



The *context* files

The five things your AI cannot work out on its own.

TIME NEEDED

One afternoon,
one folder

YOU LEAVE WITH

Five files,
written not planned

FROM

bosio.digital
AI consulting

Everyone agrees context is the differentiator.

Almost nobody says what to write.

RSM surveyed 1,030 mid-market executives in July 2026: 97% were satisfied with AI's business value, and 36% had it embedded in core processes.

People are happy with AI and still not running on it. The distance between those two numbers is almost entirely undone writing. So this guide skips the argument.

A prompt and your context are not the same thing

A prompt is what one person asks, once. Context is what the system already knows every time anyone asks anything: your standards, your history, your boundaries, your way of doing things. Prompting is a personal skill that leaves when the person leaves. Context is an asset that stays.

And it is less technical than it sounds

Context engineering is writing things down, in plain language, in files. That is genuinely most of it. The sophistication is in what you choose to write and who keeps it true, not in the tooling.

Which is why the market's answer keeps missing. Search for how to do this and you get platforms that promise to index the documents you already have. Those tools are fine. But no tool can tell you what your pricing floor is, or who may approve an exception, or which client facts must never appear in another client's room. That is not a retrieval problem. It is a writing-down problem, and it is yours.

WHAT YOU NEED

An afternoon, one folder, and five plain text files. Not a wiki, not a database, and not a tool you have to evaluate first.

Five files. Five questions. Same order.

Working alone these are five short files in one folder. At five hundred people they are five domains with owners and review cycles. The questions do not change.

THE FILE		THE QUESTION IT ANSWERS
1	Who we are and what we actually do	What do you sell, to whom, and what do you deliberately not do
2	How we sound	What would you never say
3	What things cost and who can decide	Where is the floor, and who may go below it
4	What never leaves the room	Which information crosses which wall
5	What "done" looks like	What makes the work finished and good

THE VARIABLE IS NOT HEADCOUNT

It is how many people have to agree. A consultant working alone needs the same five files as a three-hundred-person firm. They are simply faster to write, because only one person has to be right. Everything that gets harder later gets harder for that reason and no other.

Two rules before you write anything

Write the counter-examples

For all five, the useful content is at the edges. "We are professional and friendly" tells the model nothing, because every company believes that. The negative space defines the shape.

If you cannot answer one of them, that is the finding

Most people discover that one or two have never actually been decided, usually the money one and the sign-off one. The AI has been guessing at them. So, quietly, have some of your people.

Who we are and what we actually do.

Not the marketing version. This is the orientation file, and it is the one people over-write. Keep it short. A page is plenty.

What we sell, in one sentence a customer would recognize

Who we sell it to. Be specific enough to exclude someone

What we deliberately do not do, and turn down

Three things a new hire always gets wrong in their first month

1

2

3

WHY THE NEW-HIRE QUESTION EARNS ITS PLACE

It is the fastest extractor of tacit knowledge you have. Everything a new person reliably gets wrong is something true about your company that nobody wrote down. The AI is a new hire who will get those same three things wrong, forever, until you write them here.

How we sound.

Your voice, defined by contrast. A paragraph of description is worth less than five examples of things you would never say.

Five things we would never say or write

1

2

3

4

5

One real before and after. Something written badly, then the same thing written the way we actually write

Before

After

A WORKED EXAMPLE, FROM OUR OWN FILE

"We never use exclamation marks, we never say 'excited to share', and we never describe our own work as innovative." Instantly usable. "We are professional and approachable" is not.

WRITE THIS ONE FIRST

This file and file 5 produce a visible improvement in the output within minutes, and that matters more than you would think. It is what makes you keep going.

What things cost, and who can decide.

The file most companies have never actually written, and the one whose absence is most expensive.

Our standard pricing, as a rule rather than a past example

Discount authority. Who can give what, without asking

The floor. Below this we walk away

What needs a second signature

What nobody may commit to without the owner

WHY THIS CANNOT BE SKIPPED

Without it, the AI infers your pricing from your last proposal, which is a number you charged once, under conditions it cannot see. That is not a rule. It is an accident being promoted to a policy.

If two people in your company would answer the floor question differently, you have just found something worth more than the AI project.

What never leaves the room.

Name the walls explicitly. This is the file whose failures are silent and expensive. Nothing tells you it went wrong until it is already outside.

Client and customer walls. Whose facts never appear in whose work

People data. What never enters a document that goes outside

Commercial data. What stays internal regardless of who asks

What may be used to improve the system, and what may not

Who may approve an exception, and how it gets recorded

THE TEST FOR THIS FILE

A wall you have named is a wall the system can hold. A wall you assume everybody understands is a wall that holds until the first time somebody is in a hurry.

What "done" looks like.

For each thing you produce regularly, what makes it finished and good. Not a checklist of steps. A standard a reviewer could apply and reach the same verdict you would.

WHAT WE PRODUCE	IT IS DONE WHEN	IT IS NOT DONE IF

FILL THE THIRD COLUMN FIRST

People find it far easier to say what would make them reject something than to describe what would make them approve it. Write the rejections, then read them backwards into the standard.

WHAT THIS LOOKS LIKE RUNNING

VISUAL FACILITATORS, a Hamburg firm working worldwide in visual facilitation, runs a quality system that checks every output before it ships. Every person on the team has their own AI carrying the same company knowledge, with their individual voice and role on top. The firm owns and extends the system itself, on its own European storage. A first proposal draft went from 90 minutes to 30, and came back sharper. The standard is what makes the second half of that sentence true.

Five files in a folder do nothing until *the AI reads them.*

Every system does this in one of three ways. You do not need to know which product you are on to pick one.

① A workspace that remembers

Most assistants let you create a project or workspace, attach files to it once, and have every conversation inside it read them. Attach all five. Use this if your team already works in a single tool.

② A file the tool picks up on its own

Anything that works inside a folder or a repository reads a conventionally named instructions file at the root. In Claude Code that file is **CLAUDE.md**. Other tools use their own name for the same idea. The five files live beside it, and the root file says which to read and when.

③ Paste it at the top

Works everywhere, needs no setup, costs ten seconds. If you are one person doing this on a Monday, start here. Do not let choosing between the other two become the reason you never start.

WHICH ONE, IN ONE LINE

Paste for the first week. A workspace once you know it is working. A root file once the work happens in folders rather than in chats.

THE PART THAT DOES NOT TRANSFER

Your five files are yours. The way a given product loads them is that product's, and it changes. Keep them as plain text in a folder you control, and moving tools costs you an afternoon instead of the asset.

WHY THERE ARE NO SCREENSHOTS HERE

Menu paths date within two quarters, and a stale click path teaches you the whole document is old. The three mechanisms above have not changed in years.

"Write less" is the wrong lesson.

In late 2026 Anthropic removed roughly 80% of Claude Code's own system prompt. The lesson most people took from it was to write less. The right lesson is more demanding, because it asks you to sort rather than to cut.

	WHAT IT IS	WHAT TO DO
Scaffolding	Everything that exists because a model used to be weaker. The workaround for the thing it kept getting wrong. The reminder repeated in three files because it ignored the first two. The fence you built after a bad output in March.	Test the deletion rather than assume the fence is still needed. Most were written against a model that no longer exists.
Non-derivable	Everything the model cannot work out no matter how capable it becomes. Your voice. Your money boundaries. Your walls. Your sign-off. Your definition of done.	Keep, and keep current. This is the entire asset.

WHY THESE FIVE

The five files are all non-derivable. That is why they are the five.

Anthropic's own guidance says to keep files lightweight and avoid overconstraining, with a qualifier most write-ups dropped: "except in highly important areas." In a codebase the important areas are narrow. In a company, the important areas are most of what makes the company yours.

Scaffolding is a depreciating asset. No model release will ever hand you the other one.

Voice first. Money on day two.

The order to write them in

How we sound, then what "done" looks like, then who we are, then money, then the walls. Voice and done first because they change the output immediately. Money and confidentiality on day two, when you have proof it is worth the effort.

Who owns it

One person, not a committee. At small size the right number of owners is one, and it should be whoever has the most opinions about the work being right. In most firms that is the founder, which is inconvenient, because the founder is busy. It is still the answer: the context is mostly in their head, so the extraction has to come from them.

How it stays current

Fifteen minutes a month, in the owner's own calendar, answering one question: what did I correct the AI on this month that I am going to have to correct again? Write that down. That single habit is the difference between a folder that compounds and a folder that rots.

What changes as more people have to agree

WHO HAS TO AGREE	WHAT THE WORK ACTUALLY IS
One person, up to about 25	Extraction. It is in one head and needs to get onto a page. An afternoon.
Roughly 25 to 100	Arbitration. It is in several heads, they disagree, and writing it down forces someone to decide. Each file needs a named owner.
Past 100	Governed infrastructure. Access rules, review dates, and a record of who changed what.

THE FAILURE AT THE SMALL END

It is not building the wrong thing. It is building nothing while researching the right way to build it.

How you know it is working.

Open a new AI session with no setup. Ask it to draft something typical for your business: a client email, a proposal section, a scope note.

Generic

IF THE OUTPUT COULD HAVE COME FROM ANY COMPANY IN YOUR INDUSTRY, YOUR CONTEXT IS NOT LOADED, OR IT IS NOT SPECIFIC ENOUGH

Yours

IF IT SOUNDS LIKE YOU ON A GOOD DAY, IT IS WORKING

Run it monthly, at the same time as the fifteen-minute review.

Four ways this goes wrong

Writing the marketing version

File 1 becomes an About page. It should read like a briefing for someone who starts Monday.

Describing instead of contrasting

Adjectives in file 2 do nothing. Only the counter-examples and the before-and-after carry information.

Leaving money and sign-off for later

These are the two that were never decided, which is exactly why they get postponed. They are also the two whose absence costs real money.

Treating it as a document rather than a habit

A context folder written once and never corrected is worth less than no folder, because people trust it.

Block an afternoon. Write two files.

If you are the owner

Block an afternoon this week and write files 2 and 5 yourself. Do not delegate them. They are the two that are most in your head and least in anyone else's.

If you lead a function

Write file 5 for the three things your team produces most often. Then compare your third column with what your people actually reject. The gap is the conversation.

If you are the person keeping the system

Ask for the monthly fifteen minutes on the owner's calendar, and the authority to write down what comes out of it. Without both, this becomes an archive.

Where this came from

All of it is published in full, and free, at bosio.digital: how to give your AI the five things it cannot work out on its own, why "write less" is the wrong lesson, your AI does not know your business, and the data readiness myth.

Other guides in this set

Where to Put Your First AI Agent. The AI Budget Worksheet, on what an engagement actually costs in year one. Twelve Questions to Ask an AI Consulting Firm before you sign.

IF YOU GET STUCK ON ONE OF THE FIVE

Send us the one you cannot answer, or take 30 minutes with us. If we are not the right fit for where you are, we will say so, and point you somewhere better.

bosio.digital/contact

Sources: RSM, Middle Market AI Survey, July 2026, 1,030 executives. Anthropic engineering, on the Claude Code system prompt reduction and on skills guidance. Client figures as published at bosio.digital/clients.