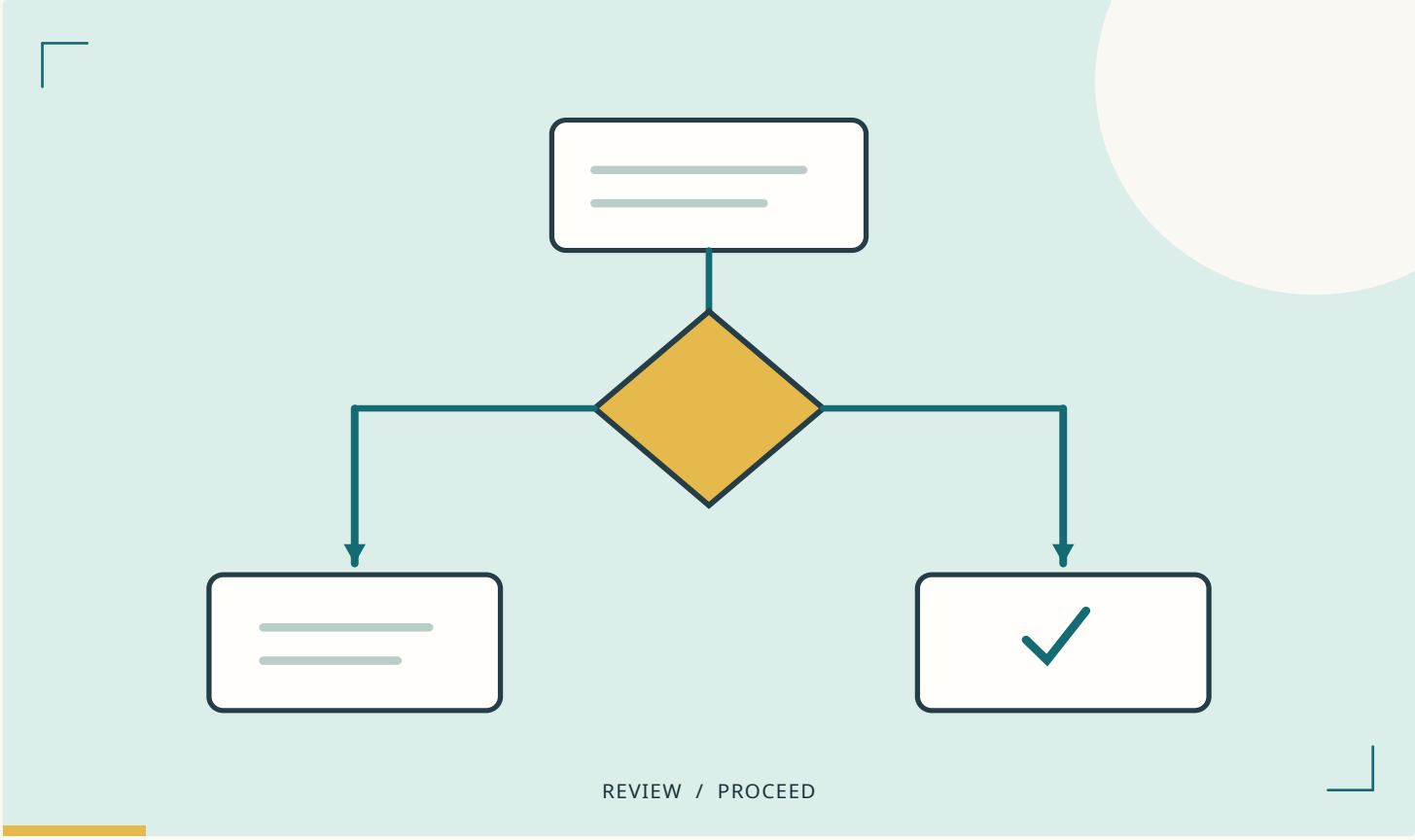
HUMAN REVIEW GATES

Defining Human Review Gates for AI Work



AI & DATA

Defining Human Review Gates for AI Work

Learning outcomes

- Route eight scenarios through proportionate human review gates.
- Distinguish source verification from release authorization.
- Keep high-impact personal decisions out of automatic approval.
- Specify gate triggers, roles, evidence, states and permitted transitions.
- Apply multi-gate precedence without confusing factual support with completed human review.
- Bind approval to exact artifact, source revision, audience and action.
- Evaluate conditional, stale and wrong-role approvals using concrete evidence.
- Produce a reproducible route ledger without making sensitive high-impact personal judgments or claiming unperformed release.

Original teaching material. All organizations, people, records and operational rules in exercises are synthetic. Use only the supplied non-sensitive inputs. This is a practice course, not an accredited or regulated certification.

Study the foundations, work through the applied units, complete the practice tasks, and then attempt the assessment. Use only the supplied fictional practice data. Keep your original inputs and your revised work separate.

Core concepts

A gate answers a specific question

Human review is most useful when the reviewer knows what they are checking and what authority they have. One gate verifies factual support; another authorizes an external communication; another determines that a task is unsuitable for automatic decision. Do not collapse these into a generic human in the loop checkbox. A reviewer who checks arithmetic has not necessarily approved publication. Define the trigger, reviewer role, evidence needed and allowed next action for each gate.

Separate low-risk drafting from release

A source-supported internal summary can proceed after a factual check under the fictional workflow. An unsupported claim pauses for verification even if the summary will stay internal. An external message or new commitment requires the responsible owner’s approval before release. The absence of personal data does not automatically make publication authorized. Conversely, a draft can be prepared without sending it. This distinction lets a team benefit from assistance while keeping consequential boundaries visible.

Do not automate sensitive high-impact judgments

The exercise includes hiring and financial-eligibility scenarios based on sensitive personal information. These must not be automatically approved by the AI workflow. Route them to the appropriate authorized human process and do not use a model score as a substitute for that process. This course does not teach legal compliance or professional eligibility decisions. Its operational lesson is to recognize when a seemingly routine workflow has crossed into a consequential judgment about another person and to stop the automatic decision path.

Make the gate auditable

The routing record should name the case, trigger, required reviewer, evidence and permitted status. A status such as awaiting approval is not approval. After a reviewer acts, record the actual decision and any conditions. Do not assume silence means consent. For this practice packet, learners only design the route; they do not send messages or make real decisions. Check that every scenario reaches an appropriate path and that no external release or high-impact judgment slips through a generic quality score.

Worked example

Demonstration: a correct internal draft is later proposed as a public announcement. The factual gate may already be satisfied, but the destination changed, so release authorization is still required. Reusing the draft does not reuse approval that was never given for public sharing.

Foundation practice

Fictional policy

Check and internal: Human checks source facts; may retain as internal low-risk draft

Verify first: Pause and resolve unsupported claim or arithmetic discrepancy

Owner approval: Prepare draft; responsible owner must approve external release or commitment

Do not automate: Do not make automated high-impact personal judgment; route to authorized human process

Cases

id	scenario
G1	Internal summary of supplied non-sensitive counts, all facts supported.
G2	Summary includes a claim absent from the packet.
G3	Draft is ready to email outside the organization.
G4	AI recommends whether a person should be hired based on private health details.
G5	Internal summary has an arithmetic discrepancy.

id	scenario
G6	Public post includes a new commitment on behalf of the team.
G7	AI asks to approve financial eligibility from personal assets.
G8	Internal meeting note is source-supported and contains no personal data.

Foundation tasks

Guided preparation

Instructions

Read the complete input pack. Create the output columns specified in the capstone, keep the original IDs, and state the unit of work. Complete the first two records manually before using an optional AI assistant. Record any assumption that could change the result.

Independent capstone

Instructions

Draw or describe the routing flow and classify all eight scenarios. Include trigger, reviewer role, required evidence and allowed next action. Target seven correct routes, with no automatic approval of high-impact personal decisions.

Peer verification and revision

Instructions

Exchange only your student work with a peer. Ask the peer to trace two ordinary results and every exception back to the supplied input. Record the exact item corrected, the original value, revised value and reason. Submit the revised artifact plus a short explanation of the most important remaining limitation.

01 Replace the human-in-the-loop checkbox with a decision boundary

A review gate is a defined boundary between an allowed state and a more consequential next action. It asks a specific question, names a person with relevant authority, identifies the evidence they need and limits what may happen after their decision. “A human looked at it” does not establish any of those things. A person can inspect grammar without verifying facts, and verify facts without having authority to publish.

The foundation gives four routing labels. CHECK_AND_INTERNAL means a human checks source facts before retaining a low-risk internal draft. VERIFY_FIRST pauses unsupported claims or arithmetic discrepancies. OWNER_APPROVAL requires the responsible owner before external release or a new commitment. DO_NOT_AUTOMATE removes sensitive high-impact personal judgments from the automated decision path and sends the matter to an authorized

human process. These labels describe different questions, not four levels of a single confidence score.

The Harbor exhibit worked case makes those distinctions operational. An AI-assisted notice summarizes attendance for a fictional public exhibition. An operations analyst can verify the count. The exhibit director can authorize a specific communication or commitment. Neither role is treated as automatically authorized to make a hiring or financial-eligibility judgment from sensitive information. The task is process design using synthetic records; you will not send the notices or decide any person's outcome.

A useful gate prevents both accidental release and unnecessary paralysis. If a source-supported internal note needs only an analyst check, routing it to every senior manager wastes attention. If a public promise needs owner approval, treating a spell-check as sufficient creates a hidden commitment. Proportionate review means matching the gate to the proposed action and its evidence, not assuming that all AI output is equally risky or equally safe.

Throughout the course, separate three things: the content's quality, the authority to act and the record of what actually happened. A correct draft can be unauthorized. An authorized action can still depend on a condition that has not been met. A completed review can be recorded incorrectly. Designing a reliable workflow requires all three dimensions to be visible, especially when a draft changes audience or purpose after the first review.

The output is a routing and evidence design. The course's fictional policy is deliberately explicit so every case can be checked. It is not a statement of legal sufficiency or a universal rule for real organizations. The transferable skill is learning to ask which boundary is being crossed and what evidence would actually justify crossing it.

02 Start with the proposed action, not the model score

Before choosing a gate, identify what the workflow is trying to do next. Retaining an internal draft, sending an email to a partner, publishing a public post, promising replacement equipment and recommending a person's eligibility are different actions. They may use the same paragraph, but their consequences and required authority differ. A model confidence score does not tell you which action the user or organization authorized.

The Harbor packet records destination and action separately. A destination names who can receive or see the result; an action names what the system would do with it. external:partner-list plus send_email is a transmission to a defined external audience. internal:exhibit-team plus retain_draft is an internal storage or drafting step. A public destination with publish_post crosses a publication boundary even when the text contains no names or sensitive data.

Classify the real purpose, not only the file's label. A document named draft can contain an instruction to send a promise immediately. Conversely, a completed-looking message can remain a private draft awaiting approval. The workflow should read the requested next action and the current state rather than infer authority from a polished appearance, a final filename or the absence of visible errors.

Some cases lack enough context to route safely. R10 has a supported attendance claim but a null destination. The appropriate immediate state is AWAITING_CONTEXT, with VERIFY_FIRST used as the foundation-compatible holding route. The missing information is not a factual error in the count. It is an unresolved destination needed to determine the release gate. Name the reason so the person receiving the task knows what to clarify.

Do not replace unknown with internal as a convenient default. That could permit retention or release under a lower gate when the intended audience is external. Likewise, do not assume send_email means a public broadcast; it may be a narrowly defined recipient. Ask for the smallest missing decision: the exact intended destination and action. Keep supported drafting work separate from the blocked release step while the answer is pending.

A complete intake row therefore includes case ID, artifact ID and version, source ID and revision, destination, proposed action, commitment indicator and any high-impact personal-decision trigger. These fields are not bureaucratic decoration. They are the facts that determine which gate applies and whether an earlier review still matches the current work.

E02-X1 Diagnose missing routing context

Inspect R03 and R10 in assets/harbor_cases.json. Identify the field that prevents R10 from reaching the same release route as R03. Do not invent its value.

Submit: Two intake decisions and one clarification question.

03 Write a gate specification that someone can execute

A gate specification needs more than a label. State the trigger, the question, the reviewer role, the required evidence, the possible decisions and the allowed next action. For factual review, the trigger is an AI-assisted factual draft ready for checking. The question is whether each claim agrees with the supplied source. The reviewer is the operations analyst. The evidence includes source records, calculations and the exact candidate version.

For release review, the trigger is an external communication or a new commitment. The question is whether the responsible owner authorizes the exact content for the exact destination and action. The reviewer is the exhibit director in this fictional policy. Evidence includes the checked artifact, the audience, any proposed commitment and any conditions that must be met. An approval for internal retention does not answer the external release question.

Specify what reject, request_changes and pending mean. A rejected release must not proceed. A request for changes creates a new candidate that needs applicable review. Pending means no decision has been established. Silence is not an implicit fourth approval state. If the workflow needs a reminder or escalation procedure, it should be separately authorized and documented; the absence of a response does not justify bypassing the gate.

The allowed next action should be narrow. After a factual pass, an internal low-risk draft may be retained as checked. An external draft moves to owner review; it is not sent. After an

owner approval whose scope matches and whose conditions are met, the release can be authorized under the local policy. Authorization is a state, not evidence that the send action actually occurred.

Use explicit evidence requirements to prevent ceremonial review. “Check the notice” gives the reviewer no test. “Recalculate 12 plus 8 from H1 and compare the result with attendance_total in R03 version 1” is testable. “Approve the message” is incomplete without content, destination and action. A good specification makes it possible for a reviewer to say exactly what they have and have not decided.

A gate also needs a failure path. If the source is missing, the analyst cannot establish correctness. If the owner role is vacant, an editor cannot inherit approval authority merely because a deadline is near. Hold the affected action and route the missing evidence or authority to the appropriate process. Continue independent low-risk preparation where allowed, but do not disguise a blocked gate as completed work.

Consider two review requests for the same R03 notice. Request A says, “Please check this when you can.” Request B says, “For R03-notice version 1 using H1 revision 1, confirm that attendance_total 20 equals North 12 plus East 8. Record facts_verified or identify the discrepancy. This request does not authorize external sending.” Request B tells the analyst what to compare, what result to record and which authority is not being requested. That precision makes the resulting event interpretable.

The owner request should be different: “Please decide whether R03-notice version 1, supported by H1 revision 1 and the completed analyst check, may be emailed to external:partner-list using send_email. Record approve, deny or request_changes, and state any release conditions.” It should include the exact content. Asking the owner merely to approve the total would answer the wrong question. The same number appears in both requests, but the decision boundary is not the same.

Now imagine the owner responds, “The attendance count looks right.” That response supports the owner's view of the fact but is not an explicit release authorization under the declared policy. Do not translate it into approve because approval was what you hoped to receive. A useful clarification is whether the owner authorizes the exact proposed email to the stated destination. The system should retain the actual response rather than replace it with a stronger interpretation.

If the owner says, “Approve this version for the partner list, but do not include a free-kit promise,” the condition affects content. A draft that already contains the promise cannot be released unchanged. The workflow needs a revised artifact and applicable review of that exact version, or an unambiguous scope that already identifies the permitted final text. The course's exact-version policy does not allow an editor to assume that every later rewrite is approved simply because it tries to comply with a condition.

These examples show why permission language should be specific without becoming unnecessarily long. The request must identify the artifact, destination, action and consequential commitment, while the factual evidence can be a concise supporting packet. A reviewer should not need to reconstruct intent from several unrelated messages. The goal is

not to make every interaction formal; it is to make the meaning of the decision recoverable when the workflow later asks whether it may act.

04 Separate source verification from release authorization

Source	Raw record	Supported interpretation
H1 revision 1	North 12 visitors; East 8 visitors	Total 12 + 8 = 20
H2 revision 1	Demonstration date 2026-11-15; no clock time stated	Date supported; opening time unknown

Factual review asks whether a claim is supported. Release authorization asks whether the organization permits an action with that content. The two questions require different evidence. H1 supplies North attendance of twelve and East attendance of eight. Their sum is twenty. That arithmetic can establish the attendance claim, but it cannot establish permission to email the claim to partners or promise a benefit to attendees.

In R01, the attendance_total is twenty, the destination is internal and the analyst has recorded a matching factual review. The local policy therefore allows CHECKED_INTERNAL. In R03, the same correct claim is directed to an external partner list. Its factual review is complete, but no owner approval is recorded. The correct state is AWAITING_OWNER. Matching content does not make the two cases equivalent because their proposed actions differ.

An approval record is not a source of factual truth. Suppose the director approves a notice saying twenty-two visitors when the source records twelve and eight. The approval cannot make twelve plus eight equal twenty-two. Under the declared policy, the factual discrepancy must be resolved before release consideration. Authority to publish and accuracy of the statement are separate gates that both need to be satisfied where applicable.

Conversely, a correct source citation is not an approval record. A notice may cite every sentence and still lack permission for external transmission. This is particularly important when staff reuse a checked internal summary as a public announcement. The source work can remain useful, but the destination change triggers a different authorization question. Reuse evidence only for the question it actually answers.

The reviewer interface should show these dimensions separately. A single green approved badge hides whether it means facts verified, owner authorized, conditions satisfied or message sent. Use distinct fields or states with clear meanings. A recipient should not need to infer a release decision from an analyst's initials in the margin of a calculation sheet.

This separation also supports efficient review. The owner need not redo every addition if an accountable analyst has provided a reproducible calculation, but the owner still must assess the proposed communication and commitment. The analyst need not decide organizational release strategy to check a total. Clear boundaries let each reviewer contribute the evidence and authority appropriate to their role.

E04-X1 Same fact, different gate

Use H1, R01 and R03. Recalculate the attendance total and state the current status of each artifact.

Submit: Calculation, two statuses and the reason they differ.

05 Resolve unsupported claims before routing toward release

An unsupported claim is not necessarily false. It is a claim that the supplied evidence does not establish. R06 says the exhibit opens at 09:00, while H2 states a demonstration date and no opening clock time. The correct diagnosis is unsupported opening time, not proven wrong opening time. That difference matters because the remedy is to obtain authoritative evidence or remove the unsupported claim, not to invent a different hour.

An arithmetic discrepancy is more specific. R02 says attendance_total is twenty-two, while H1 gives twelve plus eight. The supplied data establish twenty, so the candidate is inconsistent with its source. A reviewer can repair the total directly from the packet, retaining the source and calculation. The correction still creates a new candidate version that needs a recorded factual check before being treated as complete.

VERIFY_FIRST can therefore contain several blocker types: unsupported assertion, arithmetic mismatch, missing source revision or missing routing context. Keep the blocker detail beside the route. A single label helps sort work, but the person resolving it needs to know whether to recalculate, locate evidence or clarify the audience. Treating every verification hold as a request for a higher confidence score would not address any of these actual gaps.

Do not let an external destination hide the first blocker. R07 is an external email with an incorrect total. It ultimately needs owner approval, but the immediate next gate is verification. Record both the current gate and the later required gate: VERIFY_FIRST now, OWNER_APPROVAL after a supported, fact-checked artifact exists. This avoids the mistaken conclusion that passing verification alone will authorize sending.

Repair should stay within the evidence. For R06, an acceptable provisional draft can omit the unsupported opening time while retaining the supplied demonstration date, if that edit meets the intended drafting task. If the requested notice must include an opening time, it remains incomplete until an authoritative source supplies one. A reviewer should not quietly redefine the deliverable just to clear the queue.

Keep a record of what changed and why. “R02 total changed from 22 to 20 using H1 North 12 plus East 8” is reproducible. “AI fixed it” is not. The revised artifact and factual-review record should match the same version. Otherwise, a later release reviewer may see approval for one text and a different text waiting to be sent.

06 Make multi-gate precedence explicit

Many real tasks trigger more than one gate. A public post can contain an unsupported fact and a new commitment. A sensitive personal decision can also contain an arithmetic error. A routing design needs a declared precedence so the immediate action is unambiguous, while preserving later requirements. Without that order, different operators may choose whichever label seems easiest to clear.

The expansion policy uses this sequence. First, stop the automated high-impact personal-decision path when the sensitive-basis trigger applies. Second, hold unknown destination or action context. Third, resolve unsupported claims or arithmetic discrepancies. Fourth, require matching human factual review. Finally, require owner authorization for an external action or new commitment; otherwise retain the checked internal draft. This is a local teaching sequence, not a universal compliance rule.

The first step is not a request for a more accurate automated eligibility result. In a sensitive hiring or financial-eligibility case, the workflow routes the matter to an authorized human process and does not recommend the personal outcome. Correcting a count or removing a name would not necessarily change that purpose. The routing decision concerns what the system is being asked to decide, not only the surface appearance of its input.

For ordinary communications, unresolved facts precede owner release review because the owner needs a stable artifact. An owner can request preparation or decide a policy question while facts are being checked, but that is not a final authorization of an unknown corrected message. Keep those discussions separate from the release record. A shortcut that approves “whatever the analyst fixes” obscures the exact consequential content being authorized.

Use two fields where useful: next_gate and remaining_gates. R07 has next_gate VERIFY_FIRST and remaining_gates containing factual check and owner release approval. R03 has next_gate OWNER_APPROVAL because its facts are already checked. R01 has no external-release gate under the local scope. This representation prevents a queue label from being misread as the complete history or all future requirements.

Test precedence with deliberately combined cases. A route engine that returns OWNER_APPROVAL whenever destination is external may ignore a sensitive high-impact personal judgment or an incorrect total. A route engine that returns CHECK_AND_INTERNAL whenever facts are supported may ignore a new commitment. Order the questions by the policy, then verify both ordinary cases and combinations that would expose a premature return.

E06-X1 Route a combined blocker

R07 has attendance_total 22, H1 says North 12 and East 8, the destination is external:partner-list, and no reviews or approval exist. Specify the immediate gate and complete allowed sequence.

Submit: One ordered route with no automatic send.

07 Keep sensitive high-impact judgments outside automatic approval

Some requests ask the workflow to make a consequential judgment about a person rather than prepare routine content. The foundation includes hiring based on private health information and financial eligibility based on personal assets. The local rule is DO_NOT_AUTOMATE: do not make the automated personal judgment, and route the matter to the appropriate authorized human process. The course supplies no actual health or financial records and no personal decision should be attempted.

Recognize the combination of purpose and basis. “Summarize a public exhibition schedule” is a content task. “Recommend whether this person should be hired using private health details” is a high-impact personal decision using sensitive information. A polished summary format does not remove the consequential purpose. Likewise, changing approve to recommend does not necessarily make the automated judgment benign if the recommendation is intended to determine the outcome.

A human rubber stamp does not transform an automatic decision into a meaningful authorized process. If the reviewer sees only a score and clicks accept without relevant evidence or authority, the underlying problem remains. This course does not specify professional hiring, lending or medical procedures. It teaches recognizing that the routine content workflow is not the right place to make those decisions.

The routing artifact should state the trigger without copying unnecessary sensitive details. In R05, the request summary already identifies private health information and hiring purpose. That is enough to explain the DO_NOT_AUTOMATE route. There is no need to create a sample diagnosis, infer a disability or predict job performance. Data minimization here means recording what supports the routing decision and avoiding irrelevant personal content.

Allowed preparation must remain bounded. A workflow can identify that an authorized human process is needed, preserve the request identifier and explain why no automated outcome was produced. It should not fill the gap with an unofficial risk ranking, a suggested rejection or a confidence score. A route to human review is not permission to perform the prohibited judgment first and attach a disclaimer afterward.

In assessment, treat an automatic approval of either sensitive high-impact case as a mandatory revision regardless of the total rubric score. This is an explicit course gate, not a claim about every possible policy. The point is that a high average on routine routing cannot offset a failure at the exact boundary the workflow was designed to protect.

08 Assign reviewers by authority and competence

A reviewer role should be justified by the question being asked. The operations analyst can reproduce counts and compare claims with the source register. The exhibit director can authorize public communications and commitments under the fictional policy. A person can hold more than one role in a real organization, but the record should still show which authority they exercised for each decision. Job seniority alone does not imply every permission.

Competence and authority are different. Someone may understand the arithmetic but lack permission to publish. Someone may have release authority but need a specialist to assess technical accuracy. A robust gate can require both contributions without making one person's signature stand for both. The evidence package should let the release owner see that factual review occurred and what limitations remain.

Q03 in the capstone has a public commitment and an approve event from the operations analyst. That event is not a valid owner authorization under the declared role map. Do not treat a matching word approve as sufficient. The reviewer identity and role must match the gate. The correct response is to seek the exhibit director's release decision, not to reinterpret the analyst's signature as a promotion.

Avoid circular approval. If an automated drafting component assigns itself a reviewer label or writes “approved by owner” into the message, that text is not an authenticated decision event. The teaching packet separates the artifact from the approval record so learners can inspect the difference. In a real implementation, identity and event authenticity need appropriate system controls beyond the simplified JSON examples here.

An unavailable owner is a blocker, not a reason to widen authority silently. Record the status as awaiting the authorized role. If the organization has a formally delegated substitute, the delegation should be part of the policy evidence and scoped to the action. The supplied packet has no substitute role, so learners must not invent one to finish the case.

The gate designer should also avoid impossible workload assumptions. Requiring a director to recheck every ordinary sum may burden the scarce role without improving the actual authorization decision. Give the owner concise evidence, exact content and the specific commitment question. Let the analyst's reproducible check answer the factual question, while preserving the owner's independent responsibility to decide whether the action is authorized.

E08-X1 Evaluate a wrong-role approval

Q03 proposes a public replacement-kit commitment. Its approval event says approve but role is operations analyst. The policy names exhibit director as release owner.

Submit: Validity decision, next role and permitted current action.

09 Bind approval to an exact artifact and destination

An approval should identify what was approved, for whom and for what action. In this course the binding includes artifact ID, artifact version, source revision, destination and action. Each element prevents a different kind of accidental reuse. The artifact ID distinguishes notices. The version distinguishes edited text. The source revision identifies the facts being relied on. The destination and action define the authorized transmission or retention.

R08 has a matching approve event from the exhibit director for its current notice version, source revision, partner-list destination and send_email action. Its factual review also matches, and no conditions remain. Under the fictional policy it can reach AUTHORIZED_RELEASE. That state does not say an email was sent; it says the release gate is satisfied for the specified action.

R09 has current artifact version two but approval for version one. The claim remains twenty, yet the approval scope is stale. The packet does not say whether other wording or commitments changed between versions. The safe routing result is AWAITING_OWNER for the current version. Do not infer that unchanged numeric facts make all content changes immaterial when the declared policy requires exact-version approval.

Destination changes are equally important. Approval to send a notice to a partner list is not approval to publish it on a public page. An internal-retention approval is not external-send approval. The content can remain byte-for-byte identical while the audience changes. A review system that binds only the text hash and ignores destination would miss that expansion of exposure.

A source revision change can also invalidate a decision even when the artifact's visible claim appears correct. In Q07, the current source is revision two and the approval record refers to revision one. The local rule requires matching source revision so the owner knows which evidence supports the current artifact. That is a declared conservative condition, not a claim that every source edit in every organization must trigger full reapproval.

Make scope mismatches visible rather than merely labeling approval invalid. “Artifact version 1 approved; current version 2” gives the owner a concrete reason to review again. “Wrong destination: internal retention approved, external email requested” identifies the missing action. Specific mismatch records reduce confusion and prevent a colleague from trying to fix the wrong field.

10 Represent states and transitions without implying completion

A workflow state describes the current evidence and permission, while a transition describes what event can legitimately change that state. AWAITING_FACT_CHECK becomes CHECKED_INTERNAL only when a matching analyst review exists and the work remains internal with no new commitment. For external work, a factual pass moves the artifact toward AWAITING_OWNER, not directly to sent. The event must answer the question attached to the state boundary.

Keep requested, approved and executed separate. An owner request at 09:03 says someone asked for authorization. An approval at 09:07 says the owner granted the specified permission. A recorded successful transmission event would establish execution within this simulation, but would not prove recipient receipt or reading. The supplied event packet contains no send event. Reporting the notice as sent after L3 would therefore claim an action not recorded in the evidence.

The synthetic R03 event sequence begins with factual review, then a pending release request, then an owner approval for partner-list email. L4 later changes the destination to a public exhibit page. That later change makes the prior destination-specific approval inapplicable to the new release request. The correct end state is awaiting owner authorization for the public destination, even though an earlier approval event remains in the history.

Event order and source dates serve different purposes. The observed_at timestamps say when the fictional workflow events were recorded. The demonstration date inside H2 describes the exhibit schedule. Do not sort policy authority by whichever date looks latest without knowing what the date represents. A later source event may require content review; a later routing event may change destination; neither should be silently conflated with the other.

State transitions should preserve history. Mark an earlier approval superseded for the new scope rather than deleting it. The record explains why a case that once looked ready is now blocked. This is especially useful when someone asks why work returned to review: the reason is not that the previous reviewer failed, but that the proposed action changed after their decision.

A transition table should include invalid transitions too. Pending to authorized without an approval event is invalid. Fact-checked to sent without owner authorization is invalid for external work. Authorized for version one to sent version two is invalid under the exact-version rule. Negative examples expose the assumptions that a happy-path diagram often leaves hidden.

E10-X1 Read the review event history

Use assets/review_events.json L1–L4. State the status after each event and whether any send is established. Treat L3 as matching the original R03 partner-list scope with source revision 1.

Submit: Four states and one execution conclusion.

11 Treat conditions as open requirements, not decorative comments

An owner can approve a release subject to a condition. That decision is not equivalent to unconditional permission. Q05's approval requires inventory_confirmed, while the evidence list is empty. The correct status is AWAITING_CONDITIONS. The owner has made a conditional decision, but the workflow lacks the evidence needed to cross the release boundary.

A condition should be concrete enough to verify. “Release when inventory is confirmed” needs a specified confirmation record or responsible role in a real workflow. The simplified course packet represents condition identifiers and matching evidence identifiers. An evidence entry is not invented automatically because the condition seems likely. Missing evidence remains missing, and the release stays blocked.

Separate three states: condition not yet checked, checked and unmet, and checked and satisfied. The packet's empty evidence means satisfaction has not been established; it does not prove inventory is unavailable. This distinction mirrors unsupported factual claims: absence of evidence is not proof of the opposite, but it is sufficient to withhold a permission that depends on positive evidence.

If a condition is ambiguous, clarify it before treating it as satisfied. “Proceed when ready” does not say who decides readiness or what readiness requires. A vague condition should not be interpreted by the drafting model as blanket discretion. Ask the owner to specify the condition or supply a bounded operational interpretation through the authorized process. The review system should show the ambiguity as a blocker.

Do not drop conditions when summarizing approval. A short status like approved can encourage premature execution if the condition lives only in a hidden comment. Show the relevant outstanding condition beside the action button or release record. An audit should be able to trace the eventual transition to the evidence that satisfied the condition, not only to the original approval timestamp.

Conditional approval can be efficient when the content and destination are fixed and the remaining fact is narrow. It is not a mechanism for authorizing any later content change. If the artifact or destination changes while a condition is being checked, the scope-binding rules still apply. The workflow must satisfy both exact authorization scope and conditions, rather than treating either as a substitute for the other.

For the Quay inventory_confirmed condition, this local policy assigns evidence supply to the inventory coordinator. The release owner remains the exhibit director. The inventory coordinator establishes the narrow inventory condition; that role does not acquire authority to approve a different public commitment or destination. Keep the evidence event and release decision separate even when both are needed for the same transition.

12 Design evidence packets that support real review

A reviewer should receive the smallest complete packet needed for the gate. For the Harbor fact gate, that means the source ID and revision, raw counts, candidate claim, calculation and artifact version. For the release gate, add the exact proposed message, destination, action, new commitment if any and factual-review status. Do not bury the decision among irrelevant model logs or unrelated project documents.

The source register includes both raw text and a machine-readable facts index. The index helps locate expected values, but reviewers should still be able to reproduce twelve plus eight from the raw H1 text. A copied expected_total field alone can hide an upstream error if both the candidate and checker use the same mistaken value. Independent checking means tracing to the underlying records, not merely comparing two cached summaries.

Evidence should be sufficient for an adverse decision as well as a pass. R02's mismatch record should show source twelve and eight, candidate twenty-two, derived twenty and the requested correction. R06's record should show that H2 states a date but no opening time. The reviewer can then explain why the next action is verification without guessing whether the missing claim is true in the real world.

Keep sensitive content out of routine queues unless necessary and authorized. The high-impact cases need only a routing trigger and a pointer to the authorized process in this exercise. Displaying imagined personal health or asset details would add exposure without improving the process-design answer. The course deliberately supplies no such details, so learners should not create them for realism.

A packet should also distinguish observed events from proposed next steps. “Owner approval required” is a requirement; “owner approved version one for partner-list email” is an event; “send after approval” is a proposed action. Mixing these in one notes column makes it easy to mistake a plan for completion. Use explicit fields or clearly labeled sentences so another operator can tell what evidence currently exists.

The review record does not need a sophisticated platform to be useful. A structured document or spreadsheet can work if IDs, versions, roles, decisions, conditions and sources are explicit. Tooling should support the gate design, not define it accidentally. A beautiful dashboard cannot compensate for an approval that lacks destination scope, and a plain table can be reliable when the rules and evidence are clear.

E12-X1 Prepare a factual evidence packet

For R02, supply the exact source locator, raw values, candidate value, independent calculation, discrepancy and allowed next action.

Submit: A six-field evidence packet.

An evidence packet can fail by omission even when every included item is correct. Suppose a release request contains R04's accurate attendance total but omits its proposed free-guide-kit commitment. The owner might approve the informational notice without ever seeing the resource promise. The approval should not be stretched to cover hidden content. The reviewed artifact must be the same substantive content that would be released, including additions that appear in a footer, subject line or attachment.

Attachments and links can therefore be part of the artifact scope. In this simplified packet, each case is one structured notice and no separate attachment is proposed. In a real workflow, adding a spreadsheet or replacing a linked document could change what the recipient receives even when the email body is unchanged. A gate specification should make that scope explicit rather than assume the body text represents the entire communication. This is a design consideration, not permission to share any additional files.

The same principle applies to a batch summary. If a request says “Approve the Quay notices” without identifying membership, a later-added case can accidentally inherit the earlier decision. A stable batch manifest can list artifact IDs, versions, source revisions and destinations. A reviewer then knows the exact set being authorized. The course avoids this

ambiguity by evaluating each Q case independently, which also makes the authorized-release count easy to reconcile.

Evidence freshness should be tied to the actual claim and action, not a generic age threshold. The current Q2 source explicitly identifies revision two and the current collection date. A reviewer should use that version because the packet declares it current, not because its date happens to be numerically largest. If two conflicting sources lacked an authority rule, the correct next step would be verification of which source governs. Choosing whichever one makes the draft pass would hide the conflict.

A compact handoff can preserve these distinctions with a simple statement: “Checked content: Q06-notice version 1. Source: Q1 revision 1. Destination/action: external:venue-contact, send_email. Owner: exhibit director. Decision: approve. Conditions: none. Execution: not recorded.” Every field has a role in determining the permitted state. The statement does not imply that a different notice, attachment or audience is covered merely because it belongs to the same project.

13 Avoid review theater, automation bias and rubber stamping

A gate can exist on paper while failing in practice. If a reviewer sees a confident model summary but not the source, they may approve fluent errors. If every request is marked urgent, reviewers may click through without checking. If the interface preselects approve, a nominal human decision may become a default action. The design should make the substantive review question easier to answer than to overlook.

This course does not measure real reviewer behavior. The following are design diagnostics, not claims that a particular team has failed. Ask whether the reviewer can see the source span, reproduce the calculation, identify the destination and distinguish pending from approved. Ask whether an unmet condition remains visible. If the information needed for the decision is missing, adding another approval checkbox does not solve the evidence problem.

Automation bias can appear when a score substitutes for source checking. A ninety-nine-percent model confidence does not establish that H1 totals twenty-two or that the director authorized a public commitment. Confidence concerns the model's estimate, not the organization's evidence or authority. The gate should require the relevant observable facts even when the model sounds certain.

Human authority can also be overstated. An analyst's fact check does not become a release approval because the analyst is experienced. A director's approval does not make a wrong total correct. A supervisor's instruction to hurry does not supply a missing source. Keeping each event tied to its question prevents social pressure from silently changing the gate's meaning.

Use review prompts that demand a concrete comparison. “Confirm the total against North and East counts” is better than “Does this look right?” “Approve this version for this partner list and action” is better than “Okay to proceed?” A reviewer can still make a mistake, but

the workflow at least presents the decision in a form that supports scrutiny and leaves a useful record.

Finally, preserve the option to say not enough evidence. A review system that allows only approve or reject can pressure reviewers to choose a false certainty. Pending clarification or request changes is a legitimate state when the task is incomplete. It should block the consequential action while allowing a precise explanation of what evidence would let the work continue.

14 Test routing rules with positive and negative cases

A routing design should be tested before people rely on it. Start with positive controls that should progress under the declared policy: a supported internal draft with matching analyst review, and a supported external artifact with matching owner approval and no unmet conditions. Then add negative controls that differ in one gate-relevant field. This makes it easier to see which boundary the rule actually enforces.

R08 is the worked positive release control. Change only its approval artifact_version from one to zero, and it should no longer authorize the current artifact. Restore that field and change destination, and authorization should again fail. Restore scope and set decision to pending, and it should fail for a different reason. Each mutation tests a specific requirement rather than merely generating another random bad case.

Test wrong reviewer role and unmet conditions separately. An approve event by the operations analyst does not satisfy the exhibit-director gate. A correct director approval with inventory_confirmed outstanding does not satisfy the conditions gate. If the routing function checks only whether approval exists, both errors would pass. Negative tests demonstrate why the detailed fields matter.

Test ordinary cases too. A design that blocks every case avoids unauthorized release but fails the useful-work requirement. R01 should remain a checked internal draft without unnecessary owner review. R08 should reach authorized release under the local policy. Proportionate gates need both correct blocking and correct progression; otherwise the workflow encourages people to bypass it because it never finishes legitimate work.

Do not confuse a routing test with execution. The verification script reads supplied JSON and calculates expected states; it does not send email, publish posts or make personal decisions. A PASS result establishes that the declared checks hold for these synthetic fixtures. It does not prove that a production integration authenticates reviewers, protects credentials or enforces access controls. Those would need separate evidence.

Document test scope in the handoff. Include the current policy version, case IDs, expected and observed routes, known limitations and the specific negative mutations. If policy changes, rerun relevant cases rather than assume an earlier green report covers a new authority model. A test suite is valuable because it makes the intended boundaries explicit and repeatable, not because it replaces judgment about new tasks.

E14-X1 Specify three negative approval tests

Start with R08 as the valid release case. Make three independent hypothetical mutations: approval decision pending; approval role operations analyst; approval destination public:other-page. State expected status and reason for each.

Submit: Three one-field mutation records.

Worked ID	Next gate	Current status
R01	CHECK_AND_INTERNAL	CHECKED_INTERNAL
R02	VERIFY_FIRST	AWAITING_VERIFICATION
R03	OWNER_APPROVAL	AWAITING_OWNER
R04	OWNER_APPROVAL	AWAITING_OWNER
R05	DO_NOT_AUTOMATE	HUMAN_PROCESS_ONLY
R06	VERIFY_FIRST	AWAITING_VERIFICATION
R07	VERIFY_FIRST	AWAITING_VERIFICATION
R08	OWNER_APPROVAL	AUTHORIZED_RELEASE
R09	OWNER_APPROVAL	AWAITING_OWNER
R10	VERIFY_FIRST	AWAITING_CONTEXT

15 Manage queues without turning urgency into permission

A queue helps organize pending work, but queue priority is not authorization. A director can need to review a time-sensitive announcement first without the announcement being approved already. An analyst can prioritize a disputed total without changing the source evidence. Keep urgency, ownership and gate status in separate fields so a high-priority label cannot accidentally unlock release.

Distinguish waiting for evidence from waiting for a decision. R06 waits for support for an opening time or a source-preserving revision. R03 waits for an owner decision on an already checked external notice. R10 waits for destination context. These cases may all look pending on a dashboard, but their next useful actions differ. A good queue makes the missing item visible and assigns the right resolver.

Do not invent service-level commitments or completion times. The fictional policy does not promise that the director will respond within an hour. A routing design can record when a request was made and how long it has waited, but it should not turn an observed delay into an authorized bypass. If a real organization wants escalation rules, those rules need explicit ownership and permission rather than an improvised exception.

A stale queue entry can also be misleading. If a corrected artifact replaces the version awaiting approval, the owner must see the current version and the earlier request should be marked superseded or updated according to the workflow. Approving the old version while a new one is sent creates a scope mismatch. Version tracking is therefore a practical queue-management tool, not just an archival concern.

For batch work, do not let one cleared item release the entire batch unless approval explicitly covers the batch's exact contents and destination. R08's valid approval does not authorize R09. A batch-level approval would need an identifiable membership and version so adding a new notice cannot silently inherit permission. The course cases use item-level approvals to make this boundary easy to inspect.

A useful pending-work summary says what can proceed and what cannot. For example: “The internal count summary is checked; the external notice awaits the exhibit director; the opening-time claim needs source evidence.” That is more actionable than “Three AI tasks pending.” It respects the user's attention while preserving the specific boundary that protects each task.

16 Independent capstone: Quay kit-dispatch communications

Quay is a fictional exhibition-loan program preparing kit-dispatch communications. The independent packet contains eight cases, Q01 through Q08, and the source register includes packing counts and a revised collection date. Use the same declared Harbor-Gates-1 role and approval policy for this exercise. Do not infer new authority from the different project name.

The raw Q1 source says seven kits were packed in the morning and five in the afternoon. The correct daily total must be derived from those values. Q2 is source revision two with a current collection date of December 9; its text also mentions the prior December 8 date. Distinguish the current fact from historical context. A source register can contain both without making both equally current for the candidate notice.

For each case, submit next_gate, current_status, trigger, reviewer_role, evidence_needed and allowed_next_action. Where multiple gates apply, add remaining_gates in order. Do not merely choose one of four labels and stop. Your routing record must explain what the next reviewer needs and what remains prohibited at the present state.

The packet deliberately includes a supported internal draft without factual review, an incorrect external total, a public commitment approved by the wrong role, a sensitive high-impact financial-eligibility request, a conditional approval lacking evidence, a fully matched approval, an approval bound to an old source revision and an internal-retention approval reused for external email. Each tests a different boundary. Equal-looking approve fields should not lead to equal decisions.

Provide a transition plan for Q02 after its arithmetic repair, Q05 after its condition is satisfied and Q07 after an owner reviews the current source revision. State which additional evidence would change each current status. Do not fabricate that evidence or report the changes as

already performed. A plan describes future permitted work; the current route must remain grounded in the supplied records.

Finally, report the gate counts and identify every case that is currently authorized for external release. Count statuses from your row-level ledger rather than guessing from how many approval objects exist. Include a statement that no emails, posts or financial decisions were executed as part of this exercise. The artifact is a reproducible gate design and case evaluation, not a real operational release.

E16-X1 Route the full Quay packet

Use assets/quay_capstone.json, source_register.json and gate_policy.json. Produce all eight routing rows, gate counts, current authorized-release IDs and the Q02/Q05/Q07 transition plans.

Submit: A complete routing ledger and bounded decision memo.

17 Write a gate decision memo that stays within the evidence

A final memo should make the next decision easy without hiding the record-level reasoning. Start with the policy version and packet scope. State how many cases are checked internal drafts, awaiting factual work, awaiting owner action, awaiting conditions or routed away from automation. Name the specific artifacts authorized for release, if any, and keep that state separate from completed execution.

Summaries should be traceable to the ledger. If the memo says one artifact is authorized, the reader should be able to find its matching factual review, owner role, approval tuple and satisfied conditions. If the memo says a high-impact case is routed to a human process, it should not also contain an automated eligibility recommendation. Consistency between the summary and detailed rows is part of the review design.

Explain the most important blocker in operational terms. “Q05 lacks evidence for inventory_confirmed” is more useful than “Q05 is risky.” “Q07 approval names source revision one while the current artifact uses revision two” tells the owner exactly what needs rechecking. Specific evidence gaps support action; vague risk language encourages either overreaction or dismissal.

The memo should also state what the checks do not establish. The supplied records do not authenticate real people, prove a production system's access controls or record actual message delivery. They provide fictional facts and decisions for testing the route logic. A local verification result supports that logic on these files, not a claim that the entire organization is compliant or that the AI system is universally safe.

When a later event arrives, update the affected row and then recompute the summary. Do not preserve an old authorized count after a destination change or a new condition. Likewise, do not keep a case blocked after valid matching evidence has arrived merely because its first route was a hold. Gates should respond to evidence under stable rules, not become permanent labels attached to a case.

The core habit is to ask exactly what has been established. Is the claim supported? Has the right person checked this version? Does the owner authorize this action for this destination? Are conditions satisfied? Has execution actually happened? Keeping those questions separate makes human review meaningful, proportionate and auditable instead of a decorative step in an otherwise automatic workflow.

Assessment

Choose one best answer per question. Questions 1–10 check the foundation concepts; questions 11–20 apply the extended study units. Complete and submit the practical tasks separately. A strong quiz score does not replace correct practical work.

1. Under the supplied routing policy, a checked internal summary is proposed for public posting. What remains necessary?

A. A higher model confidence score.

B. No further gate if names are absent.

C. Another arithmetic pass alone.

D. Release authorization for the public action.

2. H2 gives a date but no opening time; the draft says 09:00. What is the next route?

A. OWNER_APPROVAL as a substitute for evidence.

B. DO_NOT_AUTOMATE because every unknown is high impact.

C. CHECK_AND_INTERNAL because the claim is plausible.

D. VERIFY_FIRST because the time is unsupported.

3. A release request has decision pending. What has been established?

A. No owner approval has yet been recorded.

B. Approval of the current artifact only.

C. Approval unless an objection arrives.

D. Permission to send a smaller audience.

4. Under the supplied routing policy, a public post adds a free-kit commitment. Which gate addresses that authority?

A. Fact checking attendance totals alone.

B. Proofreading the promise for clarity.

C. A model rating the promise as helpful.

D. Responsible-owner approval of the exact commitment.

5. A request asks for a hiring recommendation from private health details. What should this workflow do?

A. Remove names and make the same recommendation.

B. Route to an authorized human process without an automated outcome.

C. Produce the ranking but label it provisional.

D. Seek a higher model confidence score.

6. Which gate description is operationally complete?

A. A named reviewer and a green checkbox.

B. A deadline and a destination.

C. Trigger, reviewer, evidence, decision and allowed next action.

D. A source link and a model score.

7. The operations analyst verifies 12 + 8 = 20. What authority does that event establish?

A. Approval for all later versions of the notice.

B. Factual review within the analyst role, not release permission.

C. Public release because the arithmetic is correct.

D. Any commitment based on the count.

8. A complete email is stored as a private draft. Has it been sent?

A. No; preparation and transmission are separate actions.

B. Yes, if its subject line is final.

C. Only if an owner approval is attached.

D. Yes, if all facts were checked.

9. An approved partner email is changed to a public post. What should happen?

A. Reassess authorization for the new destination and action.

B. Remove the source revision from the record.

C. Reuse approval because the text is unchanged.

D. Treat the public post as internal retention.

10. What does the course practical deliver?

A. Messages sent to the partner list.

B. Actual approvals on behalf of the exhibit director.

C. A routing design and synthetic case ledger.

D. Financial-eligibility decisions for real people.

11. Under the supplied routing policy, r07 has wrong arithmetic and an external destination. What is the immediate gate?

A. OWNER_APPROVAL with no later fact check.

B. CHECK_AND_INTERNAL because the draft is unfinished.

C. VERIFY_FIRST, with owner approval still required later.

D. DO_NOT_AUTOMATE solely because arithmetic is wrong.

12. R09 current version is 2; owner approval names version 1. What follows under this policy?

A. The analyst can rename the approval version.

B. The notice is already sent.

C. Owner approval is needed for the current version.

D. Approval transfers if the count is unchanged.

13. Q05 approval requires inventory_confirmed but evidence is empty. What is its status?

A. AUTHORIZED_RELEASE.

B. HUMAN_PROCESS_ONLY.

C. CHECKED_INTERNAL.

D. AWAITING_CONDITIONS.

14. R10 has a supported count and a null destination. What should the workflow do?

A. Hold for destination clarification.

B. Send to the last-used partner list.

C. Assume the internal team is intended.

D. Mark all release gates passed.

15. Which source supports the Quay packed total?

A. Q1 morning 7 plus afternoon 5.

B. The candidate claim of 13.

C. The number of approval objects.

D. The date in source Q2.

16. An analyst writes approve on a public commitment. The release owner is the director. What is the correct interpretation?

A. It counts because approve is explicit.

B. It counts because analysts know the data.

C. It replaces the factual review record.

D. It does not satisfy the release-owner gate.

17. Which event establishes that a message was actually transmitted?

A. A request for owner approval.

B. A recorded successful transmission event for that action

C. A matching owner approval.

D. A draft marked ready.

18. Why test R08 after changing only the approval destination?

A. To improve the source attendance data.

B. To make a new owner decision.

C. To isolate whether destination binding is enforced.

D. To prove all production security controls work.

19. Q07 uses current source revision 2 but approval references revision 1. What remains needed?

A. Use the older date to fit the approval.

B. Owner review matching the current source revision and artifact scope.

C. Assume a newer source is always covered.

D. Delete the current date so revisions no longer differ.

20. Why must a routing test include valid progression cases as well as blocked cases?

A. To replace human judgment entirely.

B. To show the design permits authorized work as well as blocking invalid actions.

C. To increase the number of approved items regardless of policy.

D. To avoid checking negative cases.

Sources and further reading

NIST AI Risk Management Framework
https://airc.nist.gov/airmf-resources/airmf/

NIST AI RMF Playbook: Govern
https://airc.nist.gov/airmf-resources/playbook/govern/