

FREE DOWNLOAD

TAKE THE IDEAS · AI STRATEGY & PROMPT ENGINEERING

How to Build an **AI Agent** With a Simple MD File

A beginner-friendly guide to creating your own AI agent instructions using a single Markdown (.md) file — no coding experience required.

THE BASICS

What Is an "Agent" File, Really?

An AI agent is just a set of written instructions that tells an AI tool how to behave, what its job is, and what rules it must follow — every time, consistently. Instead of re-explaining yourself in every chat, you write it once in a plain text file, and any AI tool that supports it will read those instructions automatically.

The most common format for this today is a Markdown file (**.md**) — a simple text format used across the AI industry. You may see it called **AGENTS.md**, a "system prompt file," or a "Gem/agent instructions" file, depending on the tool — but the idea is the same: one file = one clearly defined AI helper.

WHY THIS MATTERS

- You stop repeating the same instructions every time you open a chat.
- Your AI agent behaves consistently — same tone, same rules, same output style.
- You can hand the same file to a teammate, or reuse it in Gemini Gems, GenSpark, ChatGPT, or any coding assistant that reads instruction files.
- You build a reusable "AI employee" for one specific job (customer replies, content drafts, research summaries, code reviews, etc.).

Writing Your First Agent File

1 Pick One Clear Job For Your Agent

Don't try to build one agent that does everything. Start narrow: "answers customer FAQs," "drafts weekly social captions," or "reviews my code for bugs." A focused agent performs far better than a general one.

2 Create a Plain Text File Named `agent.md`

Open any text editor (Notepad, TextEdit, VS Code, or even Google Docs saved as plain text) and save a new file as **agent.md**. The ".md" extension just tells tools it's a Markdown file — you don't need any special software to create one.

3 Fill In These Five Sections

1. **Role** — Who is this agent? ("You are a customer support assistant for a small bakery.")
2. **Goal** — What is it trying to accomplish? ("Answer questions about hours, orders, and allergies.")
3. **Rules** — What must it always do or never do? ("Never promise a refund. Always ask for the order number.")
4. **Tone** — How should it sound? ("Warm, brief, and friendly — 2-3 sentences max.")
5. **Knowledge** — Any facts it needs to know (hours, prices, policies, FAQs).

A Ready-to-Use Template

Copy this into your agent .md file and replace the bracketed text with your own details.

```
# Agent: [Name Your Agent]

# -- ROLE --
You are [describe the role, e.g. "a scheduling assistant for a
solo consultant"]. Your job is to [main responsibility].

# -- GOAL --
Your primary goal is to [what success looks like, e.g. "help
visitors book a discovery call without needing a human to
reply first"].

# -- RULES --
- Never [something it must not do, e.g. "invent pricing"].
- Always [something it must always do, e.g. "ask for the
visitor's timezone before suggesting a time"].
- If you don't know the answer, say so and suggest [fallback
action, e.g. "emailing rafael1robles4@gmail.com"].

# -- TONE --
[e.g. "Warm, concise, professional. 2-4 sentences per reply.
No corporate jargon."]

# -- KNOWLEDGE --
- [Fact 1, e.g. "Business hours: Mon-Fri, 9am-5pm MT"]
- [Fact 2, e.g. "Services offered: X, Y, Z"]
- [Fact 3, e.g. "Contact: email / phone"]
```

Keep each section short and specific. Vague instructions ("be helpful") produce vague results. Specific instructions ("always ask for the order number before answering a refund question") produce reliable, repeatable behavior.

Where to Use Your Agent File

Gemini Gems

Paste the contents of your agent .md file directly into the instructions box when creating a new Gem in Gemini — it becomes the Gem's permanent behavior.

GenSpark / ChatGPT / Claude

Paste it as a "custom instructions" or "system prompt" when starting a new project or custom assistant — most modern AI platforms support this exact pattern.

Coding Agents (Antigravity, Cursor, etc.)

Save it as AGENTS.md in the root of your project folder. Coding agents automatically read this file to understand your project's rules, structure, and how you want changes made — this is now an open, widely-adopted standard across AI coding tools.

Want Help Building Yours?

Writing an effective agent file for your exact business is a skill — the difference between a generic response and one that sounds like it truly knows your business.

Email your pain points to rafael1robles4@gmail.com and Rafael will map out a personalized AI blueprint — including a custom agent file — for your business.