THE OPERATOR ENGINE

Set Up Claude Like an Operator — The Complete System

Component 2 of the system. The Engine is the machinery your Operator Profile feeds: a folder architecture that keeps everything in its place, one master prompt that runs the same routine on every task, and a finishing pass that keeps the writing human. Set it up once; it runs every day after.

Part 1 — The 4-folder architecture

Everything in the Engine lives in your Claude project, in four folders. Each one has a job:

Folder	What lives there	Why
About Me	Your `about-me.md` and `anti-ai-style.md` files (Component 1)	Claude reads these before it works. They are your identity and your voice, in permanent form.
Projects	Briefs, notes, and source material for each thing you're working on	The facts of the task live here, separate from who you are.
Templates	Anything you reuse — email formats, post structures, checklists	Your proven shapes, ready to point Claude at instead of reinventing them.
Outputs	Finished work	What shipped, kept apart from drafts so you can always find the final version.

The rule that makes it work: your About Me files never move and never get buried in chat history. When Claude starts a task, it reads them from the folder — every session, not just the first one.

If you haven't built the folders yet, the Setup Sprint (Component 3) walks you through it on Day 1.

Part 2 — The Master Prompt

This is the heart of the Engine: one prompt you paste at the start of a task. Copy it exactly, save it somewhere you can grab it in two seconds (the top of your Templates folder works), and add your task after it.

> **THE MASTER PROMPT — copy and paste:**

> Before you start, read my about-me file and my style file in this project. Then ask me up to 5 questions whose answers would most change how well you do this task. Wait for my answers. Then do the task, run the humanizer pass from my style file, and show me the result — not your process.

Why it's built this way:

- *"Read my about-me file and my style file"* — Claude starts from who you are and how you write, not from a blank chat's guesses.
- *"Ask me up to 5 questions whose answers would most change how well you do this task"* — instead of you trying to write one perfect prompt that covers everything, Claude finds the gaps itself and asks about the ones that actually matter. Answer them briefly and honestly; vague answers get vague work.
- *"Wait for my answers"* — Claude doesn't guess past the questions. The task starts after you've answered.
- *"Run the humanizer pass"* — the finishing step in Part 3 runs on every draft, automatically.
- *"Show me the result — not your process"* — you get the finished piece, not a tour of how it was made.

Part 3 — The humanizer pass: the standing final step

Every draft the Engine produces goes through the **5-step humanizer pass** from your style file before you see it. It's written into the Master Prompt, so it happens every time — but know what it's doing:

1. Read it out loud in your head. Where would a person run out of breath or get bored? Fix there first.
2. Cut every banned phrase from your style file — replace it with the plain word a person would say.
3. Add one real detail only you would know: a name, a place, a mistake you made. Generic is the tell.
4. Vary the rhythm: if three sentences in a row are the same length, break one.
5. Check the last line. End on the point, not a summary of the point.

When an output still sounds off, the fix is almost never a better prompt mid-chat. It's a file edit: add the phrase that annoyed you to your banned list, or sharpen a rule in your about-me file. Fix the file once and it's fixed in every future run.

Part 4 — The daily ritual: /OPERATE

The Engine runs on one command. When you sit down to work:

1. Open your project.
2. Type **/OPERATE**
3. State the task in plain words.

/OPERATE is your signal to run the whole routine: the Master Prompt fires, Claude reads your two Profile files, asks its questions, does the work, humanizes it, and hands you the result. You don't re-paste the Master Prompt each time once Claude knows the routine in your project — type the command, state the task, answer the questions, review the result, save it to Outputs. One boundary, honestly: /OPERATE works where the routine lives — inside your project, where the Sprint set it up. In a brand-new chat outside the project, Claude hasn't met your files yet; paste the Master Prompt there instead, or better, keep your work in the project where your Profile does its job.

Keeping the Engine current:

- New project? Its brief goes in **Projects**.
- A format you liked? Save it in **Templates** so next time starts from your best version.
- Something about you changed — new role, new offer, new rule? Edit `about-me.md` once. Every run after that uses the new version.
- A phrase you never want to see again? It goes on the banned list in `anti-ai-style.md`. Once. Forever handled.

The Engine is not software and it isn't magic. It's a structure: your identity in files, one prompt that enforces the routine, and a finishing pass that protects your voice. Run it daily and Claude stops being a blank chat you manage — and starts working like it knows you.