FREE FIELD SHEET · Nº 004
ISSUED 26 SEP 2026

KEYWORD
AGENT

THE AI AGENT BLUEPRINT

How to scope one that survives contact with a real business

CHECKS	DO FIRST	PARTS	COST
31	7	6	Free



PREPARED BY
Hamza Jameel
10+ yrs · $5M+ ad spend managed · 150+ page-one rankings

Most AI agent projects fail at the scoping stage, not the building stage. This is the sequence I use, and the questions that kill bad ideas early, before anyone has written code.

ON THIS SHEET

1	Before you build anything	6 CHECKS
2	Choose the pattern	4 CHECKS
3	Data and grounding	5 CHECKS
4	Guardrails	7 CHECKS
5	Testing	4 CHECKS
6	Running it	5 CHECKS

TICK IT ON YOUR PHONE
youtube.hamzajameel.com/free/AGENT

PART 1 OF 6 — 6 CHECKS

Before you build anything

If you cannot answer these, you are not ready. Most projects that die in month three died here, silently, in week one.

- [ ] 1.1 DO FIRST

Name the one task, precisely

'Handle customer queries' is not a task. 'Answer delivery-status questions using our order database' is.

- [ ] 1.2 DO FIRST

Count how many times a week that task happens

Under ten and a human is cheaper. Be honest about the number.

- [ ] 1.3

Write down what a correct answer looks like

If you cannot describe correct, you cannot test it and you will never know if it works.

- [ ] 1.4

Write down what a wrong answer costs

This decides how much human review the design needs. A wrong delivery date is an annoyance; a wrong price is a refund.

- [ ] 1.5

Identify where the data lives, and whether you can actually reach it

Most agent projects stall on access to a system nobody has API keys for.

- [ ] 1.6

Decide who owns it when it breaks

An agent with no owner degrades quietly until someone notices months later.

PART 2 OF 6 — 4 CHECKS

Choose the pattern

Pick the simplest one that solves the task. Complexity is where these projects go to die.

- [ ] 2.1

Retrieval, answers questions from documents you control

Simplest and most reliable. Start here unless you have a reason not to.

☐ 2.2

Tool-using, reads and writes to real systems

Stock levels, bookings, CRM records. Needs the guardrails in section 4.

☐ 2.3

Workflow, a fixed sequence with AI at specific steps

Most business value sits here and it's the least glamorous.

☐ 2.4 DO FIRST

Autonomous multi-step, decides its own sequence

Genuinely useful in narrow cases. Also where most budget is wasted. Justify it.

PART 3 OF 6 — 5 CHECKS

Data and grounding

☐ 3.1 DO FIRST

Source documents are current and someone owns updating them

An agent answering from last year's pricing is worse than no agent.

☐ 3.2

Chunking matches how the content is structured, not a fixed size

☐ 3.3

Every answer can cite which source it came from

Non-negotiable for anything customer-facing. It's also how you debug.

☐ 3.4 DO FIRST

The agent says 'I don't know' rather than inventing

Test this deliberately with questions you know aren't covered.

☐ 3.5

Personal data handling reviewed before launch

What gets sent to the model provider, what gets logged, what gets retained.

PART 4 OF 6 — 7 CHECKS

Guardrails

Everything here is about the gap between a demo and a system you'd leave running unattended over a weekend.

- [ ] 4.1 DO FIRST
Actions that cost money or are irreversible need human approval

- [ ] 4.2
Rate limits on every external call

- [ ] 4.3
A hard spend cap on model usage
A loop without a cap can produce a genuinely alarming invoice overnight.

- [ ] 4.4
Every action is logged with inputs and outputs
Without a log you cannot explain what happened, and you will be asked.

- [ ] 4.5
A kill switch that a non-technical person can operate

- [ ] 4.6
Prompt-injection tested
If the agent reads user-supplied content, someone will try to instruct it through that content.

- [ ] 4.7
Behaviour defined for when a tool call fails
Silent failure is worse than a visible error.

PART 5 OF 6 — 4 CHECKS

Testing

- [ ] 5.1 DO FIRST
50 real questions collected before launch, with expected answers
From actual support tickets or emails, not imagined ones.

☐ 5.2

Re-run that set on every prompt change

Prompt edits regress silently. This is the whole safety net.

☐ 5.3

Edge cases tested deliberately: hostile, ambiguous, out-of-scope

☐ 5.4

A human reviews the first 100 live interactions

You will learn more in that week than in a month of planning.

PART 6 OF 6 — 5 CHECKS

Running it

☐ 6.1

Cost per interaction tracked from day one

The unit economics decide whether this scales or quietly becomes a liability.

☐ 6.2

Resolution rate measured against the human baseline

☐ 6.3

Escalation path to a person is obvious and easy

Trapping someone in a loop with a bot costs more goodwill than the agent saves.

☐ 6.4

Monthly review of what it got wrong

The failures are the roadmap.

☐ 6.5

Model version pinned, upgrades tested before switching

A provider update can change behaviour without warning.

THE QUESTION THAT SAVES THE MOST MONEY

'What happens if it gets this wrong?' Ask it about every capability before you build it. Half the features people request stop being worth it once the cost of being wrong is written down next to them.

WANT ME TO LOOK AT YOURS?

If you have gone through this and want a second pair of eyes, book a free call. We open the account together and look at what is actually happening. No deck, no pitch.

- **Book a free call** hamzajameel.com/book-a-call/
- **Free SEO audit** hamzajameel.com/free-seo-audit/
- **Instagram** @ihamza.jameel
- **YouTube** @ihamzajamil

SOURCES

1. Built from production agent deployments at hamzajameel.com/ai-agent-development/