

INVESTOR HANDBOOK • OCTOBER 2026 EDITION

The Investor's Guide to AI Technical Due Diligence

A practical handbook for assessing an AI company before you invest: what to inspect, what evidence to request, which red flags matter, and the questions to put to founders.

CONTENTS

What this guide covers

BEFORE YOU START

How to use this guide	3
Executive summary	4
Why AI diligence is different	5
The diligence process	6
Data room request list	7

THE TEN AREAS

01	Architecture	8
02	Model dependency	10
03	Proprietary data	12
04	Defensibility	14
05	Evaluation	16
06	Security	18
07	Scalability and reliability	20
08	Inference economics	22
09	Technical team	24
10	IP, legal and regulation	26

TOOLS

Recommended frameworks	28
The scorecard	29
Red-flag checklist	32
Questions to ask founders	33
The decision and the first 100 days	35
Glossary	36
Sources	37

BEFORE YOU START

How to use this guide

AI technical due diligence answers one question: does the technology support the investment case, at the cost, risk and speed the deal model assumes? This guide gives you a structured way to answer it.

Who it is for

Partners, principals and analysts at venture capital, growth equity and private equity firms; accelerator and studio teams; and corporate development teams buying or investing in AI companies. You do not need to be an engineer. Each chapter tells you what to ask for, what good looks like, and when to bring in a technical specialist.

How it is organised

The core of the guide is ten areas of diligence. Each chapter follows the same structure, so you can use it as a working checklist:

SECTION	WHAT IT GIVES YOU
Direct answer	The question this area must answer, in one or two sentences.
What good looks like	The evidence a well-run company can usually produce.
Red flags	Signals that warrant deeper work, a condition, or a price change.
Evidence to request	Specific documents, data and demonstrations to add to the data room.
Tests to run	Checks your team or adviser can perform rather than accept on trust.
Questions for founders	Open questions, with an example of a strong and a weak answer.

Scoring

Score each area from 1 to 4, consistent with the [Kesho AI technical due diligence scorecard](#):

1 · UNPROVEN

Claims without evidence.

2 · FRAGILE

Works, but depends on people, luck or conditions that may not hold.

3 · CREDIBLE

Evidenced and repeatable, with known gaps.

4 · DEFENSIBLE

Evidenced, repeatable and hard for others to copy.

Record your confidence in the evidence alongside each score. Do not average away a critical failure: missing data rights, a serious security exposure or unsupported AI claims can be deal-level issues even when the overall score looks healthy.

Calibrate to stage

A seed company will not have an ISO certificate or a mature evaluation platform. That is normal. What matters is whether the team knows where its gaps are, and whether the gaps are consistent with the amount being invested and the plan for the money. The [technical team](#) chapter includes a stage calibration table.

EXECUTIVE SUMMARY

A strong demo proves a workflow can work once

Diligence must establish whether it works repeatedly, within stated limits, under production conditions, at a cost the business can sustain, on data the company has the right to use, and in a way competitors cannot easily copy.

AREA	THE QUESTION TO ANSWER
Architecture	Is this a controlled system, or a demo held together by prompts and people?
Model dependency	Could the company survive a provider changing price, behaviour, terms or availability?
Proprietary data	Does the company have the rights to its data, and does that data measurably improve the product?
Defensibility	What stays hard to copy if a competitor had the same model tomorrow?
Evaluation	Can the company prove, with a test set and a baseline, that the product works and is not getting worse?
Security	Has the company addressed AI-specific attacks such as prompt injection and excessive agent permissions?
Scalability	Will quality, latency and margin hold at the volume in the plan?
Inference economics	What does one successful task cost, and how does margin behave under stress?
Technical team	Can this team keep the product working and improving, without depending on one person?
IP and legal	Does the company own what it sells, and do its claims and use cases comply with the law?

FIVE QUESTIONS THAT MATTER MOST

1. Show me the test set you use to decide whether a release is better or worse.
2. What does one successful task cost you today, including human review?
3. What happens if your main model provider changes price or behaviour next quarter?
4. Which data do you have the contractual right to use for training and improvement?
5. What would a competitor with the same model still be unable to copy?

TYPICAL DEAL-LEVEL ISSUES

- AI capability claims that the product or evidence does not support.
- No rights to the data the product depends on.
- No evaluation evidence beyond demos and testimonials.
- Gross margin that turns negative at planned volume, with no credible path back.
- Agents with broad permissions and no controls on what they can do.
- Core code, models or data not assigned to the company.

CONTEXT

Why AI diligence is different

Standard technical diligence assumes software behaves the same way every time, costs little to run per user, and is protected by the code itself. AI products break all three assumptions.

\$285.9B

US private AI investment in 2025, across 1,953 newly funded AI companies. Stanford AI Index 2026.

88%

of organisations report using AI, up from 78% a year earlier. Stanford AI Index 2026 and 2025.

362

documented AI incidents, up from 233 the year before. Stanford AI Index 2026.

>280×

fall in the cost of GPT-3.5-level inference, November 2022 to October 2024. Stanford AI Index 2025.

5.4%

score gap between the top and tenth-ranked model, down from 11.9%. The top two were 0.7% apart. Stanford AI Index 2025.

1 in 3

Agents rose from 12% to about 66% task success on OSWorld, yet still fail roughly one attempt in three. Stanford AI Index 2026.

Behaviour is probabilistic

The same input can produce different outputs, and quality can change when a provider updates a model, even if the company changes nothing. Stanford calls this unevenness the *jagged frontier*: a model can win a gold medal at the International Mathematical Olympiad yet read an analogue clock correctly only about half the time. Only testing on the company's own tasks tells you which side of the frontier its product sits on.

Costs scale with usage

Every request incurs model, retrieval and often human review costs. Gross margin depends on how customers use the product, not just on how many there are.

Models are converging

With leading models this close, access to a good model is rarely a moat on its own. Defensibility moves to data, workflow, distribution and trust.

Regulators are watching AI claims

In March 2024 the US SEC settled with two investment advisers, Delphia and Global Predictions, for a combined \$400,000 over false and misleading statements about their use of AI. The SEC warned that public issuers must also be careful about claims of AI adoption. In September 2024 the FTC's *Operation AI Comply* included DoNotPay, which marketed a "robot lawyer" without testing whether its output matched a human lawyer's. It agreed to pay \$193,000. As the FTC Chair put it: "there is no AI exemption from the laws on the books."

Implication for investors. Treat every AI claim in a pitch deck as a hypothesis to be tested against the product, the code and the evaluation data. Overstated AI is now a legal risk, not only a valuation risk.

PROCESS

The diligence process

Run AI technical diligence in five phases. Go deep only where the investment thesis depends on the answer. A seed review may take days; a growth or buyout review may take several weeks.

PHASE	PURPOSE	MAIN ACTIVITIES	OUTPUT
1. Screen	Decide whether AI is core to the thesis and whether claims are plausible.	Desk review of deck, website and product. One founder call. Run the AI-washing test on page 30.	Go or no-go for deeper work; scope.
2. Request	Get the evidence before the meetings, not during them.	Issue the data room request list. Agree access to code, logs and evaluation data under NDA.	Populated data room.
3. Inspect	Understand how the system really works.	Architecture walkthrough. Code and infrastructure review. Interviews with the CTO, an engineer, the evaluation owner and a customer-facing lead.	Findings by area.
4. Test	Verify rather than accept.	Re-run evaluations. Review sampled outputs with a domain expert. Rebuild the cost model. Review security against OWASP and MITRE ATLAS.	Evidence-backed scores.
5. Decide	Turn findings into a decision and a plan.	Complete the scorecard. Agree conditions, price implications and post-investment priorities.	Decision, conditions, 100-day plan.

Who should be involved

- **Deal lead** owns the thesis and the questions that matter to it.
- **Technical adviser** with hands-on AI engineering experience reviews code, architecture, evaluation and security.
- **Domain expert** judges whether outputs are actually correct for the customer's work.
- **Legal counsel** reviews data rights, IP assignment, licences and regulatory exposure.
- **Finance** connects the cost model to the deal model.

Principles that save time

- Ask for evidence that already exists. A company that must create it for the data room is telling you something.
- Prefer live demonstrations on your own examples over prepared demos.
- Interview beyond the founders. Engineers describe how things really work.
- Separate normal early-stage trade-offs from issues that change value.
- Write findings as decisions: what it means for price, terms or the plan.

DATA ROOM

Data room request list

Send this list early. Mark items as required or optional depending on stage. Missing items are findings in their own right.

Product and architecture

- ☐ Current architecture diagram, including model calls, retrieval, tools, data stores and human review points
- ☐ Model inventory: every model, version, provider and purpose
- ☐ Prompt and configuration registry with change history
- ☐ Dependency manifest or software bill of materials
- ☐ Incident log and postmortems for the last 12 months
- ☐ Product roadmap with technical dependencies

Models and providers

- ☐ Contracts and terms with model and cloud providers
- ☐ Monthly usage and spend by provider and model
- ☐ Rate limit and quota agreements
- ☐ Evaluation results comparing alternative models

Data

- ☐ Data inventory: sources, legal basis, permitted uses, retention
- ☐ Data licences and customer contract templates (data clauses)
- ☐ Data protection impact assessments where required
- ☐ Lineage from datasets to model versions
- ☐ Labelling process and vendor contracts

Evaluation

- ☐ Evaluation methodology, test sets and metric definitions
- ☐ Results over time, by release and by customer segment
- ☐ Release gate criteria and recent pass/fail records
- ☐ Production monitoring dashboards and human review samples

Security and governance

- ☐ Security policies, recent penetration test and remediation status
- ☐ AI threat model covering prompt injection, data leakage and agent permissions
- ☐ Access control model for tools and data available to agents
- ☐ Vulnerability disclosure policy and AI incident response plan
- ☐ AI governance policy, risk register and any certifications (ISO/IEC 27001, ISO/IEC 42001, SOC 2) with scope
- ☐ Customer security questionnaires completed in the last year

Economics

- ☐ Provider invoices for 12 months, and any credits with expiry dates
- ☐ Cost per task or per customer, by feature where available
- ☐ Usage distribution: median and heaviest customers
- ☐ Pricing model and gross margin by segment
- ☐ Human review and implementation effort per customer

Team

- ☐ Organisation chart with tenure and roles
- ☐ Ownership map of critical systems
- ☐ Attrition for the last 24 months
- ☐ Contractors and outsourced development
- ☐ Hiring plan linked to the roadmap

Legal and IP

- ☐ IP assignment agreements for founders, staff and contractors
- ☐ Open-source and open-weight model licence register
- ☐ Material customer contracts: AI, liability and data terms
- ☐ Regulatory assessment, including EU AI Act risk classification
- ☐ Public and investor-facing AI claims, with supporting evidence

01

Architecture

Architecture diligence asks whether the product is a system the company controls, or a demo held together by prompts and people. Map every component between the user and the model, and find where quality, cost and failure are decided.

Why it matters

In most AI products the model is only one component. Retrieval, orchestration, tool use, guardrails, evaluation, monitoring and human review decide whether the output is reliable. These components are also where most of the company's own engineering, and therefore most of its defensible work, usually lives.

The UK Code of Practice for the Cyber Security of AI expects developers to keep an audit trail of models, datasets and prompts, and an audit log of changes to system prompts and model configuration. Those are reasonable expectations of any company asking for institutional capital.

WHAT GOOD LOOKS LIKE

- An architecture diagram that matches production, including model calls, retrieval, tools, data stores and human review points.
- Model access sits behind an internal layer, so models can be swapped without rewriting the product.
- Prompts, model versions and configuration are versioned, reviewed and deployed like code.
- Every model call is logged with input, output, latency, cost, model version and outcome.
- Deterministic code handles what should be deterministic: calculations, permissions, routing and pricing.
- Separate environments, with customer data kept out of development.

RED FLAGS

- The diagram was drawn for the data room, and nobody can trace a real request through it.
- Prompts are edited directly in production with no review or history.
- Permissions, pricing or compliance decisions are left to the model.
- No logging of model inputs and outputs, or logs with no retention policy.
- Each large customer runs on its own custom fork.
- Manual steps by staff are hidden inside what is sold as automation.

EVIDENCE TO REQUEST

- Architecture diagram and a walkthrough of one live request, end to end.
- Read-only repository access or a guided code review.
- Dependency manifest or software bill of materials.
- Prompt and configuration registry with change history.
- Model inventory: model, version, provider, purpose.
- Incident log and postmortems for the last 12 months.
- Infrastructure-as-code and environment separation.

TESTS TO RUN

- Trace three real customer requests from input to output, including logs and cost.
- Ask an engineer, not the CTO, to change a prompt and ship it. Observe the review, test and rollback steps.
- Ask how long it takes to roll back a model version, then ask when it last happened.
- Compare the number of manual steps in delivery with what the deck describes.

Questions for founders

- Walk me through what happens, component by component, when a customer submits a request.
- Which parts are decided by deterministic code and which by the model? Why did you draw the line there?
- How are prompts and model configurations versioned, reviewed and rolled back?
- Which parts of the system would you rebuild if you had six months and no customer commitments?
- What happened in your last significant production incident, and what changed afterwards?
- Where do people step in today, and what share of requests need them?

Question: "What would break if your model provider changed its model tomorrow?"

STRONG ANSWER

"We pin versions. New versions run against our regression suite in staging before we switch. Last quarter we moved to a new version in two weeks; here is the evaluation report."

WEAK ANSWER

"Nothing. Models are a commodity." No version pinning, no regression suite, no example of a past change.

Specialist view

Architecture review is where a technical adviser adds the most value per hour. A short code and infrastructure review often reveals whether the company is building a product or delivering bespoke services, and whether the roadmap is realistic with the current codebase.

Related: [AI production release gate](#) · [Software value and AI clones](#)

02

Model dependency

Most AI companies build on models they do not own. That is normal. The diligence question is whether the company can survive its provider changing price, behaviour, terms or availability, and whether it could move if it had to.

Why it matters

Leading models are converging. The Stanford AI Index 2025 found the gap between the top and tenth-ranked model fell from 11.9% to 5.4%, and the gap between leading open-weight and closed models narrowed from 8% to 1.7% on some benchmarks. That cuts both ways. Switching is more feasible, but a company can no longer claim that its choice of model is a moat.

Dependency also runs deeper than the API. The Stanford AI Index 2026 notes that one foundry, TSMC, fabricates almost every leading AI chip. Few startups can mitigate that risk, but investors should know which layers of the supply chain the company controls and which it does not. Providers may also release features that overlap with the products built on them.

WHAT GOOD LOOKS LIKE

- A dependency map: every model and provider, what it does, and the fallback.
- At least one tested alternative for critical paths, with comparative evaluation results.
- Version pinning, and evaluations re-run before adopting a new release, as the UK AI cyber security code expects.
- Provider terms reviewed for training use of customer data, retention, rate limits, service levels, price-change notice and termination.
- A clear view of what the provider could ship that would overlap with the product.

RED FLAGS

- One provider, no fallback and no evaluation of alternatives.
- The product's value is the base model's capability plus a thin prompt.
- Customer data sent to providers on consumer terms or without a data processing agreement.
- Fine-tuned models that can only run on one provider's platform.
- Rate limits or quotas that cap growth and have not been negotiated.

Dependency types and what to check

APPROACH	CONTROL	SWITCHING COST	DILIGENCE FOCUS
Closed model via API	Low	Low to medium	Provider terms, version changes, rate limits, data use, price exposure and fallback.
Fine-tuned on a provider platform	Medium	Medium to high	Can the tuned model or training data be exported? Is the gain over the base model still measured?
Open-weight, self-hosted	High	Medium	Licence terms, hosting cost, GPU availability, security patching and in-house skills.
Own trained model	Highest	High	Training data rights, compute cost, evidence of an advantage over off-the-shelf models, and key-person risk.
Multiple models with routing	Medium to high	Low	Routing logic, evaluation per route, and whether the added complexity is justified.

EVIDENCE TO REQUEST

- Provider contracts and terms of service in force.
- Monthly usage and spend by model for 12 months.
- Evaluation comparisons against at least one alternative.
- Records of past model migrations and their impact.
- Rate limit, quota and capacity agreements.

TESTS TO RUN

- Ask the team to run the evaluation suite against an alternative model during diligence.
- Model the margin impact of a 50% provider price rise.
- Check what the provider's terms permit for customer data.

Questions for founders

- If your main provider doubled prices or retired your model next quarter, what would you do, and how long would it take?
- Have you run your evaluation suite on another model? Show me the result.
- What do your provider terms say about using your customers' data for training?
- Which features on your provider's public roadmap overlap with your product?
- What share of your cost of goods goes to a single provider?

Question: "Why would a customer not just use the model provider's own product?"

STRONG ANSWER

"Because we sit inside their system of record, carry their approvals and audit trail, and our evaluations show a measurable gain on their tasks. Here is the comparison."

WEAK ANSWER

"Our prompts are better." Or: "They would never build for our niche."

03

Proprietary data

Proprietary data is the most commonly claimed AI moat and the least often verified. Test three things: rights, meaning the company may lawfully use the data for this purpose; contribution, meaning the data measurably improves the product; and exclusivity, meaning competitors cannot get the same thing.

Why it matters

Data claims often drive valuation, but rights failures can force a company to stop using data, retrain models or renegotiate with customers. The law on training AI with copyrighted material remains unsettled. The US Copyright Office released a pre-publication version of its report on generative AI training in May 2025. Personal data brings data protection obligations wherever it is used.

The UK AI cyber security code expects developers to document the sources of training data, including fine-tuning data and human feedback. For public data it expects at least the source URL and the date obtained, so poisoned data can be traced. Few early companies do this. Those that do are easier to underwrite.

WHAT GOOD LOOKS LIKE

- A data inventory: each source, how it was obtained, the legal or contractual basis, permitted uses and retention.
- Customer contracts that explicitly allow the intended use, such as product improvement or cross-customer learning.
- Lineage from each model or index version back to the data behind it.
- An ablation showing performance with and without the proprietary data.
- A working feedback loop: usage produces labelled data that measurably improves quality.

RED FLAGS

- "We have millions of records," with no rights analysis.
- Scraped data with no record of source or date.
- Customer contracts that are silent on, or prohibit, training and cross-customer use.
- Personal data used beyond its original purpose.
- The "unique" dataset is licensed non-exclusively or is publicly available.

The data value test

TEST	QUESTION	EVIDENCE
Rights	May the company use this data to train, fine-tune, ground or evaluate models, and to sell the result?	Licences, customer contracts, privacy notices, legal opinion.
Provenance	Can the company show where each dataset came from and when?	Data inventory, collection logs, lineage records.
Quality	Is the data accurate, current, labelled consistently and representative of customers?	Labelling guidelines, agreement rates, coverage analysis.
Contribution	How much worse is the product without it?	Ablation results on the evaluation set.
Exclusivity	Could a well-funded competitor obtain equivalent data within a year?	Source analysis, exclusivity terms, market alternatives.
Renewal	Does the dataset grow and improve with use, or decay?	Volume and quality trends, feedback capture rates.

EVIDENCE TO REQUEST

- Data inventory and lineage records.
- Data licences and the data clauses in customer contract templates.
- Data protection impact assessments where required.
- Labelling process, guidelines and vendor contracts.
- Ablation and data-contribution analyses.

TESTS TO RUN

- Pick five datasets from the inventory and trace each to its contract or licence.
- Read three signed customer contracts for data-use terms, not just the template.
- Ask for evaluation results with the proprietary data removed.

Questions for founders

- Which data do you have the contractual right to use for training and product improvement? Show me the clause.
- How much worse is the product without your proprietary data? How do you know?
- Could a competitor buy, license or collect equivalent data? What would it cost them?
- What would you do if a major customer asked you to delete their data from your models?
- Where did your public or scraped data come from, and do you record source and date?

Related: [AI due diligence and data lineage](#)

04

Defensibility

Ask what would remain hard to copy if a competitor had the same model tomorrow. In AI, the defensible part is rarely the model. It is usually workflow integration, proprietary feedback data, distribution, trust and switching costs.

Why it matters

Two trends weaken traditional software moats. Models are converging, so access to capability is widely available. And AI is making software cheaper to build: Stanford reports that performance on SWE-bench Verified, a coding benchmark, rose from 60% to near 100% in a single year. If code is cheaper to write, code alone protects less.

That does not make AI companies indefensible. It means the source of defensibility must be identified and evidenced, rather than assumed from the technology.

MOAT	EVIDENCE THAT IT IS REAL	HOW IT ERODES
Workflow embedding	Deep integration with systems of record; daily use by named roles; product sits in the approval path.	Incumbent platforms add similar AI features.
Feedback data	Measured quality gains from usage data that competitors cannot access.	Data is not exclusive, or gains plateau.
Distribution	Efficient acquisition, channel partnerships, strong retention and expansion.	Channels are rented, not owned.
Trust and evidence	Approved by regulated customers; audit trails, assurance and certifications competitors lack.	Standards become common and cheap to meet.
Switching costs	Configuration, historical data, integrations and trained users that are costly to replace.	Data export and migration tools reduce friction.
Technical depth	Own models or systems with proven cost, latency or quality advantages.	Falling model prices narrow the advantage.

The thin-wrapper test

A "thin wrapper" adds a user interface and prompts to a model it does not control. Thin wrappers can be good businesses for a time, but they are priced differently. Five questions separate them from defensible products:

1. If the model provider launched this feature, what share of customers would leave within a year?
2. Could a capable team rebuild the core product in three months with public models?
3. Does the product get measurably better with usage, in ways competitors cannot replicate?
4. Is the company the system of record, or a layer on top of someone else's?
5. Are customers paying for an outcome, or for access to a model?

WHAT GOOD LOOKS LIKE

- The founders can name the moat and show evidence for it.
- Retention and expansion data support the claimed switching costs.
- Quality advantages are measured against general-purpose alternatives.
- The roadmap deepens the moat rather than adding features.

RED FLAGS

- "Our prompts are proprietary" as the main defence.
- First-mover advantage claimed without retention evidence.
- No comparison against the provider's own product.
- Moat depends on a partnership that can be cancelled.

Questions for founders

- What would a competitor with the same model and \$10 million still be unable to copy?
- How does your product compare with the general-purpose assistant your customers already have?
- What happens to quality as usage grows? Show me the trend.
- Which customers have churned, and why?
- What would it cost a customer to replace you?

Related: [Software value and AI clones in enterprise due diligence](#)

05

Evaluation

Evaluation is the evidence that the product works. Ask for the test set, the metric, the baseline and the trend, not a demo. A company that cannot show how it measures quality cannot show that quality is improving, or even stable.

Why it matters

Public benchmarks measure general capability, not performance on a customer's work. The jagged frontier means a model can excel at hard tasks and fail at simple ones, and agents still fail roughly one attempt in three on structured benchmarks. Only the company's own evaluations, built from its customers' tasks, tell you where its product sits.

Evaluation is also a legal matter. The FTC's action against DoNotPay rested on the company not testing whether its "robot lawyer" performed at the level of a human lawyer before claiming that it did. Untested claims of performance are a liability.

WHAT GOOD LOOKS LIKE

- A versioned test set drawn from real customer tasks, labelled by domain experts and held out from tuning.
- Baselines: the previous release, a simpler approach and, where relevant, human performance.
- Metrics tied to the cost of errors, broken down by customer segment and task type.
- A release gate: no version ships if it regresses on agreed thresholds.
- Production monitoring with sampled human review and documented failure modes.
- Any model-as-judge scoring calibrated against human reviewers.

RED FLAGS

- Only public benchmark scores, or only demos and testimonials.
- "95% accurate" with no definition of the task, data or metric.
- The test set is the same data used to tune prompts.
- No breakdown by segment; averages hide failing groups.
- Model-as-judge scoring never checked against people.
- Nobody owns evaluation.

The evaluation evidence ladder

LEVEL	WHAT EXISTS	WHAT IT LETS YOU CONCLUDE
1 • Anecdote	Demos, case studies, customer quotes.	The product can work. Nothing about how often.
2 • Offline test	A test set and metric, run occasionally.	Performance on a sample at one point in time.
3 • Controlled release	Versioned test set, baselines, release gate and human-calibrated scoring.	Quality is measured and protected release to release.
4 • Outcome evidence	Production monitoring linked to customer outcomes and error costs.	The product delivers value in real use, and you would know if it stopped.

Seed companies often sit at level 2. A Series B company selling into regulated or high-stakes work should usually be at level 3 or above.

EVIDENCE TO REQUEST

- Evaluation methodology and metric definitions.
- Test sets, with how they were built and labelled.
- Results over time, by release and segment.
- Release gate records, including failed releases.
- Production monitoring and human review samples.

TESTS TO RUN

- Review 50 random outputs with a domain expert.
- Re-run the evaluation suite yourself, or watch it run.
- Bring ten of your own examples and test them live.
- Ask for the worst-performing segment and why.

Questions for founders

- Show me the test set you use to decide whether a release is better or worse.
- What is your baseline, and how much better are you than it?
- Where does the product perform worst, and what do you tell customers about that?
- When did a release last fail your gate, and what happened?
- How do you know quality has not drifted in production this month?

Question: "How accurate is the product?"

STRONG ANSWER

"On our 1,200-case test set from six customers, 91% of extractions are correct against expert labels, up from 86% two releases ago. It drops to 78% on handwritten documents, which we flag for review."

WEAK ANSWER

"Over 95%. Customers love it." No test set, no definition, no segment view.

Related: [AI agent evaluation framework](#) · [AI production release gate](#)

06

Security

AI products add attack surfaces that standard security reviews miss: prompts that carry instructions, retrieved documents that carry attacks, agents with permissions, and model outputs passed into other systems. Assess them against the OWASP Top 10 for LLM Applications and MITRE ATLAS, on top of normal security diligence.

Why it matters

Documented AI incidents rose from 233 to 362 in a year, according to the Stanford AI Index 2026. Security failures in AI products can leak customer data, expose confidential instructions, or let an attacker steer an agent into taking actions. For enterprise sales, security evidence is often what decides whether deals close.

Three references to use

REFERENCE	USE IN DILIGENCE
OWASP Top 10 for LLM Applications (2025)	A checklist of the most important risks in applications built on large language models. Use it to structure questions and test cases.
MITRE ATLAS	A living knowledge base of adversary tactics and techniques against AI systems, based on real attacks and red-team demonstrations. Use it to judge whether the threat model is realistic.
UK Code of Practice for the Cyber Security of AI (2025)	13 principles across design, development, deployment, maintenance and end of life. Use it as a baseline of reasonable practice.

WHAT GOOD LOOKS LIKE

- An AI-specific threat model, reviewed when the system changes.
- Agents and tools granted only the permissions they need, with human approval for consequential actions.
- Independent security testing that includes AI attacks.
- A published vulnerability disclosure policy and an AI incident plan.

RED FLAGS

- Security review ignores prompts, retrieval and agent actions.
- Agents with write access to customer systems and no approval step.
- Secrets or customer data placed in system prompts.
- No limits on usage per user or per key.

OWASP Top 10 for LLM Applications 2025: what to ask

RISK	WHAT TO ASK OR CHECK
LLM01 Prompt injection	How do you stop instructions hidden in user input, documents or web pages from changing the system's behaviour? Show test cases.
LLM02 Sensitive information disclosure	Can one customer's data appear in another's output? How are personal data and secrets filtered?
LLM03 Supply chain	Which models, datasets, plugins and libraries do you depend on, and how are they vetted and updated?
LLM04 Data and model poisoning	How do you check the integrity of training, fine-tuning and retrieval data?
LLM05 Improper output handling	Is model output validated before it reaches databases, code execution, browsers or other systems?
LLM06 Excessive agency	What can agents do without human approval? Are permissions scoped to the minimum needed?
LLM07 System prompt leakage	Would anything harmful be exposed if the system prompt leaked? Are secrets kept out of it?
LLM08 Vector and embedding weaknesses	Are retrieval indexes separated by customer and permission? Can users retrieve documents they should not see?
LLM09 Misinformation	How are incorrect or fabricated outputs detected, flagged or reviewed before they cause harm?
LLM10 Unbounded consumption	Are there limits on usage, cost and request rates per user to prevent abuse and runaway bills?

EVIDENCE TO REQUEST

- Recent penetration test, scope and remediation status.
- AI threat model and red-team results.
- Agent permission model and approval workflow.
- Vulnerability disclosure policy and incident plan.
- Certifications with their scope.

QUESTIONS FOR FOUNDERS

- What is the worst thing an attacker could make your system do?
- Who last tested your product for prompt injection, and what did they find?
- Which actions require a human to approve them?
- How would you know if customer data leaked through a model output?

Related: [AI agent risk assessment](#) · [Minimum viable AI governance pack](#)

07

Scalability and reliability

AI systems scale differently from ordinary software. Cost grows with usage, latency depends on external providers, and quality can drift without any code change. Test whether the system can handle the growth in the plan without breaking margin, latency or quality.

Why it matters

Growth plans usually assume more customers, larger customers and new features such as agents. Each puts different pressure on an AI system. Larger customers bring security reviews, single sign-on, audit logs and data residency requirements. Agents multiply the number of model calls per task. A system that works for twenty pilot customers can fail commercially at two hundred.

Watch for services disguised as software. If every new customer needs weeks of custom prompts, data mapping or manual review, growth adds people, not margin.

WHAT GOOD LOOKS LIKE

- Load testing at a multiple of the plan's peak volume.
- Known latency at the 95th percentile, with a budget per step.
- Fallbacks for provider outages and graceful degradation.
- Queues, caching and rate-limit management in place.
- Quality drift monitoring in production.
- Tenant isolation and a tested recovery to a known good state.
- Implementation effort per customer falling over time.

RED FLAGS

- Never load-tested; peak latency unknown.
- Single provider, single region, no outage plan.
- Onboarding each customer needs bespoke engineering.
- Quality degrades silently and is found by customers.
- Enterprise requirements on the roadmap but not designed for.
- Provider rate limits already reached at current volume.

Connect the growth plan to the system

PLAN ASSUMPTION	TECHNICAL QUESTION	EVIDENCE
Ten times more customers	Do cost, latency and provider quotas hold? What breaks first?	Load test results, quota agreements, cost model.
Move to enterprise	Are single sign-on, audit logs, permissions, data residency and security reviews supported?	Completed security questionnaires, enterprise feature list.
New regions	Can data and processing stay in region? Which regulations apply?	Hosting design, provider regions, regulatory assessment.
Agentic features	How many model calls per task, and what is the failure and cost profile?	Agent evaluation, cost per task, permission model.
Larger inputs	How do longer documents or histories affect cost, latency and accuracy?	Performance by input size.
New verticals	Does the product generalise, or does each vertical need new data and evaluation?	Results by segment; effort to launch the last vertical.

EVIDENCE TO REQUEST

- Load test results and capacity plan.
- Uptime, latency and error-rate history.
- Provider outage runbook and recovery tests.
- Implementation hours per customer over time.

QUESTIONS FOR FOUNDERS

- What breaks first at ten times today's volume?
- What happened during your provider's last outage?
- How long does onboarding take now, compared with a year ago?
- How do you detect quality drift before customers do?

Rule of thumb. If implementation effort per customer is flat or rising, model the business as partly services. That affects gross margin, valuation multiples and the hiring plan.

08

Inference economics

In AI companies, cost of goods rises with every request. Rebuild gross margin from the cost of one successful task, not from the provider's price list, and stress-test it against usage, provider pricing and human review.

Why it matters

Model prices have fallen fast. The Stanford AI Index 2025 found the cost of GPT-3.5-level inference fell more than 280-fold between November 2022 and October 2024, with hardware costs falling about 30% a year and energy efficiency improving about 40% a year. But falling unit prices do not guarantee falling costs per task. Newer products often use more capable models, more steps, longer inputs and agents that make many calls per task.

Two further distortions are common. Provider or cloud credits can make current margins look better than they will be once the credits expire. And blended averages can hide heavy users or segments that lose money.

```
Cost per successful task =  
(model cost per attempt + retrieval and infrastructure + tool and API fees + human review)  
÷ success rate
```

Human review is often the largest item, not tokens. A product that needs a person to check 30% of outputs has very different economics from one that needs 5%.

WHAT GOOD LOOKS LIKE

- Costs tagged by customer, feature and model.
- Margin known per segment, not only blended.
- Pricing linked to the main cost drivers.
- Usage limits on heavy users and free tiers.
- A plan to route simpler tasks to cheaper models, backed by evaluation.

RED FLAGS

- Margin calculated before credits expire.
- Flat-fee pricing with unlimited usage.
- Human review cost excluded from cost of goods.
- No view of the heaviest customers' cost.
- Plan assumes model prices keep falling to break even.

Worked example

Illustrative figures for a document-processing product priced at \$1.00 per completed task. Replace with the company's own data.

COST LINE	ASSUMPTION	PER ATTEMPT
Model calls	6 calls × \$0.012	\$0.072
Retrieval and infrastructure	Hosting, vector search, storage	\$0.010
Human review	10% of tasks × \$1.50 per review	\$0.150
Cost per attempt		\$0.232
Success rate	85% of attempts complete correctly	÷ 0.85
Cost per successful task	Gross margin on AI cost of goods: 73%	\$0.273

Stress test

SCENARIO	COST PER SUCCESSFUL TASK	GROSS MARGIN
Base case	\$0.273	73%
Provider raises prices 50%	\$0.315	68%
Heavy users need 3× the model calls	\$0.442	56%
Human review rises to 30% of tasks	\$0.626	37%
Cheaper model at 40% of cost; success falls to 80%	\$0.236	76%

In this example, the biggest threat to margin is not provider pricing but human review. That is common, and it is why evaluation and economics should be reviewed together.

EVIDENCE TO REQUEST

- Provider invoices for 12 months and any credits, with expiry dates.
- Cost per task or per customer, by feature.
- Usage distribution, including the heaviest customers.
- Human review rates and cost.

QUESTIONS FOR FOUNDERS

- What does one successful task cost you, including human review?
- What is the margin on your ten heaviest customers?
- What is your margin once current credits run out?
- How does your pricing protect you if usage per customer doubles?

09

Technical team

Assess whether the team can keep the product working, improving and safe after the money arrives, and whether that capability depends on one or two people.

Why it matters

AI products need a combination of skills that is still scarce: product engineering, machine learning, data engineering, evaluation, security and domain knowledge. In early companies, much of the critical knowledge about prompts, models and data sits in a few heads. The Stanford AI Index 2026 also reports that the number of AI researchers and developers moving to the United States has dropped sharply, a reminder that hiring plans should not assume easy access to talent.

WHAT GOOD LOOKS LIKE

- A technical leader who explains trade-offs and limits honestly, including what does not work.
- Skills that match the product: engineering, AI, data, evaluation and security covered for the stage.
- Named owners for evaluation, security and each critical system.
- Documentation and on-call arrangements that do not rely on one person.
- Domain experts involved in defining quality.
- A hiring plan tied to the roadmap and budget.

RED FLAGS

- One person holds all knowledge of models, prompts or data pipelines.
- Founders overstate capability, and engineers contradict them.
- High recent attrition in engineering.
- Core development outsourced, with unclear IP assignment.
- Nobody can explain why a known failure happened.
- Roadmap requires skills the team does not have and has no plan to hire.

Who to interview

PERSON	WHAT YOU LEARN
CTO or technical co-founder	Strategy, trade-offs, honesty about limits.
A senior engineer	How things really work, quality of practices, morale.
Evaluation or ML owner	Whether quality is measured or assumed.
Implementation or customer success lead	Hidden manual work, customer pain, onboarding effort.

Stage calibration

Use this table to judge whether gaps are normal for the company's stage. Apply it across all ten areas, not only the team.

AREA	PRE-SEED AND SEED	SERIES A	SERIES B AND LATER
Architecture	Simple, understood by founders; prompts in version control.	Model access abstracted; logging on model calls.	Documented, observable, recoverable; change control.
Evaluation	A small expert-labelled test set.	Versioned test set, baselines, release gate.	Production monitoring linked to outcomes.
Data	Rights understood; basic inventory.	Contracts permit intended use; lineage started.	Full inventory, lineage and impact assessments.
Security	Basic controls; secrets out of prompts.	AI threat model; scoped agent permissions; pen test.	Independent AI testing; incident plan; certification where customers need it.
Economics	Rough cost per task known.	Cost by customer and feature; credits understood.	Segment margins; pricing linked to cost drivers.
Team	Founders can build and explain the core.	Named owners for evaluation and security.	No single point of failure in critical systems.

Key-person test

For each critical system, ask: if this person left tomorrow, who could keep it running, and for how long? Record the answer in a simple table: system, owner, backup, documentation. Gaps become retention or hiring conditions.

Questions for founders

- If your lead ML engineer left tomorrow, what would stop working?
- What is the hardest technical problem you have solved, and what did you get wrong first?
- Which roles must you hire in the next 12 months for the roadmap to be credible?
- Who outside the founding team has written significant parts of the core code?
- Why have engineers left in the last two years?

10

IP, legal and regulation

Confirm that the company owns, or has the right to use, what it sells: code, models, data and outputs. Then check that its claims and use cases fit the laws that apply. AI adds new questions to standard IP diligence.

What AI changes

ISSUE	WHAT TO KNOW
Copyright in AI outputs	In January 2025 the US Copyright Office concluded that AI outputs can be protected only where a human author has determined sufficient expressive elements. Prompts alone are not enough, but using AI as a tool does not bar protection. Assets generated mainly by AI may not be protectable.
Training data	The legal position on training with copyrighted works is unsettled. The Copyright Office released a pre-publication version of its training report in May 2025. Check licences and exposure to claims.
Open-weight model licences	Licences differ. Some restrict certain uses, add conditions above usage thresholds, or require attribution. Confirm the licence of every model in production permits the company's commercial use.
IP assignment	Confirm assignment from founders, employees, contractors and outsourced developers, including for prompts, evaluation sets and fine-tuned models.
AI claims	The SEC and FTC have acted against false or unsupported AI claims. Compare the deck, website and contracts with the evidence.
Customer contracts	Check liability caps, warranties on accuracy, indemnities for AI output, and data-use terms.

EU AI Act: is the use case high-risk?

The EU AI Act applies to companies placing AI systems on the EU market or whose system outputs are used in the EU. Annex III lists high-risk areas, including biometrics; critical infrastructure; education and vocational training; employment and worker management; access to essential private and public services, including credit scoring and some insurance pricing; law enforcement; migration and border control; and the administration of justice and democratic processes. High-risk systems carry obligations covering risk management, data governance, documentation, human oversight, accuracy and robustness.

EU AI Act timeline

DATE	WHAT APPLIES
1 August 2024	The Act entered into force.
2 February 2025	Prohibited AI practices and AI literacy obligations.
2 August 2025	Obligations for general-purpose AI models, governance and penalties. Models already on the market before this date have until 2 August 2027.
2 August 2026	The remainder of the Act, except as amended below.
2 December 2027	High-risk obligations for Annex III systems, as amended by the Digital Omnibus.
2 August 2028	High-risk obligations for Annex I systems (AI in regulated products), as amended.

Maximum penalties (Article 99)

INFRINGEMENT	UP TO
Prohibited AI practices	€35 million or 7% of worldwide annual turnover
Most other obligations	€15 million or 3% of worldwide annual turnover
Incorrect or misleading information to authorities	€7.5 million or 1% of worldwide annual turnover

For larger undertakings the higher amount applies. For SMEs, including start-ups, the lower amount applies.

EVIDENCE TO REQUEST

- IP assignment agreements for everyone who contributed code or models.
- Open-source and model licence register.
- Material customer contracts.
- Regulatory assessment and EU AI Act classification.
- Evidence behind each public AI claim.

QUESTIONS FOR FOUNDERS

- Which of your use cases could fall under EU AI Act Annex III?
- Which models do you use under open licences, and have you checked the terms?
- What evidence supports each AI claim on your website and in this deck?
- Do your contracts warrant accuracy, and who bears the cost of AI errors?

This chapter is not legal advice. Involve counsel for jurisdiction-specific review. Related: [EU AI Act compliance roadmap](#)

TOOLS

Recommended frameworks

You do not need to adopt a framework to run diligence, but recognised frameworks make questions sharper and findings easier to defend. Use them as references, not as checklists to be completed in full.

FRAMEWORK	WHAT IT IS	HOW TO USE IT IN DILIGENCE
NIST AI Risk Management Framework 1.0 (2023)	Voluntary US framework organised around four functions: Govern, Map, Measure and Manage.	Structure governance questions. Ask how the company identifies, measures and manages AI risks, and who is accountable.
NIST AI 600-1 Generative AI Profile (2024)	A companion to the AI RMF focused on risks specific to generative AI.	Check generative AI risks such as fabricated outputs, data privacy and information security are covered.
ISO/IEC 42001:2023	The first international, certifiable standard for an AI management system, published December 2023.	Ask whether the company is certified or aligned, and check the scope of any certificate.
OWASP Top 10 for LLM Applications (2025)	The ten most important security risks for applications built on large language models.	Use as the security checklist in Chapter 06.
MITRE ATLAS	A knowledge base of adversary tactics and techniques against AI systems, drawn from real attacks and red teams.	Test whether the threat model and red-team scope are realistic.
EU AI Act	EU regulation classifying AI by risk, with obligations and penalties.	Classify use cases, identify obligations and assess exposure.
UK Code of Practice for the Cyber Security of AI (2025)	13 principles across the AI lifecycle, intended to inform a global ETSI standard.	A baseline of reasonable security practice, including documentation of data, models and prompts.

Read certificates carefully. An ISO/IEC 42001 or ISO/IEC 27001 certificate is evidence of a management system within a defined scope. It is not evidence that a particular model is accurate, that its data rights are clean, or that its margins hold. Check what the scope covers, and test the product regardless.

Related: [NIST AI RMF guide](#) · [ISO/IEC 42001 implementation guide](#)

TOOLS

The scorecard

Score each area from 1 to 4 using the descriptions below. Record the evidence and your confidence. A score of 1 in data rights, security or AI claims can be a deal-level issue on its own.

AREA	1 · UNPROVEN	2 · FRAGILE	3 · CREDIBLE	4 · DEFENSIBLE
Architecture	Cannot explain how a request flows; no logging.	Works, but relies on manual steps and unversioned prompts.	Documented, versioned, observable; models swappable.	Mature change control; rollback proven; engineering is a source of advantage.
Model dependency	One provider; terms unread; no fallback.	Alternatives identified but untested.	Tested fallback; pinned versions; terms reviewed.	Proven migrations; dependency is a choice, not a constraint.
Proprietary data	Rights unclear; no inventory.	Rights mostly clear; value not measured.	Rights evidenced; contribution measured.	Exclusive, renewing data with measured advantage.
Defensibility	Thin wrapper; no named moat.	Moat claimed but not yet evidenced.	Moat evidenced by retention or switching costs.	Several reinforcing moats with evidence.
Evaluation	Demos and testimonials only.	Occasional offline tests.	Versioned test set, baselines, release gate.	Production outcomes measured and linked to value.

AREA	1 · UNPROVEN	2 · FRAGILE	3 · CREDIBLE	4 · DEFENSIBLE
Security	No AI threat model; agents unrestricted.	Basic controls; AI risks partly covered.	Threat model, scoped permissions, independent testing.	Security evidence is a sales advantage with demanding customers.
Scalability	Never tested beyond pilots.	Handles today's load; bespoke onboarding.	Load-tested to plan; fallbacks; falling onboarding effort.	Scales with improving margin and reliability.
Inference economics	Cost per task unknown.	Known on average; heavy users or credits hide losses.	Segment margins known and stress-tested.	Structural cost advantage over competitors.
Technical team	Cannot build or explain the core.	Capable, but critical knowledge sits with one person.	Named owners, backups and a credible hiring plan.	Team quality attracts talent and customers.
IP and legal	IP unassigned or licences breached.	Gaps that can be fixed before close.	Clean IP; regulatory position understood.	Compliance and evidence ahead of the market.

How this relates to the seven-dimension scorecard

The [Kesho scorecard article](#) uses seven dimensions: customer value, data, model evidence, architecture, security and governance, economics and team execution. This guide expands them into ten areas for a full review. Model dependency and defensibility extend architecture and data; scalability extends architecture; IP and legal extends security and governance. Customer value is assessed through evaluation, defensibility and commercial diligence.

The AI-washing test

Run this before committing time to full diligence. Each "no" is a question to resolve.

- ☐ Can the company show where AI sits in the product, and what it does there?
- ☐ Does the product do what the deck says, when tested on your own examples?
- ☐ Is the share of work completed without human involvement known and disclosed?
- ☐ Are performance claims backed by an evaluation with a defined test set and metric?
- ☐ Do the website, deck, contracts and engineers describe the product consistently?
- ☐ Would each AI claim survive scrutiny from a regulator or a customer's lawyer?

WORKSHEET

Scoring worksheet

Company: _____ Stage: _____ Reviewer: _____ Date: _____

AREA	SCORE 1-4	CONFIDENCE	KEY EVIDENCE	CONDITION OR ACTION
Architecture				
Model dependency				
Proprietary data				
Defensibility				
Evaluation				
Security				
Scalability				
Inference economics				
Technical team				
IP and legal				
AI-washing test				

DEAL-LEVEL ISSUES

Any area scored 1 where the thesis depends on it.

RECOMMENDED OUTCOME

- ☐ Proceed
- ☐ Proceed with conditions
- ☐ Reprice or restructure
- ☐ Do not proceed

TOOLS

Red-flag checklist

One red flag is rarely decisive. Several in the same area, or any that undermine the thesis, should change the scope, the terms or the decision.

Claims and evidence

- ☐ AI claims the product or evidence does not support
- ☐ Accuracy quoted with no test set or definition
- ☐ Only demos, testimonials or public benchmarks
- ☐ Founders and engineers describe the product differently

Architecture and dependency

- ☐ Nobody can trace a real request through the system
- ☐ Prompts changed in production with no history
- ☐ Single provider, no tested fallback
- ☐ Value is mostly the base model plus prompts
- ☐ Manual work hidden behind "automation"

Data and IP

- ☐ No data inventory or rights analysis
- ☐ Customer contracts silent on training use
- ☐ Scraped data with no source records
- ☐ Unassigned IP from contractors or founders
- ☐ Open-weight model licence not checked

Security

- ☐ No AI-specific threat model
- ☐ Agents can act without approval or scoped permissions
- ☐ Secrets or customer data in system prompts
- ☐ No usage limits per user or key
- ☐ Retrieval not separated by customer or permission

Scale and economics

- ☐ Cost per successful task unknown
- ☐ Margins depend on credits that will expire
- ☐ Human review excluded from cost of goods
- ☐ Flat pricing with unlimited usage
- ☐ Never load-tested; bespoke onboarding per customer

Team and regulation

- ☐ One person holds critical model or data knowledge
- ☐ High recent engineering attrition
- ☐ Possible EU AI Act high-risk use, not assessed
- ☐ Contracts warrant accuracy without evidence

TOOLS

Questions to ask founders

Ask open questions and request evidence for every answer. The best follow-up is usually "Show me." Numbered for reference in meetings and notes.

Architecture

1. Walk me through what happens when a customer submits a request.
2. What is decided by code, and what by the model?
3. How are prompts and model versions reviewed and rolled back?
4. What would you rebuild with six months and no customer commitments?

Model dependency

5. What happens if your main provider doubles prices or retires your model?
6. Have you run your evaluations on another model? Show me.
7. What do provider terms allow for your customers' data?
8. Why would customers not use the provider's own product?

Proprietary data

9. Which data may you use for training? Show me the clause.
10. How much worse is the product without your proprietary data?
11. Could a competitor obtain equivalent data, and at what cost?
12. What would you do if a major customer asked you to delete their data?

Defensibility

13. What could a competitor with the same model not copy?
14. How do you compare with general-purpose assistants customers already have?
15. Does quality improve with usage? Show me the trend.
16. What would it cost a customer to replace you?

Evaluation

17. Show me the test set you use to judge a release.
18. What is your baseline, and by how much do you beat it?
19. Where does the product perform worst?
20. When did a release last fail your gate?
21. How do you detect drift in production?

Security

22. What is the worst thing an attacker could make your system do?
23. Who last tested you for prompt injection, and what did they find?
24. Which actions need human approval?
25. How would you know if customer data leaked through an output?

Scalability

26. What breaks first at ten times today's volume?
27. What happened during your provider's last outage?
28. How has onboarding time changed over the last year?

Inference economics

- 29. What does one successful task cost, including human review?
- 30. What is the margin on your ten heaviest customers?
- 31. What is your margin after current credits expire?
- 32. How does pricing protect you if usage per customer doubles?

Technical team

- 33. If your lead ML engineer left tomorrow, what would stop working?
- 34. Which hires does the roadmap depend on?
- 35. Who outside the founding team wrote core code?
- 36. Why have engineers left in the last two years?

IP, legal and regulation

- 37. Is all code, model and data work assigned to the company?
- 38. Which open-licensed models do you use, and under what terms?
- 39. Which use cases could be high-risk under the EU AI Act?
- 40. What evidence supports each AI claim in this deck?

Closing questions

- 41. What do you know about your product's weaknesses that we have not asked about?
- 42. What would you spend the first \$1 million of this round on, technically?
- 43. Which customer would give us the most honest view of the product?

How to read the answers

SIGNAL	USUALLY MEANS
Specific numbers, offered with caveats	The team measures and understands its product.
Evidence produced within a day	The practice exists; it was not created for the data room.
Founders volunteer weaknesses	Lower risk of surprises after investment.
Answers that change between meetings	The underlying facts are not known or are being managed.
"We'll have that soon"	Treat as not yet existing. Make it a condition if it matters.
Deflection to vision or market size	The question touched a real gap. Follow up.

TOOLS

The decision and the first 100 days

Diligence is only useful if it changes a decision or a plan. Translate findings into one of four outcomes, then into the first priorities after investment.

OUTCOME	WHEN TO USE IT	TYPICAL ACTIONS
Proceed	Evidence supports the thesis; gaps are normal for the stage.	Share findings with the founders as a roadmap.
Proceed with conditions	The thesis is credible, but specific controls, hires, rights or fixes are needed.	Conditions precedent or post-close covenants with dates and owners.
Reprice or restructure	Cost, dependency or delivery assumptions differ materially from the deal model.	Adjust valuation, tranche funding against milestones, or change structure.
Do not proceed	A foundational claim is unsupported, or the downside cannot be controlled.	Document the reasons; revisit if evidence changes.

Common conditions

- Update customer contracts to secure data-use rights the product depends on.
- Test a fallback model and record evaluation results.
- Introduce a release gate and a versioned test set.
- Scope agent permissions and add human approval for consequential actions.
- Complete IP assignment from contractors and former staff.
- Retention arrangements for key technical staff, and named backups.
- Correct AI claims in marketing and investor materials.

The first 100 days

DAYS 1–30

- Agree the findings and owners with the founders.
- Close any critical security or rights gaps.
- Set up cost and quality reporting to the board.

DAYS 31–60

- Release gate and test set in place.
- Fallback model tested.
- Key hires started.

DAYS 61–100

- Margin by segment reported.
- Governance pack in place for enterprise sales.
- Re-score the ten areas.

Related: [The AI-ready portfolio framework for investors](#)

REFERENCE

Glossary

TERM	MEANING
Ablation	A test that removes one component, such as a dataset, to measure how much it contributes.
Agent	An AI system that plans and takes actions, such as calling tools or updating records, to complete a task.
Baseline	The reference point for comparison: a previous version, a simpler method or human performance.
Evaluation (eval)	A repeatable test of AI output quality against a defined test set and metric.
Fine-tuning	Further training of an existing model on specific data to change its behaviour.
Foundation model	A large model trained on broad data and adapted to many tasks. Often accessed through an API.
Guardrails	Rules or controls that limit what an AI system can output or do.
Inference	Running a trained model to produce an output. The main running cost of most AI products.
Jagged frontier	The uneven pattern of AI capability: excellent at some hard tasks, poor at some simple ones.
Model-as-judge	Using one model to score another's outputs. Needs checking against human reviewers.
Open-weight model	A model whose trained parameters are published, so it can be run independently, subject to its licence.
Prompt injection	An attack that hides instructions in input or content to change an AI system's behaviour.
Release gate	Criteria a new version must meet before it can be deployed.
Retrieval-augmented generation (RAG)	Retrieving relevant documents and supplying them to a model so it can answer from them.
System prompt	Instructions given to a model by the developer to set its behaviour and rules.
Thin wrapper	A product that adds an interface and prompts to a model it does not control, with little else.
Token	A unit of text processed by a model. Providers usually price by tokens in and out.
Vector database	A store of numerical representations of content, used to find similar items for retrieval.

REFERENCE

Sources

All sources accessed in October 2026.

1. Stanford HAI. *The 2026 AI Index Report*. hai.stanford.edu/ai-index/2026-ai-index-report
2. Stanford HAI. *The 2025 AI Index Report*. hai.stanford.edu/ai-index/2025-ai-index-report
3. OWASP GenAI Security Project. *OWASP Top 10 for LLM Applications 2025*. genai.owasp.org/llm-top-10
4. MITRE. *ATLAS: Adversarial Threat Landscape for Artificial-Intelligence Systems*. atlas.mitre.org
5. UK Department for Science, Innovation and Technology. *Code of Practice for the Cyber Security of AI*, 31 January 2025. gov.uk/government/publications/ai-cyber-security-code-of-practice
6. NIST. *Artificial Intelligence Risk Management Framework (AI RMF 1.0)*, NIST AI 100-1, 2023. doi.org/10.6028/NIST.AI.100-1
7. NIST. *Generative Artificial Intelligence Profile*, NIST AI 600-1, 2024. doi.org/10.6028/NIST.AI.600-1
8. ISO. *ISO/IEC 42001:2023 Artificial intelligence: Management system*. iso.org/standard/42001
9. EU Artificial Intelligence Act. *Implementation timeline*. artificialintelligenceact.eu/implementation-timeline
10. EU Artificial Intelligence Act. *Article 99: Penalties*. artificialintelligenceact.eu/article/99
11. EU Artificial Intelligence Act. *Annex III: High-risk AI systems*. artificialintelligenceact.eu/annex/3
12. US Securities and Exchange Commission. *SEC Charges Two Investment Advisers with Making False and Misleading Statements About Their Use of Artificial Intelligence*, 18 March 2024. sec.gov/newsroom/press-releases/2024-36
13. US Federal Trade Commission. *FTC Announces Crackdown on Deceptive AI Claims and Schemes*, 25 September 2024. ftc.gov/news-events/news/press-releases/2024/09
14. US Copyright Office. *Copyright and Artificial Intelligence*, Parts 1–3. copyright.gov/ai
15. US Copyright Office. *Copyright Office Releases Part 2 of Artificial Intelligence Report*, 29 January 2025. copyright.gov/newsnet/2025/1060.html
16. Kesho Partners. *AI technical due diligence: separating a demo from a defensible product*. keshopartners.com/insights/ai-technical-due-diligence-scorecard

Independent technical due diligence for AI investments

Kesho Partners helps venture, growth and private equity investors understand the technology behind a deal: architecture, data, models, evaluation, security, economics and team. We separate material risk from normal early-stage trade-offs, and turn findings into conditions, price implications and a practical 100-day plan.

If you are assessing an AI company and want an independent view before you commit, we can help with a focused review or a full diligence engagement.

keshopartners.com/technical-due-diligence