PUBLIC SAMPLE FORMAT

Agent Build Spec

A build-ready specification for one custom AI agent: the job it owns, tools it can use, records it can read, outputs it drafts, and review rules it must obey.

■ SAMPLE ONLY - NOT A CLIENT SPEC

AUDIENCE	PURPOSE	OUTPUT
Client owner and build team	**Make the agent buildable**	**Job, tools, prompts, logs, evals**

WHAT THE AGENT IS RESPONSIBLE FOR

The agent job

This sample assumes one focused workflow has already been selected. The spec turns that workflow into a practical build plan the client and builder can review.

Spec rule: the agent should have a narrow job, named inputs, named outputs, limited tool access, observable logs, and clear rules for when it must stop and ask a human.

AGENT JOB

Prepare one repeated workflow

The agent gathers approved context, summarizes the state, drafts the next action, and presents it for review.

INPUTS

Only trusted source records

The spec names the exact forms, records, tickets, orders, docs, sheets, or systems the agent can read.

OUTPUTS

Drafts, summaries, and tasks

The agent produces reviewable work, not irreversible decisions. Every output should show what sources it used.

EXCLUSIONS

What the agent must not do

The spec blocks final sends, money movement, public publishing, and regulated-data handling unless explicitly approved in scope.

Spec fields we confirm

TOOLS

The exact systems the agent can read from or draft into, with read and write permissions separated.

REVIEW RULES

The events that require human approval, escalation, source checks, or a hard stop.

EVALS

Small tests that show whether the agent follows instructions, uses sources, and flags missing context.

WHAT THE AGENT CAN TOUCH

Tool and data access

The spec draws a bright line between read-only context gathering, draft creation, and any action that could affect a customer, account, order, record, or public page.

ACCESS POSTURE

Start read-only wherever possible

The first version should read approved context, draft reviewable outputs, and log its reasoning. Write actions come later, only after the workflow is trusted.

ALLOWED

Safe first permissions

- Read named records and docs
- Draft responses, briefs, and tasks
- Attach source links and confidence flags
- Log what happened after each run

BLOCKED

Actions held for review

- Final customer messages
- Refunds, order edits, and invoices
- Public publishing and campaign sends
- Legal, medical, tax, or regulated decisions

WHAT WOULD GO INTO THE BUILD

Agent build spec components

The build spec keeps the agent narrow, testable, and reviewable before it enters a real business workflow.

01 **JOB STATEMENT.** Define the one workflow the agent helps with, the output it prepares, and the human who owns approval.

02 **TOOL CONTRACTS.** List each tool, the permission level, the fields available, and what the agent can never change.

03 **PROMPT STACK.** Capture the agent role, source rules, tone rules, escalation rules, output format, and refusal behavior.

04 **LOGS AND EVALS.** Define what gets recorded, how outputs are checked, and which test cases must pass before launch.

05 **LAUNCH CHECKLIST.** Confirm access, test data, human reviewer, rollback plan, monitoring cadence, and post-launch owner.

The point: a strong spec prevents vague agent builds. Everyone can see what the agent does, where it stops, and how it will be judged.

BUILD RISKS

Risk flags we would call out

A useful build spec is honest about anything that could make the agent unreliable, unsafe, hard to monitor, or hard for the team to trust.

UNCLEAR JOB

The agent owns too much

If the job statement includes several unrelated workflows, the first build should be split before implementation starts.

MISSING EVALS

No tests means no trust

The spec should include sample inputs, expected behavior, bad cases, and pass rules before any live workflow depends on the agent.

PERMISSIONS

Access is too broad

The safest spec starts narrow. Broad write permissions create risk before the team has evidence the agent behaves correctly.

HANDOFF

No owner after launch

If no one will review outputs, mark corrections, and watch the logs, the build needs a simpler launch plan.

What the client gets next

BRIEF	A written implementation spec that names the agent job, sources, tools, prompts, outputs, logs, evals, and human approval rules.
BUILD	If approved, the spec becomes the working plan for implementation, test cases, access setup, and client review.
LAUNCH	The first live version starts with limited access, visible logs, human review, and optional post-launch support.