docWeb

Practical guide · Version 1.1 (EN)

AI, agents, and you

A plain-language guide to use AI seriously — and still move forward

For curious beginners, small organisations,
and anyone tired of empty talk.

> ***AI moves fast. Judgement keeps it on course.***
>
> *Without understanding, you are guessing. With clear limits, you can build.*

This paper is deliberately simple. No jargon for show — practical checkpoints, in the spirit of **docWeb** : understand, verify, stay in control.

1. What everyone should know

Artificial intelligence is a **remarkable tool**. It is neither a diploma, nor a profession, nor a guarantee.

Today, many people speak with confidence about subjects they have never really practised. AI amplifies that: in a few seconds it can produce professional-looking text, a piece of code, or “expert advice”.

The problem is not using AI. The problem is **believing it replaces understanding**.

> ***Nobody becomes an accountant, developer, designer, lawyer, or vet***
>
> *just because a model produced a fluent answer.*

The right attitude is to:

1. **Get informed**
2. **Build the necessary basics**
3. **Understand enough** of what AI proposes
4. **Check and correct** what is wrong
5. **Test** before you trust
6. **Stay in control**

You do not always need to understand every detail or every line of code. You do need to understand the result well enough — its limits and the checks you ran — so you are not working blind.

Knowledge is not a luxury for specialists. It is a **treasure** — and the only lasting safety net.

2. Hallucination: the word you need

When AI is wrong, it does not always say so. It can **invent**, **distort**, or build an explanation that sounds perfectly logical but matches neither the facts nor your situation. It can also mix truth and falsehood with great confidence.

That is called a **hallucination**. It is not science fiction. It is a concrete risk:

- a reference or court decision that does not exist;
- a figure presented as exact, but wrong;
- an invented software function;
- a rule that applies in another country or an old version;
- a plausible explanation that is off-context;
- a mix of true and false that is hard to spot.

> ***Confidence is not proof.***
>
> *The remedy is not fear of AI. The remedy is verification.*

Simple rule: the more important a decision is, the more the answer must be **explained, checked, and controlled** — not merely copied.

3. The cat example — and why it matters

Situation A — inside your zone of understanding. You know a topic in broad terms — for example small-business bookkeeping. You use AI to move faster, look up information, review a form, or structure your work. You compare answers with official documents, ask questions, correct, and learn. You **progress**. The tool increases your capacity.

Situation B — outside your zone. Your cat is sick. Even if AI produces a very convincing diagnosis, you do not play the vet. It may help you describe symptoms, prepare questions, or recognise urgency. It must not lead you to give a risky treatment.

The same logic applies everywhere:

Situation	Sound approach with AI
You have basics	Accelerate, learn, correct, deliver
You know little	Learn first, verify, avoid irreversible decisions
Stakes are high	Check official sources and get validation if needed
Health, law, security, or critical money	Competent advice, stronger verification, human decision

Humility is not a brake. It is what lets you move without turning a mistake into a disaster.

4. Chatbot or agent: not exactly the same

A **chatbot** mainly answers questions. It produces text, explanations, images, or code that you then choose to use or not.

An **agent** can go further. Depending on its tools, it may:

– read or modify files;
– search for information;
– use software or a website;
– manipulate data;
– run code;
– send a message;
– create a task;
– publish or deploy a service.

An agent generally combines:

1. a **goal**;
2. **instructions**;
3. an AI model to reason and choose actions;
4. **tools** to act;
5. **permissions** that define what it is allowed to do;
6. controls to observe, stop, or correct its work.

This is not magic. It is a system that observes, decides, and acts within a defined scope.

> *A chatbot can give a bad answer.*
>
> ***A poorly scoped agent can turn a bad answer into a bad action.***

5. What an agent can do — and what it does not guarantee

What an agent can do

- Quickly produce a first base: page, component, function, or work plan
- Explore several paths
- Structure an idea or project
- Draft documentation
- Generate and run tests
- Hunt for an error across a set of files
- Automate repetitive tasks
- Prepare an action before your validation
- Work while you keep the wheel

What an agent does not guarantee

- That its goal was perfectly understood
- That the information used is accurate or up to date
- That the result truly fits your context
- That the code is safe in every case
- That its tools will work as expected
- That it will catch all of its own mistakes
- That an action can easily be undone
- That it will take responsibility for consequences

An agent can fail in several ways: hallucinate, misunderstand the goal, pick the wrong tool, fail mid-operation, or technically succeed at an action that was not desirable.

> ***The agent accelerates and can act.***
>
> ***You set the limits, verify, and remain responsible for the decision.***

6. Giving tools also means giving power

The more tools and permissions an agent has, the more useful it can be. The more an error can also have consequences.

It does not necessarily need:

– access to all your files;

– to read all your email;

– to know all your passwords;

– to change production directly;

– to send a message without confirmation;

– to permanently delete data.

Good practice is to give it **only what it needs for the task at hand**.

A few simple rules:

1. **Limit permissions** to the strict minimum
2. **Separate** test and production environments
3. **Require confirmation** before a sensitive action
4. **Keep a trail** of important actions
5. **Plan a rollback**: backup, version, or trash
6. **Protect secrets**: passwords, keys, personal data
7. **Stop the agent** when its behaviour becomes opaque

> *The more an agent can act, the more its permissions must be limited,*
>
> *its actions auditable, and important decisions subject to human confirmation.*

Control is not watching every second. It is building a setup where error stays **visible, limited, and recoverable**.

7. Small business: no full team? Method still applies

A small organisation often cannot afford a full-time designer, a large-agency dev team, a specialist for every topic, or an expensive “AI consultant” for every decision. That is not shameful. That is reality.

It can still:

Step	Detail
1. Define the need clearly	For whom? Why? With what limits?
2. Set sound basics	Clarity, structure, readability, security, data protection.
3. Build with AI and agents	Pages, functions, mockups, scripts, documentation.
4. Understand the result enough	What was done, for what purpose, with what risks.
5. Correct bad assumptions	Hallucinations, omissions, off-topic, context errors.
6. Test	Normal cases, edge cases, possible failures and their impact.
7. Integrate cleanly	Connect the result without weakening the whole system.
8. Decide go-live as a human	Never settle for: “AI said it was fine.”

You are not less professional because you use agents. You are professional if you apply real discipline, know your limits, and own what you ship.

8. Surface and depth — no self-deception

Surface. Text, layout, first mockup, illustration, interface prototype: AI is often very fast here. In minutes it can produce something that **looks professional**. A surface error is usually visible and often fixable without major damage.

Depth. Business logic, data, access rights, payments, configuration, security, servers, backups: these show less, but hold the system up. A deep error can stay silent for weeks, then become expensive when discovered.

Even design is not only decoration. A pretty page without clear structure, accessibility, or consistency quickly becomes a burden.

> *The criterion is not the fashionable language.*
>
> ***Do I understand this result well enough to correct it, test it, and own it?***

If yes: move with AI. If no: learn first, shrink the scope, or get help on the critical zone.

9. The docWeb method in 8 steps

Keep this near the screen:

1. **Frame** — What do I want? For whom? What limit?
2. **Assess risk** — What if the result is wrong?
3. **Check the basics** — Do I have the minimum to judge the result?
4. **Limit access** — Which tools and data does the agent actually need?
5. **Generate or act** — Use AI inside that scope.
6. **Read and correct** — What looks false, vague, or off-context?
7. **Test and verify** — What proves it works and does not break things?
8. **Validate and control** — Human decision, clean integration, rollback planned.

It is simple. It is not always easy. But each well-run project leaves you **more competent than before** — not only more dependent on a tool.

10. Three levels of control

Not every action needs the same supervision.

Level	Example	Reasonable control
Low risk	Draft, summary, mockup	Review before use
Moderate risk	Code changes, data, internal automation	Test, review changes, keep a previous version
High risk	Payment, deletion, publish, official message, health or security	Explicit human confirmation and stronger verification

The goal is not to slow every task. The goal is to place human control where it matters. You can let an agent prepare a sensitive action. You should not necessarily let it trigger it alone.

11. Motivation without magic

You do not need to be a genius. You need **curiosity**, **honesty**, and a bit of **methodical courage**.

› AI can save you weeks.

› It can also cost you months if you validate too quickly.

› A well-scoped agent can do impressive work.

› An overpowered, poorly controlled agent can amplify a simple mistake.

› Truly competent people are not those who talk AI with the most confidence.

› They are those who deliver, verify, and can say: “Here I need to learn or ask for help.”

The real asset is not the perfect prompt. It is **knowledge and method** — enough to spot an overconfident answer and stop a bad action before it lands.

12. Summary

Idea	Line to keep
Improvisation	AI does not hand you a ready-made profession.
Hallucination	Confidence is not proof.
Chatbot	It mainly proposes an answer.
Agent	It can turn a decision into an action.
Permissions	Give only the access needed.
Small business	No full team does not mean no standards.
Method	Frame → limit → understand → correct → test → validate.
Healthy limit	Outside your domain, do not play the expert.
Control	A sensitive action must stay verifiable and recoverable.
Cap	Stay in charge.
Hope	Knowledge remains the real lever.

Go further with docWeb

Explain without unnecessary jargon. Show what AI and agents can really do — and what they do not replace. Learn to use them without acting blind.

Fewer promises. More proof. More understanding. More control.

docweb.io · docweb.fr

Use AI. Do not fake expertise.

Keep learning. Stay in control.

docWeb

Written to be read, shared, discussed — and challenged.