Give AI a SHIFT.

The delegation brief for getting finished work instead of another very articulate status update.

4 SYSTEMS + 5 WORKING PROMPTS

THE ACTUAL PROBLEM

Most prompts describe a topic. Real delegation describes a finish line.

"Research the market and make me a strategy."

That sounds clear because a human can quietly fill in the missing context. A system cannot know which sources count, what the artifact should look like, what it may change, or when it must stop.

SHIFT PREMISE

Do not ask for help. Define a shift of work with an artifact, a source boundary, a finish line, and a handback condition.

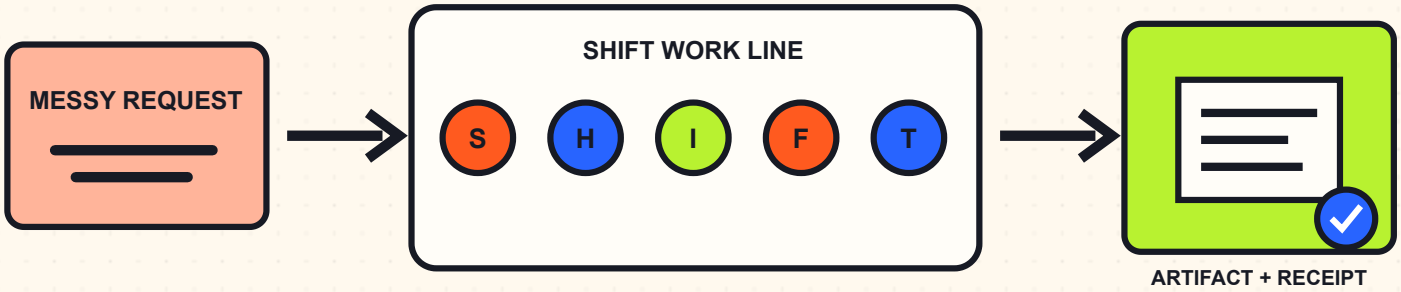
- **A named artifact**, not a vague outcome.
- **Approved evidence**, not ambient internet memory.
- **Visible constraints**, not assumed judgment.
- **Acceptance checks**, not "looks good."
- **A stop rule**, not unlimited autonomy.

SHIFT does not make a model infallible. It makes the work auditable, because you can see what it was told to make, which evidence it was allowed to use, and what "done" had to mean.

Framework language on this page is bytethewayyy's operating method, not a vendor capability claim.

SHIFT turns a request into a work contract.

S	H	I	F	T
SPECIFY	**HAND OVER**	**IDENTIFY**	**FIX**	**TRIGGER**
Name the exact artifact, audience, format, and destination.	List the sources, date window, access path, and source hierarchy.	Make boundaries, exclusions, permissions, and risks explicit.	Define objective checks that prove the artifact is finished.	State exactly when to pause, retry, escalate, or hand back.



OLD 3-PART BRIEF

Outcome maps to Specify + Fix.

OLD 3-PART BRIEF

Sources maps to Hand over.

OLD 3-PART BRIEF

Stop rule maps to Identify + Trigger.

THE UNIVERSAL PROMPT

Copy this before you specialize it.

SHIFT MASTER BRIEF

```
S | SPECIFY THE ARTIFACT
Create [exact artifact] for [specific audience and decision]. Deliver it as [file type, native app, dimensions, or
repository path] named [filename]. The artifact must contain [required sections] and should be written in [voice or
operating standard]. Save or return it at [destination].

H | HAND OVER THE SOURCES
Treat these as authoritative, in this order: (1) [source/path/date range], (2) [source/path/date range], (3) [approved
external sources]. Build a source map before drafting. Attach a source link, file path, or line reference to every
load-bearing factual claim. If sources disagree, preserve the conflict and trigger the handback rule. Do not fill gaps
from memory.

I | IDENTIFY CONSTRAINTS
You may [permitted actions]. You may not [publish, send, spend, delete, change credentials, modify production, contact
people, or expand scope]. Do not add packages, create accounts, alter source files, or infer sensitive facts unless I
explicitly authorize it. Keep changes limited to [scope].

F | FIX THE FINISH LINE
The work is complete only when: [objective check 1]; [objective check 2]; [objective check 3]; all links open;
calculations reconcile; required tests pass; and the final artifact has been reviewed against [template, rubric, or
examples]. Run the checks yourself, then report the result of each one. "Draft complete" is not a finish condition.

T | TRIGGER A HANDBACK
Pause and return a decision request if [source conflict], [missing permission], [material ambiguity], [risk
threshold], or [scope threshold]. Retry a failed reversible step no more than [N] times, changing one documented
variable per attempt. After the cap, stop. Otherwise continue until the artifact, verification results, and handback
receipt are complete.

HANDBACK RECEIPT
Return: artifact location; sources used; checks run and results; changes made; assumptions avoided; unresolved items;
approvals still required.
```

USE CONCRETE NOUNS

"A 10-slide launch brief" beats "a presentation." "The July 1 to July 14 support export" beats "customer feedback."

USE OBSERVABLE CHECKS

"Every claim has a source link" beats "make it accurate." "All repository tests pass" beats "make sure it works."

SAFETY WITHOUT BABYSITTING

Write the stop rule before you start the shift.

Trigger a handback when...

- [ ] Two authoritative sources conflict on a material fact.
- [ ] A required source is missing, inaccessible, or stale.
- [ ] The next action sends, publishes, spends, deletes, or changes access.
- [ ] The task would touch credentials, personal data, production, or legal language.
- [ ] The scope exceeds the named folders, files, records, or budget.
- [ ] A deterministic retry cap is exhausted.

Keep working when...

- [ ] A reversible formatting choice can be resolved from the supplied standard.
- [ ] A failed check has a clear, low-risk fix inside the approved scope.
- [ ] The source hierarchy already resolves a disagreement.
- [ ] A missing detail can be marked explicitly without inventing it.
- [ ] The system can verify the artifact using the checks you supplied.

Five acceptance checks that actually work

1
EXISTENCE
The named file exists at the named location.

2
COVERAGE
Every required section or case is present.

3
EVIDENCE
Claims point back to approved sources.

4
FUNCTION
Links, formulas, tests, or branches work.

5
BOUNDARIES
Nothing outside scope was changed.

Deterministic retry pattern: attempt counter starts at 0, one variable changes per attempt, the same check runs again, and the path stops after attempt 3 whether the work passed or failed.

CW

SYSTEM 01

ChatGPT Work: connected-app campaign brief

Best fit: a multi-source knowledge task that must end in polished native artifacts, not a chat summary. [1][2]

PASTE INTO CHATGPT WORK

```
S | SPECIFY
Create a campaign decision pack for Loomiq's AI Agent Operations offer. Deliver (1) a 10-slide Google Slides deck named
"AI Agent Operations - July Campaign Brief" and (2) a Google Sheet named "July Campaign Source Map." The audience is a
founder choosing one campaign angle for the next 30 days. Slides: decision; audience tension; 3 evidence-backed pains;
proposed promise; proof assets; channel plan; 3 content concepts; lead magnet; risks; next 7 actions. Use the visual
template in @Google Drive /Content OS Automation Engine/Brand/Templates.

H | HAND OVER
Use only: @Slack messages from #customer-signal and #sales from June 15 through July 15, 2026; @Google Drive /Content OS
Automation Engine/Research from the same date range; and @Gmail threads carrying the label "Agent Pilot." Build the source
map first with columns: claim, exact excerpt or datum, source URL, source date, confidence, deck location. Deduplicate
repeated claims. A claim appearing in multiple messages is not automatically true. Use approved web research only to
verify a claim already present in those sources, and link the primary source.

I | IDENTIFY
Read only. Do not send Slack messages or email, edit source files, publish, purchase media, create calendar events, or
change positioning outside the evidence. Do not invent customer quotes, metrics, market size, or urgency. Separate
verbatim excerpts from your synthesis.

F | FIX
Complete only when all 10 slides exist; every factual claim links to a source-map row; every recommended angle cites at
least 3 independent source items; the deck contains one clear decision rather than a list of options; no slide exceeds 45
words; all links open; spelling is checked; and the final recommendation includes one reason not to choose it.

T | TRIGGER
Pause if the Drive template is missing, fewer than 12 distinct source items exist, two authoritative sources materially
conflict, or a source contains sensitive customer information that should not appear in the deck. Otherwise continue
through source map, outline, deck, verification, and receipt. Do not ask for approval on reversible formatting choices
that the template resolves.

RECEIPT
Return both file links, source count by app, checks and results, material conflicts, excluded sensitive items, and the one
decision still requiring human approval.
```

Capability basis: OpenAI documents connected apps, long-running work, scheduled work, and native artifact creation. The delegation structure and acceptance checks are bytethewayyy's method. [1][2]

CC

SYSTEM 02

Claude Code: scoped bug fix with receipts

Best fit: repository work where the system can inspect code, edit files, run project commands, and report the diff. [3][4]

PASTE INTO CLAUDE CODE

```
Work in the current repository. Start in Plan Mode and inspect AGENTS.md, package.json, the nearest tests, and the
implementation path before proposing a change.

S | SPECIFY
Fix the dashboard defect where choosing a new date range updates the visible label but leaves the KPI request on the
previous range. The finished artifact is a minimal code patch plus a regression test. Preserve the current UI and API
contract.

H | HAND OVER
Treat repository code, repository instructions, existing tests, and package scripts as authoritative, in that order. Find
the date-range state owner, request-key construction, and any existing test for query invalidation. Cite file paths and
line numbers in the plan. Derive commands from package.json or project docs instead of assuming a package manager.

I | IDENTIFY
Do not add or upgrade packages, change environment files, touch database schema, rename public APIs, rewrite adjacent
state management, or modify more than 5 files. Do not use destructive git commands. Preserve user changes already in the
worktree. You may add one focused test helper only if an existing helper cannot express the case.

F | FIX
Before editing, reproduce the bug with a failing automated test that proves the request uses the stale range. Then make
the smallest coherent fix. Completion requires: regression test passes; relevant test suite passes; typecheck and lint
pass if the repository defines them; no snapshot changes unrelated to this defect; diff contains no debug output, disabled
checks, or broad formatting churn; and a final manual trace shows UI selection -> state -> request key -> API parameters
all carry the same range.

T | TRIGGER
Stop before editing if the defect cannot be reproduced, if fixing it requires an API or schema change, if repository
instructions conflict, if secrets appear in output, or if the patch would exceed 5 files. Run no more than 3 fix-and-test
cycles. In each cycle, record the failed assertion and change one causal hypothesis. After cycle 3, stop with evidence
instead of weakening the test.

HANDBACK RECEIPT
Return: root cause; files changed with one-line reasons; commands run and exact outcomes; before/after behavior; remaining
uncertainty; worktree items you intentionally did not touch. Do not commit unless explicitly asked.
```

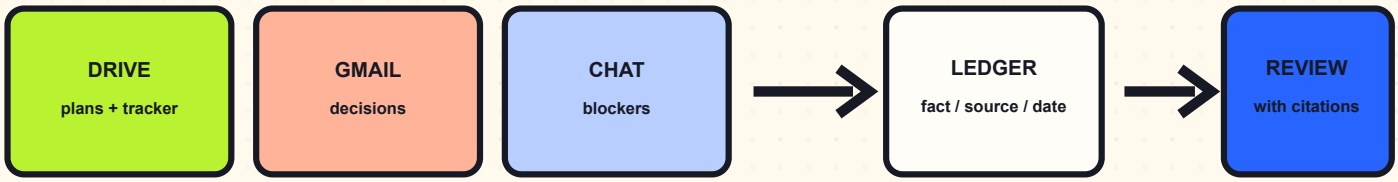
Capability basis: Anthropic documents repository inspection, edits, command execution, Plan Mode, tool controls, and bounded turns. The five-file scope and three-cycle cap are explicit operator choices. [3][4]

GM

SYSTEM 03

Gemini: cited operating review from Workspace

Best fit: synthesis across Google Workspace sources where every conclusion can point back to the listed source. [5][6]



PASTE INTO GEMINI

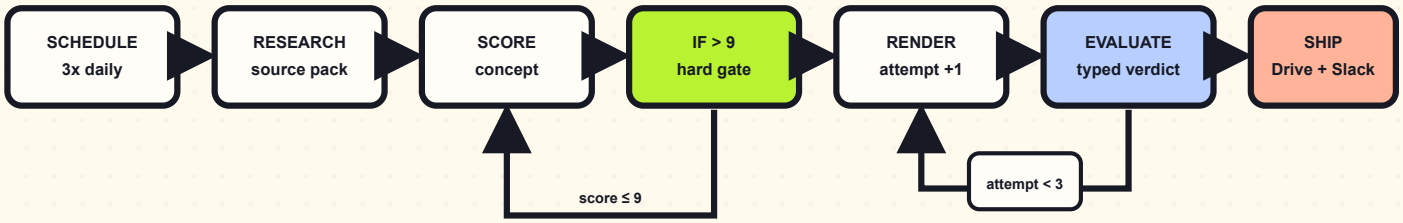
```
Select @Google Drive, @Gmail, and @Google Chat when prompted so the named Workspace sources are explicit.

S | SPECIFY
Create a weekly operating review for Loomiq covering July 6 through July 12, 2026. Generate a Google Doc named "Loomiq
Weekly Operating Review – 2026-07-12" with: executive readout; commitments made; work completed; missed commitments;
active blockers; decisions required; metric table; and next-week owner/action/date plan. The audience is the founder
making Monday priority decisions, so lead with exceptions and decisions, not activity volume.

H | HAND OVER
Use only: @Google Drive /Loomiq/Operating Plan/Q3 Plan, /Loomiq/Trackers/Weekly Metrics, and meeting notes dated July 6
through July 12; @Gmail threads from the same window with labels Client, Product, or Finance; and @Google Chat messages in
the Loomiq Ops space from the same window. Create an evidence ledger first with: statement, source title, source link,
source date, owner, and whether it is fact, commitment, or interpretation. Review the source links Gemini lists, and omit
any statement whose source cannot be opened.

I | IDENTIFY
Do not send email or Chat messages, change tasks or calendar events, edit source documents, infer intent, or turn a
tentative discussion into a commitment. Do not include personal information unless it is necessary to an operating
decision. Mark missing metric values as "not reported" rather than estimating them.

F | FIX
Complete only when every commitment has owner and date; every metric shows current value, prior value, target, and source;
every missed commitment has a cited status; duplicate updates are consolidated; interpretations are labeled; all links
open; the seven-day window is enforced; and the review ends with at most 5 decisions ordered by consequence and urgency.
Export the finished review to Drive and retain the evidence ledger as an appendix.

T | TRIGGER
Pause if a requested Workspace source is unavailable, dates or owners conflict, a metric definition changes during the
week, or more than 3 decisions lack a clear owner. Otherwise continue through ledger, synthesis, verification, document
generation, and receipt.

HANDBACK RECEIPT
Return the Doc link, source count by app, omitted unsupported statements, unresolved conflicts, checks run, and the exact
decisions requiring the founder.
```

Capability basis: Google documents explicit Workspace app selection, source review, and file generation or export. Its help pages also warn that generated content can be inaccurate, which is why this brief requires a source ledger. [5][6]

SYSTEM 04

n8n: deterministic content production loop

Best fit: recurring work where routing, retries, caps, error handling, and side effects must be visible as nodes. [7][8][9]



PASTE INTO N8N AI WORKFLOW BUILDER

Build a visible, deterministic n8n workflow named "Content OS – Research to Render – v1". AI may generate research, concepts, captions, and evaluations, but AI must not control routing, retry counts, approval boundaries, or daily caps. Do not use an AI Agent node for flow control.

S | SPECIFY
Run at 08:00, 13:00, and 18:00 America/Chicago. Produce up to 3 distinct 9:16 reel renders per calendar day, then stop. Save each research packet, concept evaluation, prompt, render, and final receipt under Google Drive /Content OS Automation Engine/YYYY-MM-DD/. Post the final MP4, caption, score breakdown, and Drive link to Slack only after all gates pass. Create drafts only. Never publish to a social platform.

H | HAND OVER
Add a Config node containing timezone, score_threshold=9.0, max_daily_renders=3, max_attempts=3, target_width=1080, target_height=1920, and Drive folder path. Research through Perplexity with primary links, then verify load-bearing claims through OpenAI or Gemini. Normalize output to strict JSON: concept_id, topic, audience_pain, promise, hook, storyline[6], visual_plan[6], caption, lead_magnet_keyword, sources[], novelty_score, value_score, evidence_score, visual_score, total_score. Reject malformed JSON before scoring.

I | IDENTIFY
Use explicit Schedule Trigger, Edit Fields, HTTP Request, Code, If, Switch, Loop Over Items, Wait, Google Drive, Slack, and Error Trigger nodes. Credentials must use n8n credentials or environment variables, never literal secrets. Add an idempotency key of date + concept_id + stage. Do not overwrite existing Drive files, send duplicate Slack messages, expose credentials in logs, or continue after a cap. Keep generation and evaluation prompts in named Edit Fields nodes so they are inspectable.

F | FIX
Gate 1: total_score must be strictly greater than 9.0. Gate 2: daily successful_render_count must be below 3. Gate 3: render must be 1080x1920, 24–30 seconds, decodable, non-black, and have no frozen segment longer than 1.5 seconds. Gate 4: evaluator returns exactly pass, render_fail, or concept_fail plus reasons. pass -> Drive + Slack; render_fail -> increment attempt and rerender with one prompt variable changed; concept_fail -> increment attempt and return to research for a different concept. The same attempt counter applies across both branches and stops at 3. After every side effect, write status and idempotency key to the tracker.

T | TRIGGER
Use If and Switch nodes for every branch. Stop when successful_render_count=3, attempt_count=3, required credentials are missing, verification finds a material unsupported claim, or Drive/Slack returns a non-retryable error. Error workflow must record execution ID, node, status code, concept_id, attempt, and next safe action. No branch may loop without incrementing a visible integer counter.

RECEIPT
End each execution with JSON containing execution_id, concepts_researched, gate_failures, renders_attempted, renders_shipped, Drive URLs, Slack timestamps, source URLs, retry history, and final stop_reason. Add node notes that map each cluster to S, H, I, F, or T.

Make the logic visible enough to audit in 30 seconds.

Control plane

Schedule Trigger
Three fixed runs.

Config
Thresholds and caps.

Daily Counter
Persistent success count.

Idempotency Check
No duplicate side effects.

Work plane

Research HTTP
Source packet.

Verify HTTP
Claim check.

Code: Normalize
Strict schema.

Render HTTP
ComfyUI or renderer.

Proof plane

Media Probe
Size, duration, decode.

Evaluate HTTP
Typed verdict.

Drive + Slack
Artifacts and notice.

Execution Receipt
What happened.

Exact branch table

CONDITION	NEXT NODE	COUNTER	ALLOWED SIDE EFFECT
score ≤ 9.0	New concept or Stop	concept_attempt + 1	Write evaluation only
score > 9.0 and daily < 3	Render	render_attempt + 1	Create local render
verdict = render_fail and attempt < 3	Tweak one render variable	same attempt + 1	Replace failed draft only
verdict = concept_fail and attempt < 3	Research different concept	same attempt + 1	Write new research packet
verdict = pass	Drive, then Slack	daily_success + 1	Upload and notify
attempt = 3 or daily = 3	Stop with receipt	none	Write stop reason only

One counter, both failure branches. Otherwise a concept can fail three times, switch branches, and quietly earn three more render attempts.

n8n documents explicit loop, branch, error, and execution primitives. The node arrangement above is the SHIFT implementation for the Content OS requirement. [7][8][9]

CHOOSE THE RIGHT SHIFT

Same framework, different execution surface.

SYSTEM	USE IT WHEN	SOURCE HANDOFF	FINISH PROOF	CRITICAL STOP RULE
ChatGPT Work ARTIFACTS	Several apps feed one polished decision artifact.	Name apps, folders, channels, labels, and dates.	Native file links, source map, link check, content rubric.	Stop before send, publish, spend, or positioning change.
Claude Code REPOSITORIES	The work lives in code and can be tested.	Repository rules, implementation, tests, and scripts.	Reproduction, passing tests, typecheck, narrow diff, receipt.	Stop on scope expansion, secrets, schema, or package changes.
Gemini WORKSPACE	Gmail, Drive, and Chat contain the operating record.	Explicit Workspace apps, source titles, date windows.	Evidence ledger, open source links, complete owners and dates.	Stop on inaccessible sources, conflicting owners, or metric definitions.
n8n RECURRING	The process repeats and needs visible control flow.	Config node, schemas, credentials, and source APIs.	Execution data, counters, files, IDs, and error workflow.	Stop on caps, non-retryable errors, or missing credentials.

Use a conversational system when...

The work benefits from synthesis, editorial judgment, and iterative artifact creation, while the user can inspect the sources and final file.

Use a workflow system when...

The same inputs and branches repeat, side effects matter, and a human should be able to inspect counters, errors, and execution history without reading a model's reasoning.

The framework stays stable. What changes is the proof surface: links for knowledge work, tests for code, ledgers for Workspace synthesis, and execution receipts for automation.

Why the short version breaks.

GENERIC

"Research our customers and make a content strategy with viral ideas."

- ☐ No artifact or audience.
- ☐ No source boundary or date window.
- ☐ "Viral" has no measurable acceptance check.
- ☐ No permission boundary.
- ☐ No stop or handback condition.

SHIFTED

Create a 6-week content decision sheet for a founder choosing what to produce, with 18 concepts grouped by customer pain and 3 fully scripted pilots.

Use only the last 45 days of saved customer calls, support threads, and posts that exceeded each creator's 90-day median views by at least 2x. Link every source and keep observed performance separate from interpretation.

Do not claim an idea will go viral, fabricate performance, publish, or contact anyone.

Complete when each concept has one pain, one outcome, one source, one visual mechanism, one lead-magnet CTA, and a score with written evidence.

Stop when fewer than 8 distinct customer pains exist, evidence conflicts materially, or the third research pass still produces no concept above 9.0.

Three upgrades that create most of the difference

01

REPLACE ADJECTIVES

"High quality" becomes named sections, dimensions, files, tests, source counts, or word limits.

02

EXPOSE JUDGMENT

Ask the system to separate evidence, interpretation, recommendation, and uncertainty.

03

CAP PERSISTENCE

Specify a retry counter and what changes, then require a failed receipt after the cap.

Do not write "keep trying until it works." Write what counts as a reversible retry, how many attempts exist, what variable may change, and what evidence must return when it still fails.

Do not accept "done." Ask for the receipt.

1. ARTIFACT
Exact file, URL, repository path, execution ID, or draft location.

2. SOURCES
What was used, what was excluded, and where each load-bearing fact came from.

3. CHANGES
Files, records, nodes, settings, or sections created or modified.

4. CHECKS
Each acceptance check, how it ran, and its actual result.

5. RETRIES
Attempt count, causal hypothesis, single variable changed, and outcome.

6. BOUNDARIES
What the system deliberately did not touch because it was out of scope.

7. UNCERTAINTY
Unsupported statements, missing evidence, unresolved conflicts, and residual risk.

8. DECISION
The one human approval or judgment still required, phrased as a specific choice.

FINAL QA PROMPT

```
Audit the completed artifact against the SHIFT brief before handing it back. Create a table with one row for every explicit requirement and columns for: requirement; evidence location; pass/fail; exact defect; corrective action. Do not mark a requirement passed because the artifact looks plausible. Open links, inspect file paths, run defined tests, reconcile calculations, and verify that no excluded scope changed. Correct reversible failures within scope, rerun the same check, and record the attempt. After the retry cap, stop and return the failed receipt. End with exactly one status: PASS, PASS WITH DISCLOSED LIMITATION, or FAIL.
```

The receipt is not extra paperwork. It is the artifact that tells you whether supervision can safely decrease next time.

SOURCES + FIELD NOTES

What is documented, and what is SHIFT.

Primary sources read for this guide

1. **[1] OpenAI, ChatGPT Work.** Connected apps, longer-running work, scheduled tasks, and artifact creation. openai.com/index/chatgpt-for-your-most-ambitious-work/
2. **[2] OpenAI Help, creating and editing files.** Native document, spreadsheet, and presentation outputs when the relevant app is enabled. help.openai.com/en/articles/20001278
3. **[3] Anthropic, Claude Code CLI reference.** Plan permission mode, tool controls, max turns, and structured output. docs.anthropic.com/en/docs/claude-code/cli-usage
4. **[4] Anthropic, Claude Code common workflows.** Repository context, edits, testing, debugging, and git-oriented work. docs.anthropic.com/en/docs/claude-code/common-tasks
5. **[5] Google, Gemini connected Workspace apps.** Explicit app selection, supported source retrieval, listed-source review, and documented limitations. support.google.com/gemini/answer/14959807
6. **[6] Google, creating files in Gemini.** Supported generated file types and Drive export. support.google.com/gemini/answer/13275745
7. **[7] n8n Docs, Loop Over Items.** Batch iteration and loop behavior. docs.n8n.io/.../splitinbatches/
8. **[8] n8n Docs, error handling.** Error workflows and Error Trigger. docs.n8n.io/flow-logic/error-handling/
9. **[9] n8n Docs, workflow executions.** Execution history and retrying failed workflows. docs.n8n.io/workflows/executions/all-executions/

Documented capability

Statements about what a product can access, create, run, or expose come from the vendor sources listed here and are bounded by account, app, permission, and product availability.

SHIFT framing

The five-part method, prompt structures, stop rules, acceptance checks, node arrangement, receipts, and examples are bytethewayyy's operating framework. They are recommendations, not vendor guarantees.

Your next move

Choose one real task, fill the master brief, and keep the first receipt. The second shift should reuse what the first receipt taught you about sources, boundaries, and checks.

KEEP THE FIELD GUIDE

Save this. Then make your AI clock in.

Prepared July 15, 2026. Product behavior can change, so verify current account capabilities against the linked official documentation before operational use.