

AI Daily Stack by Dr. Sergey Morozov

How to start building agentic systems and automations

This is a learning document: it will help you start building automations into your daily routine, even if you are not yet working with Claude Cowork at the level of agentic systems.

The typical way people use AI is **point to point**: you ask a question, you get an answer. The real leap is where you build **projects and agentic systems**: the AI works with your folders of files, keeps **file versions**, and you always have access to everything you work with.

No coding required. The main language of interaction is an ordinary conversation: the model understands **practically any spoken language**. In my experience my **productivity has grown at least tenfold**, and it is less about saving time than about volume and quality: the time you free up can go into new work or into what matters more, including personal life and sport.

I currently run about **38 skills**, **11** scheduled automations and **19** on demand, around **30 external integrations** and **7 delivery channels**, and one person runs all of it.

What changes: you stop doing the same thing by hand, results accumulate in your files and projects instead of getting lost in chats, and you delegate a whole workflow rather than a single answer.

How to think

- **Build projects, not one-off chats.** Folders of files, versioning, and constant access to everything you work with.
- **Think automation from the start.** Run every action through one question, will I repeat this, and build the repeat into Claude Cowork or Claude Code.
- **You rent the model, you own the system.** The model changes every quarter, while your real advantage is the files, rules and connections you accumulate. Grow those.
- **You stay the editor.** AI makes mistakes and forgets, so check the output and stay in control (downsides and what to do about them are at the end).

Steps to get started

- 1. Take a real task.** A payment, a tax return, or a trip plan with a dozen stops and conditions.
- 2. Describe it by voice.** You can dictate straight into Claude, ChatGPT or Gemini, or through a separate input like Wispr Flow. Ask it to assemble the prompt, check it, then run.
- 3. Connect data and services.** **MCP connectors** to anything: PubMed, email, calendars, sources.
- 4. Let the agent work on its own.** In **act mode** it runs the steps without asking for approval at each one, with access to your files and version history.
- 5. Get the result in a convenient format.** For anything you want to share, ask for an **HTML page**: you can simply send it as a file over WhatsApp or Telegram.
- 6. Check the result.** Ask the AI to re-check itself and take a screenshot of its own work for self-verification.
- 7. Save the process as a Skill,** so you can run it again without explaining from scratch.

AI Daily Stack by Dr. Sergey Morozov

How to start building agentic systems and automations

How to talk to the model: the CIDI framework

CIDI stands for **Context, Instructions, Details, Input**.

- **Context (C)**: who you are and what the task is, the role and the goal.
- **Instructions (I)**: exactly what to do, step by step.
- **Details (D)**: constraints, format, what to avoid.
- **Input (I)**: which folder and files you are working with.

A ready template from my work: abstract reviewer

"You are a peer reviewer (**C, context**). Check the manuscript against the **CLAIM and PRISMA** checklists and verify all the statistics, find errors and weak spots. Then search **PubMed** for similar publications and build a comparison table of my work against the comparable ones, with columns: method, sample, metrics, result, difference from my work. Take only **Q1** journals and papers from the last five years, prioritise highly cited ones, at most five, open access only (**I, instructions**). Give a verdict at the level of a major revision and a point-by-point list of edits, no fluff (**D, details**). The manuscript and data are in the [your_project] folder in Cowork (**I, input**)."

Tools: platforms and services

Tool	What it is and how I use it	Level	Link
Agentic platforms			
Claude · Cowork and Claude Code	My main platform. Cowork for tasks with files and projects, Claude Code for code and pipelines.	Must	claude.ai
Claude in Excel and PowerPoint	I use Claude right inside spreadsheets and presentations. I build slides through Claude Cowork and Claude for PowerPoint, it is excellent.	Should	claude.ai
ChatGPT and Codex	I do not use it, but it is a good alternative to Claude.	Could	chatgpt.com
Open solutions	An alternative for control and privacy: open-source models (Llama, Mistral) run locally via Ollama, LM Studio, OpenWebUI.	Could	ollama.com
Other services I use			
Wispr Flow	voice input (you can also dictate straight into Claude, ChatGPT, Gemini)	Should	wisprflow.ai
Nano Banana	image and portrait generation	Could	gemini.google/overview/image-generation/
Gemini	audio analysis; video via MCP from Claude Code	Could	gemini.google.com

AI Daily Stack by Dr. Sergey Morozov

How to start building agentic systems and automations

Tool	What it is and how I use it	Level	Link
NotebookLM	when I need to work strictly from the documents I provide and make audio and video overviews. I run it through the Claude integration	Should	notebooklm.google.com
Granola	turns on automatically for meetings in my calendar, I dictate notes; then an automatic export from Granola into Claude and a Notion database	Should	www.granola.ai
Perplexity	fast search on any topic	Could	www.perplexity.ai
Telegram Bot API	my personal assistant bot	Could	core.telegram.org/bots

Level by the MSCW matrix: **Must** required · **Should** desirable · **Could** optional.

Use cases: what exactly I automate

Use case	How it works
Scheduled	
Radiology AI digest weekly	PubMed → journal shortlist → scoring → Exa (smart web search) and KOLs (opinion leaders) on LinkedIn → QA (fact-check) → PDF in EN and FR → Telegram, website, audio overview and social posts
Idea digest daily	sources from the last day → digest → PDF in Telegram + inbox in Notion + drafts for LinkedIn
Morning brief daily	calendar + tasks + email → triage by work stream → priorities and KPIs → PDF to phone
Meetings to Notion daily	meeting notes → dedup (remove duplicates) → classification → summary → saved to Notion as a contact base
French lesson daily	news at my level → lesson and vocabulary → I record audio → Gemini scores pronunciation → the lesson adapts
LinkedIn analytics monthly	export and scrape of posts → trend file
On demand	
Photo annotation	selected photos from the Photos app on my phone → uploaded into NotebookLM on the laptop → I label them → Cowork agents with access to the labels and photos → a story with pictures
Family calendar	one request in Cowork → events from the French school lesson timetable and the public holiday calendar → refined events in my calendar
Conference abstract	data and the specific conference template → a structured abstract
Reviewer response	the editor letter and the manuscript → an edit table → a point-by-point reply and checklist
Literature screening	a list of abstracts → several AI reviewers → include or exclude → final human check

AI Daily Stack by Dr. Sergey Morozov

How to start building agentic systems and automations

Use case	How it works
Expense note	PDF and photos of receipts from the task folder → currency conversion → an expense sheet

Skills: nothing complicated

When you have worked on a task for a while and set everything up the way you want, just say create a skill, and next time you will not have to explain it again. A skill is a file with your custom rules: what to include and what to exclude, what format to return, which sources and connections to use. That is why it is so specific, it holds your way of working. A few of mine:

- **Abstract factory:** my rules for checking against the CLAIM and PRISMA checklists, templates for specific conferences, my review cycle, and lists of what to include and exclude.
- **Weekly AI report:** my journal allow-list, scoring rules, card format, and connections to PubMed, Exa and Telegram.
- **Post publishing:** my authorial voice and rules (no em dash, text length, hashtag set), image hosting and queuing into Buffer.

Delivery channels

Telegram or Slack (main mobile, files and pushes), Notion (databases and ideas), Gmail (drafts only), Buffer (social scheduling), personal website, email newsletter, Google Calendar.

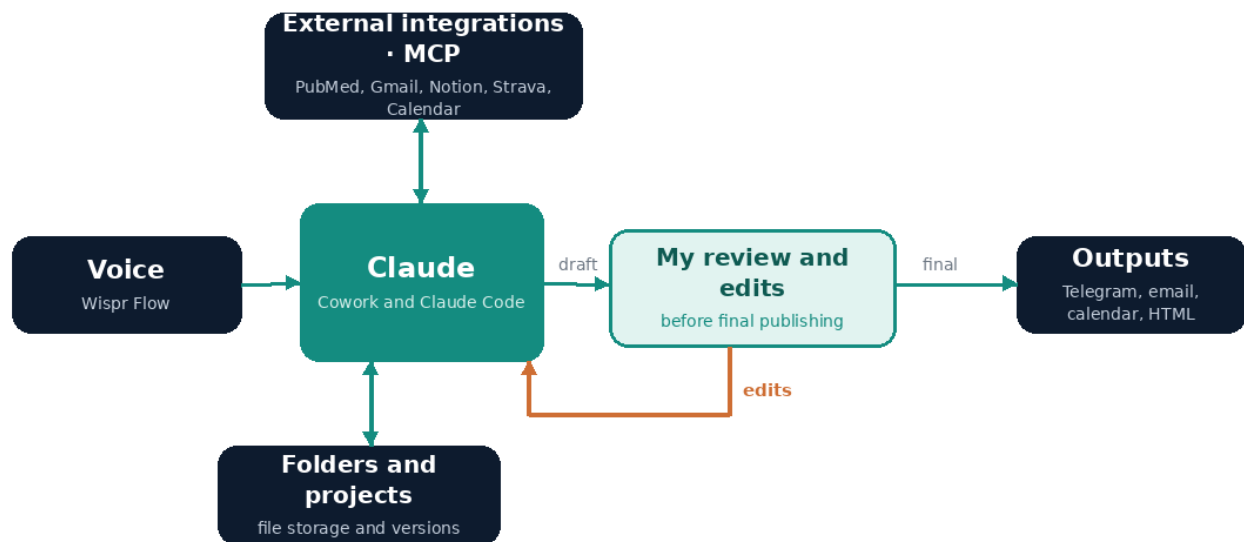
Honest downsides and what to do about them

Problem	What to do
Hallucinations	The model confidently invents facts and figures. Ask for sources, verify key numbers by hand, do not publish without checking.
Context window limit	It does not all fit in one session, important things get pushed out. Break the task up, keep context in project files, feed only the pieces you need.
Forgetting between sessions	The model does not remember past conversations. Record rules and agreements in project files, and start with a short context.
Over-engineering	It often gives a bulky solution. Explicitly ask for a simple version and limit the scope.
Addictive	It is easy to spread yourself across ten projects at once. Limit the number of parallel tasks.

AI Daily Stack by Dr. Sergey Morozov

How to start building agentic systems and automations

How it all connects: my workflow



The loop: draft → my review and edits → and only then the final output

Where to start right now

Open Claude Cowork, pick a working folder with the documents you use, and just start talking out loud about what you want to get. Record as detailed a description as you can, and at the end say: based on this, assemble a prompt and suggest which actions on this task you can carry out for me. And do not forget to set up integrations with the services I use. You will add to them as you go.

Dream big and do not be afraid to set the most ambitious tasks: you can now build almost anything with AI tools.

Sergey Morozov · drsergeymorozov.com · LinkedIn: sergey-morozov