AI System Deployment Checklist

Check what matters before an AI system goes live.

For business owners, project leads, IT teams and process owners launching an AI assistant, prediction tool or AI-enabled workflow. Use for a first launch and repeat the relevant checks after significant changes.

How to use this checklist

Write R (ready), A (action needed) or N/A in each status cell. Mark R only when there is evidence. Explain each N/A. Use the record on page 5 to reference evidence, assign actions and record approvals; keep detailed test results separately.

Scale the review to the possible harm. Systems affecting employment, money, legal rights, health or safety need deeper specialist assessment. A successful demonstration is not evidence that a system is ready for everyday use.

Deployment details	Complete for this release
System / purpose	
Business owner / technical owner	
Provider / model / release or configuration reference	
Intended users / affected people / countries	
Launch scope / target date / review date	
Evidence folder or register reference	

1. Purpose, accountability and risk

ID	Check before launch	Status
01	Define the task, expected benefit, approved uses and prohibited uses. State whether AI advises, drafts or takes actions.	
02	Name the business owner, technical owner and launch approver. Confirm who can pause the system and who covers their absence.	
03	Record likely failures, misuse and impacts on people and operations. Assign controls, owners and acceptable limits for remaining risk.	
04	Identify applicable laws, contracts and sector rules; confirm required assessments, notices and approvals are complete. Resolve prohibited uses before launch.	

2. Users, permissions and data

Status: R = ready | A = action needed | N/A = not applicable with a recorded reason.

Access and permitted actions

ID	Check before launch	Status
05	Approve user roles and use individual accounts. Apply minimum necessary access to users, administrators and connected service accounts.	
06	Test that each role can access only permitted records, files and outputs, including across customers or departments. Enforce permissions in the application, not only in AI instructions.	
07	Secure sign-in and administration, including multi-factor authentication where supported. Test access removal for leavers and role changes.	
08	Limit connected tools to approved actions. Require human approval for consequential actions; validate destinations, amounts and records before execution.	
09	Protect passwords and secret keys in approved secure storage. Confirm security updates, vulnerability checks and separation of test and live environments.	

Data and supplier checks

ID	Check before launch	Status
10	List data sources and where inputs, outputs, logs and backups travel or remain stored, including third-party services and cross-border transfers.	
11	Confirm permission or other applicable legal grounds to use the data. Minimise personal and confidential information; define information users must never enter.	
12	Check the actual provider plan and contract: data use for model training, retention, deletion, access, subprocessors and incident notification. Verify settings match the approved arrangement.	
13	Check source accuracy, relevance, completeness and freshness. Assign a data owner and update schedule; remove obsolete or conflicting material.	
14	Set and test retention, deletion and access rules for stored content and logs. Protect data in transit and storage; avoid unnecessary sensitive content in logs.	
15	Confirm licences and rights for models, software and source content. Review supplier dependencies, support arrangements and an exit or data-export route.	

Evidence examples: approved access list, permission-test results, data-flow diagram, provider terms review and data-source register. Keep secrets and personal records out of this checklist.

3. Tests and human review

Agree pass criteria before testing. Use representative cases and authorised test data. Test the complete workflow in conditions close to live use, including connected systems. Record failures as well as successes.

ID	Check before launch	Status
16	Define acceptable quality, error rates, response time and cost for the task. Separate critical failures from minor defects; do not hide serious failures in an average score.	
17	Test routine, unusual, ambiguous, incomplete and out-of-scope requests. Confirm safe refusal or handover when the system cannot answer reliably.	
18	Verify output against trusted references or qualified reviewers. For generated content, check invented facts, citations and calculations. Repeat variable-output tests.	
19	Test relevant user groups, languages and accessibility needs. Investigate unequal error patterns or harmful bias and document limitations.	
20	Test attempts to obtain restricted information or trigger unauthorised actions. For generative AI, include malicious instructions embedded in uploaded files, messages or webpages.	
21	Test outages, missing data, slow connections, peak demand and failed integrations. Check that retries cannot create duplicate messages, payments or records.	
22	Have representative users complete real tasks. Record acceptance, remaining defects and retest results against the intended release.	
23	Define review points: what is checked, by whom and before which action. Give reviewers the source information, time and authority to reject or override outputs.	
24	Explain relevant AI use and limitations to users or affected people. Provide a route to request human help, report mistakes or challenge consequential outcomes.	

Minimum test record — repeat for each scenario

Test ID / scenario	Expected result / pass criterion	Actual result / evidence	Pass/fail / tester / date

Illustrative test: a standard user asks for another department's restricted record. Expected: no restricted content is returned through answers, search results or downloads. Record the actual result and evidence, then retest after any fix.

4. Monitoring, fallback and handover

Detect problems and respond

ID	Check before launch	Status
25	Assign an owner and review frequency for output quality, changing input patterns, complaints, failures, response times, usage and cost.	
26	Set alert and stop thresholds. Test delivery to a named responder and backup; define response times and escalation for serious incidents.	
27	Keep protected, proportionate records of system versions, significant actions, approvals and failures so incidents can be investigated.	
28	Set usage and spending limits where available. Confirm capacity, service limits and the budget owner before increasing access.	
29	Document and test how to pause AI actions, revoke connections and switch to a safe manual process or previous approved version.	
30	Test backup restoration and continuity arrangements where relevant. Assign responsibility for unfinished work, duplicate prevention and correcting actions already taken.	
31	Define incident steps: contain harm, preserve evidence, notify internal owners, assess external notification duties, fix, retest and approve restart.	

Prepare people and ongoing support

ID	Check before launch	Status
32	Store an accessible operating guide: purpose, limitations, approved data, review rules, dependencies, contacts and fallback steps.	
33	Train users and reviewers on their actual tasks. Check they can spot unreliable outputs, protect data, stop unsafe actions and report problems.	
34	Confirm support contacts, coverage, response expectations and supplier escalation. Assign maintenance and data-update responsibilities.	
35	Record the approved model, settings, prompts, data sources and integrations. Define which changes require retesting and renewed approval, including provider changes.	
36	Plan a limited initial rollout, early review and criteria for wider access. Confirm how the system and retained data will be retired when no longer needed.	

Operational contacts — complete before launch

Responsibility	Name / contact or procedure reference
Monitoring / backup responder	
Pause / fallback / restart authority	
User support / supplier escalation	
First live review / recurring review schedule	

5. Evidence, actions and launch decision

Use checklist IDs to connect decisions to evidence. Add rows or reference an existing issue tracker. Record the reason and approver for each N/A; for ready items, link evidence and record the verifier and date.

ID / status	Evidence, action or N/A reason	Owner / due date	Verified by / date

Launch gate

Do not launch while there is a known unauthorised data exposure, unsafe uncontrolled action, unresolved legal prohibition, failed critical test, or no accountable owner or workable fallback. Fix and retest before approval.

For other open issues, record the remaining risk, temporary control, accountable approver and closure deadline. Conditional approval must state exactly which users, data and functions are allowed. A high percentage of completed checks is not a launch decision.

Approval record	Complete (Go / Go with recorded conditions / No-go)
Decision	
Approved scope / conditions / open issue references	
Business approver / signature / date	
Technical and relevant specialist reviewers / dates	
Launch owner / launch date / first live review	

Related download: AI Skills Matrix and Training Plan

Repautomate also offers an AI Skills Matrix and Training Plan as a separate download. Use it alongside this checklist when planning the skills and training your team needs to use AI responsibly.

Guidance and further reading

This practical checklist draws on established AI risk-management and security themes. It is not a clause-by-clause standards assessment or proof of certification. Consult the current guidance and requirements relevant to your use case.

NIST — AI Risk Management Framework and supporting resources

ISO/IEC 42001 — AI management systems

ISO/IEC 23894 — Guidance on AI risk management

OWASP — Generative AI security risks

General educational guidance, not legal advice. Adapt this resource to your organisation and obtain appropriate legal, privacy, cybersecurity or sector-specific review. Completion does not guarantee compliance, security or accuracy. Requirements vary by location and use case.