Hector Garcia

The 8 tool stack to build an AI agent in 2026

TOOLS · 7 MIN READ · OCT 28, 2026

An AI agent is not one tool. It is a brain, a place to remember things, a way to talk to people and a way to take action, all wired together. Below is each tool from the reel, the exact job it does, how a single customer message moves through all 8, and a weekend plan to build your first agent without writing code from scratch.

THE STACK AT A GLANCE

JOB	TOOL	WHAT IT DOES IN YOUR AGENT	COST TO START
Brain	Claude	Reads the message, decides what to do and writes the reply in a natural voice	Free plan, API is pay per use
Builder	Claude Code	Writes the custom code for you when you describe what the agent should do	Needs a paid Claude plan
Workflows	n8n	Connects the agent to your email, calendar, CRM and spreadsheets so it can actually do things	Free to self host, cloud has a free trial
Memory	Supabase	Stores customers, past conversations and your business info so the agent never starts from zero	Free tier
Voice	ElevenLabs	Turns the agent's replies into a human sounding voice for phone calls	Free tier
Chat	Voiceflow	The chat widget on your site where customers talk to the agent	Free plan
Hosting	Vercel	Puts your agent's site and endpoints online with a public link	Free hobby plan
Monitoring	Langfuse	Logs every question and answer so you can spot wrong replies and fix the prompt	Free cloud tier, open source

HOW ONE MESSAGE MOVES THROUGH THE STACK

A customer opens the chat on your site (**Voiceflow**) or calls your number (**ElevenLabs** handles the voice). Their message gets sent to a workflow in **n8n**.

n8n looks the customer up in **Supabase**, so the agent knows who they are and what they asked last time. Then it passes the message plus that context to **Claude**, which decides what to answer and whether an action is needed, like booking a slot or sending a quote.

If there is an action, n8n does it: creates the calendar event, updates the CRM, sends the email. The reply goes back to the chat or the call, the conversation gets saved to Supabase, and **Langfuse** records the whole exchange so you can review it later.

Claude Code and **Vercel** sit behind all of this. Claude Code writes any custom piece n8n can't do with its built in nodes, and Vercel puts that code online.

BUILD YOUR FIRST AGENT IN A WEEKEND

Do these in order. Each step works on its own before you add the next one.

1. Pick **one** job for the agent, for example answering the 10 most common questions for a clinic and booking appointments. One job, done well, beats a do everything agent
2. Write the agent's instructions and test them in Claude by chatting with it as if you were a customer. Fix the instructions until 9 out of 10 answers are right
3. Create a free Supabase project with two tables: customers and conversations. Add your business info (hours, prices, services) as a third table
4. Build an n8n workflow: webhook in, look up the customer in Supabase, send everything to Claude, save the reply, send it back
5. Connect the front door: a Voiceflow chat on your site first. Add ElevenLabs voice only after chat works
6. Turn on Langfuse and read the first 20 real conversations. Every wrong answer is a line to add to the instructions
7. Only when you hit something n8n can't do, open Claude Code, describe the missing piece and deploy it on Vercel

PROMPTS TO GET STARTED

Fill in the brackets and paste them as is.

THE AGENT'S BRAIN (PASTE INTO CLAUDE)

```
You are the assistant for [business name], a [type of business] in [city].

Your job:
- Answer customer questions about [services, prices, hours, location].
- Book appointments when someone is ready, by collecting [name, phone, preferred day
and time].

Business info:
[paste hours, prices, services, address, policies]

Rules:
- Only use the business info above. If you don't know, say a person will follow up
and ask for their phone number.
- Keep replies under 3 sentences and sound like a friendly front desk person.
- Never promise discounts, medical advice or anything not listed above.

When a booking is ready, end your reply with:
BOOKING: name | phone | day | time
```

PLAN THE BUILD (PASTE INTO CLAUDE CODE)

```
I want to build an AI agent for [business type] that [one job, e.g. answers questions
and books appointments].

My stack:
- Brain: Claude API
- Workflows: n8n
- Memory: Supabase
- Chat: Voiceflow
- Hosting: Vercel
- Monitoring: Langfuse

Task:
1. Propose the Supabase tables I need with their columns.
2. List the n8n workflow steps from incoming message to reply.
3. Tell me which parts need custom code and which n8n can do on its own.
4. Build the custom parts one at a time, and wait for me to confirm each one works
before moving on.
```

FIX A BAD ANSWER (USE WITH LANGFUSE LOGS)

```
Here are my agent's instructions:
[paste instructions]

Here is a conversation where it answered wrong:
[paste the conversation from Langfuse]

What the right answer should have been:
[the correct answer]

Task:
1. Explain why the agent answered wrong.
2. Give me the smallest change to the instructions that fixes it without breaking
other answers.
3. Give me 3 similar test questions to check the fix.
```

BEFORE YOU PAY FOR ANYTHING

Note: every tool here has a free way to start except Claude Code, which needs a paid Claude plan. Build the whole first version on free tiers and only upgrade the tool you actually outgrow. Prices change often, so check each pricing page before upgrading.

Don't stack two tools that do the same job. One memory, one chat, one workflow tool. Every extra tool is one more thing that can break.

QUICK LINKS

Brain: Claude
Builder: Claude Code
Workflows: n8n
Memory: Supabase
Voice: ElevenLabs
Chat: Voiceflow
Hosting: Vercel
Monitoring: Langfuse

WANT MORE LIKE THIS?

I break down a new AI tool or build every week on Instagram @hex.gar. Follow along if you want the next one.

Run a business and would rather have an agent built for you? Book a free call here.

WORK WITH ME

Business owner looking to automate?

hexgarcia.com/free-call

Questions? Reply to the newsletter or reach out via email.

Hector Garcia
@hex.gar · contact@hexgarcia.com