BUILT TO STAND

FREE GUIDE



HOW WE BUILD WITH CLAUDE

THE SETUP

The folder, the rules file, the memory and the daily loop behind everything we ship. **Set it up once and Claude stops chatting and starts building.**

GUI, MATHEUS & BIEL · TYOHAN

BUILT IN EUROPE · 2026

READ THIS FIRST

MOST PEOPLE CHAT WITH CLAUDE. A FEW BUILD WITH IT.

Same model, same price. The difference isn't a magic prompt. It's the **setup**: where Claude works, what it already knows before you type, and how you check what it did. That's what this guide gives you.

CHATTING	BUILDING
Lives in a browser tab	Lives in a folder on your computer
Forgets you every new chat	Reads your rules and notes every time it starts
Hands you code to copy and paste	Writes, runs and fixes the files itself
"Done" means it answered	"Done" means you checked that it works
Ends in a conversation	Ends in something live

WHAT THIS EXACT SETUP BUILT FOR US

tyohan.com
Our brand site, from brand book to live.

A video studio
Script to narrated, edited video.

A design library
References that became a taste guide.

A personal OS
Tasks, money and training in one app.

3D demos
A restaurant you walk through in the browser.

EVERY ONE OF THEM STARTED THE SAME WAY: AN EMPTY FOLDER AND ONE SENTENCE.



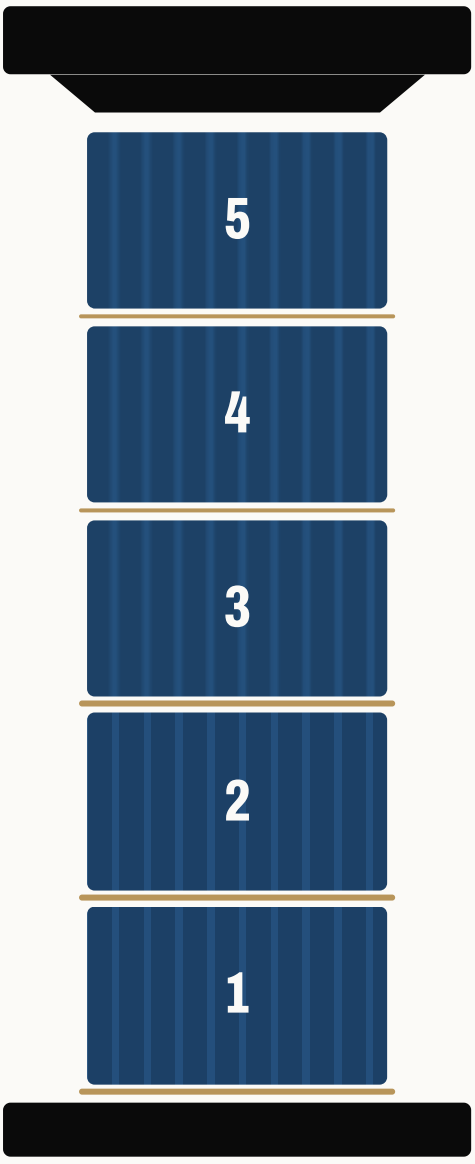
How to use this guide

Read it with Claude open. There are **5 pieces**, one per page, and each ends with **one thing to do**. About an hour, start to finish. At the end you'll have the same setup we use every day.

THE MAP

FIVE PIECES. STACKED LIKE A COLUMN.

Each piece sits on the one below it. Skip one and the whole thing wobbles: Claude forgets, guesses, or breaks what was working. Put all five in place and it stands. That's where the name comes from.



05

The loop

Ask small, check, ask the next thing. Five requests you'll use every single day.

04

The memory

A map of your projects and a note to tomorrow, so no session starts from zero.

03

The rules file

CLAUDE.md: you teach Claude how you work once, and it reads it every time.

02

The folder

One project, one folder, three small files. Claude works much better with a place to live.

01

Claude on your computer

Out of the browser tab, into a folder where it can create, run and fix real files.

01 PIECE ONE

CLAUDE ON YOUR COMPUTER

In a browser tab, Claude can only **talk** about your project. On your computer, pointed at a folder, it can **work** on it: create the files, run them, see the error and fix it. That one change is most of the difference.

THREE WAYS TO USE CLAUDE

BROWSER

claude.ai

Great for thinking, writing and questions. It can't touch the files on your computer.

WHAT WE USE

Desktop app, Code tab

Claude works inside a folder you choose. You see every change and approve what matters.

TERMINAL

Claude Code

The same engine in a command line. Same results; pick it if you already like the terminal.

WHICH PLAN

Start on a paid individual plan: the free one runs out fast once Claude is doing real work. We moved up a tier when we started building every day. Prices change, so check **claude.com/pricing** the day you sign up.

TEN-MINUTE INSTALL

01 Download the Claude desktop app for Windows or Mac and sign in.

02 Create an empty folder called **Claude** in your documents. Everything you build will live in it.

03 Open the **Code** tab and choose that folder.

04 Make your first request (below) and watch the file appear.

DO THIS
5 MINUTES

Your first request. Swap the part in brackets:

```
Create a file called hello.html: a page with [your name] in
huge letters on a dark background. Then open it in my browser.
```

02

PIECE TWO

ONE PROJECT, ONE FOLDER

Claude reads the folder it's working in. A messy folder makes a confused Claude. A clean one, with three small text files, makes it feel like it's been on the project for months. This is close to how ours looks:

```
Claude/
├─ CLAUDE.md          the map of every project (piece 4)
├─ site/              our brand site
│  ├─ README.md       what it is, how to open it, where we stopped
│  ├─ TESTS.md        how to check it still works
│  └─ NEXT.md         the first step for tomorrow
├─ studio/            the video studio app
├─ library/           the design library
└─ life-os/           the personal OS
```

THE THREE FILES, IN PLAIN WORDS

README.MD

What this project is, who it's for, how to open it, and where the work stopped.

TESTS.MD

A list of "click this, this should happen". Run it before you show anyone.

NEXT.MD

What you did today, what's missing, and the first step tomorrow.

DO THIS

5 MINUTES

Pick one thing you want to build and let Claude set up its folder:

```
Create a folder called [project-name] for [what it is, who it's for]. Inside it, create README.md, TESTS.md and NEXT.md with short starting content. Don't build anything yet.
```

03

PIECE THREE

CLAUDE.MD: TRAIN IT ONCE

A file called **CLAUDE.md** inside your folder. Claude reads it every time it starts working there. It's like training a new hire once instead of every morning. Here's ours, simplified. Copy it and change the brackets:

```
# How to work in this project

## Who I am
- [Your name]. I'm not a programmer.
- Explain things in plain words.

## How we work
- Before any big change, show me the plan
  in up to 5 steps and wait for my ok.
- One request at a time.
- Don't change what I didn't ask for.
  If something else needs fixing, ask.
- When you finish, tell me exactly
  how to check that it works.

## This project
- What it is: [one line].
- Who uses it: [people]. Write the
  on-screen text for them.
- Look: [colors, fonts, mood].

## Never
- Never put passwords or keys in files.
- Never delete files without asking.
- Never run anything that costs money
  without telling me the price first.
```

Who I am
Claude adjusts how it explains. "Not a programmer" alone changes every answer.

Plan first
The most useful line in the file. A wrong plan costs one sentence to fix, not an afternoon.

Don't change what I didn't ask
Stops the classic: fix one thing, break another.

How to check
"Done" only counts when you saw it work.

Never
The rules that protect your money and your files. More on page 09.

DO THIS
10 MINUTES

Paste the template into your project folder as CLAUDE.md, fill the brackets, then ask:

```
Read CLAUDE.md and tell me in 3 lines how you're going to work
with me.
```

04

PIECE FOUR

MEMORY THAT SURVIVES ANYTHING

In November we changed plans and **lost our whole chat history**. Months of conversations, gone in a click. The projects didn't stop for a single day, because the things that mattered were never in the chat. **They were in files.**



Chats are temporary. Files stay. So the memory lives in two places: a **map** at the top of your Claude folder, and the **NEXT.md** inside each project.

THE MAP: THE TOP OF YOUR CLAUDE.MD

```
## My projects
| Project | Folder   | What it is          | Where it stopped        |
| Site    | site/    | My brand site, live | New services section    |
| Studio  | studio/  | Video app           | Voice fixed; test next  |

When you finish an important step, update that project's line.
```

THE RITUAL: HOW EVERY SESSION STARTS AND ENDS

Start

```
Read CLAUDE.md and the README and NEXT
of [project]. Tell me where we stopped
and what's pending before touching
anything.
```

End

```
Update NEXT.md: what we did today,
what's missing, the first step
tomorrow. Then update this project's
line in the map.
```

DO THIS
5 MINUTES

Add the map to the CLAUDE.md at the top of your Claude folder, with one line per project. Run the End request before you close today.

05

PIECE FIVE

THE LOOP: ASK SMALL, CHECK, NEXT

People who get stuck ask "build me an app" all at once and get a mess. People who build ask for one small thing, check it, and ask for the next. These five requests are that loop, in words that work.

1	2	3	4	5
PLAN	**ONE STEP**	**CHECK**	**FIX**	**CLOSE**
See the path first	Small enough to check	Click it yourself	Understand, then repair	Leave a note for tomorrow

Plan first
anything bigger than one screen

```
I want [what you want]. Before writing anything, tell me in up to 5 steps how you'd do it. Don't start until I say ok.
```

One step
right after the plan

```
Ok. Do only step 1. When you finish, stop and tell me how I can check it worked.
```

Check
when it says "done"

```
Before saying it's done, write in TESTS.md how I check it: what I click and what should happen.
```

Fix
when something breaks

```
I got this error: [paste it all]. First explain in plain words what happened. Then fix it and tell me how to check.
```

Close
end of every session

```
Update NEXT.md: what we did today, what's missing, the first step tomorrow.
```

BONUS
MOST USEFUL

For any screen or feature, talk like the owner of the business, not like a programmer. Claude translates it into code:

```
Who uses it: [a waiter, on a phone]. What they see: [the tables, each with a color]. When they tap one: [its order opens]. It has to be: [fast with one hand].
```

KEEP IT STANDING

FIVE RULES WE NEVER BREAK

Every one of these exists because something went wrong once. Put them in your CLAUDE.md and you skip the expensive lessons.

01 Keys and passwords never go in the chat or in a file

API keys are like your card number. Keep them in your computer's environment variables and tell Claude the name of the variable, never the value.

02 Know the price before anything runs

Image, voice and video APIs charge per use. Make Claude tell you the estimated cost and wait for your ok. A loop that runs all night can cost real money.

03 Save versions

If something breaks, you go back to the last version that worked instead of untangling it.

```
Set up git in this project, explain it to me in two lines, and save a version
every time something works.
```

04 Don't touch what I didn't ask for

Small change, small scope. If Claude thinks something else needs fixing, it asks first.

05 Check it on a phone before anyone sees it

Most people will open your work on a phone first. This one catches most problems early:

```
Check this screen as a 375px-wide phone. What breaks, gets cut off or is too
small to tap? List it, then fix it.
```



WHAT'S NEXT

THE FOUNDATION IS DONE.

With these five pieces, Claude stops chatting and starts working with you. But a foundation is not a building. This is what the setup **doesn't** teach on its own, and what we'll build together, step by step:

Your site, live
From a brand to a page that looks expensive, on your own domain.

Your first app
A tool that solves a real problem of yours, and grows without breaking.

Video, images and 3D
Video made of code, AI images without burning money, 3D in the browser.

Turning it into a service
Finding a local business, the demo that sells, pricing and delivering.

DEC 5

20:00 BRASÍLIA · 23:00 DUBLIN

Built to Stand opens. You're on the list, so you get the link first, before anyone else, with the launch bonus. Keep an eye on your inbox.