

PROJECT 042 - CLIENT BRIEF

AI Agent + Tools + Human Approval System | Business Simulation

Project Type	AI Operations / Human-in-the-Loop Automation
Core Stack	n8n • PostgreSQL • Slack • LLM / AI Agent
Primary Users	Operations team, managers, supervisors
Business Goal	Give teams fast AI-assisted access to operational data while keeping consequential database changes under human control

CLIENT SCENARIO

A growing operations team manages customer and order information in PostgreSQL. Staff frequently need to check order status, retrieve customer details, and answer operational questions. Managers also receive requests to change important order states such as moving an order from Processing to Ready for Dispatch or Delivered.

The client wants employees to interact with these systems through a simple AI assistant instead of manually searching database records. However, the business does not want an AI agent to make consequential database changes without authorization.

BUSINESS PROBLEM

- Operational information is stored in structured systems but requires manual searching or technical access.
- Routine lookup requests consume staff time and slow down responses.
- An unrestricted AI agent could create operational risk if it directly modifies order records.
- Managers need a clear approval checkpoint for protected actions.
- The business needs traceability showing what was requested, who approved or rejected it, what action was executed, and the final result.

CLIENT REQUIREMENTS

Requirement	Expected Behavior
Conversational access	Staff can ask natural-language questions about orders, customers, and operations.
AI tool selection	The agent chooses the correct lookup or intelligence tool based on the user's request.
Protected writes	Order-status changes cannot be executed directly by the AI agent.
Human approval	Protected changes create a Pending approval and are sent to Slack for Approve / Reject.
Validation	The system confirms that an approval is still Pending before executing a decision.
Auditability	Approved and rejected outcomes are recorded with request and execution context.

PROPOSED SOLUTION

Build a two-workflow AI operations system. Workflow 01 acts as the conversational AI layer, using PostgreSQL tools for order lookup, customer lookup, and operations intelligence. When a user requests a protected order-status change, the agent creates a Pending approval rather than modifying the database.

Workflow 02 acts as the deterministic authorization layer. It validates the approval record, sends an interactive Slack approval card, receives the human decision, validates the request again, routes the decision, executes only the authorized outcome, sends confirmation, and writes the result to the audit log.

SUCCESS CRITERIA

- Successful implementation means:
- Staff can retrieve operational data through natural language.
 - The AI agent uses the correct business tools instead of inventing answers.
 - Protected database updates always require human authorization.
 - Approval and rejection paths behave deterministically.
 - The order remains unchanged while approval is Pending or when a request is Rejected.
 - Approved changes are executed and confirmed.
 - Every consequential action produces an auditable record.

DELIVERABLES

- Workflow 01 - Main AI Operations Agent with reasoning model, conversation memory, and PostgreSQL tools.
- Workflow 02 - Human Approval & Order Update workflow with Slack interaction, validation, controlled execution, and audit logging.
- PostgreSQL approval and activity-log records supporting traceability.
- Tested order lookup, customer/operations retrieval capability, approval creation, Approved branch, and Rejected branch.

- Portfolio documentation including workflow screenshots, case study, workflow notes, architecture/future roadmap, and client brief.

BUSINESS VALUE

The system combines the speed and flexibility of an AI agent with deterministic controls around sensitive actions. Employees receive faster access to operational information, managers retain authority over consequential changes, and the organization gains a structured audit trail rather than relying on untracked manual decisions.

SCOPE & PRODUCTION CONSIDERATIONS

This project is a portfolio/client simulation using representative business data. A production deployment would additionally require organization-specific access controls, authenticated webhooks, Slack permission design, least-privilege database credentials, error and retry handling, monitoring, data-retention rules, and separate development/staging/production environments.