stamp and a notice when you open a copy, and fit it to phones.

HANDOFF @cursor → @gemini · 16 MESSAGES

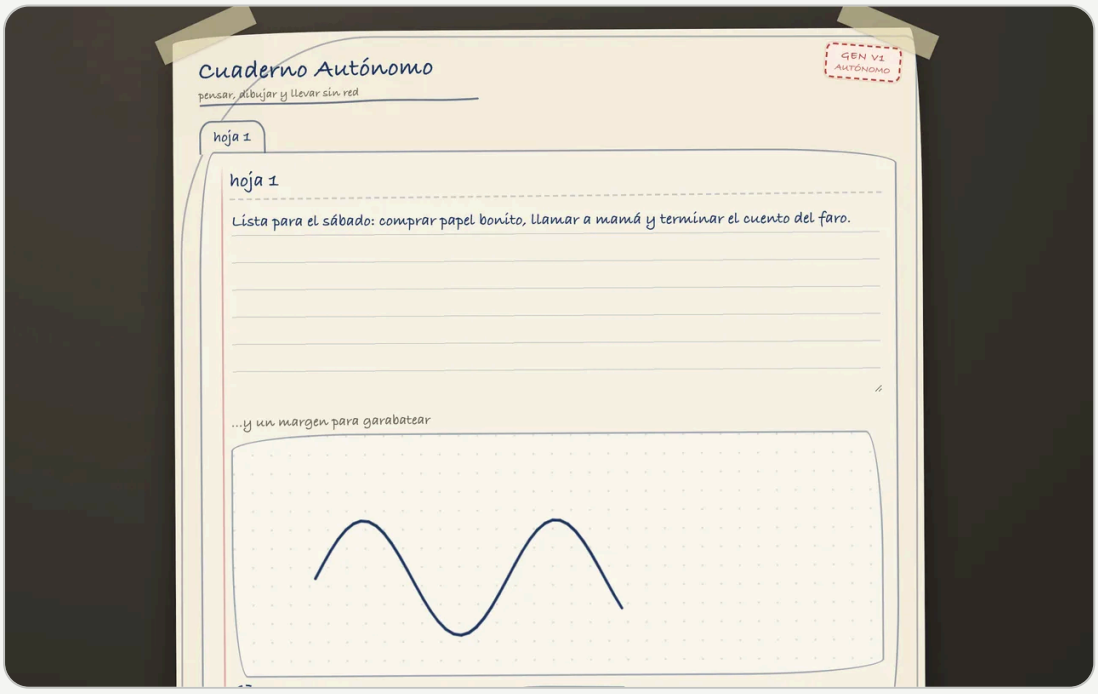
5 @codex #3 Edit

```
@codex #3 Add a credits note to app.html that tells how the idea was born and what each AI did in this conversation.
```

Codex wrote how the app was born, with what each AI proposed and did, and tucked it inside «look at the HTML's guts» so it doesn't spoil the surprise.

HANDOFF @gemini → @codex · 18 MESSAGES

What you'll end up with: app.html, an «autonomous notebook»: you write and doodle, and when you save you take the file with everything inside.



This is how it came out for me, untouched. Yours will come out different, because each AI does it its own way.

Open it and try it ↗ It’s the file that came out of the test, untouched and in Spanish.

If you don’t like something, ask any AI for the change or click Undo. Gemini needs a Google AI Studio key (step 4). In these tests Cursor could only read: that’s why it designs and reviews, and the others write.

THE CHEAT SHEET

All of MADRE on one page.

Keep it handy. It's also in the PDF.

In the Terminal

`npx @jossuealcala/madre`
Opens Home, with all your projects.

`npx @jossuealcala/madre start`
Opens the project of the folder you're in.

`npx @jossuealcala/madre doctor`
Tells you which AIs MADRE found and which ones are signed in.

`Ctrl + C`
Shuts MADRE down, like the power button in the bar.

In the text box

`@agent`
Who you're writing to.

`#0 … #4`
What you let it do.

`!file`
Which file you hand over.

`/`
The command menu.

`@madre, ask the team`
Round table with all of them.

Shortcuts

`Enter`
Sends the message.

`⌥ Enter`
New line (Alt + Enter on Linux).

`⌘ ,`
Opens Settings (Ctrl + , on Linux).

`?`
The gestures and shortcuts sheet.

The modes

`#0`
Incognito

`#1`
Chat only

`#2`
Create

`#3`
Edit

`#4`
Full access

WHEN YOU WANT MORE

To keep learning.



The Coordinator An AI that can look into your projects, only the ones you connect. Ask it from any project with `/bishop`.



Modules Extra features you turn on with a switch, like GitHub, Suggested effort or the Test browser.



Make your module ↗ Add features to MADRE with a single file.



The Memory map What your project remembers, in 3D: decisions, facts, preferences and open questions.



The copilot An AI works on your project with permission #3 for as long as you choose, without asking you to confirm every message.



Technical reference ↗ All the options, commands and limits, in case you want to know every detail.