

HR Recruitment Automation Case Study

A seven-workflow n8n system for job intake, resume screening, human approval, interview booking, reporting, and operational monitoring

Designed and built independently by John Rey Cretesio

Category	Details
Project type	Independent portfolio demonstration
Role	Architecture, workflow development, database design, integrations, testing, and documentation
Platform	Self-hosted n8n, PostgreSQL, and Google Workspace
Reviewed scale	7 coordinated workflows and 195 nodes
Current status	Portfolio-ready demonstration; not certified for live applicant data

This case study explains how I designed the workflows as one coordinated recruitment system, how I kept AI recommendations subordinate to human decisions, and how I handled duplicate processing, state changes, external-service failures, reporting, and operational alerts.

Project Overview

Recruitment involves more than one isolated task. A usable process must connect job requirements, incoming applications, screening decisions, candidate communication, interview availability, calendar events, reporting, and exception handling. Disconnected inboxes and spreadsheets force HR staff to copy information repeatedly, verify status manually, and repair partial failures.

I designed this portfolio system as one coordinated workflow architecture. PostgreSQL serves as the shared system of record, while n8n orchestrates Gmail, Google Drive, Google Calendar, Google Sheets, OpenAI-assisted extraction, deterministic business rules, and human approval steps.

The language model does not make the final hiring decision. It produces structured evidence, deterministic logic calculates the weighted score, and an authorized HR reviewer makes the final shortlist or rejection decision.

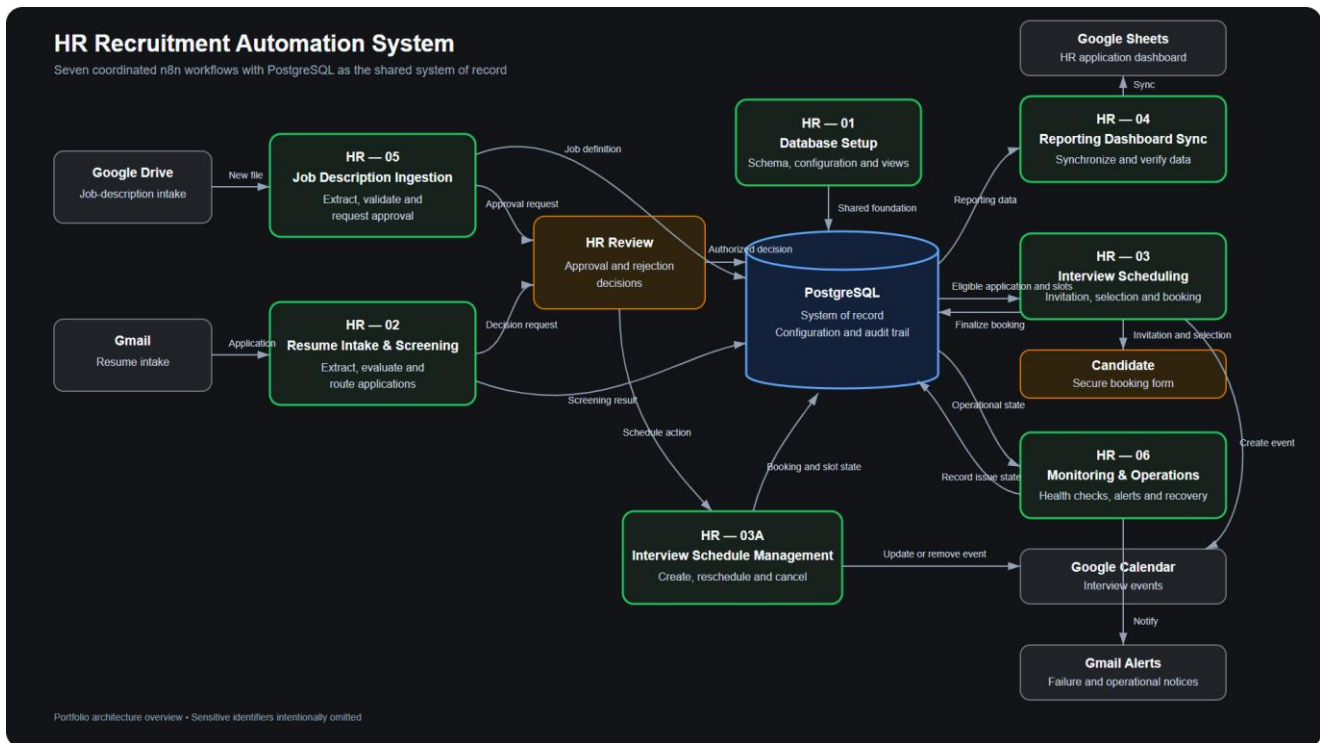


Figure 1. Seven n8n workflows coordinate intake, screening, scheduling, reporting, and monitoring through a shared PostgreSQL foundation.

Recruitment Process Challenges

- Convert job-description documents into consistent, versioned selection criteria.
- Reject invalid, unreadable, incomplete, and duplicate submissions before unnecessary AI processing.
- Keep automated screening explainable and subordinate to human judgment.
- Keep PostgreSQL and Google Calendar aligned when one service succeeds and another fails.
- Give HR a useful reporting view and actionable alerts instead of silent workflow failures.

The main technical problem was maintaining reliable state across asynchronous email, file, database, webhook, calendar, and spreadsheet operations.

Workflow Architecture

The latest reviewed exports contain the following workflow inventory.

ID	Workflow	Nodes	State
HR 01	Database Setup	18	Manual deployment
HR 02	Resume Intake and Screening	45	Active
HR 03	Interview Scheduling	23	Active
HR 03A	Interview Schedule Management	45	Active
HR 04	Reporting Dashboard Sync	10	Active
HR 05	Job Description Ingestion	40	Active
HR 06	Monitoring and Operations	14	Active

Database and Configuration Foundation

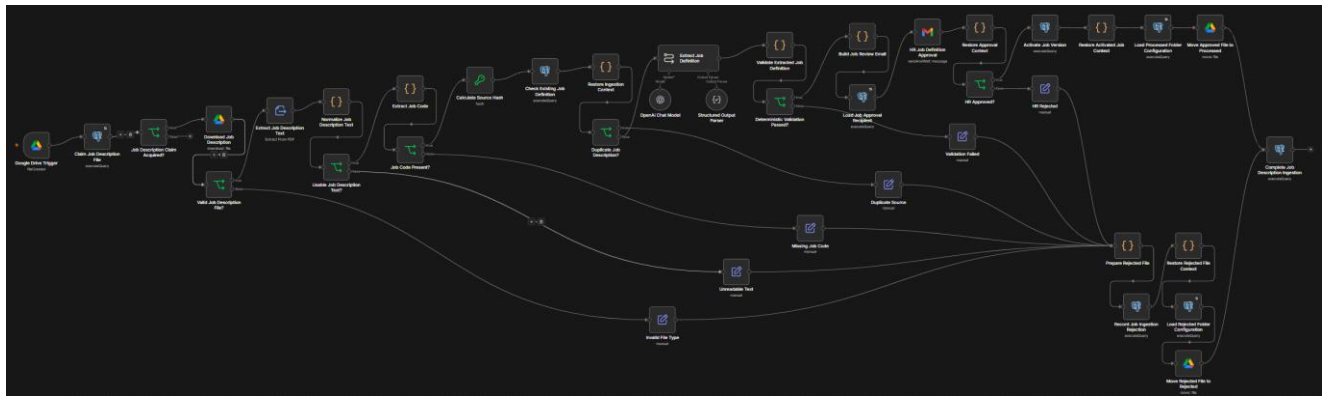
HR 01 Database Setup is a manual deployment workflow with 18 nodes. It creates the shared tables, indexes, views, workflow configuration, intake-claim tables, authorized HR-user records, operational-alert state, scheduling structures, and append-only audit configuration.

Non-secret deployment values such as the business timezone, recipient addresses, Calendar identifier, reporting spreadsheet identifier, reservation duration, and alert cooldown are loaded from PostgreSQL instead of being repeated across operational workflows. The incoming Drive folder remains trigger-level configuration because the trigger must know what to monitor before runtime database queries can execute.

Job Description Ingestion and Versioning

HR 05 Job Description Ingestion watches a configured Google Drive folder. Its 40 nodes implement a guarded ingestion pipeline.

1. Claim the file before expensive processing begins.
2. Validate the file and extract selectable text.
3. Normalize the content and identify the job code.
4. Calculate a source hash and check for an existing definition.
5. Use structured AI extraction to prepare job criteria.
6. Apply deterministic validation to the extracted structure.
7. Send a human approval request.
8. Activate the approved version and move the file to the processed folder.
9. Route invalid, unreadable, duplicate, rejected, or incomplete files through a recorded rejection path.



Resume Intake Screening and Human Review

HR 02 Resume Intake and Screening is the largest workflow, with 45 nodes. It starts from Gmail and claims the message before downloading and processing the attachment. This prevents retryable or duplicate executions from repeating extraction and AI evaluation unnecessarily.

The workflow validates the attachment, extracts text, identifies the job code, loads the active job version, checks for an existing application, and classifies the submission as new, duplicate, or requiring reapplication review. Invalid file types, unreadable resumes, missing job codes, and inactive job codes receive explicit responses.

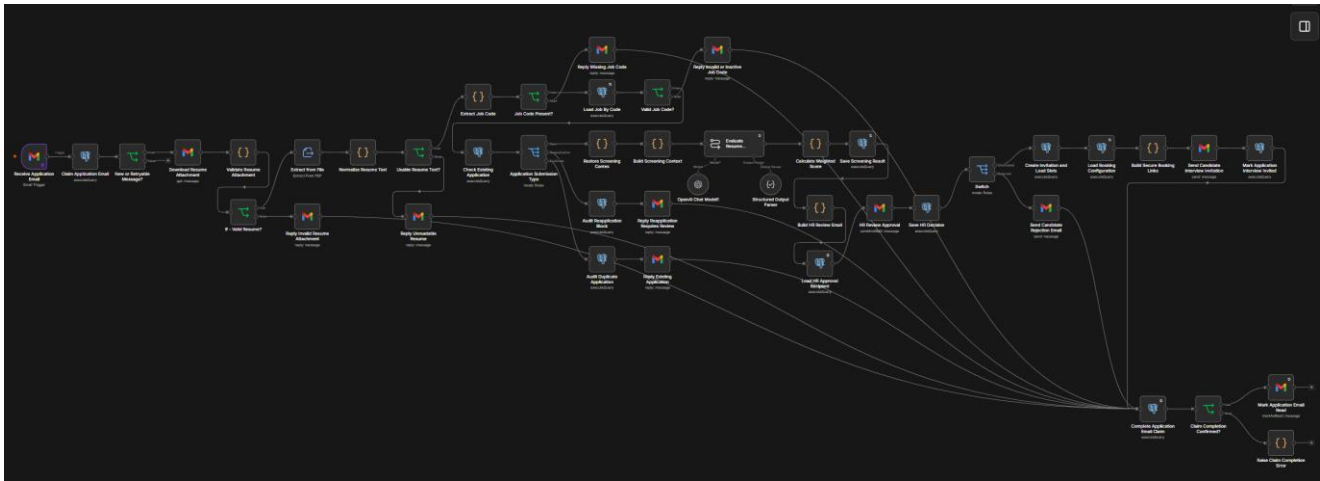


Figure 3. Resume intake includes validation, duplicate controls, structured evaluation, human review, candidate communication, and claim-completion handling.

For valid new submissions, the AI stage produces structured evidence against the active job criteria. A separate code step calculates the weighted score. This division keeps evidence identification separate from deterministic scoring rules.

Weighted evaluation rubric

Deterministic scoring based on the active job-version criteria.

Criterion	Weight	Rating	Points	Evidence
n8n Workflow Development	30%	5 / 5	30	• Designed and maintained more than 25 multi-step n8n workflows. • HR automation system using n8n and PostgreSQL
API and Webhook Integration	20%	5 / 5	20	• Built REST API and webhook integrations for internal business applications. • Created API requests and transformed JSON responses.
JavaScript and Node.js	15%	5 / 5	15	• Developed Node.js applications

Figure 4. The review output presents criterion weights, ratings, awarded points, and resume evidence instead of only an opaque score.

The final review states that the system recommendation is advisory. HR inspects the evidence and chooses Reject or Shortlist.

actions.

Ambiguities and review notes

- No direct examples of specific authentication methods (e.g., OAuth flows) used for APIs—resume states REST/API integration but not auth details.
- No code samples or links to repositories to verify complexity of SQL transactions or JavaScript implementations.
- Scale and production environment details (deployment, concurrency load) are not fully quantified.

Final decision required

Review the evidence and select the final HR decision below. The system recommendation is advisory and does not replace human judgment.

Generated by the HR Recruitment Automation portfolio system.

Reject

Shortlist

Figure 5. Human approval remains the final decision point after automated evidence extraction and scoring.

Candidate Interview Selection and Booking

After HR shortlists an application, the system creates an expiring invitation, loads eligible interview slots, and emails secure booking options to the candidate.

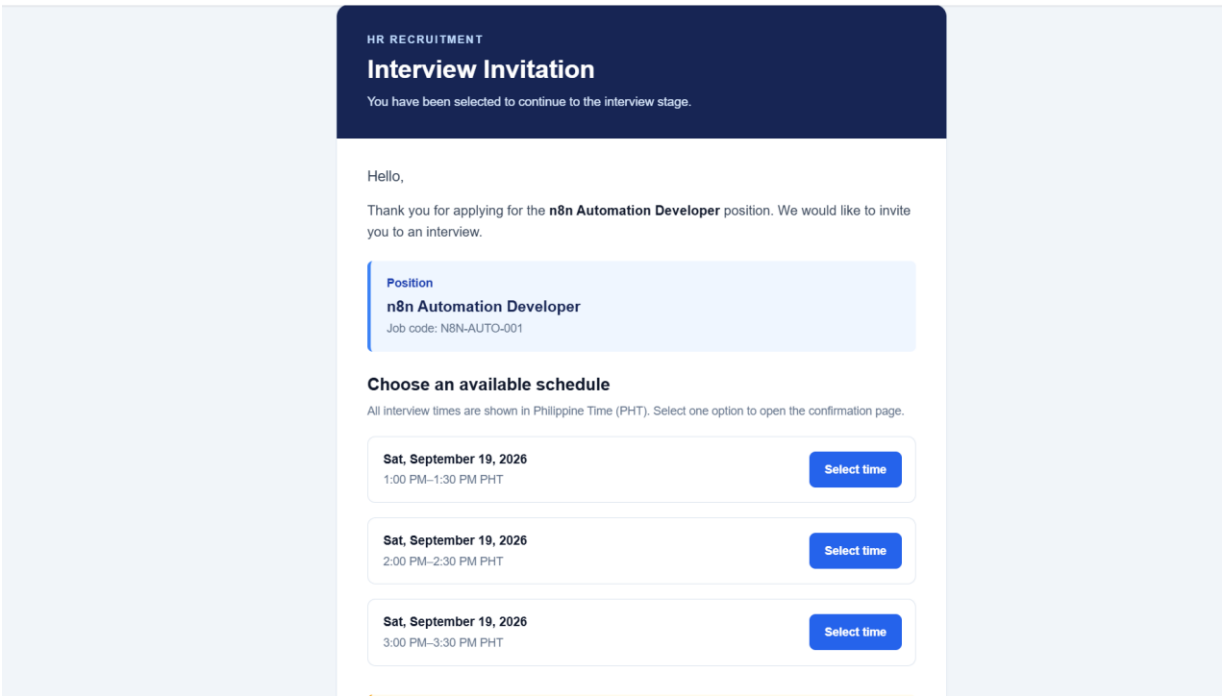


Figure 6. The candidate sees only available interview times in the configured Philippine timezone.

Selecting a time opens a confirmation page through a read-only GET request. The reservation occurs only after the candidate presses Confirm interview, which submits a POST request. Link previews and automated scanners therefore cannot change booking state merely by opening the URL.

Confirm your interview

Please review your selected schedule before confirming.

Position: n8n Automation Developer

Date and start: September 19, 2026 at 02:00 PM

End: 02:30 PM

Timezone: Asia/Manila

Confirm interview

Your schedule is not reserved until you press "Confirm interview."

Figure 7. The confirmation step separates viewing a proposed slot from committing the booking.

HR 03 Interview Scheduling uses PostgreSQL state predicates and locking to reserve a slot atomically. It creates a Google Calendar event with a deterministic identifier and then finalizes the booking in PostgreSQL. If Calendar creation or database finalization fails, compensation paths release the reservation or remove the unfinalized event.

This is compensation-based consistency rather than a distributed transaction. It reduces the likelihood of abandoned reservations and orphaned external events while acknowledging the boundary between PostgreSQL and Google services.

HR Controlled Schedule Management

HR 03A Interview Schedule Management provides controlled actions for adding availability, cancelling available slots, rescheduling booked interviews, and cancelling interviews. Its 45 nodes combine form input, authorization checks, PostgreSQL mutations, Calendar operations, verification, rollback, scheduled cleanup, and manual-review outcomes.

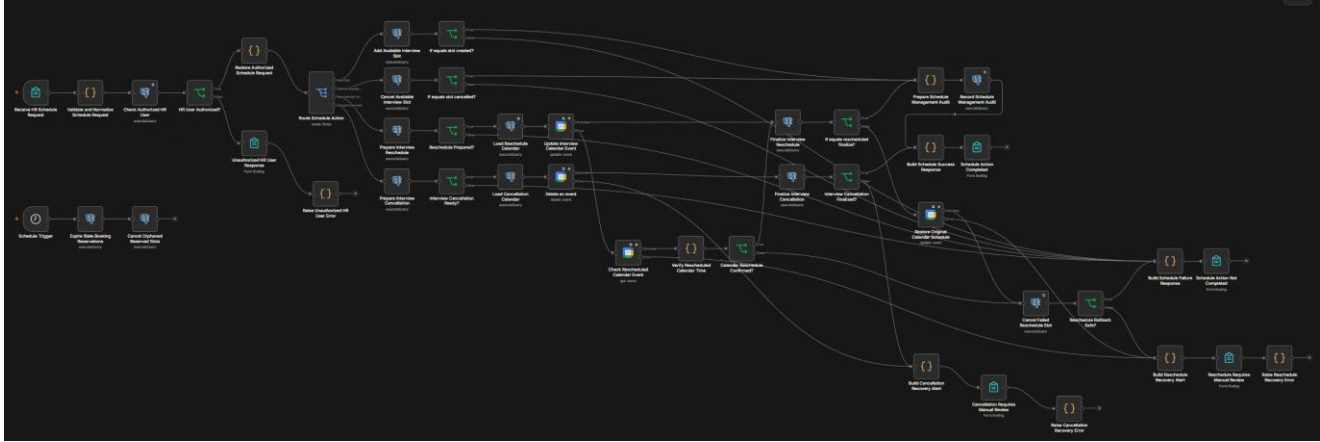


Figure 8. Schedule management includes authorized actions, stale-reservation cleanup, Calendar verification, rollback paths, and explicit manual-review outcomes.

The form is protected by Basic Authentication and an active HR-email allowlist. This is adequate for a controlled demonstration, but it is not equivalent to enterprise single sign-on or identity derived from cryptographically verified user claims.

Reporting Dashboard Synchronization

HR 04 Reporting Dashboard Sync loads application and interview-slot views from PostgreSQL and upserts them into Google Sheets. It reads the application sheet back and verifies synchronization instead of assuming a successful API response produced the expected dashboard state.

Monitoring and Operational Recovery

HR 06 Monitoring and Operations combines an n8n Error Trigger with scheduled PostgreSQL health checks. It builds sanitized failure notifications, marks or releases failed intake claims, fingerprints operational issues, persists issue state, suppresses repeat alerts during a configurable cooldown, and records notification timestamps only after email delivery succeeds.

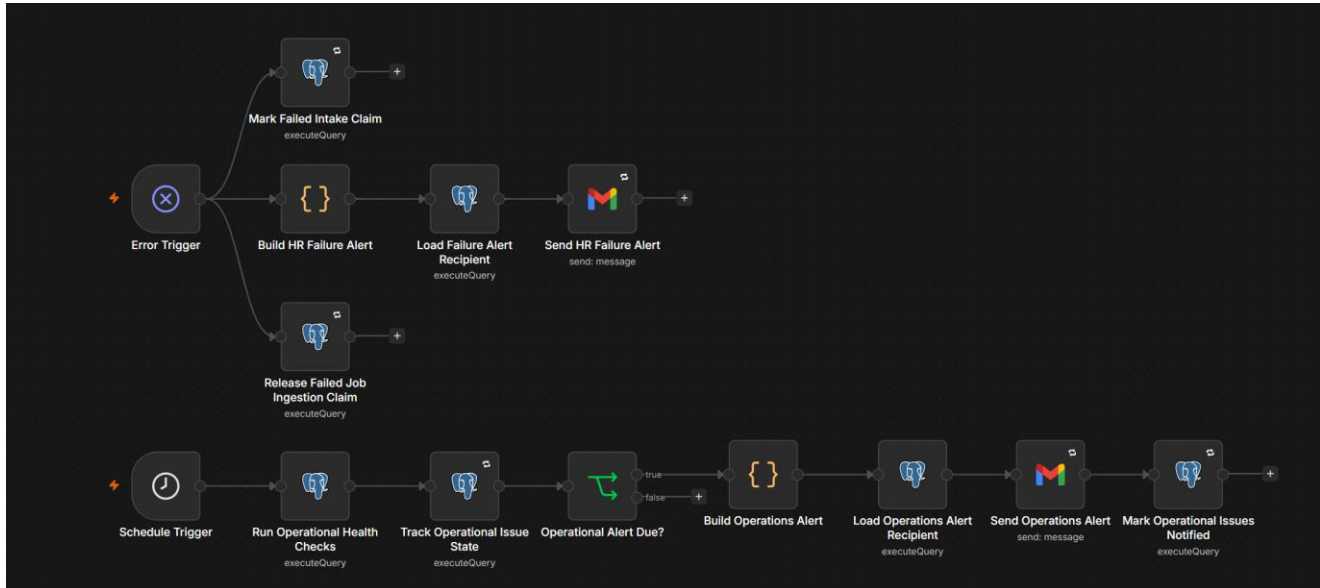


Figure 10. Failure handling and scheduled health checks share persisted alert state while keeping applicant content and credentials out of notification emails.

Engineering Decisions

Claim before expensive processing

Gmail messages and Drive files are claimed in PostgreSQL before text extraction or language-model processing. This reduces duplicate work during retries and provides explicit completion or release behavior.

AI evidence with deterministic scoring

The model performs structured extraction and identifies evidence. Code-based rules calculate the weighted result, and HR retains the final decision. This creates a clearer review trail than an unrestricted AI recommendation.

Confirmation before mutation

The booking link displays information through GET, while POST commits the reservation. This reduces accidental state changes caused by email previews and security scanners.

Compensating actions across services

PostgreSQL and Google services cannot participate in one atomic commit. Booking, rescheduling, and cancellation therefore verify external outcomes and provide rollback, cleanup, or manual-review branches when systems diverge.

Append-only audit history

Schedule-management and decision events are inserted as separate records instead of replacing the previous event. A regression test retained a slot-created event and a later slot-cancelled event for the same slot.

Reliability Security and Privacy Controls

- Parameterized PostgreSQL queries for dynamic values.
- Advisory locks and state predicates around scheduling mutations.
- Gmail and Drive intake claims before expensive processing.
- Structured-output parsing followed by deterministic validation.
- Human approval gates for screening and job activation.
- Deterministic Calendar event identifiers.
- Calendar verification, cleanup, rollback, and manual-review branches.
- Active HR allowlist checks before schedule mutations.
- Centralized non-secret workflow configuration.
- Append-only audit records for material actions.
- Operational issue fingerprints, occurrence counts, persisted state, and alert cooldowns.
- Sanitized failure alerts that exclude raw applicant content, tokens, and credentials.
- Post-write verification of the reporting dashboard.

Controlled Test Results

These results describe controlled portfolio tests, not live recruitment performance.

- The latest reviewed exports contain 7 workflows and 195 nodes.
- Six operational workflows are active; the database setup workflow is intentionally manual.
- An unauthorized HR schedule request was rejected before reaching mutation nodes.
- An authorized test created and later cancelled a slot while retaining two separate audit events.
- A reporting test synchronized 17 application rows and 16 interview-slot rows, then passed the dashboard verification branch.
- A monitoring test tracked 4 active issues and suppressed a repeated notification inside the configured 24-hour cooldown.
- The test plan included valid, low-qualification, unreadable, invalid-file, duplicate-application, booking, rescheduling, cancellation, reporting, and monitoring paths.

No production time-saving or cost-reduction metric is claimed because the system has not been measured against a live HR team. The demonstrated result is an integrated, testable architecture with observable success, rejection, recovery, and monitoring states.

Technology Stack

Category	Details
Automation	Self-hosted n8n in Docker
Database	PostgreSQL
AI processing	OpenAI chat model with structured output parsing
Email	Gmail
File intake	Google Drive
Scheduling	Google Calendar and Google Meet
Reporting	Google Sheets
Interfaces	n8n forms, webhooks, HTML email, and confirmation pages
Deployment	Custom HTTPS domain

Scope and Production Roadmap

The project demonstrates production-minded workflow design, but it should not be presented as a certified production HR platform. Before it processes real applicant data, the following work remains.

- Replace Basic Authentication and the email allowlist with enterprise single sign-on and identity derived from authenticated claims.
- Automate retention, deletion, or anonymization across PostgreSQL, Sheets, Gmail, Calendar, n8n execution data, and AI-related records.
- Protect invitation tokens from browser history, proxy logs, analytics, and referrer exposure.
- Add a durable outbox or reconciliation worker for stronger cross-service consistency.
- Standardize retry policies and add secondary monitoring for failure of the alert workflow itself.
- Complete high-volume load testing, disaster-recovery testing, security testing, and applicable legal or compliance review.
- Move database changes into version-controlled migrations with deployment and rollback procedures.

Reflection

This project reinforced that reliable automation depends on explicit state transitions, ownership, failure behavior, and evidence. The most consequential design choices were separating AI extraction from deterministic scoring, claiming work before expensive processing, and adding recovery paths around Calendar operations.

If I extend the project, I will prioritize automated applicant-data retention and authenticated per-user HR identity. These additions address the clearest gap between a controlled portfolio demonstration and a system suitable for real personnel data.

Conclusion

The completed portfolio project demonstrates my ability to design a multi-workflow n8n system that combines AI-assisted processing, human decisions, relational state, Google Workspace integrations, exception handling, and operational monitoring. The case study also states the remaining production limitations so the demonstrated engineering is clear without overstating deployment readiness.