

CASE STUDY • AI + N8N AUTOMATION

Weekly Ops Digest Automation

How one automation pattern — fetch data, synthesize with an LLM, deliver across channels — moves from a job-scraper to a weekly executive business report.

n8n + Groq

Orchestration & LLM

3 sources

Sales · Projects · Support

2 channels

Telegram + Email

~15 min/wk

saved per report

THE PROBLEM

Every Monday, someone rebuilds the same report by hand

Business data lives scattered across systems: sales in a spreadsheet, project tasks in Airtable, support tickets in an inbox. Building the weekly report means logging into each system, copying numbers by hand, and writing a summary from scratch — 45 to 60 minutes, every single week, with the risk it goes stale or simply doesn't happen if that person is out.

The problem isn't a lack of data — it's a lack of synthesis. Every system knows its own piece, but nobody assembles the full picture without repetitive manual work. And that work — gather, summarize, write — is exactly what an LLM orchestrated inside a workflow can do in seconds.

Why now: this same problem shows up, in different shapes, in almost any small or medium team — sales reports, support digests, project summaries. Solving it once, in a reusable way, is worth more than solving it for a single one-off case.

3+

different systems to check every week

~50 min

of manual work per report

0

consistency from week to week

THE SOLUTION

One workflow. Zero human intervention on the happy path.

An n8n workflow triggers once a week, pulls data from all three sources automatically, aggregates and normalizes it, asks an LLM (Groq) to draft an executive summary in natural language, and delivers it over Telegram and email. If anything fails along the way, it raises its own alert.

The architecture reuses, step for step, the same pattern already proven in an earlier job-scraping and scoring workflow: fetch data → normalize → send context to an LLM via HTTP Request → distribute the result. The domain changes — business instead of jobs — but the workflow structure stays the same. Which is, itself, part of the point: a well-designed pattern gets reused, not rebuilt.

Step	What it does
1 Fetch	Pull the last 7 days from Sheets, Airtable, and Gmail in three parallel branches.
2 Normalize	Join branches by node name and build one clean, aggregated JSON object.
3 LLM Synthesis	Send the aggregated data to Groq and get back an executive summary in Markdown.
4 Deliver	Format for each channel and send via Telegram and Gmail in parallel.

13 nodes, three blocks

Data sources → AI synthesis → Delivery. Each step resolves one concrete pain point from the old manual process.

Node	What it does	Pain it resolves
Weekly Trigger	Fires every Monday at 8am automatically.	No longer depends on one person remembering.
Sheets / Airtable / Gmail	Three parallel branches pull sales, project, and support data from the last 7 days.	No manual login to each system; each source is swappable per business.
Aggregate & Normalize	A Code node joins branches by name, computes totals, builds one clean JSON object.	Replaces manually summing columns across spreadsheets.
Groq — Generate Summary	HTTP Request to Groq with aggregated data, asking for an executive Markdown summary.	Removes the slowest, most inconsistent step: writing the summary.
Parse AI Response	Extracts LLM text, builds Markdown (Telegram) and HTML (email) versions.	One summary, formatted per channel, no extra effort.
Telegram + Gmail	Delivers the digest on both channels in parallel.	Ensures even distribution instead of a single silo.
Telegram — Alert on Failure	Separate error output notifies the admin if the Groq call fails.	Prevents a silent failure from going unnoticed.

Small choices, fewer surprises

1 Aggregate by node name, not Merge

Instead of a 3-input Merge node — which silently drops the third input if it's misconfigured — the Code node reads each source explicitly with `$( 'Node Name' ).all()`. More explicit, and it avoids a classic input-count mistake.

2 Retry on every network node

Sheets, Airtable, Gmail, Groq, Telegram, and the email send all use `retryOnFail` with 3 attempts and a 5-second wait. Most external API failures are transient — a 429, a delay — and resolve themselves without anyone noticing.

3 Separate error output on the AI node

If Groq doesn't respond, the workflow doesn't die — it routes to a distinct Telegram alert instead of the normal report, so "it worked" is never confused with "something needed attention."

TECH STACK

1 n8n

Orchestrator for the workflow

2 Groq API (Llama 3.3 70B)

Executive summary synthesis

3 Google Sheets

Sales KPI source

4 Airtable

Project / task status source

5 Gmail

Support source & delivery channel

6 Telegram Bot API

Fast delivery & alerts

BUSINESS IMPACT

Consistency is the real win

~45–60 min

saved every single week

100%

of weeks with a report — no longer
dependent on one person

2 channels

of simultaneous distribution, at no
extra effort

Beyond the time saved, the biggest benefit is consistency: the report goes out every week, at the same level of detail, independent of the mood or availability of whoever used to write it by hand. And because the whole thing lives in a versionable workflow, adding a fourth data source — a CRM, say — is a matter of adding one more branch, not redesigning the process.

REUSABILITY

One pattern, two domains

This workflow's architecture — multi-source fetch → aggregation → LLM call via HTTP → multi-channel delivery → error handling — is the exact same pattern already proven in a job-scraping and scoring workflow. The data domain and the prompt change; the structure, the nodes, and the design decisions stay identical.

RECRUITING

Job scraping + scoring

Scrapes listings, scores fit with an LLM, delivers matches.

BUSINESS OPS

Weekly executive digest

Aggregates KPIs, drafts a summary with an LLM, delivers the report.

From template to your own stack

- 1 Import *Weekly_Ops_Digest_Automation.json* in n8n (Workflows → Import from File).
- 2 Connect your own credentials for Google Sheets, Airtable, Gmail, Telegram, and the Groq API key.
- 3 Swap the three source nodes for the business's real systems — they don't have to be Sheets / Airtable / Gmail.
- 4 Adjust the prompt on the "Groq – Generate Executive Summary" node to match each team's tone and format.
- 5 Set the workflow's Error Workflow in Settings, so failure alerts land there too.

Weekly Ops Digest Automation

A case study in automation with n8n + LLM