

PRACTICAL AI AGENT GUIDE

What an AI agent actually is

A plain-English guide to the difference between a chatbot that answers and an agent that can take several authorised actions toward an outcome.

An answer is not the same as an action.

Start with the goal, define the permitted steps and keep a person accountable for what happens next.

01

Separate answer from action.

A chatbot responds. An agent may use tools and change business systems.

02

Set the authority.

Name exactly what it may do, approve and send without asking.

03

Keep ownership.

A person still explains the outcome and handles exceptions.

If the system can change something, decide who owns the change.

[Use this guide →](#)

MISSION & SCOPE

A chatbot answers. An agent can take the next step.

The useful distinction is not how human the interface sounds. It is whether the system can act across tools, records or messages after receiving a goal.

THIS GUIDE IS USEFUL IF

- You are comparing chatbot and agent products.
- A proposal promises multi-step action.
- The team cannot explain what the system may change.

THIS GUIDE WILL NOT

- Claim an agent removes all human responsibility.
- Judge a product by the label on its homepage.
- Recommend a particular vendor or capability.

How the distinction works

01

Goal

A person defines the outcome, scope and limits.

02

Actions

The agent selects and performs authorised steps using connected tools.

03

Outcome

A person reviews results, exceptions and business consequence.

**A knowledgeable receptionist can answer a question.
An agent is the receptionist who can also book, update and confirm.**

The extra value is action. The extra risk is action too. Every connected tool, permission and handoff expands what must be controlled.

CAPABILITY MAP

Six questions that reveal whether the system is really an agent.

Ignore the marketing label. Trace what happens between a goal and a completed outcome.

Follow the action trail.

If nothing changes outside the conversation, the system may still be a chatbot or assistant rather than an agent.

01

Goal

Does it work toward an outcome rather than answer one prompt?

02

Planning

Can it select more than one step or tool?

03

Action

Can it create, update, send or trigger something?

04

Authority

Which actions can happen without approval?

05

Memory

What context persists between steps or runs?

06

Exception

What happens when the next step is unclear or fails?

ANSWER VERSUS ACTION

The difference appears after the first response.

Trace the path from request to business outcome.

CHATBOT	AGENT
01 Question → answer Produces information inside the conversation for a person to use.	01 Goal → several actions May work across connected tools before returning an outcome.
02 Person takes action A person copies, books, updates, sends or decides what happens next.	02 System acts within limits The system executes authorised steps and escalates exceptions.
CONTROL POINT The answer still needs interpretation before anything changes.	OWNERSHIP TEST Name permissions, approvals, logs, fallback and final owner.

Know what can act? Audit the task before tooling →

COMMERCIAL EFFECT

Action creates leverage only when authority and fallback are clear.

The commercial question is not whether the system looks intelligent. It is whether the actions remove useful work without creating invisible exposure.

Clearly scoped actions

Useful leverage

The system handles named steps inside known limits.

Undefined authority

Uncontrolled change

The system can update, send or commit without a visible boundary.

Logged activity

Traceable outcome

People can see what happened, when and through which tool.

Invisible activity

Hard-to-explain result

A result appears with no reliable record of the path taken.

Fallback defined

Continuity

A person can take over when the agent cannot continue safely.

No exception path

Stalled process

The workflow waits, repeats or acts incorrectly when conditions change.

Ask what the system can change before asking how clever it is.

Agent value appears across repeated runs. Setup, permissions, testing and exception handling come first.

FIVE SIGNS THE AGENT IS NOT DEFINED

Problems to catch before agents act.

These faults are about scope and authority, not product terminology.

Critical

Sign 1

The 'agent' only answers questions

The business plans around actions that still require a person to copy and execute.

WHAT IT COSTS

A tool is expected to complete work it was never connected or authorised to perform.

Major

Sign 2

No one can state its unsupervised authority

Permissions were granted without documenting what the system may change.

WHAT IT COSTS

Customer, financial or operational changes happen without a clear approval rule.

Major

Sign 3

Tooling comes before task scope

The team starts with a platform and searches for work to justify it.

WHAT IT COSTS

Cost and complexity arrive before the business case or process boundary.

Moderate

Sign 4

Agent and automation are treated as identical

A fixed workflow and a flexible action system are discussed as the same thing.

WHAT IT COSTS

The wrong design is chosen because two overlapping concepts were collapsed.

Moderate

Sign 5

Nobody can explain the agent plainly

The description relies on product language instead of goals, actions and limits.

WHAT IT COSTS

Staff cannot monitor, challenge or improve a process they do not understand.

A useful definition names the goal, tools, authority and exception path.

If the explanation ends at 'it uses AI', the operating model is still missing.

CORRECTION AND SELF-ASSESSMENT

Five questions before an agent touches business systems.

Common myth: an agent removes the person. Better framing: it moves work, while accountability still needs a named owner.

01

What outcome is the agent responsible for?

Describe the completed business result, not the tool or interface.

02

Which actions can it take without approval?

List creating, updating, sending, purchasing or committing separately.

03

Which systems and data can it access?

Every connection expands both capability and exposure.

04

What evidence shows what happened?

Logs and records should support review, correction and learning.

05

Who takes over when the path breaks?

Name the person, trigger and safe state before launch.

WORKING AGENT DEFINITION

A reusable scope your team can test before buying.

Describe the operating model in plain language. If the team cannot fill this in, the proposed agent is not yet ready to assess.

01 Name the goal

Define one outcome the agent is meant to complete.

02 List the actions

Separate reading, drafting, updating, sending and committing.

03 Set permissions

Mark which actions need approval and which may run.

04 Map connections

List systems, accounts and information the agent can access.

05 Define evidence

State what log or record proves what occurred.

06 Plan exceptions

Name the trigger, safe state and human owner.

INSTEAD OF

“It will handle the process.”

No actions, limits, evidence or owner.

Reusable agent scope

GOAL:

ACTIONS:

CONNECTED TOOLS:

DATA ACCESS:

MAY RUN ALONE:

NEEDS APPROVAL:

EVIDENCE:

EXCEPTION OWNER:

DISCUSS THE SCOPE →

Turn the label into an operating model.

Map one goal, its permissions and exception path.

DUE DILIGENCE

Ask what acts, what changes and who remains accountable.

Three rules and two next steps for evaluating an agent proposal.

01

Trace the full path

Follow the goal through every action, tool and business change.

02

Separate permissions

Review reading, drafting, updating, sending and committing individually.

03

Test the exception

Ask what happens when context is missing, a tool fails or approval is denied.

An agent is an operating model, not a more impressive chatbot label.

If a provider cannot explain actions, authority, evidence and fallback plainly, keep digging.

CONTINUE LEARNING

The Automation Audit

Find which repeated tasks are worth handing over first.

[Read next →](#)

AI Agents & Automation

Continue through the rest of this collection.

[Explore collection →](#)

Define the actions before you trust the label.

A useful agent has a clear goal, bounded authority, visible evidence and a person ready to handle exceptions.

PRIMARY NEXT STEP

The Automation Audit

Find the repeated tasks that genuinely earn automation effort.

[Continue →](#)

AI Agents & Automation

See every guide in this collection.

[Explore collection →](#)

All guides

Browse every topic and problem.

[View library →](#)

Need help scoping an agent safely?

Bring one proposed use case. Map the actions, permissions, systems, evidence and exception owner.

[Discuss the scope →](#)