WUNDERLANDMEDIA

The AI Automation Scoping Worksheet

Six questions to answer in writing before a build price exists, so the project doesn't eat your margin.

By Kemal | wunderlandmedia.com

17→42%

Companies abandoning most AI initiatives, year over year

46%

Of proofs-of-concept scrapped before production

6

Failure modes to price before you quote

Both figures: S&P Global Market Intelligence / 451 Research, Voice of the Enterprise: AI, covering more than 1,000 organisations in North America and Europe. Self-reported. See page 2 for how to read the numbers people will quote at you.

1 Before You Quote: The Six Failure Modes

An AI automation brief usually arrives as one sentence and a gesture at a folder. That sentence hides six separate questions, and each one left unanswered is worth somewhere between a week and the entire margin. This document is one page per question.

THE MAP OF THE REST OF THIS DOCUMENT

1 **Nobody agreed what "working" means** — page 3
80% accuracy is a triumph on one task and a write-off on another. Decide which, in numbers, before you build.

2 **The data is worse than the client said** — page 4
Not because they lied. Nobody has looked at all of their data at once in years.

3 **"Just connect it to our system"** — page 5
The API may exist behind an IT department that needs six weeks to issue credentials. You cannot fix that with better code.

4 **Edge cases arrive as "it should obviously also handle this"** — page 6
Framed as a bug, priced as a feature, delivered as free work unless the boundary is written down.

5 **Human review is the deliverable, not the fallback** — page 7
The review workflow is usually most of the actual work. Leave it out of scope and you build it anyway, unpaid.

6 **The maintenance tail nobody prices** — page 7
Models get retired, prices change, the client's own process shifts. Someone owns day 91.

HOW TO READ THE FAILURE STATISTICS SOMEONE WILL QUOTE AT YOU

The "95% of AI projects fail" line does not say what people think

That figure comes from *The GenAI Divide: State of AI in Business 2025*, published by MIT's NANDA initiative. It is not a peer-reviewed MIT study, and it does not claim 95% of AI projects fail. It reports that 95% of the pilots it examined showed no measurable P&L impact within roughly six months. The evidence base is 52 executive interviews, a survey of around 150 leaders, and a scan of about 300 public deployments, and the report itself describes its interview findings as directionally accurate rather than drawn from company reporting. Cost savings, cycle time and churn were not counted. Do not repeat it as an observed failure rate.

17→42% The share of organisations abandoning most of their AI initiatives, year over year, per S&P Global Market Intelligence's 451 Research (Voice of the Enterprise: AI, 1,000+ organisations in North America and Europe). Still self-reported, but abandonment is something people can answer accurately about themselves.

46% Share of proofs-of-concept the average organisation scrapped before production, same 451 Research survey. This is the number worth quoting in a sales conversation, because it is about the stage you are being asked to price.

Gartner's 40% is a forecast, not a measurement

On 25 June 2025 Gartner predicted that over 40% of agentic AI projects would be cancelled by end of 2027, citing escalating costs, unclear business value and inadequate risk controls. Note the tense. Anyone quoting it as an observed rate is quoting it wrong. The useful part is the three causes: cost, ambiguity, risk. None of them is the technology failing to work. All three are scope problems.

2 Defining "Working": Writing an Acceptance Criterion

An automation that is right 80% of the time is either a triumph or a total write-off depending on the task. If nobody decides which up front, you reach the demo and discover you and the client had different projects in your heads the whole time.

THE TWO CATEGORIES, AND THE REVIEW THRESHOLD EACH IMPLIES

80% IS GENUINELY FINE

Sorting inbound email into three buckets. A message lands in the wrong folder, someone moves it, life continues.

Review threshold that follows: spot-check only. Sample a percentage weekly, no per-item review queue, no reviewer rota. The cost of an error is one drag-and-drop.

80% IS WORTHLESS

Extracting payment amounts from invoices. One wrong number in every five, in accounting. Now somebody checks all of them, which is the job you were hired to remove.

Review threshold that follows: 100% human review of anything below the confidence cutoff, and the automation must flag rather than guess. Otherwise the deliverable is negative value.

The test that sorts the two

Ask: if the system is wrong, is the error *detectable* and is it *cheap to fix*? Both yes means a percentage target is enough. If being wrong is both unacceptable and undetectable, say so on the call, because sometimes the right answer is that this task shouldn't be fully automated at all.

EXAMPLE ACCEPTANCE LANGUAGE YOU CAN LIFT

```
ACCEPTANCE CRITERION (build phase)

Measured against a sample of 200 real documents supplied by the
client during discovery, the system correctly extracts the six
named fields (invoice number, invoice date, supplier name, net
amount, VAT amount, currency) in at least 95% of cases, and
flags rather than guesses where model confidence is below the
agreed threshold.

COUNTING METHOD: a document counts as correct only when all six
named fields are correct. A flagged document is excluded from
the accuracy denominator and counted separately; the flag rate
must not exceed 15% of the sample.

The same 200-document sample is retained and re-run quarterly
as the maintenance accuracy check (see maintenance section).
```

WHY EACH CLAUSE IS THERE

- [ ] **The sample is real client data, not your examples**
  Your estimate came from the clean files they were proud of. The criterion has to be measured against what actually arrives.
- [ ] **The fields are named individually**
  "It should also get the PO number" becomes a change order instead of an assumption you absorb.
- [ ] **The flagging clause makes "I don't know" correct behaviour**
  Without it, an honest abstention counts as a miss and you are incentivised to build something that guesses confidently.
- [ ] **The counting method has a denominator**
  One wrong field out of six: is that document 0% or 83%? Write it down or you are having that argument at delivery.
- [ ] **The flag rate is capped**
  Otherwise a system that flags 90% of documents technically passes the accuracy bar while automating nothing.
- [ ] **The sample is retained for later**
  The criterion you negotiate at the start is also the maintenance tool that turns prompt drift into a number instead of a feeling.

Money-losing trap

"High accuracy" and "works reliably" are not acceptance criteria. They are outcomes, and outcomes have no boundary. A percentage with no sample, no named fields and no counting method is the same thing wearing a number.

3 Data Discovery: Ask Before You Quote

> *"I have never once opened a client's data and thought 'oh, this is cleaner than described.'"*
> **Kemal, Wunderlandmedia**

That isn't the client lying. Nobody has looked at all of their data at once in years, and the person describing it to you is describing the tidy part they touch.

THE DISCOVERY QUESTIONNAIRE — RUN THIS BEFORE A BUILD PRICE EXISTS

- [ ] **Where does this data physically live today?**
  Name every system, not the one that comes to mind first. Ask twice.
- [ ] **Has it ever been migrated, and did the migration finish?**
  "We moved CRMs in 2023" usually means two exports and a partial overlap.
- [ ] **Does anyone keep a private spreadsheet alongside the official system?**
  Ask the person who does the work, not the person who bought the software.
- [ ] **How much of it is scanned or photographed rather than digital?**
  A photo of paper is a different project from a generated PDF.
- [ ] **How many date formats, languages and currencies are in there?**
  Get an actual count from a real export, not an estimate.
- [ ] **Are there free-text notes fields, and what has ended up in them?**
  Phone numbers, addresses and half a contract, typically.
- [ ] **Can you give me a real export today, unredacted or consistently redacted?**
  The answer to this question tells you more than the answers to all the others.
- [ ] **Who owns this data internally, and who has to approve us touching it?**
  A name and a role, not a department.
- [ ] **What volume arrives per week, and how spiky is it?**
  A quarterly batch is a different architecture from a steady trickle.
- [ ] **What happens to the data today, manually, step by step?**
  If they can't describe the manual process, you cannot automate it yet.

RED FLAGS IN THE ANSWERS

WHAT YOU GET TOLD	WHAT THAT USUALLY MEANS
"It's all in the CRM, it's pretty standardised." "We can send you a few examples." "It's basically the same format every time." "IT can pull an export at some point."	Exports from two CRMs, a private spreadsheet, dates in three formats, company names with and without GmbH, scanned PDFs that are photographs of paper, and 200 records where somebody put the phone number in the notes field.

A bigger buffer is not the fix

Padding the estimate does not solve this, because messiness has a long tail and your padding is a guess against an unknown distribution. The fix is a separate, paid, fixed-scope discovery phase before you quote the build.

WHY DISCOVERY GETS PRICED SEPARATELY

- [ ] **One to two weeks, fixed price, documented deliverable**
  You get a real export, profile it, and deliver a short document: what's in there, what share is usable as-is, what cleanup would cost, and the realistic accuracy ceiling on this input.
- [ ] **The client pays for it and owns the deliverable**
  If they walk afterwards, they got value and you got paid. That is what makes it easy to sell.
- [ ] **Clause one: the build quote is valid only against the profiled data set**
  Described precisely enough to identify it later: the export, the date it was taken, the record count.
- [ ] **Clause two: data cleanup is excluded unless quoted separately**
  Otherwise you absorb the cleanup, silently, at midnight, for free.

4 The Integration Surface: An API Checklist

"Just connect it to our system." *Just* is carrying a load-bearing amount of weight in that sentence. Work through this list before the word "integration" appears in a quote.

DOES IT EXIST, AND IN WHAT FORM

- [ ] **Is there an API at all, and have you seen its base URL?**
  "Our vendor said it's possible" is not an API. Some systems end at a nightly CSV drop, which is a valid answer but a different build.
- [ ] **Is it public documentation you can read right now?**
  If the docs are behind a login you don't have, you are quoting blind.
- [ ] **Which named objects and which operations does the project touch?**
  Read and write are different permissions and often different pricing tiers.
- [ ] **Is the version you need current, deprecated or beta?**
  Building against a deprecated version puts a migration inside your warranty period.

LIMITS, TIERS AND AGREEMENTS

- [ ] **Are the rate limits published, and do they cover your volume?**
  Undocumented rate limits are discovered in production, which is the most expensive place to discover them.
- [ ] **Is API access gated behind a pricing tier the client is not on?**
  Find out who pays for the upgrade before it becomes an awkward conversation at week six.
- [ ] **Does it require a partner agreement or vendor approval?**
  Partner programmes have application timelines measured in weeks and sometimes reject you.
- [ ] **Does the vendor charge for an integration consultation?**
  On-premise systems in particular. Four figures is normal. It is the client's line item, not yours.

CREDENTIALS, ENVIRONMENTS AND THE HUMAN PATH

- [ ] **Who owns the credentials, and who can actually issue them?**
  A named person with a named date in the contract. Not "IT".
- [ ] **Is there a security review between the request and the key?**
  This is the single biggest delivery-date killer on the list, and it is not technical. Six weeks is common.
- [ ] **What is the sandbox story?**
  Is there a test environment, does it hold representative data, and does it behave like production? Building against production because there is no sandbox is a risk the client should be told about.
- [ ] **Who is on the hook for third-party fees and licence tiers?**
  Write it into the contract as the client's responsibility.
- [ ] **What happens if the integration turns out to be impossible?**
  Discovery is where a rescope should still be cheap. Say so in writing while it still is.

SCOPE WORDING

WEAK

"The system integrates with the client's CRM."

STRONG

"Integration with HubSpot via the v3 CRM API, using credentials supplied by the client by [date], covering the Contacts and Deals objects, read and create only. Credentials, licence tiers and third-party fees are the client's responsibility. If access is provided late, the delivery timeline shifts day for day."

> **The day-for-day clause is not a punishment**
> It is the thing that stops you eating a delay you had no power to prevent. Present it that way on the call and almost nobody objects to it.

5 Edge Cases and Change Control

The word "obviously" is the tell. The client considers the case so normal that it never occurred to them to mention it, which is also why they genuinely feel they aren't asking for anything new.

THE ONES THAT SHOW UP ALMOST EVERY TIME

- [ ] **Multi-currency documents**
  Changes field extraction, validation and anything downstream that sums.
- [ ] **A second language**
  Often arrives after the pilot, from one supplier, as an afterthought.
- [ ] **The supplier who sends everything as a phone photo**
  A different pipeline, not a harder version of the same one.
- [ ] **Credit notes**
  They look like invoices with the sign flipped, which is exactly why they get silently mishandled.
- [ ] **The quarterly batch as one 400-page PDF instead of 400 files**
  Splitting and boundary detection is its own piece of work.

> **You are not trying to enumerate them all**
> You can't, and that isn't the goal. The goal is a boundary, so that everything outside it is *visibly* outside it. Which means your scope describes the input universe rather than the output.

SCOPE WORDING THAT HOLDS

WEAK — DESCRIBES THE OUTPUT

"The system processes the client's incoming invoices and enters them into the accounting system."

Everything the client has ever received is now in scope, because nothing was excluded.

STRONG — DESCRIBES THE INPUT UNIVERSE

"In scope: single-page, digitally generated PDF invoices, in English or German, from the supplier list provided during discovery, one invoice per file. Any input not matching this description routes to the exception queue rather than being processed."

WHY THE EXCEPTION QUEUE CLAUSE IS THE IMPORTANT ONE

- [ ] **It converts an unhandled case from a failure into specified behaviour**
  The system did what the contract says it does. That is a very different conversation.
- [ ] **It makes extending coverage a priced change request against a written baseline**
  The baseline has to be specific enough to be boring. Specific and boring is what survives a renegotiation.
- [ ] **It gives the client a visible place where the unhandled work goes**
  Nothing disappears silently, which is what actually erodes trust.

BUDGET AN EDGE-CASE ALLOWANCE INTO THE BUILD

- [ ] **Fixed number of days for cases found during the pilot, priced in**
  The client sees a real number up front instead of a surprise later.
- [ ] **State clearly what happens when the allowance is used up**
  Further cases become change requests. Nobody feels ambushed because they were told.
- [ ] **Track the allowance visibly during the pilot**
  Report days consumed alongside progress, so the ceiling isn't a shock in the final week.
- [ ] **Define what a change request costs before you need one**
  Agreeing the mechanism while everyone is happy is much cheaper than agreeing it mid-argument.

> **Money-losing trap**
> Accepting the first two "obviously it should also" requests for free to keep the relationship warm. You have just established that the boundary is negotiable, and every subsequent request is now framed against that precedent.

6 Human-in-the-Loop and the Maintenance Tail

The extraction is the easy week

The extraction is a prompt and some plumbing. The human-in-the-loop design is usually the majority of the actual work. Leave it out of the scope and you will build it anyway in the last two weeks, unpaid, because without it the automation doesn't function.

REVIEWER ROLES AND ESCALATION — EACH ONE IS A LINE ITEM

- [ ] **How does a low-confidence result get surfaced?**
  Push, pull or queue. Decide, because it determines the whole interface.
- [ ] **Name the interface explicitly**
  A Slack approval flow, an admin screen or an email digest. Each is a different amount of work; "we'll figure it out" is the expensive one.
- [ ] **Name the reviewing role, then get the client to name the actual person**
  A role with no name attached is a role nobody has been told about yet.
- [ ] **How does a correction get back into the system of record?**
  Frequently the hardest single piece, and always underestimated.
- [ ] **Define what "resolved" means, and what happens when it isn't**
  An item sitting in the queue for nine days needs a defined destination.
- [ ] **Who gets notified when the queue backs up, and at what threshold?**
  A number and a recipient, written down.
- [ ] **What happens when the reviewer is on holiday?**
  Name the deputy in the scope. This is a client-side dependency, so it belongs in the contract, not in your head.

Say the headcount part out loud

Most clients still expect it to go to zero. Realistically you are moving someone from doing all of it to checking the 10% the system flagged. That is a genuine efficiency gain worth paying for. It is just not the demo they saw, and the gap is better closed on the sales call than at handover.

THE MAINTENANCE TAIL: FOUR THINGS THAT MOVE UNDERNEATH YOU

- [ ] **Model deprecation**
  Anthropic commits to at least 60 days notice before retiring a publicly released model, which is the good end of the industry. Sixty days is courteous and also not long if the client must schedule a change window, revalidate accuracy and get sign-off.
- [ ] **Token prices change**
  If you quoted a per-document running cost, say what happens when the underlying price moves.
- [ ] **Rate limits and API terms change**
  On the model provider's side and the integration's side, independently.
- [ ] **The client's own process changes — the sneakiest one**
  They add a field to a form, or a new supplier joins with a different format, and the automation starts being quietly wrong in a way nobody notices for a month.

Prompt drift, demystified

The prompt does not degrade. The inputs shift, or the model underneath changes, and the same prompt produces different output on the same data. Which is why the accuracy criterion from page 3 is also your maintenance tool: re-run the same 200-document sample quarterly and you have a number instead of a feeling.

WHO OWNS THIS ON DAY 91

OPTION A — RETAINER

Covers model migrations, monitoring, a quarterly accuracy check against the retained sample, and a set number of support hours. Priced monthly, scoped in writing.

OPTION B — CLEAN HANDOVER

The client takes ownership and you document exactly what they have taken on: which model is pinned, which credentials expire, what the quarterly check is, who to call. Legitimate. Just has to be written down.

There is no Option C

Leaving day 91 unspecified means "we thought you were handling that" becomes the theme of month four, usually unpaid, usually during another project's crunch week.

7 Pricing Structures Compared

> *"Fixed price is not the enemy. Fixed price on something you haven't measured is."*
> **Kemal, Wunderlandmedia**

Your effort scales with the messiness of data you haven't seen yet, and messiness has a long tail. Your estimate is based on the clean examples the client sent, because those are the ones they were proud of. Your delivery is against everything.

Structure	Where it fits	Where it fails	What makes it safe
Fixed price, no discovery **Avoid**	Only when you have already profiled this exact data set and the integration is one you have shipped before.	"Automate our invoice processing" with no accuracy criterion, no data profile and no named integration. That is an outcome, not a scope, and outcomes have no boundary.	Nothing. It is writing the client a free option on your time.
Discovery first, then fixed price **Default**	Almost every AI automation project. Phase one: paid, fixed-price, one to two weeks, documented deliverable. Phase two: fixed-price build quoted afterwards.	Rarely. The main friction is a client who wants a single number today and reads a two-phase quote as evasion.	Phase one is small enough to approve without a committee, and the build quote is valid only against the data profiled in phase one.
Phased delivery with a go/no-go **Strong**	Multiple document types or departments. Pilot one type at a fixed price, measured against the accuracy criterion, then a separately priced extension phase.	When the phases are not genuinely separable and phase one cannot deliver value alone.	Say in the proposal that the client can stop after the pilot. It reads as confidence and removes the pressure to cram everything into phase one.
Time and materials, capped **Situational**	Genuinely exploratory work where neither side can describe the finished thing. Hourly, not-to-exceed ceiling, mandatory review at 50% of budget.	Harder to sell to anyone who wants a number, and to a procurement process that requires one.	The 50% checkpoint where either side can walk. Without it the cap is just a slower way to lose the same money.

The honest line that closes the "just give me a number" conversation

The number they want requires knowing things neither of you knows yet. So you can either find them out in a small paid phase, or price the uncertainty into a much bigger fixed one. Put both options in front of them. Most people pick the small phase.

TWO MORE THINGS TO PRICE FOR

- [ ] **Running costs, stated separately from build cost**
  Tokens, hosting and any third-party licence. Client-paid, on client accounts where possible, so a price change is not your problem to absorb.
- [ ] **The maintenance decision, priced at proposal time**
  Retainer or documented handover. Deciding it later means deciding it for free.

Watch for the automation that is really a cron job

Be suspicious of work so thin it is a scheduled script with a model call in it. Those get sold at automation prices and delivered as expensive cron jobs, and the disappointment lands on the invoice.

8 The Scoping Questionnaire

Six questions answered in writing before a build price exists. Print this page, fill it in on the call, and don't quote until every line has something on it.

#	Question	What a good answer contains	Answer
1	What counts as working, and how is it measured?	A percentage, a named sample of real client data, the named fields, a counting method, a flag rate cap.	
2	What data was profiled, and by whom?	A real export, identified by date and record count, profiled in a paid discovery phase. Cleanup excluded unless quoted.	
3	What exactly does it connect to, and who supplies access?	Named system, named API version, named objects and operations, named person, named date, sandbox answer.	
4	What is inside the input boundary?	The input universe described, not the output. Everything outside routes to the exception queue. Edge-case allowance in days.	
5	Who reviews the output, and where?	Named interface, named role, named person, named deputy, definition of resolved, queue-backup threshold and recipient.	
6	Who owns it after handover?	Retainer or documented handover. Quarterly accuracy re-run against the retained sample. A name on day 91.	

BEFORE THE QUOTE GOES OUT

- [ ] **Every one of the six lines above has a written answer**
  Not a verbal one. Not "we discussed it."
- [ ] **The pricing structure matches how much you actually know**
  Discovery-first unless you have profiled this data set already.
- [ ] **Exclusions are listed explicitly, not implied**
  Data cleanup, third-party fees, licence tiers, coverage beyond the input boundary.
- [ ] **The day-for-day delay clause is in there**
  For credentials and any client-side dependency with a named date.
- [ ] **The change-request mechanism is agreed while everyone is happy**
  Much cheaper than agreeing it mid-argument.
- [ ] **You have said the headcount sentence out loud**
  From doing all of it to checking the 10% that got flagged.

> *"It isn't clever. It's just that each one, left unanswered, is worth somewhere between a week and the entire margin. What the six questions change is the category of surprise: from something you absorb to something you rescope."*
>
> **Kemal, Wunderlandmedia**

The one thing to take away

The AI part is usually the easiest week of the whole project. Projects go 40 hours over because nobody wrote down what "reads these" means, whose data it is, what happens when it's wrong, and who's on the hook when the model it runs on gets retired eleven months from now.

Next

Read the full article: **wunderlandmedia.com/scoping-ai-automation-projects-without-losing-money**
Want a second opinion on a scope before you sign it, or would you rather run discovery properly than guess? That's the kind of thing I do — **wunderlandmedia.com**