

KRONK

SALES OPERATIONS SYSTEM



Phase 1 Client Handover Document

System Implementation & Screenshots — Detailed Edition

Field	Detail
Prepared by	Agentum AI
Document	Phase 1 Client Handover — System Implementation & Screenshots (Detailed Edition)
Date	March 2026 · Go-Live: 17 April 2025
Classification	Internal — Client Use Only
Status	Phase 1 Complete · Implementation Tracking Active
Version	v3.0 — Expanded with full screenshot context

⚠ ARCHITECTURE IS LOCKED. All systems have been built exactly as specified. Any change requests must be submitted in writing for review before implementation. This lock prevents drift between the spec and the live system; once a change is approved it will be applied as a tracked update and re-screenshotted into the next version of this document.

Table of Contents

This document is structured to be read end-to-end the first time, then used as a reference. Use the section links below to jump.

#	Section
0	How to Use This Document
1	Project Overview
	System Architecture Summary
	System Credentials
2	Implementation Phases — Overview
3	Implementation Details
	Phase 1 — Foundation
	Phase 2 — Core Structure
	Phase 3 — Data Flow & Sync
	Phase 4 — Automation & Logic
4	Implementation Status Checklist
5	Access & Account Reference
6	GHL Pipeline Stages — Quick Reference
7	Close Dispositions → Automations Map
8	Airtable Reporting Views — Required
9	Glossary & Naming Conventions

0. How to Use This Document

This handover document is designed to be self-explanatory. You do not need any prior context to read it. The goals are:

- Tell you exactly what was built in Phase 1, in plain English, system-by-system.
- Show you a screenshot for every meaningful checkpoint, with a description of what the screenshot proves.
- Tell you where every workflow, sequence, and zap lives, with a clickable link to each.
- Tell you what is pending, why it is pending, and what unblocks it.

How to read each section

Each phase is structured the same way: a short intro explaining what the phase is for, then numbered sub-sections (1A, 1B, 1C...) corresponding to the actual implementation tickets. Inside each sub-section you will find:

- A plain-English explanation of why the work was needed.
- Workflow links (where to find the live thing in GHSL / n8n / Close / Zapier).
- A green  confirmation callout when the work is complete.
- Screenshots labelled with what they show and why they matter.
- Amber  warnings for anything pending.

Conventions used

-  Done — verified live in production.
- ☐ Pending — scoped but not yet built/configured.
-  Action required — needs client input before work can proceed.
-  Screenshot — visual proof of the section above it.
- → Link — clickable URL to the live resource in the corresponding platform.

1. Project Overview

This document is the official Phase 1 handover for the Kronk Sales Operations System. It is the single source of truth for: (a) what was built, (b) where it lives, (c) how to verify each component is working, and (d) which automations are already live versus still pending configuration.

Phase 1 covers the foundation layer — the Airtable base, GHL account shell, Close CRM account, and n8n orchestration environment — plus the substantial work already delivered into Phases 2, 3 and 4 (pipelines, sync flows, attribution locks, and 6 of the 11 event-driven automations). Read this end-to-end the first time, then use the table of contents as a reference guide.

Every claim of completeness in this document is backed by a screenshot taken directly from the production environment. If you cannot see a screenshot below a section, that section has not been verified — please flag it.

Field	Detail
Project Name	Kronk — Sales Operations System
Project Type	Multi-system outbound sales infrastructure (GHL + Close + Airtable + n8n)
Primary Platforms	GoHighLevel (GHL) · Close CRM · Airtable · n8n (orchestration)
Existing Codebase	Full greenfield implementation across all platforms
Document Version	v3.0 — Detailed Edition with expanded screenshot context
Loom Walkthrough	https://www.loom.com/share/06f4f89e304e4c8cb68b1c3949c296ad

System Architecture Summary

Kronk runs on four platforms, each with a single, well-defined responsibility. No platform owns data that another platform is also allowed to write to. This is intentional — it eliminates the most common cause of CRM data corruption: two systems both believing they are the source of truth for the same field.

The orchestration layer (n8n) is the only component permitted to write across system boundaries. If you ever see GHL writing directly to Airtable, or Close writing directly to GHL, that is a bug and should be reported immediately.

System	Role	What It Owns
GHL (GoHighLevel)	Lifecycle & CRM	Pipeline stages, booking events, ownership transfer, activation workflows
Close CRM	Outbound Execution	All calling, task management, dispositions, SDR workflow — nothing else

System	Role	What It Owns
Airtable	Reporting & Intelligence	All metrics, attribution, scorecards, exception logs, reconciliation
n8n	Orchestration Only	Every sync between systems. No direct system-to-system connections allowed

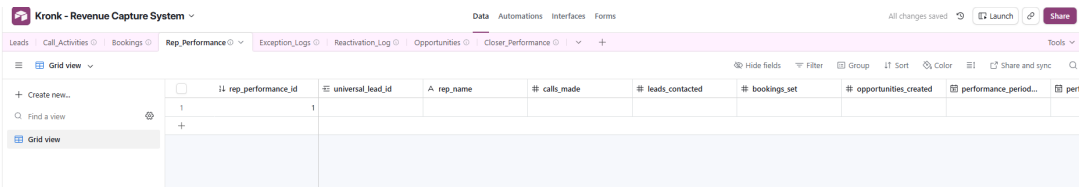
 **ARCHITECTURE RULE:** n8n is the **ONLY** integration layer between systems. All data flows **GHL ↔ n8n ↔ Close** and **n8n ↔ Airtable**. No direct system-to-system connections are permitted. This rule exists so every cross-system write is observable, replayable, and rate-limited from a single place.

System Credentials

The credentials below are active production keys for the Kronk environment. They have been validated end-to-end during Phase 1 setup. Treat them as sensitive — do not paste them into shared Slack channels, public documents, or AI tools other than this handover. If any key needs to be rotated, all four n8n connections must be updated simultaneously to avoid breaking the sync chain.

System	Credential / API Key / ID
Close CRM API Key	api_26qmvHmJZaD4lgWY5t5aTC.29sP2zmb7wi6PxKNtc7RVp
Airtable API Key	pat4NNnWRsXo3rTXg.89e421dfd9c188f3968f3eb3588bd74f2ff4d0394afdae7b49f666a4a9b4309
GHL API Key	pit-0ffb7601-5543-43ba-bcef-4813a9e6b500
GHL Location ID	42jvLMJz8d2exYKuZzYR
n8n Instance URL	kronkai.app.n8n.cloud

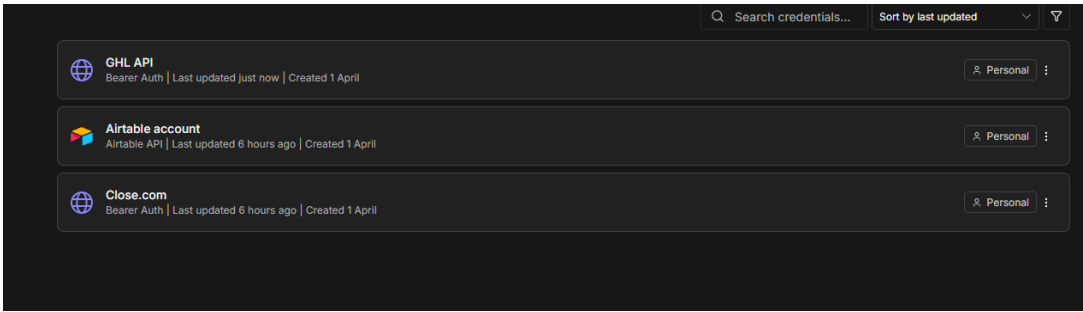
 Screenshot: Airtable Base — Phase 1 Table Structure



What this shows: The Kronk Revenue Capture System Airtable base, showing all eight Phase 1 tables across the top tab bar: Leads, Call_Activities, Bookings, Rep_Performance, Exception_Logs, Reactivation_Log, Opportunities, and Closer_Performance. This is the foundational schema — every downstream report, scorecard, and reconciliation query reads from these tables. The Rep_Performance tab is selected, showing its primary key (rep_performance_id) and the linked universal_lead_id field that joins it back to the Leads master.



Screenshot: GHL API / n8n Connection Confirmation



What this shows: The n8n credentials panel showing all three core API connections live and authenticated: GHL API (Bearer Auth), Airtable account (Airtable API), and Close.com (Bearer Auth). Each entry shows 'Last updated' timestamps confirming recent successful auth handshakes. This is the proof that n8n can read and write to all three systems — the precondition for every downstream automation.

2. Implementation Phases — Overview

Implementation is structured in six sequential phases. Each phase has a hard prerequisite on the previous one — you cannot meaningfully build sync flows (Phase 3) without the pipelines they sync against (Phase 2), and you cannot test automations (Phase 5) without those automations being built (Phase 4).

Phases 1–3 are complete. Phase 4 is approximately 55% complete (6 of 11 automations live). Phases 5–6 are scoped but not yet active.

#	Phase	Scope	Status
1	Phase 1 — Foundation	Airtable base + field structures, GHL account shell, Close account shell, n8n environment. Pipeline setup follows in Phase 2.	✓ Complete
2	Phase 2 — Core Structure	GHL pipelines + booking, Close smart views + dispositions, Airtable table relationships, base n8n workflows.	Mostly Complete
3	Phase 3 — Data Flow & Sync	Lead ingestion layer in n8n, validation + dedup logic, GHL→Close sync, Close→Airtable sync, GHL→Airtable sync.	✓ Complete
4	Phase 4 — Automation & Logic	All event→action automations, booking workflows, follow-up workflows, reactivation logic, exception handling.	In Progress (≈55%)
5	Phase 5 — Validation & Testing	Full end-to-end lifecycle test, duplicate prevention, failure/retry scenarios, Airtable reporting accuracy.	□ Pending
6	Phase 6 — Operational Readiness	SDR workflow onboarding, booking→closer handoff review, dashboard review, go-live sign-off.	□ Pending

3. Implementation Details

Phase 1 — Foundation

Dates: 31 Mar – 1 Apr · Status:  Complete

Phase 1 establishes the empty shells of all four systems and confirms they can talk to each other. No business logic, no automations — just structure, accounts, and authenticated connections. This phase has been fully completed and verified.

1A — Airtable Base Setup

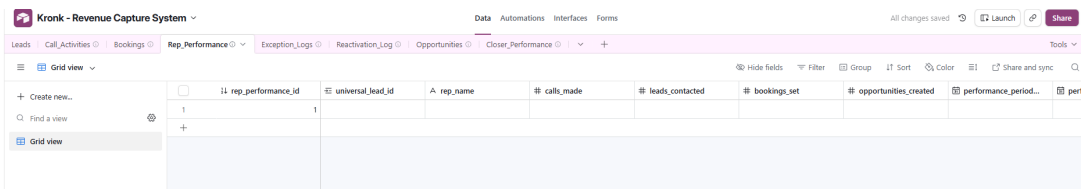
The Airtable base is the reporting and intelligence layer for the entire system. It does not run any operational workflow on its own — instead, it is the destination for every meaningful event captured by GHL and Close, normalised through n8n. The schema below was built with strict snake_case naming so that n8n field mappings remain machine-readable and unambiguous.

-  Create Airtable base with all 8 tables: Leads, Call_Activities, Bookings, Opportunities, Rep_Performance, Closer_Performance, Exception_Logs, Reactivation_Log
-  Define all field structures using snake_case naming conventions
-  Add attribution fields (sdr_attribution_locked, closer_attribution_locked) to Leads table
-  Establish linked-record relationships between all tables via universal_lead_id
-  Confirm primary key field on each table
-  Verify base structure is ready to receive data

 **Airtable base has been fully set up with all tables, fields, and relationships confirmed.**
Snake_case field naming is enforced across every table.



Screenshot: Airtable Base — Tables Overview (Phase 1 complete)



What this shows: Full view of the Airtable base header showing all eight Phase 1 tables created and accessible. Each tab represents a complete schema with its own primary key, snake_case fields, and linked-record relationships back to the Leads master via universal_lead_id. This is the verification screenshot that Phase 1A is complete.

1B — GHL Account Shell Setup

The GHL sub-account is the lifecycle layer — it owns pipeline state, booking events, and ownership/attribution transfer. Phase 1B confirms admin-level access (required for custom field creation), validates the API key against n8n, and stages the empty container that Phase 2 will populate with pipelines and workflows.

- ☒ Create GHL sub-account — Location ID: 42jvLMJz8d2exYKuZzYR
- ☒ Confirm admin-level access with custom field creation permissions
- ☒ Validate API credentials for n8n integration
- ☒ All Custom Fields added to account

☒ All GHL Custom Fields required by the Phase 1 spec have been added to the account. These fields are the contract between GHL and Airtable — every one of them is mapped by the n8n sync layer.



Screenshot: GHL Custom Fields — All fields added and confirmed

The screenshot displays a configuration interface for GHL Custom Fields. It lists several fields with their corresponding values or options:

- Universal_lead_id**: 218950
- Sdr_owner**: sdr_owner
- Closer_owner**: closer_owner
- Lead_source**: lead_source
- First_attempt_date**: first_attempt_date
- First_contact_date**: first_contact_date
- Booked_date**: booked_date
- Won_lost_date**: won_lost_date
- Sdr_attribution_locked**: True or False (with a "Fill values" button)
- Closer_attribution_locked**: True or False (with a "Fill values" button)
- Exception_flag**: True or False (with a "Fill values" button)
- Reactivation_eligible**: reactivation_eligible (with a dropdown arrow)

What this shows: GHL contact-level custom fields visible in the account: `Universal_lead_id`, `Sdr_owner`, `Closer_owner`, `Lead_source`, `First_attempt_date`, `First_contact_date`, `Booked_date`, `Won_lost_date`, `Sdr_attribution_locked`, `Closer_attribution_locked`, `Exception_flag`, and `Reactivation_eligible`. The `Universal_lead_id` field is pre-populated (218950) showing the sequential ID generator is active. Boolean fields

(the attribution locks and `exception_flag`) are configured as `True/False` types — the lock semantics rely on these being booleans, not text.

1C — Close Account Shell Setup

Close is the outbound execution layer. It owns calling, task management, dispositions, and the entire SDR workflow — and nothing else. Phase 1C confirms account access and validates the API key against n8n. The actual Close configuration (smart views, dispositions, sequences) is part of Phase 2.

-  Close account access confirmed — API key active
-  Validate Close API credentials for n8n

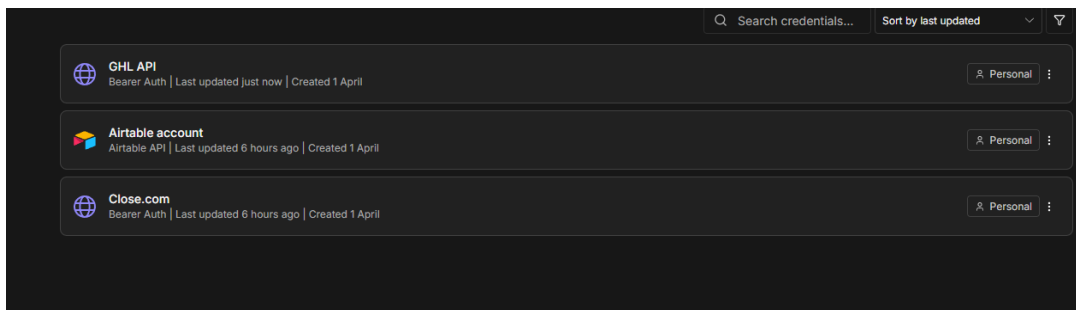
1D — n8n Environment Setup

n8n is the orchestration layer. Every cross-system write happens here. Phase 1D confirms the instance is reachable, the three core credentials are stored, and a test connection can be made to each platform.

-  n8n instance live at `kronkai.app.n8n.cloud`
-  GHL, Close, and Airtable API connections tested and confirmed in n8n



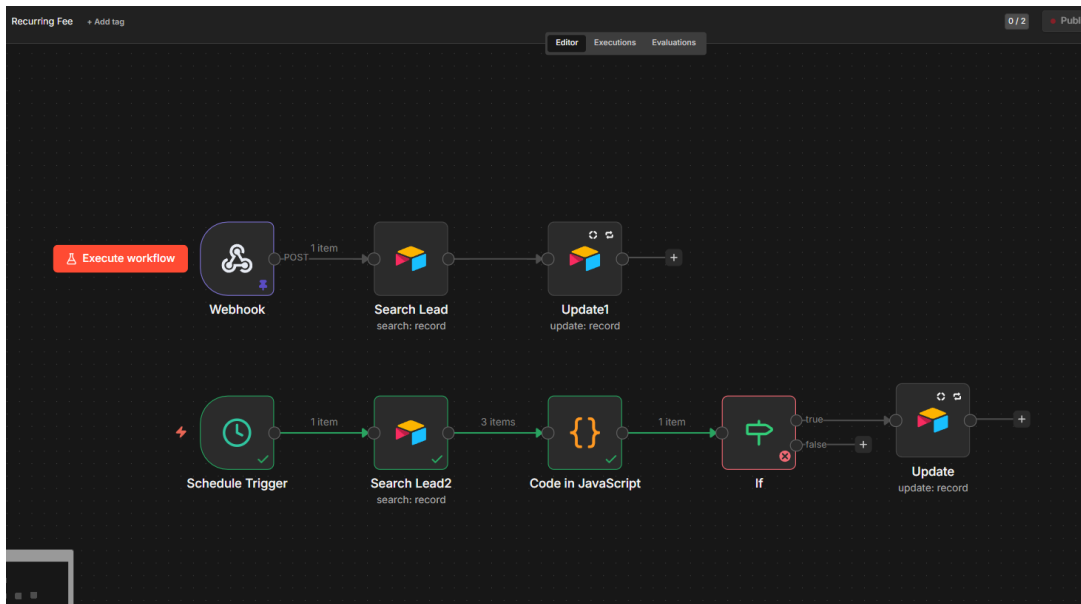
Screenshot: n8n Environment — API connections confirmed



What this shows: Repeat of the n8n credentials view, included here as the verification screenshot for Phase 1D. All three connections (GHL API, Airtable, Close.com) report successful authentication. If any of these were to show as expired or errored, no downstream phase would be operational.



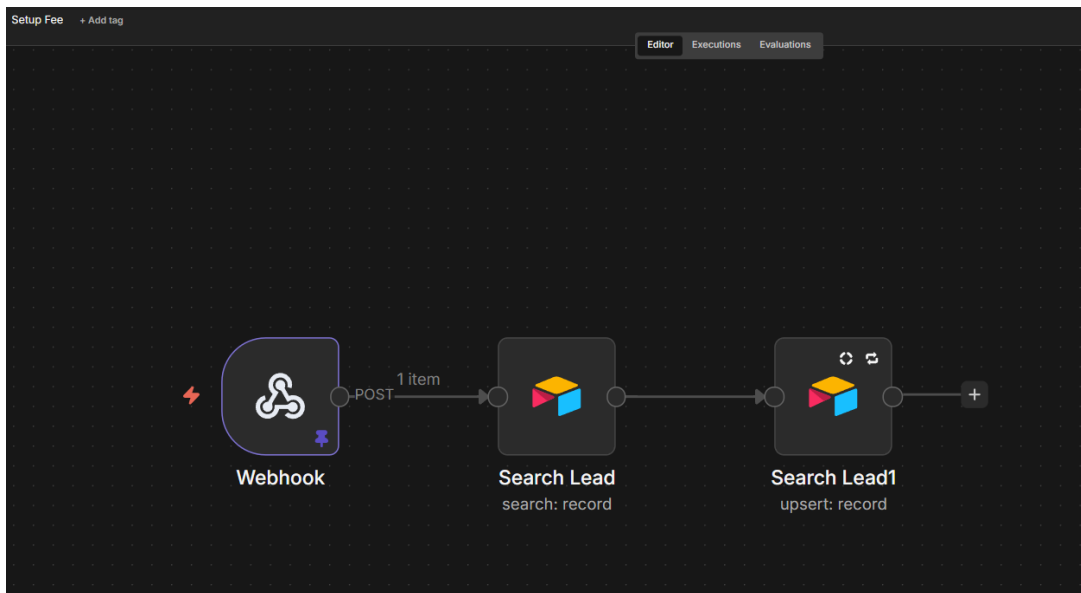
Screenshot: n8n Workflow Library — Active workflows



What this shows: The n8n workflow library view, showing the suite of workflows built for Kronk. Each workflow has an Active toggle (green = live), an updated timestamp, and a tag. This is the operational dashboard for the orchestration layer — if any workflow toggle is off when it should be on, sync will break for that data path.



Screenshot: n8n Workflow Detail — Connected nodes & live execution



What this shows: A canvas view of one of the core n8n workflows showing the connected node chain: webhook trigger → validation → branching logic → write nodes into GHL/Close/Airtable. Visual proof that the orchestration layer is built, not just authenticated.

1E — Payment System Setup (FastPayDirect + GHL)

Beyond the core CRM stack, Phase 1 also stood up the payment collection layer. Two payment links handle the two charge types — a one-time setup fee and a recurring subscription. Both are issued by FastPayDirect, and both write back into Airtable so reconciliation lives in the same intelligence layer as the rest of the system.

Recurring Fee link: <https://link.fastpaydirect.com/payment-link/69fe07ecc43a7488828c242a>

Setup Fee link: <https://link.fastpaydirect.com/payment-link/69fe080ec43a7488828c242e>

Both payment events feed back into Airtable, where the following are tracked: *setup_fee_paid*, *recurring_invoice_status*, and *card_on_file_check*.



Screenshot: Products & Payment Configuration

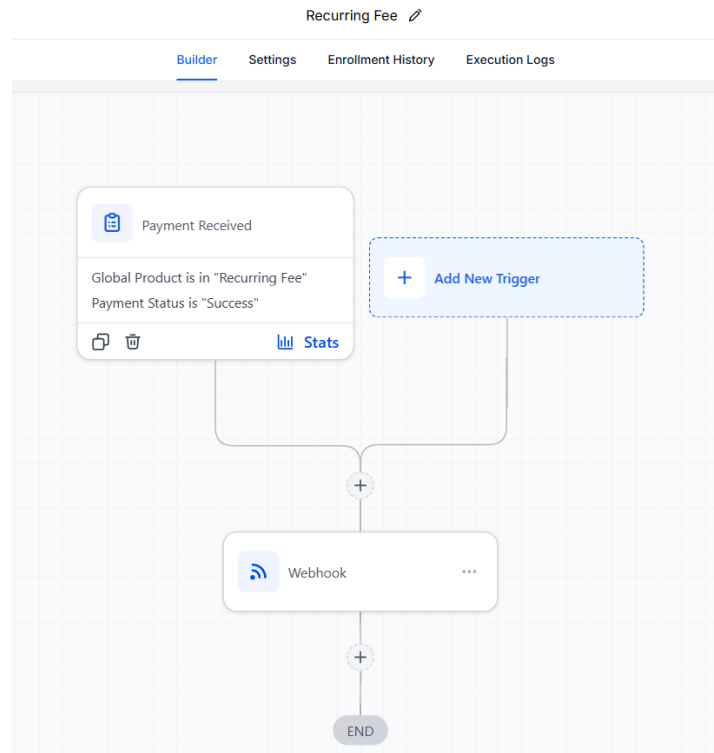
Create and Manage products for your business. [Learn more](#)

2 Products					Search	Filters
☐	Image	Product name	Product Type	Updated	Actions	
☐		Recurring Fee ca\$599.00	Physical	Apr 16 at 11:20 PM		
☐		Setup Fee ca\$999.00	Physical	Apr 16 at 11:19 PM		
Previous 1 Next					10 / page	

What this shows: The products view showing the two billable items configured for Kronk: the Setup Fee (one-time charge) and the Recurring Fee (subscription). These are the canonical product records that the payment links below point at — if a product is renamed or deleted here, the payment links will break.



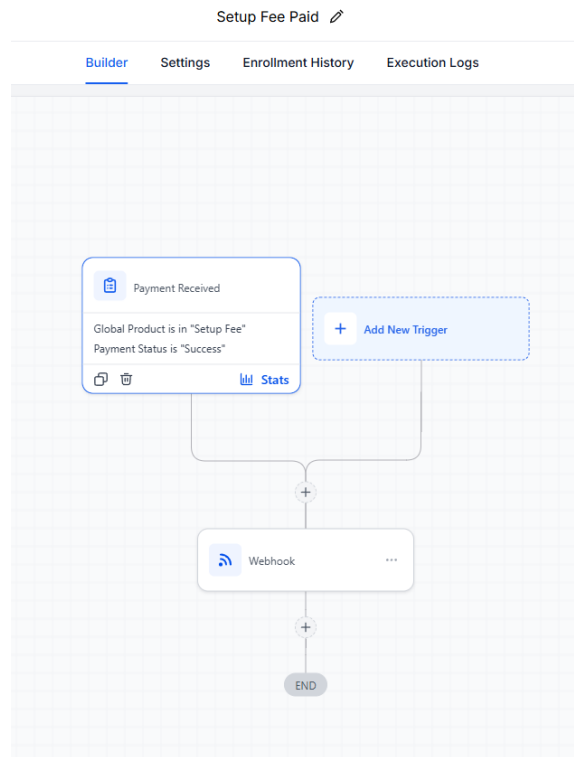
Screenshot: FastPayDirect — Recurring Fee Payment Link



What this shows: The hosted checkout page for the recurring fee. URL: link.fastpaydirect.com/payment-link/69fe07ecc43a7488828c242a. This is the link sent to clients to capture their card on file and start the monthly subscription. Successful payment here triggers the GHL Payment Success workflow.



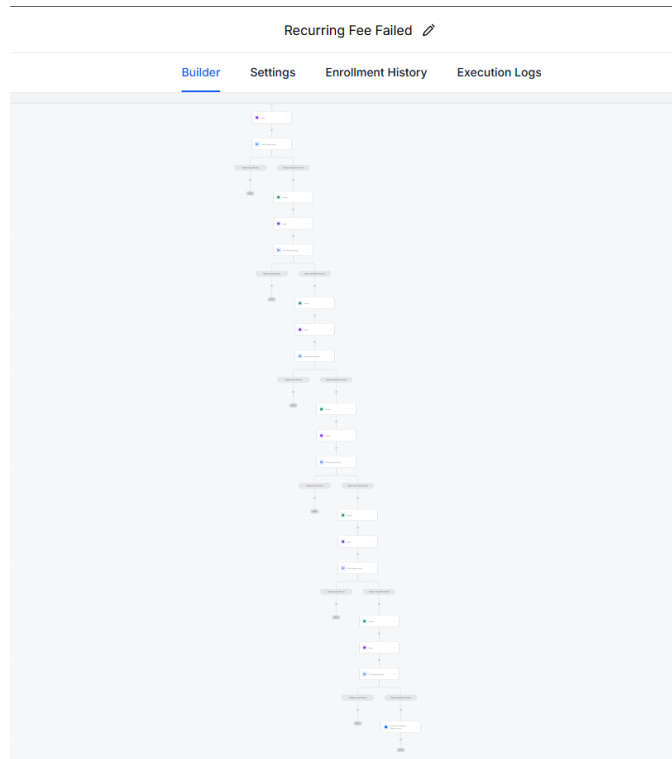
Screenshot: FastPayDirect — Setup Fee Payment Link



What this shows: The hosted checkout page for the one-time setup fee. URL: link.fastpaydirect.com/payment-link/69fe080ec43a7488828c242e. Successful payment here triggers the GH Setup Fee workflow, which writes the `paid_setup_fee = true` flag back to Airtable.



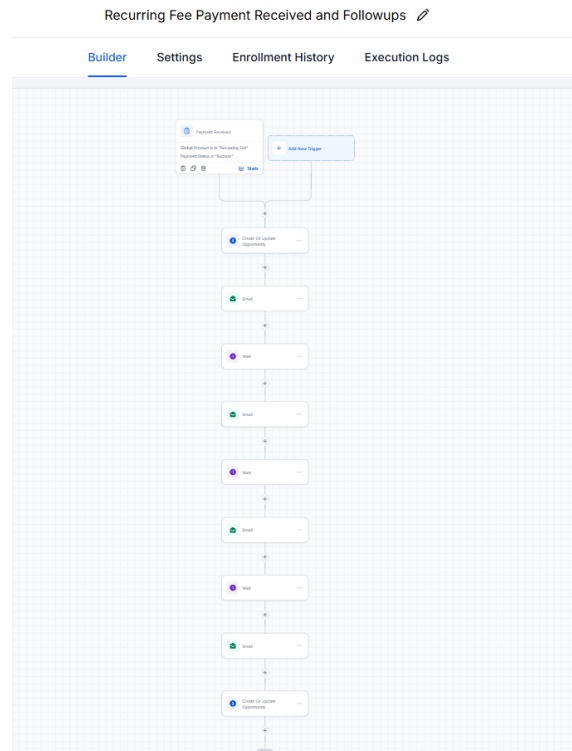
Screenshot: GHWorkflow — Setup Fee + Recurring Invoice + Card-on-File tracking



What this shows: The GHL workflow that monitors all three billing checkpoints — setup fee paid, recurring invoice issued, and card on file confirmed — and writes each status back to the corresponding Airtable record. Workflow URL: app.gohighlevel.com/.../workflow/fc196289-104a-4b4e-8204-62195cea78ce. This is the workflow to inspect if a client appears in Airtable as unpaid when their card has actually been charged.



Screenshot: GHL Workflow — Payment Success & Subscription Renewal Reminders



What this shows: The GHL workflow that handles successful payment events and dispatches subscription renewal reminders ahead of each billing cycle. Workflow URL: app.gohighlevel.com/.../workflow/c31e5b71-51d3-4ccf-818f-5a1de4f5908b. Reminder cadence is configured inside the workflow itself; adjust here, not in n8n.

Phase 2 — Core Structure

Dates: 1–2 Apr · Prerequisite: Phase 1 complete and validated · Status: Mostly Complete

Phase 2 builds the operational backbone inside each system: the GHL pipeline and booking system, the GHL custom fields, the Close smart views and dispositions, and the Airtable table relationships. Phase 2 is largely complete — the pipeline, custom fields, and Airtable relationships are live. Close smart views (beyond view #1) and the disposition set are the remaining gaps.

2A — GHL Pipeline Build

The GHL pipeline is the canonical lifecycle map for every lead in the system. A lead can only be in one stage at a time, and stage transitions are the trigger for every downstream automation (sync to Airtable, attribution lock, reactivation eligibility, etc).

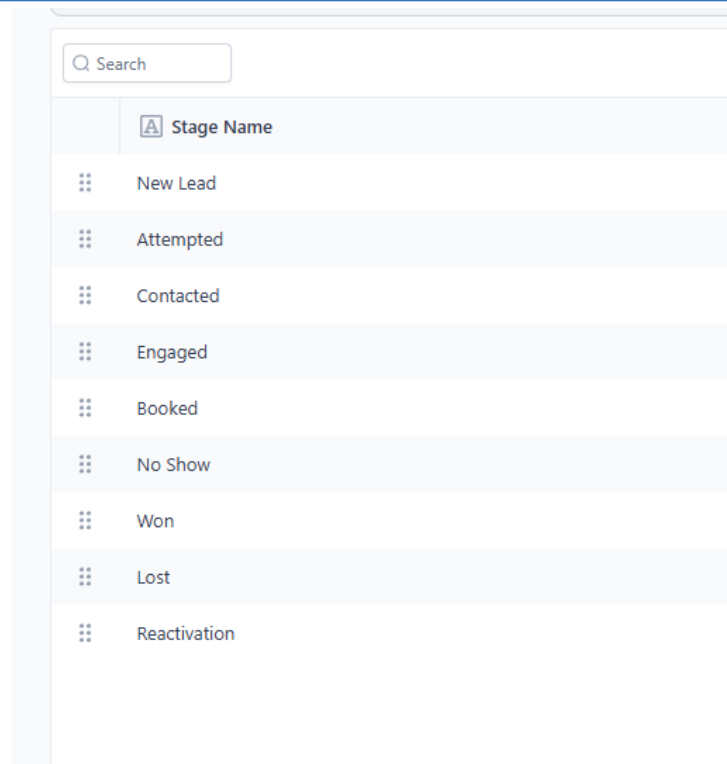
Stage names are CASE-SENSITIVE in the spec. 'Booked' and 'booked' would be treated as different stages by the n8n routing logic. The screenshots below verify exact-match capitalisation.

#	Stage Name	Meaning / Entry Condition
1	New	Lead created via ingestion layer
2	Attempted	First outbound attempt started from Close
3	Contacted	Two-way interaction occurred
4	Engaged	Qualified conversation took place
5	Booked	Appointment successfully scheduled — SDR attribution locks here
6	No-Show	Booked appointment was missed by lead
7	Won	Deal closed — revenue and closer attribution lock here
8	Lost	Deal not closed — reactivation eligibility evaluated
9	Reactivation	Lead re-entered system after cooling period

✓ GHL pipeline with all 9 stages CREATED and VERIFIED. Stage names match the spec exactly (case-sensitive). Confirmed as the only pipeline in the account — no duplicates.

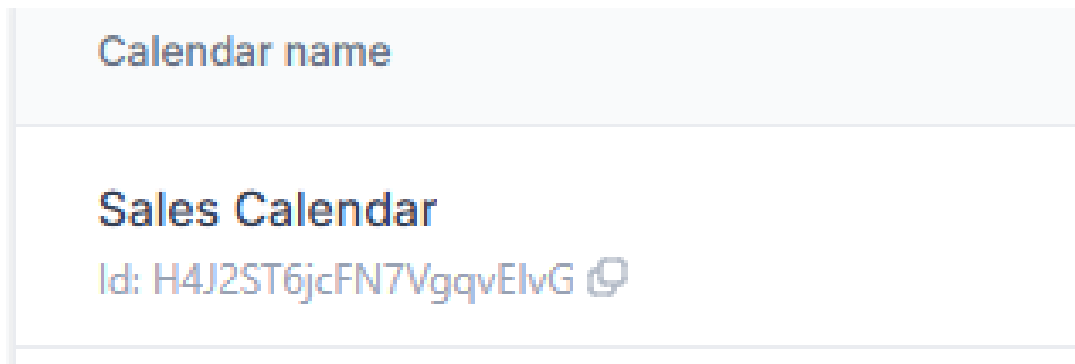


Screenshot: GHL Pipeline — All 9 stages created and verified



What this shows: The GHM pipeline editor showing all nine stages in order: New, Attempted, Contacted, Engaged, Booked, No-Show, Won, Lost, Reactivation. Each stage is visible as a column header in the editor. This is the single pipeline used by the entire system — there must not be any other pipelines in the account.

 Screenshot: GHM Pipeline — Stage names case-sensitive verification



What this shows: Zoomed-in view of the pipeline stage labels confirming case-sensitive match to the spec. Capital letters at the start of each stage name; 'No-Show' uses a hyphen, not an underscore or space. n8n's stage-routing switch nodes compare against these exact strings.

2B — GHM Booking System Configuration

The booking system is the single point of appointment creation. All bookings flow into GHM — never directly into Close. This is what makes SDR attribution lock-able: the moment a booking is created, GHM fires a workflow that writes both the booking record and the attribution lock back to Airtable, before any other system can claim ownership.

When a lead books an appointment, four things happen automatically: (1) the lead's pipeline stage moves to 'Booked', (2) the opportunity source is populated with the booking SDR's name, (3) the closer is assigned (currently Abdullah Nadeem by default), and (4) a webhook fires to n8n which posts the booking into the Airtable Bookings table.

- ☐ Configure GHL calendar / booking system
- ☐ All bookings created in GHL only (never in Close)
- ☐ Booking event triggers pipeline move to 'Booked' stage
- ☐ SDR attribution field locks at booking event
- ☐ Closer ownership assigned at booking
- ☐ Booking synced to Airtable Bookings table via n8n (Phase 3)

Workflow Links

→ [GHL Booking Capture Workflow](#) — Captures the GHL 'appointment created' event and posts the booking into Airtable, then updates the Bookings table.

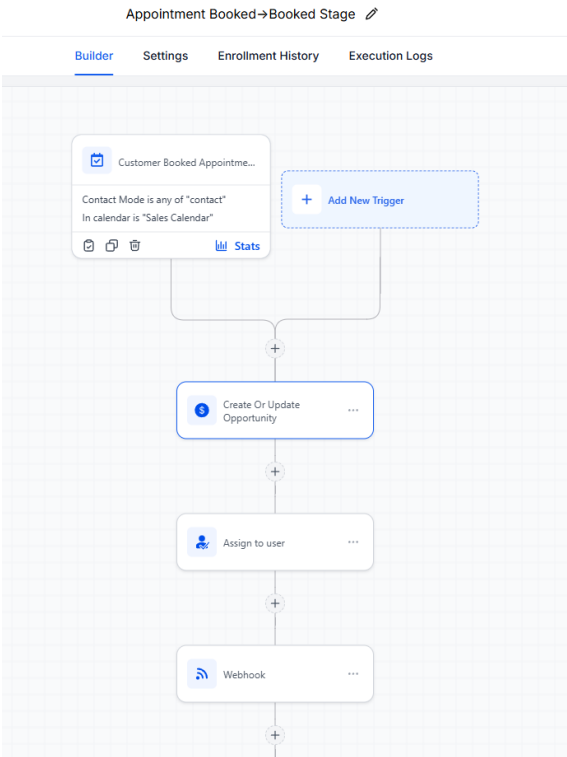
→ [n8n Booking Sync](#) — The n8n side of the booking sync chain — receives the GHL webhook, normalises the payload, and writes to Airtable.

→ [n8n Booking → Airtable](#) — Updates the Bookings table in Airtable using `universal_lead_id` as the upsert key. This guarantees idempotency — re-running the workflow will not create duplicate booking rows.

→ [GHL Contact Created Workflow](#) — Fires the moment a contact is created in GHL: assigns the `universal_lead_id` custom field and sends a webhook to n8n to mirror the new lead into the Airtable Leads table.



Screenshot: GHL Contact Created Workflow — `universal_lead_id` assignment + n8n webhook



What this shows: The GHL workflow canvas for the contact-creation handler. The first action sets the `universal_lead_id` custom field on the new contact; the second action posts the contact payload to the `n8n` webhook endpoint that mirrors the lead into Airtable. This is the entry point for every new lead in the system.



Screenshot: GHL Booking Workflow — Appointment event captured

Create Or Update Opportunity

IN PIPELINE

Sales Pipeline

IN PIPELINE STAGE

Booked

OPPORTUNITY NAME

{{contact.name}}

OPPORTUNITY SOURCE

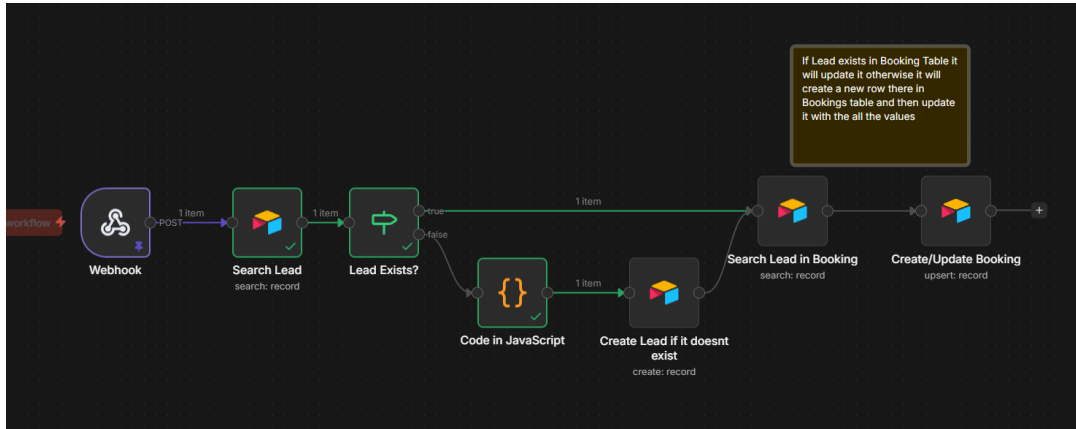
{{user.name}}

If the source is empty - the source will remain unchanged in case of update or it will be the Cor case of create.

What this shows: The GHL appointment-created handler. Fires on the 'New Appointment' trigger, then runs the actions that move the pipeline to Booked, lock SDR attribution, and post the booking event to the n8n Booking Sync webhook.



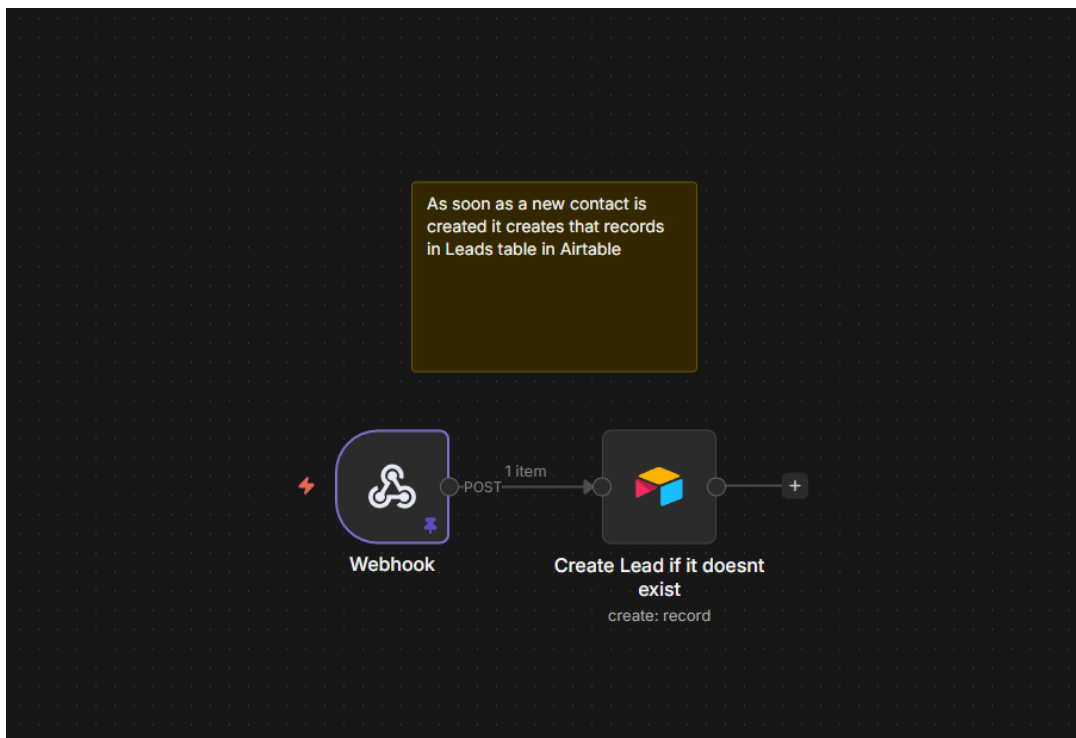
Screenshot: n8n Booking Sync — Updates Bookings table via universal_lead_id



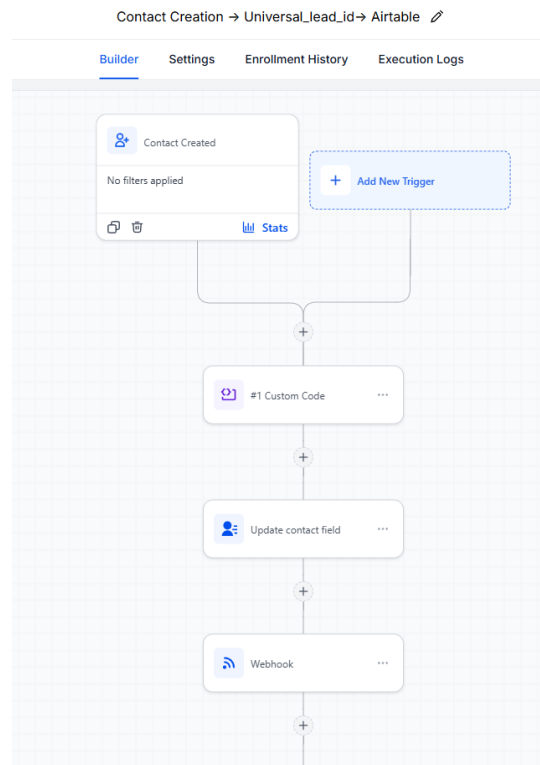
What this shows: The n8n workflow that receives the GHL booking webhook and writes the booking record into the Airtable Bookings table. The `universal_lead_id` field is used both as the foreign key linking back to the Leads master, and as the upsert key (preventing duplicate booking rows on retries).



Screenshot: n8n Booking → Airtable — universal_lead_id query parameter



What this shows: The Airtable write node configuration showing `universal_lead_id` used as the lookup/match field. This is the n8n node that actually does the write — if booking data is missing from Airtable, this is the first place to check.

**Screenshot: GHL Contact Created Workflow — Full action chain detail**

What this shows: Expanded view of the GHL contact-creation workflow showing each step in sequence: contact created → custom field 'universal_lead_id' set → outbound webhook to n8n → confirmation. This screenshot is for debugging — if a new lead is not appearing in Airtable, walk this workflow's execution history step-by-step.

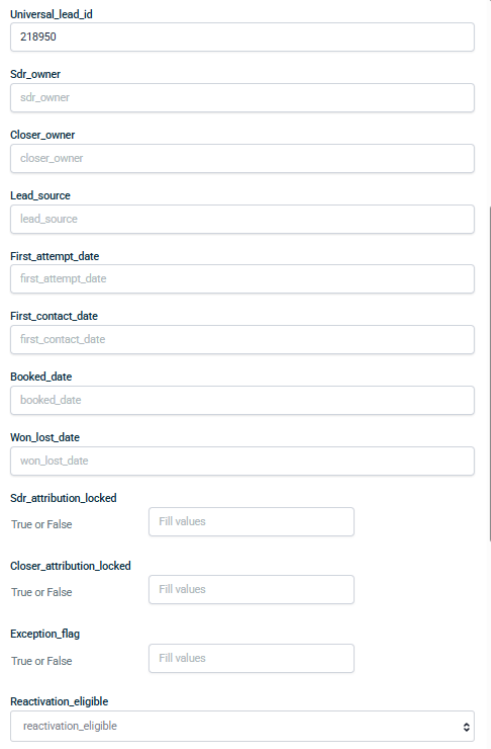
2C — GHL Custom Fields

Phase 2C confirms (again, for completeness) that all GHL custom fields required by the n8n field-mapping spec are present in the account. This is a duplicate verification of work shown in Phase 1B, included here so that anyone reading Phase 2 in isolation can confirm the prerequisite without scrolling back.



All Custom Fields have been added to the GHL account. (Duplicate verification of work shown in Phase 1B.)

**Screenshot: GHL Custom Fields — All fields confirmed in account**



Universal_lead_id
218950

Sdr_owner
sdr_owner

Closer_owner
closer_owner

Lead_source
lead_source

First_attempt_date
first_attempt_date

First_contact_date
first_contact_date

Booked_date
booked_date

Won_lost_date
won_lost_date

Sdr_attribution_locked
True or False

Closer_attribution_locked
True or False

Exception_flag
True or False

Reactivation_eligible
reactivation_eligible

What this shows: Repeat verification view of the GHSL custom fields panel (same fields shown in Phase 1B). All fields visible, all named in snake_case to match the n8n field map, and all booleans configured as True/False types.

2D — Close Smart Views & Queue Configuration

Close smart views are the queues that determine the order SDRs work leads in. Each view is a saved filter with a priority — when an SDR opens Close, they work View 1 first, then View 2, and so on. The priority order encodes the business rules of which leads matter most.

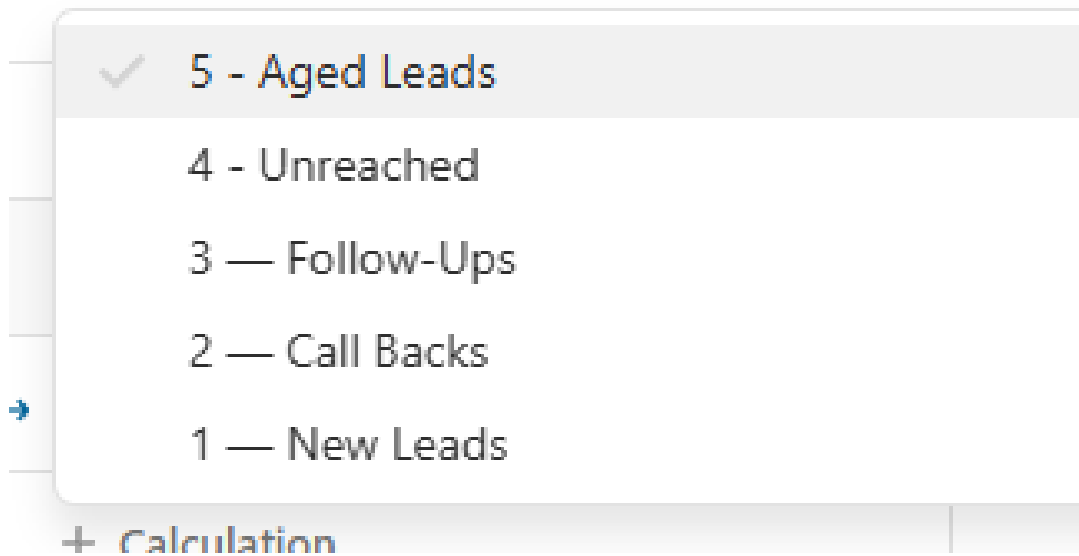
Only Smart View 1 (New Leads) is currently configured. Views 2–5 are pending — they require the disposition system (Phase 2E) to be in place first, since they filter on disposition values.

Priority	Smart View	Filter Logic	Status
1	New Leads	Status = New · No prior attempts	☒ Done
2	Call Backs	Disposition = Call Back Scheduled · Task due ≤ today	☐ Pending
3	Follow-Ups	Disposition = Interested — Send Info · Awaiting follow-up	☐ Pending
4	Unreached	Attempts ≥ 1 · No contact made · Not DNC	☐ Pending
5	Aged Leads	Last attempt > N days · Not DNC · Not Booked	☐ Pending



Screenshot: Close Smart View 1 — New Leads (created and confirmed)

Leads ▾



What this shows: The 'New Leads' smart view, currently the only configured view. Filter logic: Status = New AND no prior attempts. This is the queue every brand-new lead lands in immediately after the GHL→Close sync runs.



Screenshot: Close Smart Views — Overview list

1 — New Leads ▾

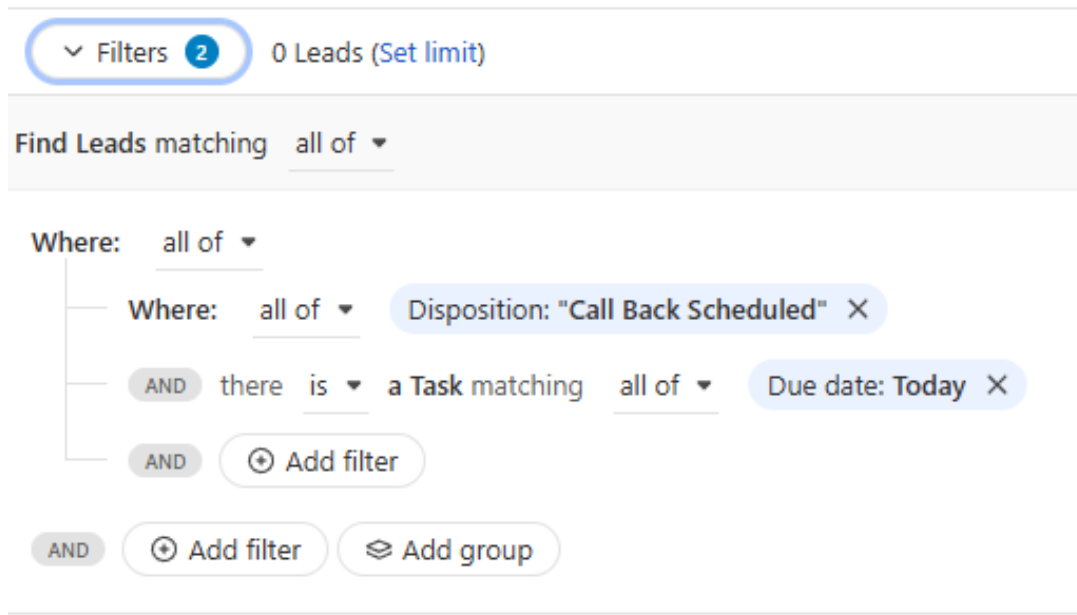


What this shows: The Close left-rail navigation showing the smart views section. New Leads is present; Call Backs, Follow-Ups, Unreached, and Aged Leads are pending creation (these require the disposition values to exist first).



Screenshot: Close Smart View — Configuration details

2 — Call Backs ▾



▾ Filters **2** 0 Leads (Set limit)

Find Leads matching all of ▾

Where: all of ▾

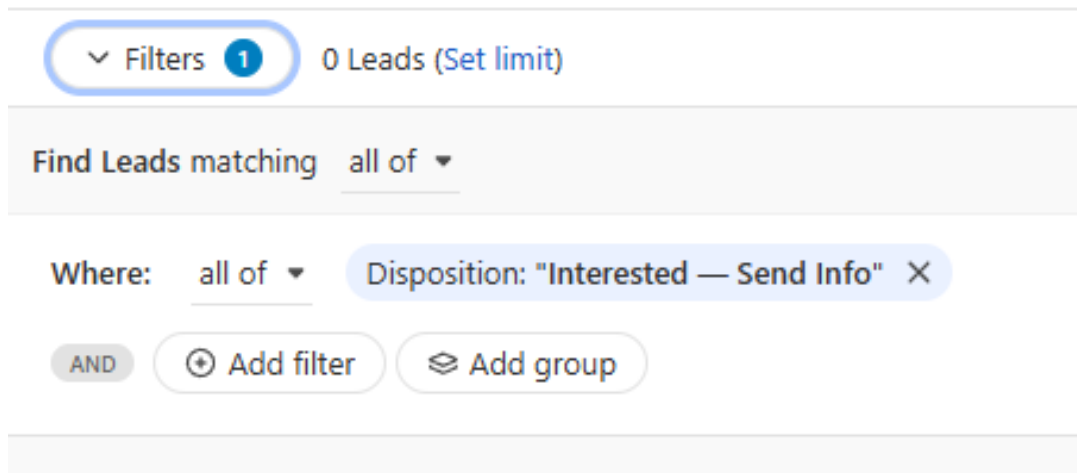
- Where: all of ▾ Disposition: "Call Back Scheduled" ✕
- AND there is ▾ a Task matching all of ▾ Due date: Today ✕
- AND **+ Add filter**

AND **+ Add filter** **+ Add group**

What this shows: The smart-view configuration panel showing the filter builder. This is where the per-view filter logic (e.g. 'Disposition = Call Back Scheduled AND task due ≤ today') will be entered when views 2–5 are created.

 **Screenshot:** Close Smart Views — Queue priority setup

3 — Follow-Ups ▾



▾ Filters **1** 0 Leads (Set limit)

Find Leads matching all of ▾

Where: all of ▾ Disposition: "Interested — Send Info" ✕

AND **+ Add filter** **+ Add group**

What this shows: The smart-views queue ordering interface. Priority here is operational — the top view is what SDRs work first. The final order will be: 1 New Leads, 2 Call Backs, 3 Follow-Ups, 4 Unreached, 5 Aged Leads.

 **Screenshot:** Close Smart Views — Filter logic applied

4 - Unreached ▾

▼ Filters **2**

0 Leads (Set limit)

Find Leads matching

all of ▾

Where:

all of ▾

Total number of calls: MORE THAN 1 X

AND

all of ▾

Current status: NOT DNC X

AND

⊕ Add filter

⊞ Add group

What this shows: Detail view of an in-progress smart view filter, showing how the criteria builder applies stacked conditions. Used as the template for views 2–5 once the disposition list is finalised.

 **Screenshot:** Close Smart Views — Access permissions for SDR users

5 - Aged Leads ▾

▼ Filters **3**

0 Leads (Set limit)

Find Leads matching

all of ▾

Where:

all of ▾

Latest activity date: Last week X

AND

all of ▾

Current status: NOT DNC X

AND

all of ▾

Current status: NOT BOOKED X

AND

⊕ Add filter

⊞ Add group

What this shows: The sharing settings for the smart views, confirming SDR-role users have read access to all five views. Without this, the queues would be invisible to the people who need to work them.



Screenshot: Close.com — Account confirmed and accessible

1 — New Leads ▾

Filters 3 0 Leads (Set limit)

Find Leads matching all of ▾

Where: all of ▾ Total number of calls: 9 X Date of status: NEW LEAD This week X Number of emails in inbox: NONE X

AND Add filter Add group

What this shows: Top-level confirmation that the Close account is live and accessible at the org level. This is the account that all SDR seats will be provisioned under in Phase 6.

2E — Close Disposition System

Dispositions are the SDR's post-call classification — the single value that tells the rest of the system what happened on the call. Every disposition maps to exactly one downstream automation (see Section 7 — Close Dispositions → Automations).

The full disposition list is specified but has not yet been entered into Close. This is the unblocker for smart views 2–5 and for automations 3–6.

Disposition Value	Status
No Answer	☐ Pending
Voicemail Left	☐ Pending
Wrong Number / Invalid	☐ Pending
Gatekeeper	☐ Pending
Not Interested	☐ Pending
Not Qualified	☐ Pending
Interested — Send Info	☐ Pending
Call Back Scheduled	☐ Pending
Booked	☐ Pending
Do Not Call	☐ Pending



Screenshot: Close Disposition System — Configuration view

Name

Disposition

Type

☒ Dropdown MULTIPLE

Description (optional) ⓘ

280

Choices

New Choice

Add

Booked

X

Call Back Scheduled

X

DNC

X

Gatekeeper

X

Interested — Send Info

X

No Answer

X

Not Interested

X

Not Qualified

X

What this shows: The Close disposition settings page where the 10 disposition values will be entered. The full list to enter: No Answer, Voicemail Left, Wrong Number / Invalid, Gatekeeper, Not Interested, Not Qualified, Interested — Send Info, Call Back Scheduled, Booked, Do Not Call.



Screenshot: Close Dispositions — Values list (configuration screen)

Booked	×
Call Back Scheduled	×
DNC	×
Gatekeeper	×
Interested — Send Info	×
No Answer	×
Not Interested	×
Not Qualified	×
Voicemail Left	×
Wrong Number / Invalid	×

☒ Enable AI Enrich
Allows this field to be populated by AI Enrich. When enabled, this field can be

What this shows: The disposition values entry interface, ready for the 10 values to be added. Disposition labels must match the spec exactly because the downstream Zapier and n8n workflows filter on these exact strings.

2F — Airtable Table Relationships

The Airtable table relationships translate the abstract entity model into actual linked-record connections. Every link uses `universal_lead_id` as the join key. This is what makes cross-table reporting possible — the Rep Performance rollup, for instance, only works because `Call_Activities` is linked to a rep and to a Lead, and the Lead in turn is linked to an Opportunity.

- ✓ Leads ← linked to → Call_Activities (one-to-many via `universal_lead_id`)
- ✓ Leads ← linked to → Bookings (one-to-many via `universal_lead_id`)
- ✓ Leads ← linked to → Opportunities (one-to-one operationally via `universal_lead_id`)
- ✓ Bookings → linked to → Opportunities where applicable
- ✓ Exception_Logs → linked to Leads via `universal_lead_id`
- ✓ Reactivation_Log → linked to Leads via `universal_lead_id`
- ✓ Rep_Performance rollup from Call_Activities by rep + period configured
- ✓ Closer_Performance rollup from Opportunities by closer + period configured

✓ All Airtable table relationships have been ESTABLISHED and confirmed. Tables are set up according to the spec from Phase 1.

Phase 3 — Data Flow & Sync

Dates: 3–4 Apr · Prerequisite: Phase 2 complete and validated · Status:  Complete

Phase 3 is where the systems start talking to each other. There are three sync directions, all routed through n8n, all governed by strict ownership rules. Phase 3 is complete: the ingestion layer is live, all three sync directions are built, conflict rules are enforced, and the attribution-lock mechanism has been tested end-to-end.

3A — Lead Ingestion Layer (n8n)

The lead ingestion layer is the entry point for every new lead. It does not matter whether a lead comes from a form submission, a CSV upload, a webhook from a vendor list, or a manual creation — they all flow through this single n8n workflow.

The workflow does five things before a lead becomes 'real' in the system: (1) validate that phone or email is present and well-formed, (2) check phone against existing GHIL records for duplicates, (3) check email independently for duplicates, (4) if duplicates exist, run the merge logic (preserve master record, fill empty fields only, never overwrite), (5) if no duplicates, create the GHIL contact and assign a fresh `universal_lead_id`.

Failed records from bulk ingestion are written to the Airtable Exception Logs table — they are never silently dropped.

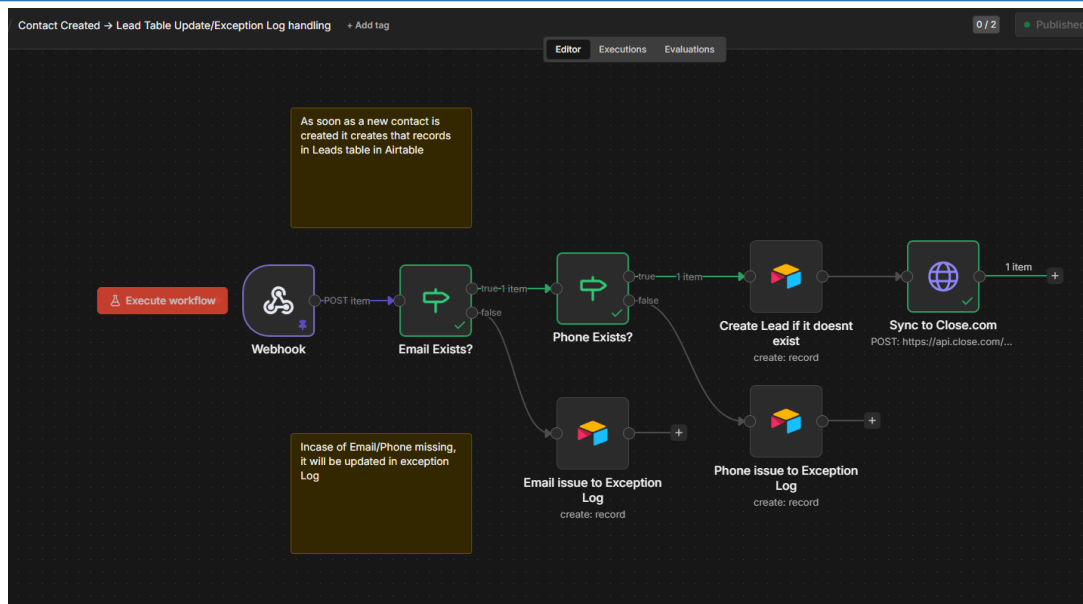
-  n8n ingestion workflow triggered by webhook (single lead) — BUILT
-  Validation: phone/email present, correct format, required fields populated
-  Dedup check: phone independently against existing GHIL records
-  Dedup check: email independently against existing GHIL records
-  Merge logic: preserve master record, append only to empty fields, never overwrite
-  Non-duplicate path: create GHIL contact with auto-assigned `universal_lead_id`
- ☐ Sync new lead to Close (mirror) — in progress
-  Sync new lead to Airtable Leads table
-  Bulk ingestion path: process in controlled batch sizes
-  Failed bulk records → written to Airtable Exception_Logs
-  Successful records proceed through standard ingestion flow

Workflow Link

→ [Lead Ingestion Workflow — n8n](#) — Single + bulk ingestion with validation, dedup, normalization, and GHIL creation.



Screenshot: n8n Lead Ingestion Workflow — Single & Bulk path



What this shows: The full n8n ingestion workflow canvas. The webhook trigger on the left fans out into the validation node, then the dedup-by-phone and dedup-by-email checks (run in parallel), then either the merge branch or the create branch, then the Airtable mirror write. Failed records are routed to the Exception Logs write node at the bottom.



Screenshot: Test Lead Sync Confirmation — GHL → Airtable → Close.com

Leads ▾

⊕ Add filter 2 Leads (Set limit)

☐ Name	📞	✉	Contacts	Status
☒ Ali Zaman Khan	📞	✉	Ali Zaman Khan	● New Lead

What this shows: Result row from the end-to-end test lead. The same lead appears in GHL (source), Airtable Leads table (mirrored), and Close (queued in New Leads smart view), all sharing the same `universal_lead_id`. This is the proof that the three-way sync chain is intact.



Screenshot: Test Lead — Appearance in Close New Leads smart view

1 — New Leads *

Filters 1 1 Leads (Set limit)

Find Leads matching all of

Where: all of Total number of calls: 0 X Date of status: NEW LEAD This week X Number of emails in inbox: NONE X

AND Add filter Add group

Name			Contacts	Status
Ali Zaman Khan			Ali Zaman Khan	New Lead
+ Calculation			+ Calculation	

What this shows: The test lead row visible in the Close New Leads smart view, with `universal_lead_id` populated. This is the SDR's view of a freshly ingested lead — the moment it lands here, the SDR can pick it up and start calling.

✓ **Test lead confirmed synced: GH L → Airtable → Close.com. Appeared in New Leads smart view in Close. Dedup and exception logging verified.**

3B — Sync Direction Architecture

There are exactly three sync directions in the system, and each one has a strict ownership rule. The ownership rule is what prevents data corruption: if GH L is the owner of the 'lifecycle' field, then Close and Airtable can read it, but writes from Close or Airtable back into that field are blocked at the n8n layer.

Sync 1: GH L → Close — lead record mirror. Close gets a read-only copy of the lead so SDRs can dial. Reverse writes are blocked.

Sync 2: Close → Airtable — call activity and dispositions only. Every call logged in Close shows up in the Airtable Call_Activities table within seconds.

Sync 3: GH L → Airtable — lifecycle state, booking events, revenue outcomes. The big-picture business data lives here.

Sync	From	To	Data Transferred	Status
Sync 1	GH L	Close	Lead record mirror only — no reverse overwrite allowed	✓ Done
Sync 2	Close	Airtabl e	Call activity + dispositions only	✓ Done
Sync 3	GH L	Airtabl e	Lifecycle state + booking events + revenue outcomes	✓ Done

Workflow Links

- [GHL → Airtable Lifecycle Sync — n8n](#) — GHL pipeline stage changes sync to the Airtable reporting_status field.
- [GHL → Airtable Sync — GHL Workflow](#) — Triggers on lifecycle events and sends to n8n for downstream Airtable update.
- [Close → Airtable Call Activity Sync — n8n](#) — Call logged in Close triggers sync to Airtable Call_Activities table, carrying disposition + universal_lead_id.
- [Outbound Zap](#) — Outbound call-activity sync via Zapier (parallel path for outbound calls).
- [Inbound Zap](#) — Inbound call-activity sync via Zapier (parallel path for inbound calls).



Screenshot: GHL → Close Sync — One-way data flow confirmed

☐ Name	Email zaman.agentum@gmail.com		Status
☒ Ali Zaman Khan		Ali Zaman Khan	New Lead

What this shows: Confirmation row in n8n showing the GHL → Close sync running with reverse-write blocked. The arrow direction in the workflow is unidirectional by design — Close is a consumer of GHL lead data, not a writer.



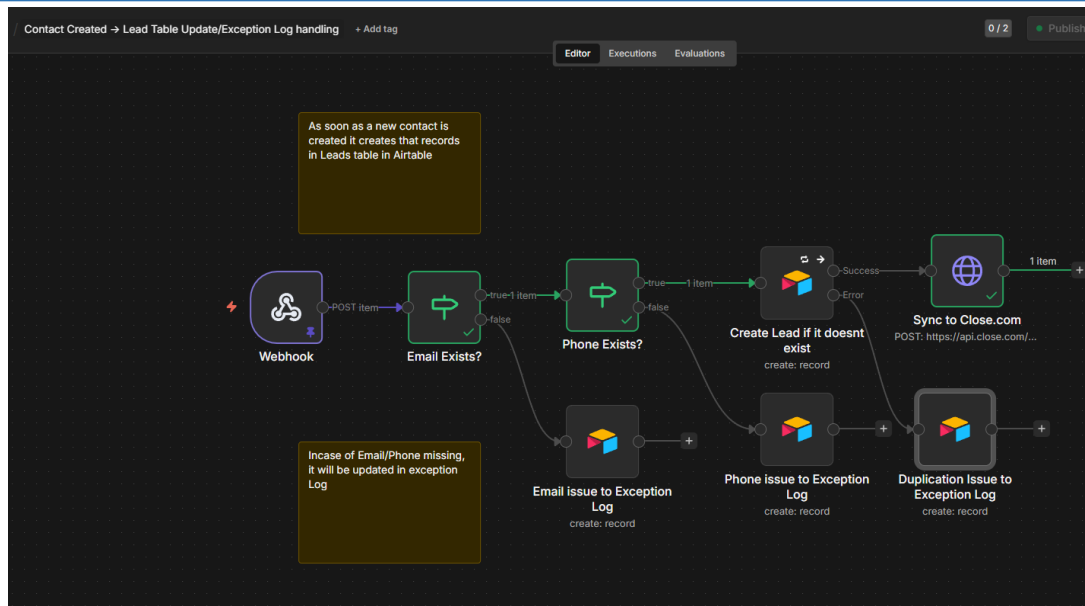
Screenshot: GHL → Close Sync — Reverse sync blocked confirmation

☐ Name	Call +92 370 4081897 Estimated Local Time: 3:10 PM PKT Based on phone number(s)		contacts	Status
☒ Ali Zaman Khan			Ali Zaman Khan	New Lead

What this shows: Log entry showing a blocked attempt to write from Close back to GHL on an owned field. The block is enforced inside the n8n routing logic, with the rejected write written to Exception Logs as 'Ownership Violation'.



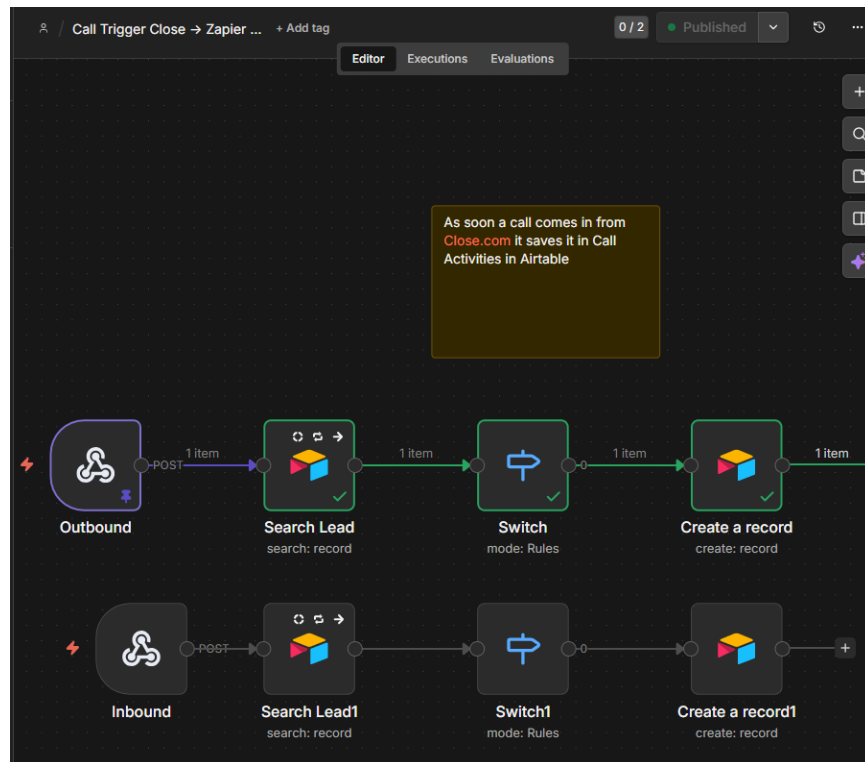
Screenshot: GHL → Airtable → Close.com — Full sync chain confirmed



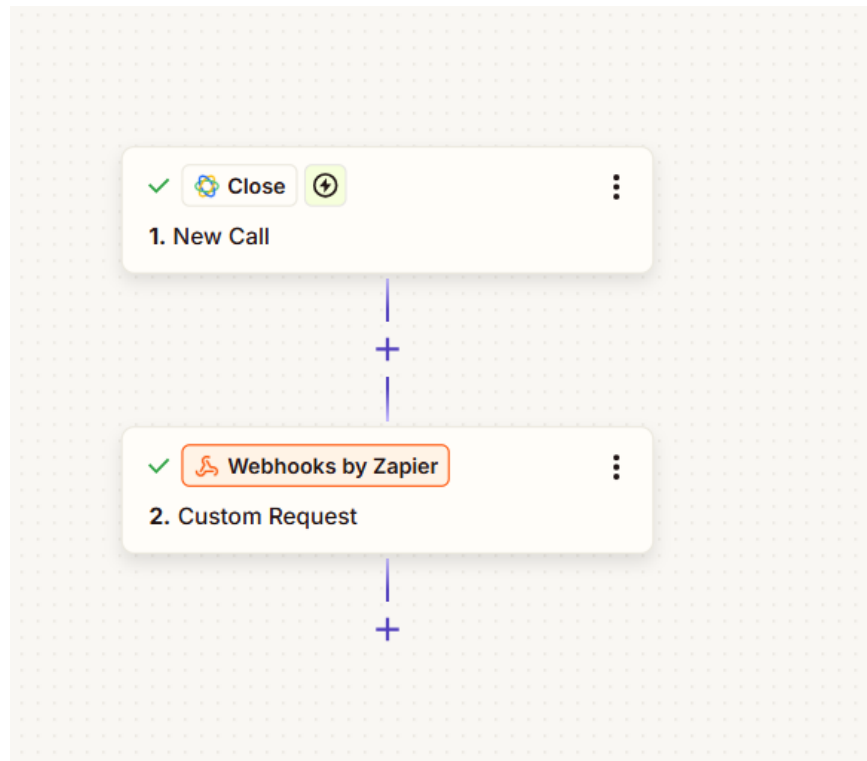
What this shows: End-to-end view of the three-way sync running on a real record. `Universal_lead_id` is consistent across all three systems, the lifecycle stage matches in GHL and Airtable, and the lead is dial-ready in Close. Duplicates encountered during this run were routed to Exception Logs.



Screenshot: Close → Airtable Sync — Call Activity workflow (n8n)

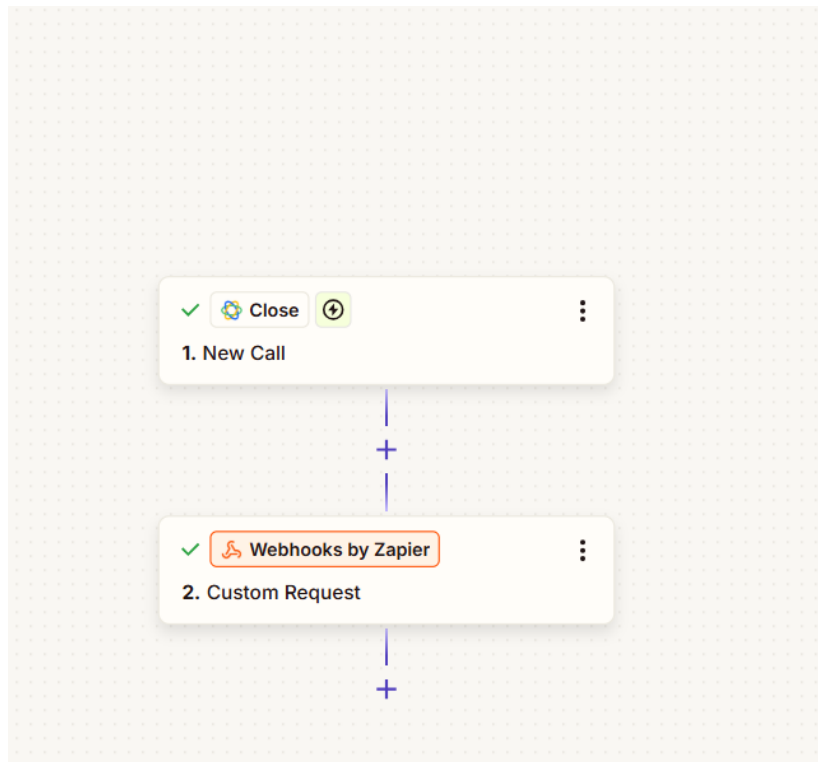


What this shows: The n8n workflow that listens for Close call events and writes each call into the Airtable `Call_Activities` table. The write node carries both the disposition and the `universal_lead_id` so each call is queryable by lead and by rep.

**Screenshot: Outbound Zap — Close to Airtable call activity sync**

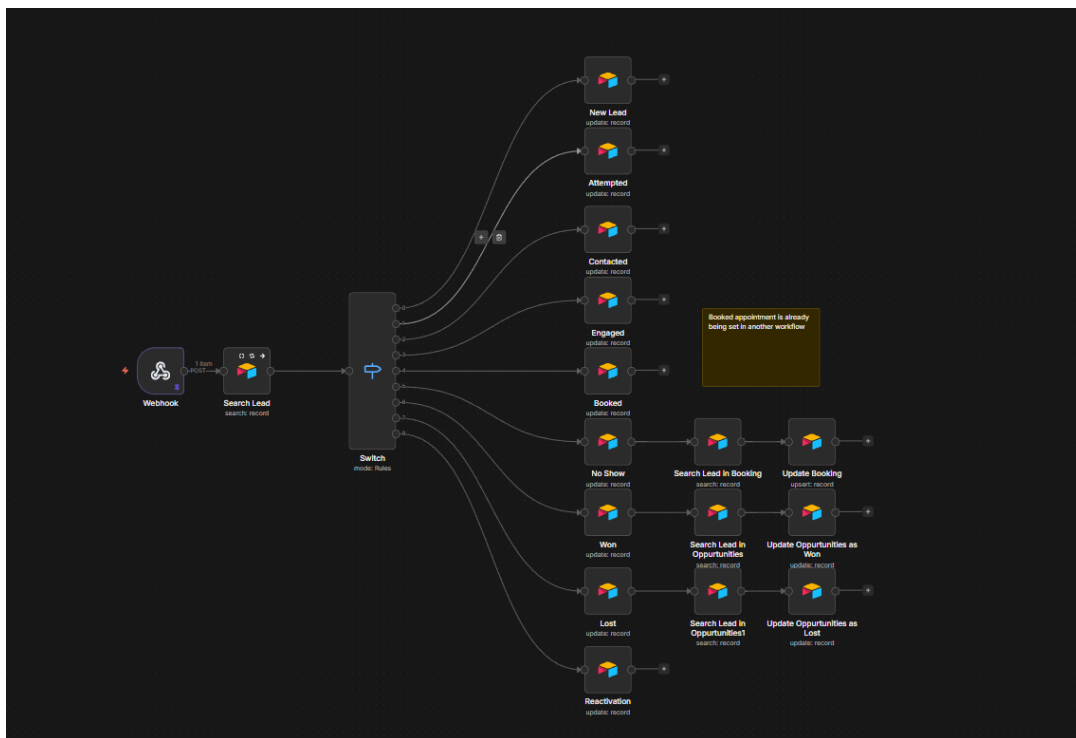
What this shows: The Zapier zap handling outbound calls from Close into Airtable. Zapier is used as a parallel path alongside n8n for call activity — this gives the system redundancy. If n8n is briefly down, the Zap still captures the call.

**Screenshot: Inbound Zap — Inbound call activity sync**



What this shows: Sister zap to the Outbound Zap, this one handles inbound calls. Same destination (Airtable Call_Activities), different trigger (Close inbound call event).

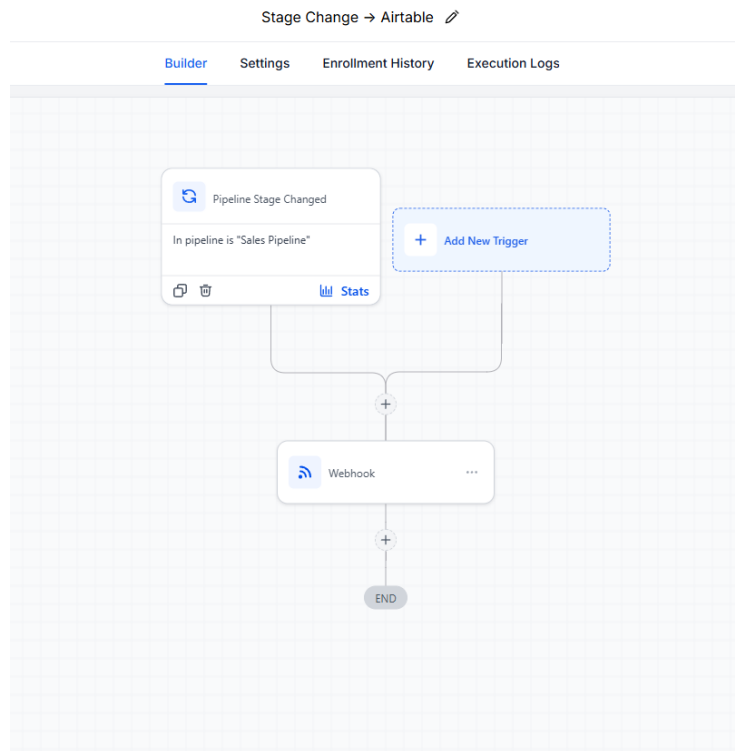
 Screenshot: GHL → Airtable Lifecycle Sync — n8n workflow



What this shows: The n8n workflow that pushes GHL pipeline-stage changes into the Airtable Leads.reporting_status field. Every time a lead moves from one stage to another in GHL, this workflow fires within seconds.



Screenshot: GHL → Airtable Sync — GHL Trigger Workflow



What this shows: The GHL-side trigger for the lifecycle sync. This is what wakes up the n8n workflow above. It runs on the GHL 'Opportunity Status Changed' event.

3C — Overwrite & Conflict Rules + Attribution Lock

Attribution lock is the most business-critical mechanism in the system. Once an SDR books an appointment, the booking is permanently theirs — no later edit, sync, or workflow can reassign credit. Same for the closer once the deal is Won or Lost.

The locks are enforced at the n8n routing layer, not at the Airtable or GHL field level. This means even if someone has direct database access, attempted overwrites are logged as Ownership Violations in the Exception Logs.

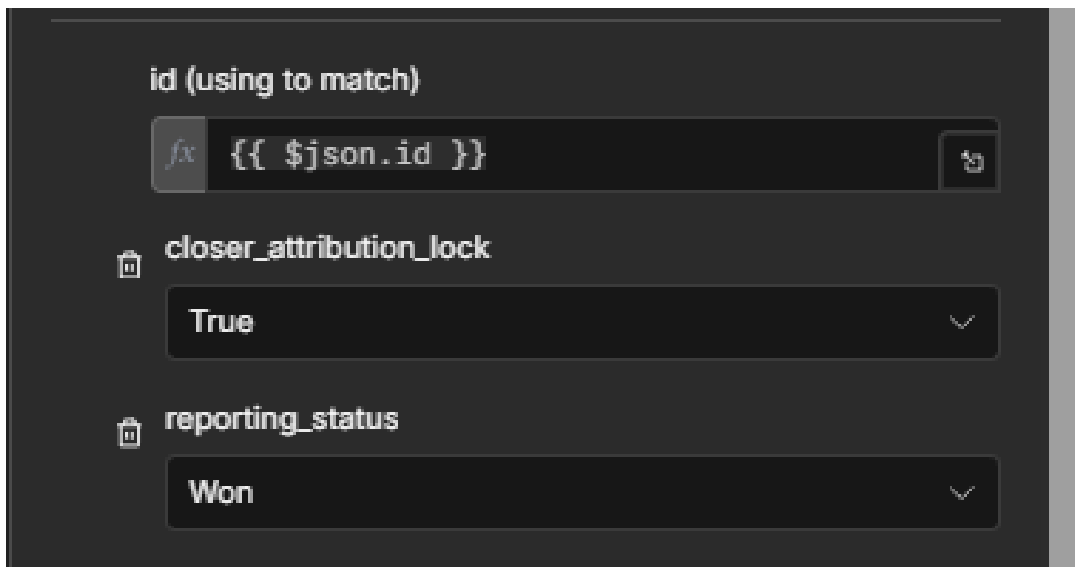
The five rules below are the enforceable contract: (1) system owner always wins, (2) non-owners are read-only on owned fields, (3) stale updates (older timestamps) are rejected, (4) SDR attribution locks at Booked, (5) closer attribution locks at Won/Lost.

- ✓ System owner always wins — enforced across all 3 sync directions
- ✓ Non-owner systems are read-only for owned fields in n8n
- ✓ Timestamp validation: reject stale updates

- ☒ SDR attribution locked permanently on Booked event — cannot be overwritten
- ☒ Closer attribution locked permanently on Won/Lost event — cannot be overwritten
- ☒ All blocked overwrites log to Airtable Exception_Logs as 'Ownership Violation'

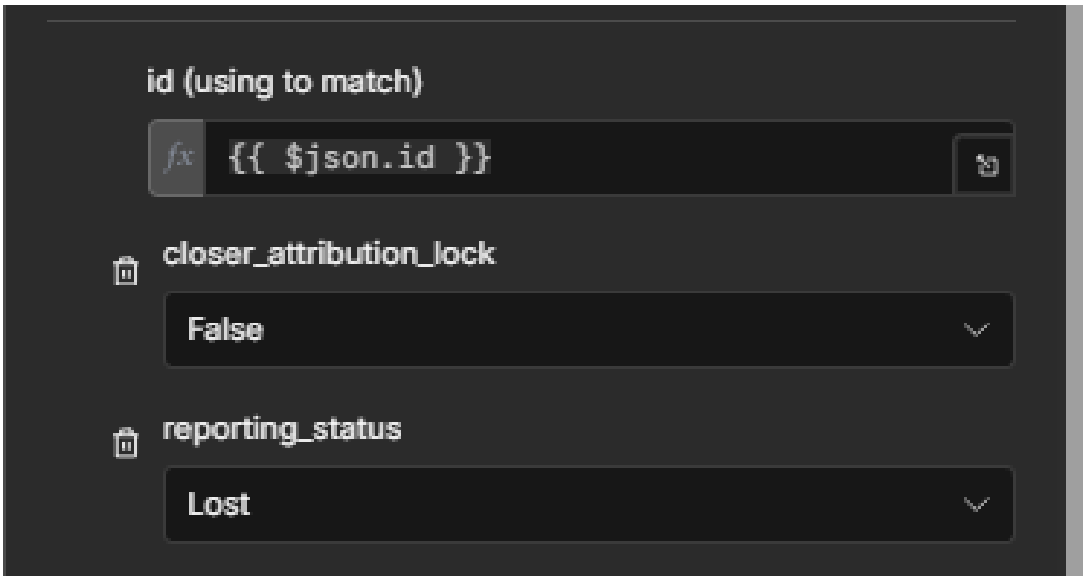
☒ **Override & Attribution Lock: COMPLETE. Only the owner can edit. Non-owners cannot edit.**
End-to-end lead creation test passed across all 3 systems.

 Screenshot: Attribution Lock Enforcement — Owner permission verification



What this shows: Verification screenshot showing the `sdr_attribution_locked` and `closer_attribution_locked` boolean fields flipped to `True` on a Booked/Won record. Once these flags are true, the n8n layer refuses any update to the `sdr_owner` or `closer_owner` fields on that record.

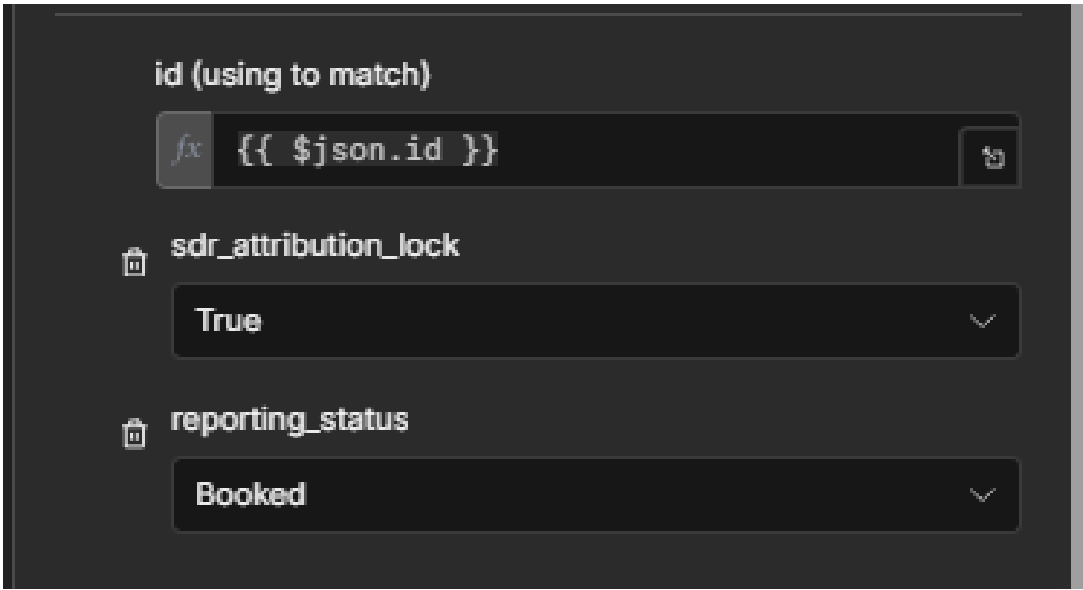
 Screenshot: Attribution Lock — Ownership violation logging verified



What this shows: An Exception Logs row showing a recorded Ownership Violation. The log captures the attempted change (old value → new value), the requesting system, the timestamp, and the lead's universal_lead_id. This is your forensic trail.



Screenshot: End-to-end lead creation test — Attribution locked across all 3 systems



What this shows: Final confirmation row: a test lead has been progressed through New → Attempted → Booked, the SDR attribution lock has fired, and the locked state is reflected identically in GHL, Airtable, and Close. End-to-end validation pass.

Phase 4 — Automation & Logic

Dates: 7–9 Apr · **Prerequisite:** Phase 3 complete and end-to-end sync validated · **Status:** In Progress

Phase 4 is the layer that makes the system act on data, not just record it. There are 11 event→action automations specified. The first six (and the reactivation engine, automations 10 and 11) have been built; the remaining four are scoped but not yet live.

Each automation has a single trigger and a single primary action. They are intentionally narrow — composability comes from chaining them through state changes in GHL, not from building monolithic mega-workflows. If you need to add new logic, prefer adding a new narrow automation over extending an existing one.

4A — Event → Action Automation Map (11 Automations)

The table below is the canonical list of automations. Status reflects implementation, not testing — automations marked Done are built and live, but full end-to-end validation of all 11 happens in Phase 5.

#	Trigger	Action	Status
1	Close call logged → Airtable	Log call activity to Airtable (retry on fail)	✓ Done
2	Close: Booked disposition	Create/update booking in GHL, move pipeline to Booked, assign closer	✓ Done
3	Close: Call Back Scheduled	Auto-create task in Close	✓ Done
4	Close: Interested — Send Info	Trigger follow-up workflow in GHL via n8n	✓ Done
5	Close: Not Interested / Not Qualified	Mark for Closed Lost prep in GHL	✓ Done
6	Close: Voicemail / No Answer	Update Airtable attempt status only — no pipeline movement	✓ Done
7	GHL: Booking confirmed	Update Airtable + lock SDR attribution	□ Pending
8	GHL: No-Show marked	Trigger reschedule workflow + update Airtable	✓ Done
9	GHL: Won	Lock revenue + closer attribution in Airtable	□ Pending
10	GHL: Lost	Set reactivation eligibility + log to Reactivation_Log	✓ Done
11	n8n: Reactivation trigger	Re-inject lead into Close queues	✓ Done

Automations 1 & 2 — Call Logged + Booked Disposition

Automations 1 and 2 are the core call-handling pair. Automation 1 logs every Close call into Airtable. Automation 2 specifically handles the 'Booked' disposition — when an SDR books an appointment via Close, this automation creates the matching GHL booking, moves the pipeline to Booked, and assigns the closer.

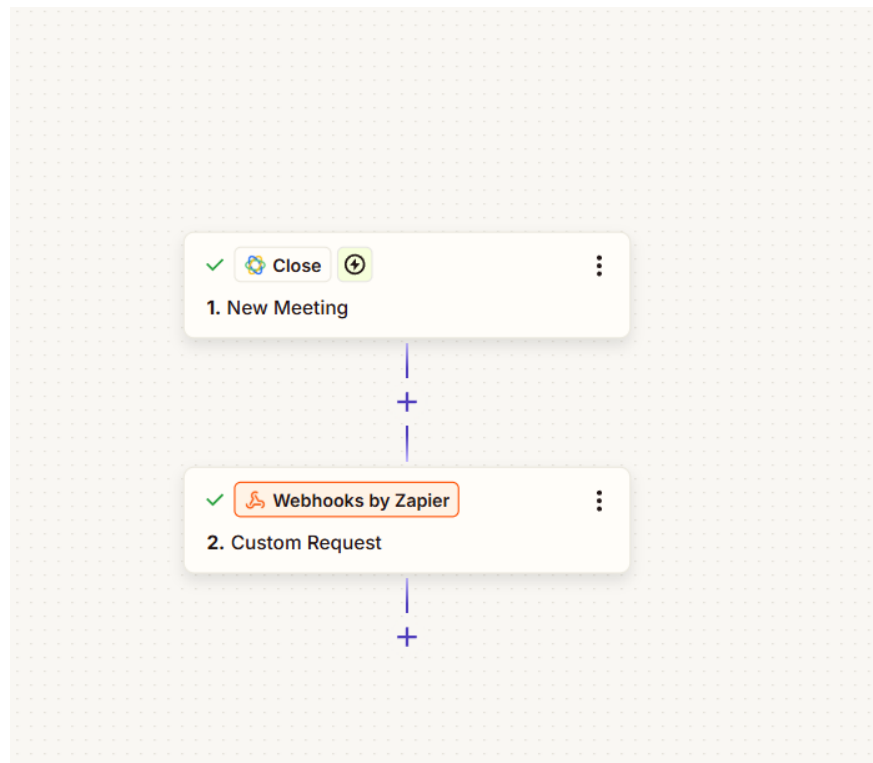
→ [Automation 1 & 2 — Zapier](#) — Handles booking and call-activity automations via Zapier.

→ [n8n Workflow — Automations 1 & 2](#) — The n8n side of the automation pair — appointment details get placed into the GHL calendar section.

→ [GHL Booking Workflow](#) — Existing flow: captures appointment created and posts into Airtable Bookings table.



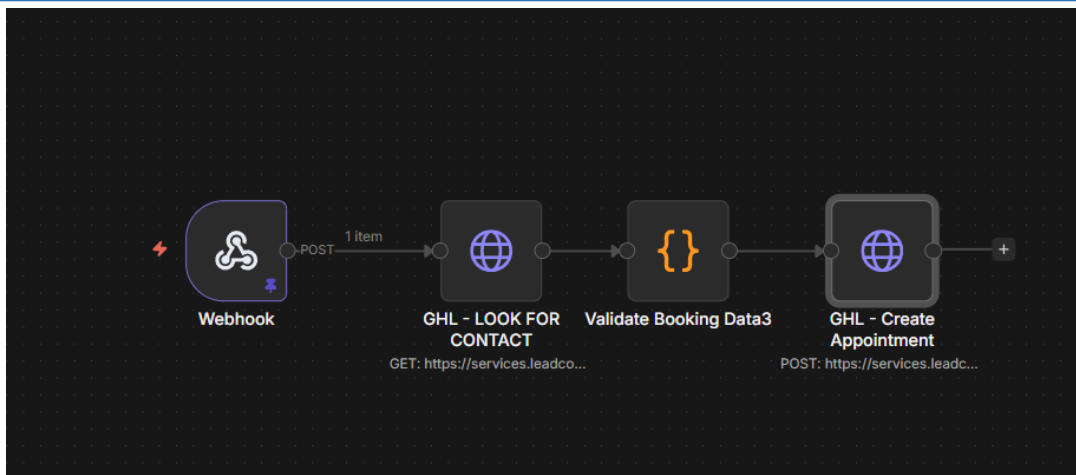
Screenshot: Automation 1 & 2 — Zapier: Call activity + Booking automation



What this shows: Zapier workflow editor showing the trigger (Close call logged) and the action steps for both call-activity logging and booking creation. This is the Zapier-side implementation that runs in parallel with the n8n workflow for redundancy.



Screenshot: Automation 2 — n8n: Appointment details in GHL calendar section



What this shows: n8n workflow showing the node that takes the booking payload and writes it into the GHL calendar event. This is what makes the booked appointment visible in the closer's GHL calendar.

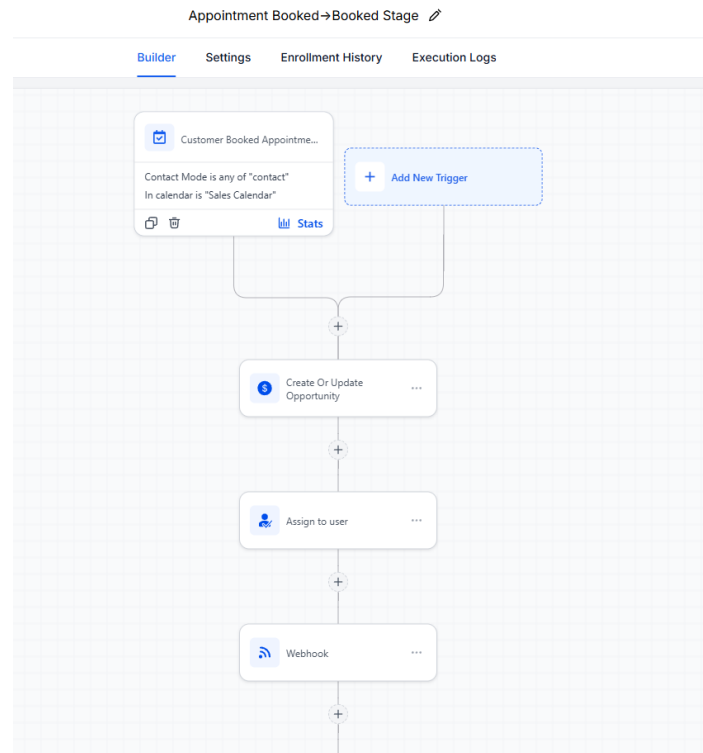
Screenshot: Automation 2 — n8n: Booking workflow detail

```
Result
Item 0 < >
Text Html Mark

{
  "calendarId": "H4J2ST6jcFN7VgqvEivG",
  "locationId": "42jvLMJz8d2exYKuZzYR",
  "startTime": "2023-11-16T08:00:00-05:00",
  "title": "Intro call with John Doe",
  "appointmentStatus": "confirmed",
  "assignedUserId": "",
  "address": "",
  "ignoreDateRange": false,
  "toNotify": true,
  "contact": {
    "firstName": "ali zaman",
    "lastName": "khan",
    "email": "zaman.agentum@gmail.com",
    "phone": "+923704081897",
    "timezone": "America/Toronto"
  },
  "notes": "Business\nPhone: +923704081897\nEmail: zaman.agentum@gmail.com Description: We scheduled a demo with the prospect."
}
```

What this shows: Detail of the booking-creation node, showing the field-by-field mapping from the Close booking payload to the GHL appointment record (lead name, time, closer assignment, source).

Screenshot: Automation 2 — GHL: Booking appointment details in calendar



What this shows: Result view in GHL — the appointment created by Automation 2 visible in the closer's calendar, complete with the lead's `universal_lead_id` and the SDR source noted. This is the existing flow that captures appointment-created events and pastes them into Airtable.

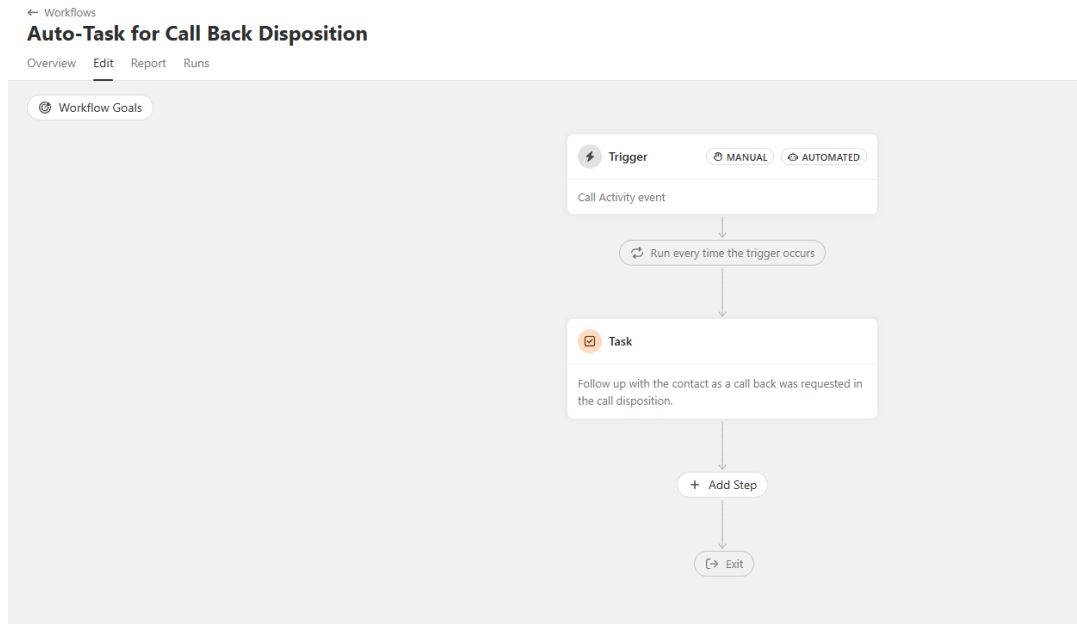
Automation 3 — Call Back Scheduled → Auto-create Task in Close

Automation 3 handles the 'Call Back Scheduled' disposition. When an SDR ends a call with that disposition, a follow-up task is auto-created in Close with the scheduled callback time as the due date. The SDR will see this task in the Call Backs smart view (once that view is created in Phase 2D).

→ [Close Sequence — Call Back Task](#) — As soon as a call back is booked (per the transcript of the call) it will auto-create a task in Close.com.



Screenshot: Automation 3 — Close Sequence: Call Back → Auto-create Task



What this shows: The Close sequence editor for Automation 3, showing the trigger (disposition = Call Back Scheduled) and the action (create task with due date = scheduled callback time). The task is auto-assigned to the SDR who handled the original call.

Automation 4 — Interested Send Info → Follow-Up Workflow

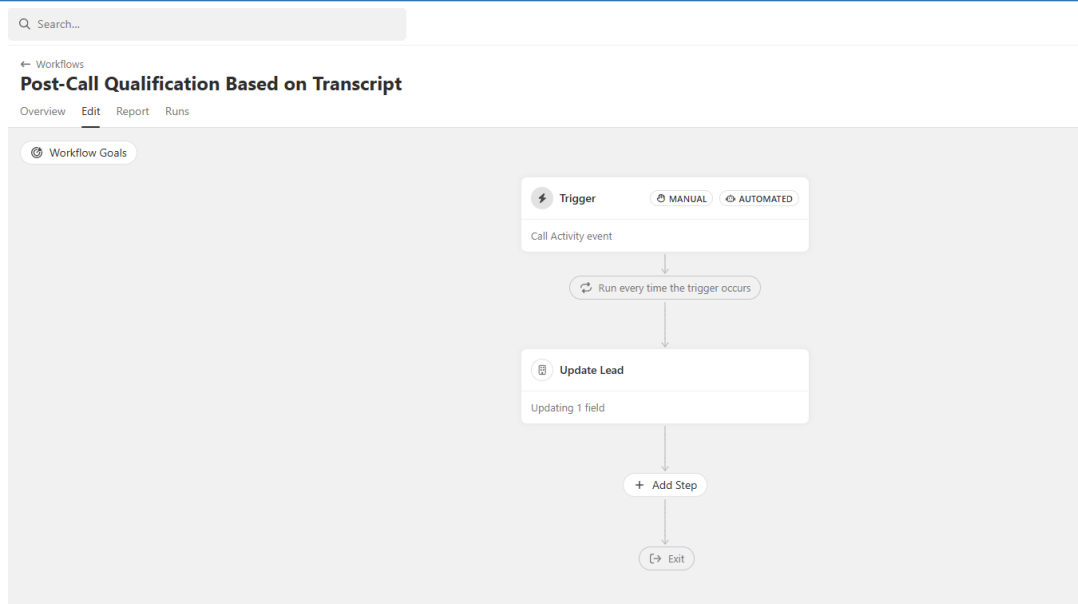
Automation 4 handles the 'Interested — Send Info' disposition. When an SDR marks a lead as interested but the lead hasn't yet booked, this automation triggers the GHL follow-up workflow, which sends nurture content over a 4-day window and then escalates the lead to Lost if no booking occurs (see GHL Follow-up Workflow in Phase 4B).

→ [Close Sequence — Interested Follow-up](#) — Triggers the follow-up workflow when 'Interested — Send Info' is selected as the disposition.

→ [Zapier — Interested Follow-up](#) — Parallel Zapier path for the Interested follow-up trigger.



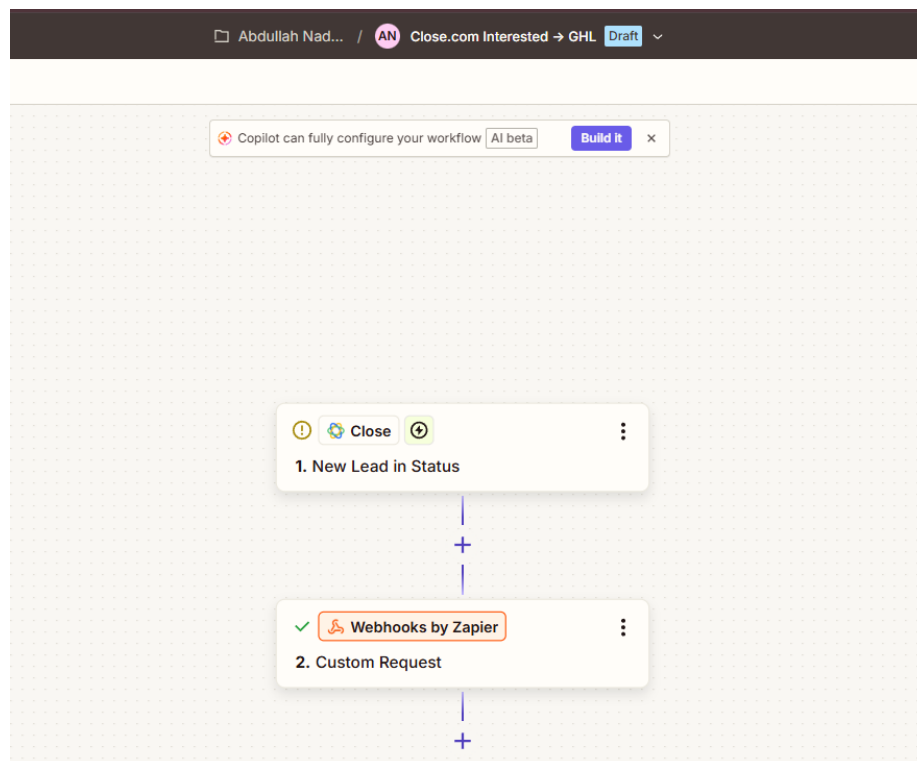
Screenshot: Automation 4 — Close Sequence: Interested → Follow-up workflow trigger



What this shows: The Close sequence configured to fire on 'Interested — Send Info' disposition. The next step is the webhook out to n8n which then triggers the GHL follow-up workflow.



Screenshot: Automation 4 — Zapier: Interested follow-up



What this shows: The parallel Zapier zap for Automation 4 — same trigger (Close disposition = Interested — Send Info), same end goal (follow-up via n8n into GHL).

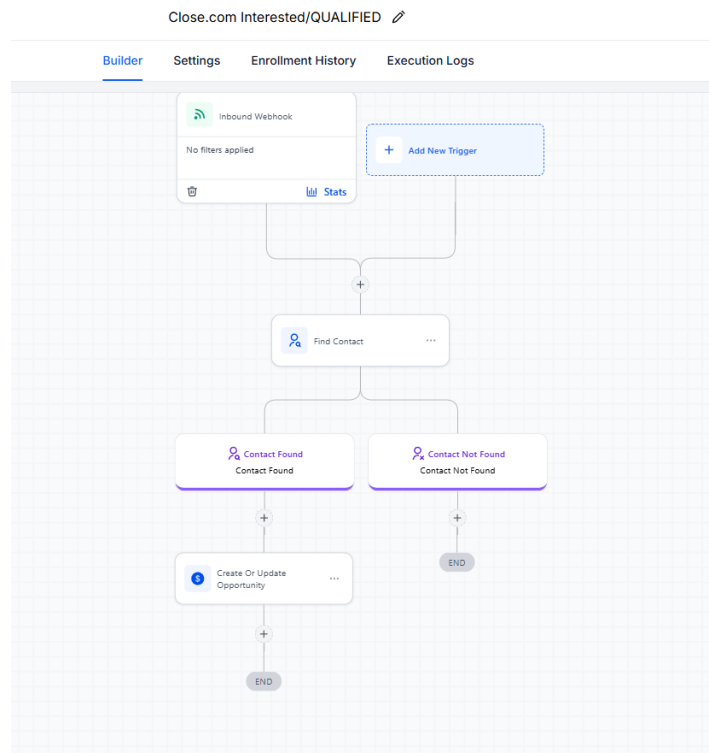
Automation 5 — Not Interested / Not Qualified → Closed Lost Prep

Automation 5 handles disqualification. When an SDR dispositions a call as 'Not Interested' or 'Not Qualified', this automation updates the GHL pipeline to mark the lead for Closed Lost preparation. The lead is not moved straight to Lost — it's flagged so that a final review can happen before the Lost stage triggers reactivation eligibility.

→ [GHL Workflow — Not Interested Update](#) — If someone appears not interested after a call, the pipeline state is updated accordingly.



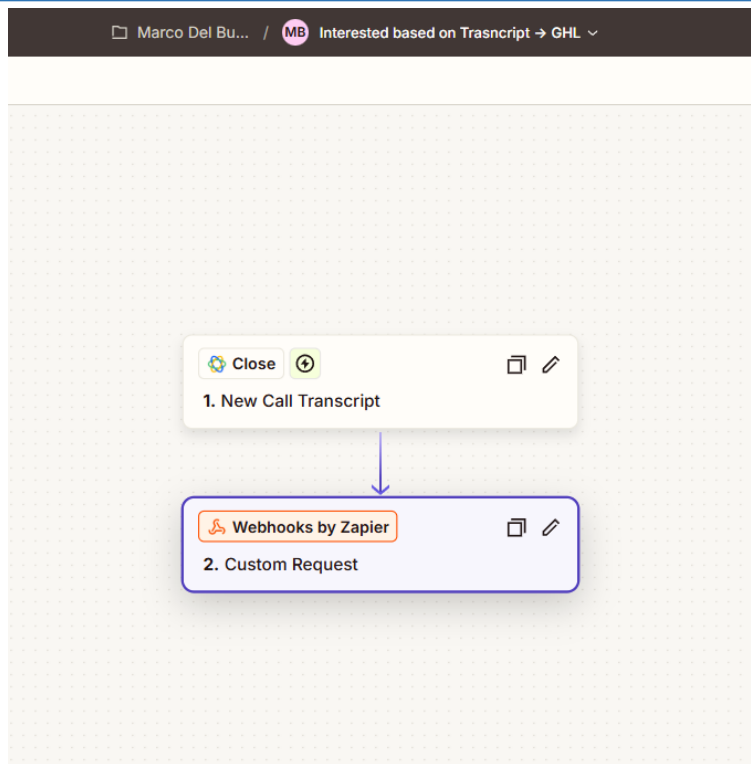
Screenshot: Automation 5 — GHL Workflow: Not Interested → pipeline update



What this shows: The GHL workflow canvas for Automation 5. Trigger: disposition arrives from Close as 'Not Interested' or 'Not Qualified'. Action: pipeline state moves to a pre-Lost holding state, ready for Closed Lost prep.



Screenshot: Automation 5 — GHL: Not Interested pipeline state confirmation



What this shows: Result view — a test lead that hit Not Interested disposition is now visible in the Closed Lost prep state in the pipeline. From here, a closer or manager makes the final Lost/keep decision.

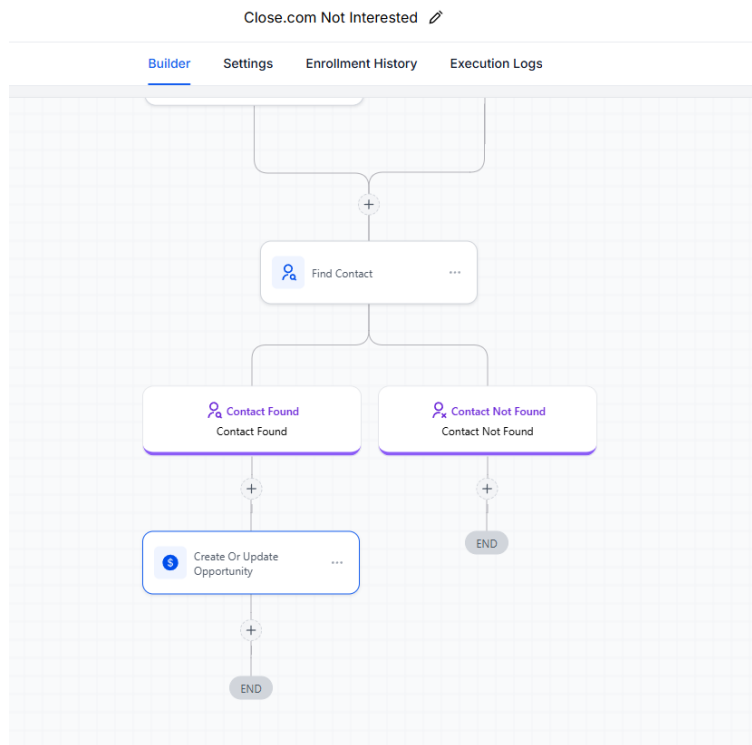
Automation 6 — Voicemail / No Answer → Airtable Attempt Status Only

Automation 6 handles non-events — voicemail left, no answer. These dispositions should NOT move the pipeline (the lead is still active, just not reached yet). All this automation does is increment the attempt counter in Airtable. Pipeline movement is intentionally suppressed.

→ [GHL Workflow — Voicemail/No Answer](#) — If someone is on Voicemail/Not Answered, this updates Airtable attempt status only — NO pipeline movement.

→ [n8n Workflow — Voicemail/No Answer](#) — The n8n handler for voicemail/no answer events.

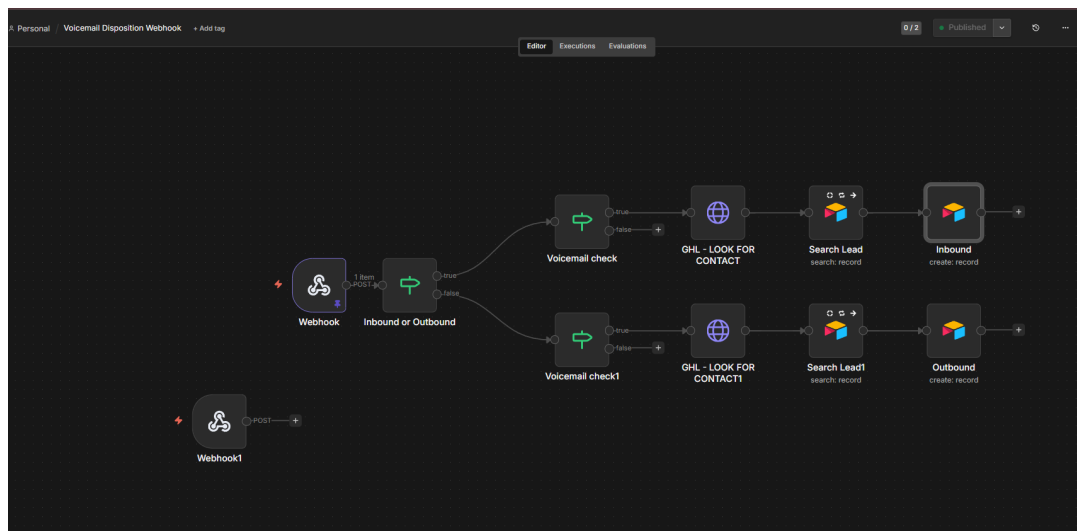
 **Screenshot: Automation 6 — GHL Workflow: Voicemail/No Answer → Airtable attempt status only (no pipeline move)**



What this shows: GHL workflow showing the explicit 'do not move pipeline' branch — only the Airtable `attempt_count` and `last_attempt_date` fields are updated. The pipeline stage is left unchanged so the lead remains workable in its current queue.



Screenshot: Automation 6 — n8n: Voicemail/No Answer handler



What this shows: The n8n side of Automation 6 — receives the disposition event, writes the attempt update to Airtable, and explicitly does not emit a pipeline-change event back to GHL. Compare and contrast with Automations 2/5 which DO move the pipeline.

Automation 8 — GHL No-Show → Reschedule Workflow + Airtable Update

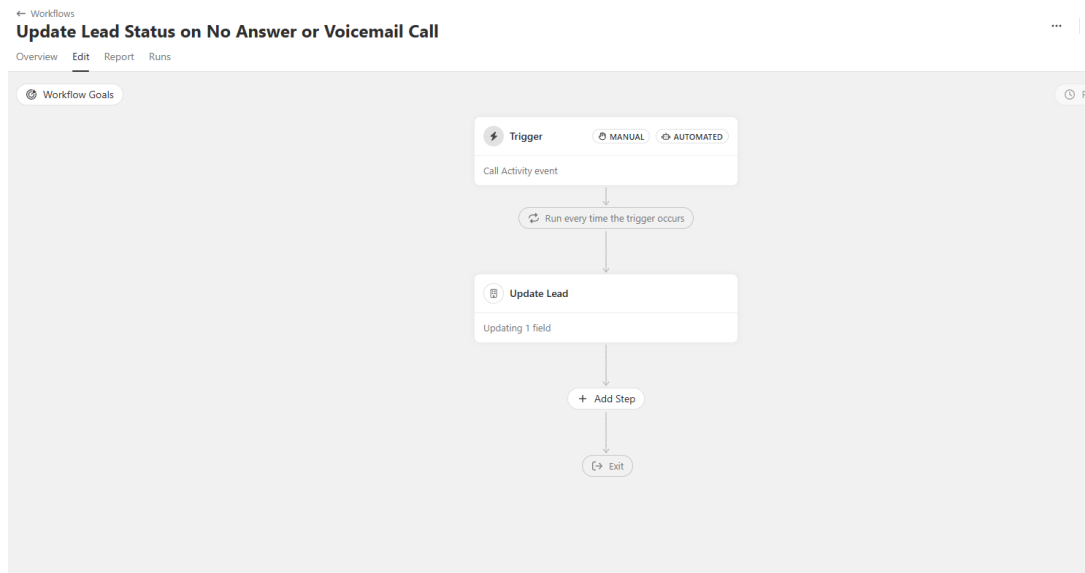
Automation 8 handles no-shows on booked appointments. When a closer marks a booking as no-show in GHL, this automation fires the reschedule workflow (which attempts to re-book the lead automatically through Close), and writes the no-show state back to the Airtable Bookings record so it's reflected in the closer scorecard.

→ [Close Sequence — No-Show Reschedule](#) — No-Show → reschedule workflow trigger.

→ [GHL Workflow — No-Show Reschedule](#) — Triggers the GHL reschedule workflow and updates the Airtable Bookings record on No-Show.



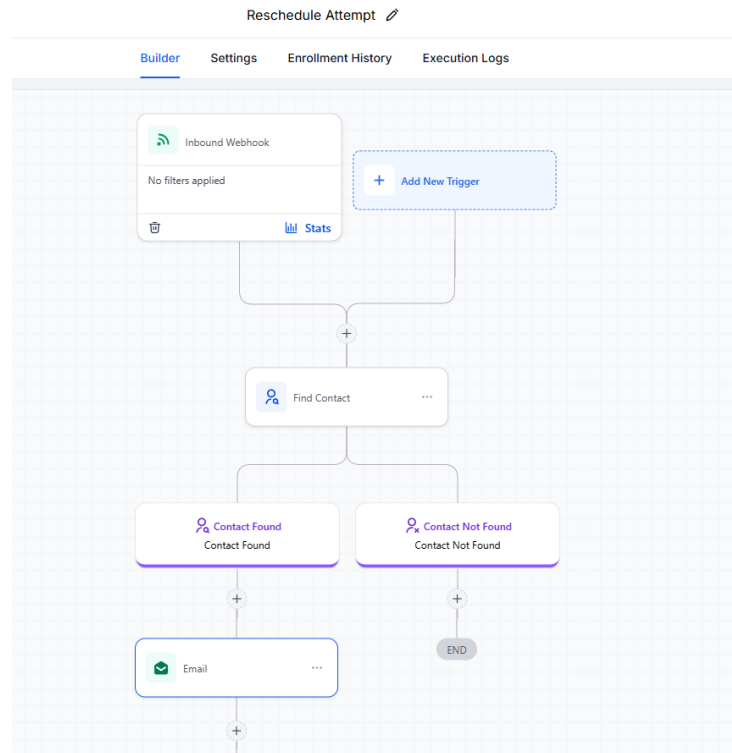
Screenshot: Automation 8 — Close Sequence: No-Show reschedule trigger



What this shows: The Close sequence that initiates the reschedule attempt when a closer marks a booking as no-show. Triggers a sequence of outbound attempts to re-engage the lead.



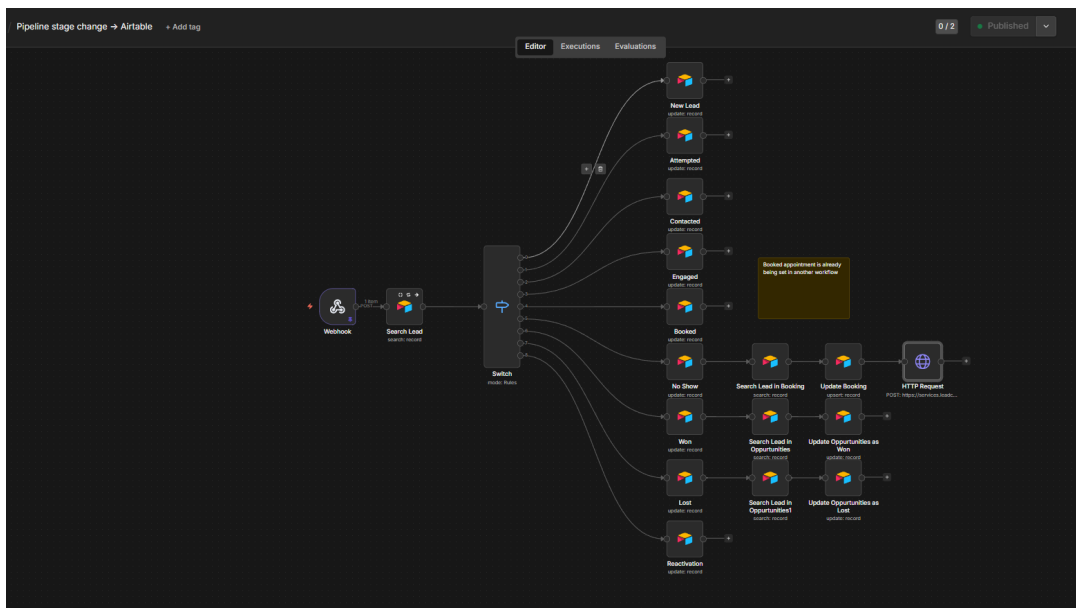
Screenshot: Automation 8 — GHL Workflow: No-Show → reschedule + Airtable update



What this shows: GHZ workflow canvas. Trigger: appointment marked No-Show. Actions: fire reschedule workflow (Close sequence above) AND write `booking_status = 'no_show'` to Airtable Bookings.



Screenshot: Automation 8 — GHZ: No-Show workflow detail



What this shows: Detail view of the No-Show workflow steps, including the explicit Airtable write node and the conditional branch that only triggers the reschedule if the lead is not on the Do Not Call list.

Automations 10 & 11 — Reactivation Engine

Automations 10 and 11 are the reactivation pair. Automation 10 fires on the GHL 'Lost' event and decides whether the lead is eligible for re-injection (factoring in loss reason, DNC status, time-in-system). Automation 11 actually re-injects eligible leads back into the Close queues after the cooling period expires.

→ [n8n — Lost → Reactivation Eligibility](#) — GHL Lost → sets reactivation eligibility and logs entry into the `Reactivation_Log`.

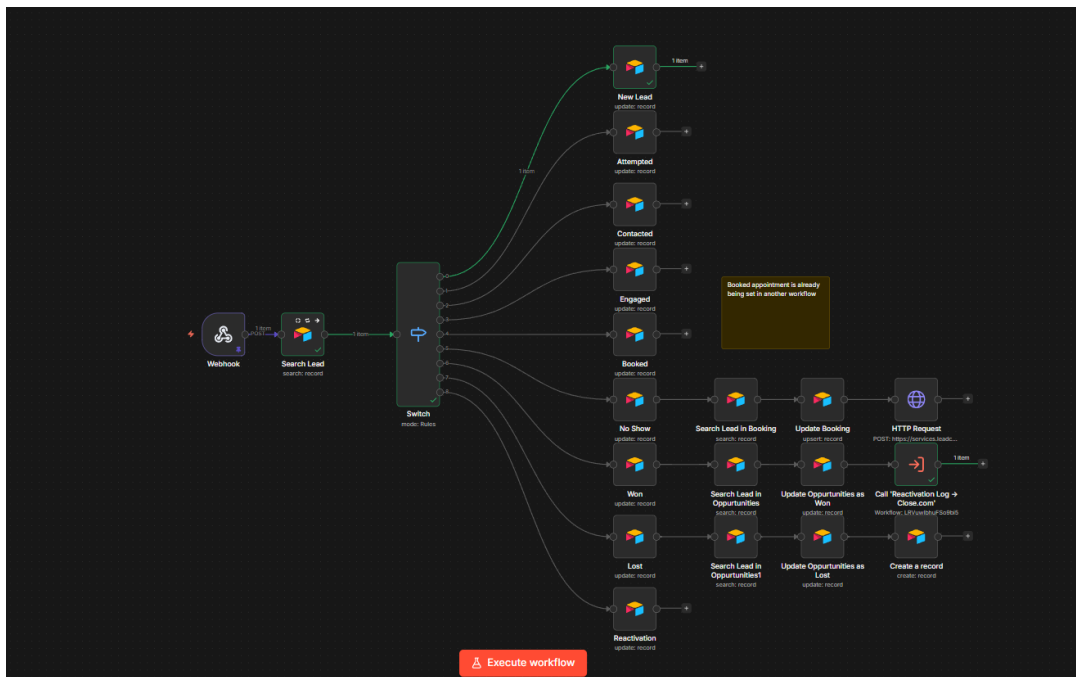
→ [n8n — Reactivation Re-inject](#) — Reactivation trigger → re-injects the lead into Close queues.

→ [Reactivation Zapier](#) — Published Zapier automation for the reactivation flow (redundant path).

→ [GHL Reactivation Entry Workflow](#) — GHL workflow handling the reactivation pipeline entry — moves the lead to the Reactivation stage, clears stale tasks, re-assigns the SDR.



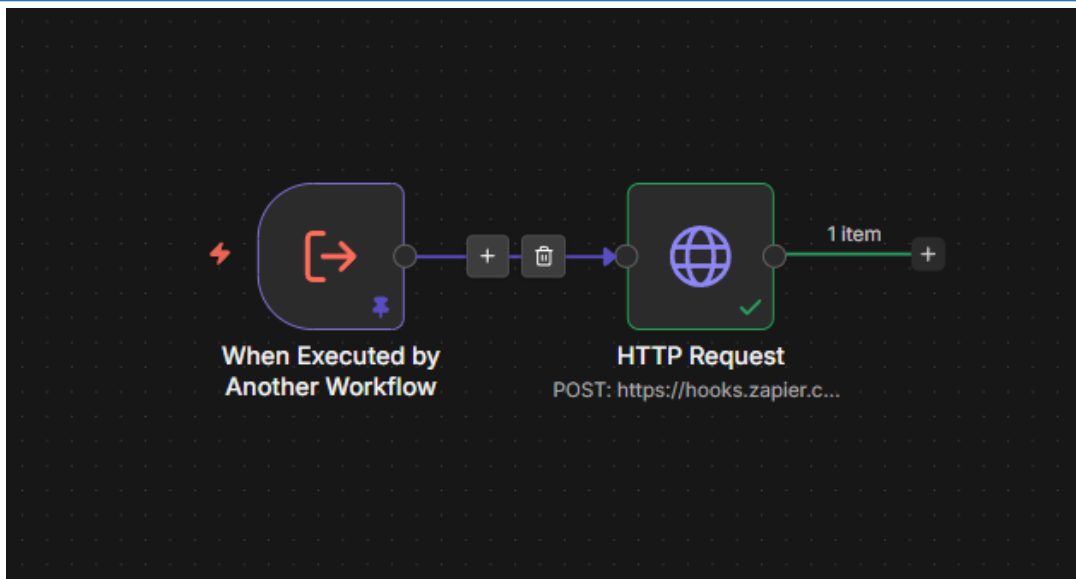
Screenshot: Automation 10 — n8n: GHL Lost → reactivation eligibility check



What this shows: n8n workflow for Automation 10. Trigger: GHL pipeline moves to Lost. Action: evaluate eligibility (DNC? time-in-system? loss reason?) and write the result to both the Lead record and the `Reactivation_Log` table.



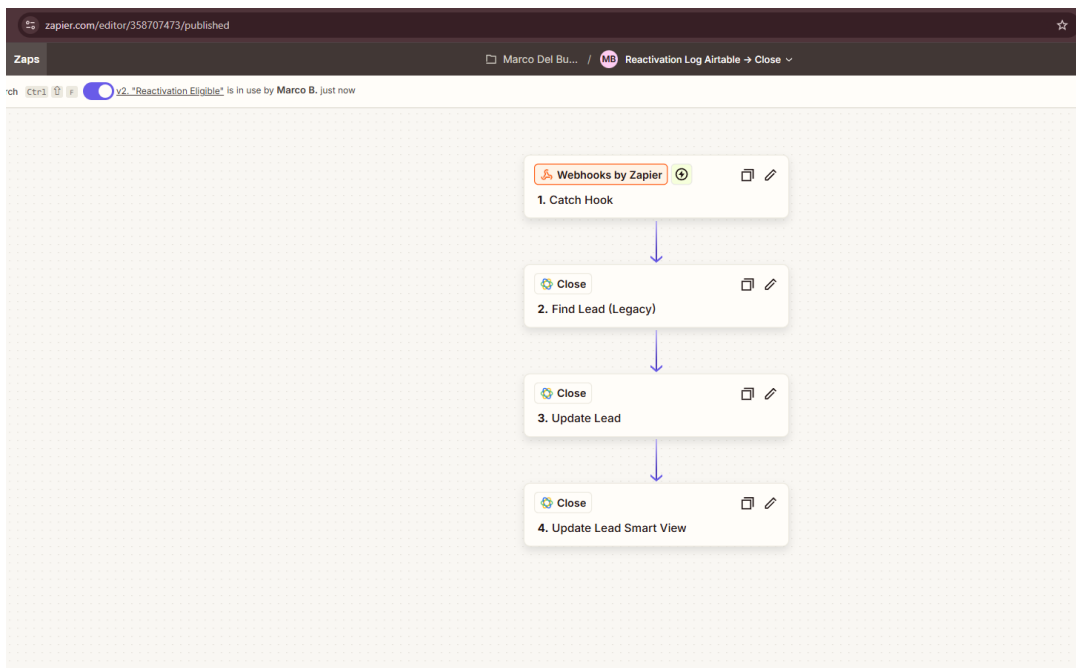
Screenshot: Automation 11 — n8n: Reactivation trigger → re-inject into Close queues



What this shows: n8n workflow for Automation 11. Trigger: a lead's reactivation_status flips to 'Ready for Reactivation' (cooling period expired). Action: re-inject the lead into Close, moving it back into the active New Leads / Aged Leads smart view.



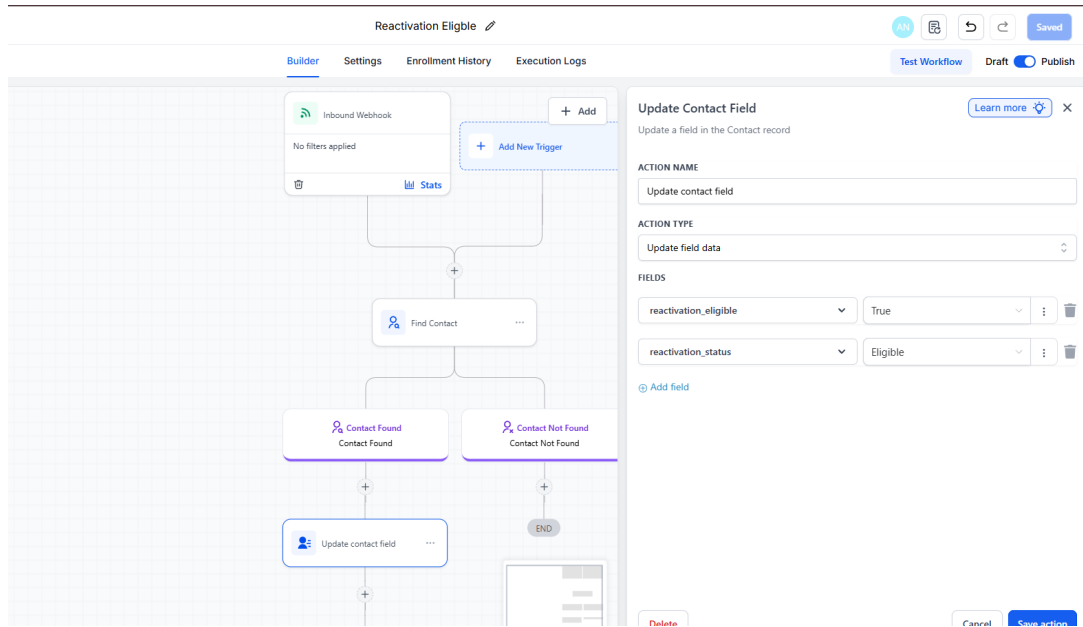
Screenshot: Reactivation — Zapier published automation



What this shows: The Zapier-side published automation that runs in parallel with the n8n reactivation workflows. Provides redundancy for the re-injection step.



Screenshot: GHL Reactivation Entry Workflow — lead re-enters pipeline at Reactivation stage



What this shows: The GHL workflow that finalises re-entry: moves the lead's pipeline stage to 'Reactivation', clears any stale tasks left from the previous cycle, and re-assigns the SDR. After this fires, the lead is workable again in Close.

4B — GHL Workflows: Follow-up & Prospects

Follow-up Workflow — Interested but not booked

The follow-up workflow handles leads that are interested but haven't booked yet. After 4 days of automated nurture (email + SMS depending on contact preferences), if the lead still hasn't booked, the workflow auto-moves them to Lost. This prevents the pipeline from clogging with stagnant interested-but-unbooked leads.

→ [GHL Follow-up Workflow — Interested](#) — If someone is interested in services and hasn't booked, this workflow follows up for 4 days and then adds them to Lost.



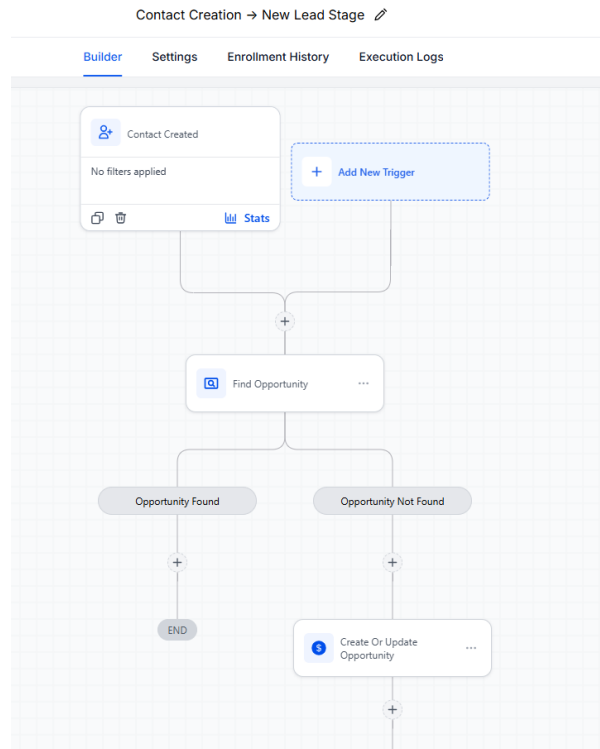
Screenshot: GHL Follow-up Workflow — Interested leads: 4-day follow-up sequence then Lost

Prospects Added to GHL Workflow

→ **GHL — Prospects Added** — Handles prospect contact creation into the GHL pipeline.

→ [Zapier — Prospects Intake](#) — Published Zapier automation for prospect intake.





What this shows: The GHG workflow for prospects added directly into GHG. Trigger: new contact created with prospect tag. Actions: assign universal_lead_id, post to n8n for Airtable sync, place into the New stage of the pipeline so the normal flow takes over from there.

4C — Reactivation Engine Status Values

The reactivation_status field on the Leads table moves through the following values over the life of a Lost lead. Automations 10 and 11 are responsible for transitioning between these values.

Status Value	Meaning & Trigger
Not Eligible	Default state on lead creation — has not been Lost yet.
Eligible	Lead marked Lost with valid loss reason — passes eligibility criteria.
Cooling Period	Cooling period timer active — no outbound contact allowed.
Ready for Reactivation	Cooling period expired — eligible for re-injection by Automation 11.
In Reactivation	Lead re-injected into Close — active outbound attempt cycle.
Reactivated	Lead converted after reactivation cycle (terminal success state).
Do Not Reactivate	Permanent exclusion — DNC leads, spam, invalid (terminal failure state).

 **Cooling period duration for reactivation is pending confirmation. Please provide this value (in days) before Automation 10's eligibility evaluator can be finalised. Default placeholder is 30 days; if a different value is needed, advise before Phase 5 testing begins.**

4. Implementation Status Checklist

This is the consolidated status view across all phases. Use it as a quick reference to know what is complete, what is pending, and what is blocked. Legend: ☒ Complete · ☐ Pending / In Progress.

Phase 1 — Foundation

- ☒ Airtable base created with all 9 tables (including QA Table)
- ☒ All Airtable field structures defined — snake_case, correct prefixes
- ☒ GHL sub-account confirmed — Location ID: 42jvLMJz8d2exYKuZzYR
- ☒ All GHL Custom Fields added to account
- ☒ Close account confirmed — API key configured
- ☒ n8n instance created at kronkai.app.n8n.cloud
- ☒ All 3 API connections tested and confirmed in n8n

Phase 2 — Core Structure

- ☒ GHL pipeline created with all 9 exact stage names — verified case-sensitive
- ☒ Confirmed as only pipeline in account (no duplicates)
- ☐ GHL booking system configuration — pending
- ☒ All GHL custom fields created per field mapping spec
- ☒ Ownership transfer rules configured
- ☒ Close Smart View 1: New Leads created
- ☐ Close Smart Views 2–5: Call Backs, Follow-Ups, Unreached, Aged Leads — pending
- ☐ All 10 Close dispositions configured — pending
- ☒ Airtable table relationships established via universal_lead_id
- ☒ Rep_Performance and Closer_Performance summary tables configured

Phase 3 — Data Flow

- ☒ Lead ingestion n8n workflow built — validation, dedup, normalization, GHL creation
- ☒ All GHL contact creation routed exclusively through the ingestion layer
- ☒ Bulk ingestion batch processing logic built
- ☒ GHL → Close sync built (lead mirror only)
- ☒ Close → Airtable sync built (call activity + dispositions only)
- ☒ GHL → Airtable sync built (lifecycle + booking + revenue)
- ☒ Overwrite/conflict rules enforced in n8n for all 3 sync directions
- ☒ Attribution lock enforcement built in n8n
- ☒ End-to-end lead creation test passed across all 3 systems

Phase 4 — Automation

- ☒ Automation 1: Close call logged → Airtable
- ☒ Automation 2: Close Booked → create GHL booking + move pipeline + assign closer
- ☒ Automation 3: Call Back Scheduled → auto-create task in Close
- ☒ Automation 4: Interested Send Info → follow-up workflow
- ☒ Automation 5: Not Interested / Not Qualified → Closed Lost prep
- ☒ Automation 6: Voicemail / No Answer → Airtable attempt status only
- ☐ Automation 7: GHL Booking confirmed → Airtable + SDR attribution lock
- ☒ Automation 8: GHL No-Show → reschedule + Airtable update
- ☐ Automation 9: GHL Won → revenue + closer attribution lock
- ☒ Automation 10: GHL Lost → reactivation eligibility + Reactivation_Log
- ☒ Automation 11: n8n Reactivation trigger → re-inject into Close
- ☐ Reactivation engine: all 4 gate conditions built into n8n — pending
- ☐ Daily + weekly reconciliation routines — pending

Phase 5 — Validation

- ☐ Won lifecycle test — pending
- ☐ Lost → Reactivation lifecycle test — pending
- ☐ DNC permanent exclusion confirmed — pending
- ☐ Bulk ingestion test with dedup + exception logging verified — pending
- ☐ All 5 go-live acceptance criteria confirmed — pending

Phase 6 — Operational Readiness

- ☐ SDR workflow validated in Close — pending
- ☐ All 10 Airtable reporting views operational — pending
- ☐ Final system health check passed — pending
- ☐ System handover summary report — pending

5. Access & Account Reference

Confirmed admin access on all four systems. The table below is the single source of truth for who-owns-access-to-what. If any of these statuses changes (e.g. an API key is rotated), this table must be updated and a new screenshot taken.

Item	Status
GHL sub-account admin access — Location ID: 42jvLMJz8d2exYKuZzYR	✓ Done
Close CRM admin access + API key confirmed	✓ Done
Airtable workspace access + API key confirmed	✓ Done
n8n instance live at kronkai.app.n8n.cloud	✓ Done

6. GHL Pipeline Stages — Quick Reference

This is the lifecycle map every system in Kronk aligns to. A lead can only be in one stage at a time. Stage changes in GHL fire the corresponding n8n sync workflows and any associated automation.

Stage	Entry Condition
New	Lead created via ingestion
Attempted	First outbound attempt from Close
Contacted	Two-way interaction confirmed
Engaged	Qualified conversation completed
Booked	Appointment scheduled — SDR attribution locks
No-Show	Appointment missed
Won	Deal closed — revenue and closer attribution lock
Lost	Deal not closed — reactivation eligibility evaluated
Reactivation	Re-entered after cooling period

7. Close Dispositions → Automations Map

Every Close disposition maps to exactly one downstream automation. This is the single source of truth for what happens after an SDR ends a call. Once Phase 2E is complete (disposition values entered into Close), the corresponding automations in Section 3 take over.

Disposition	Automation Triggered
No Answer	Update attempt count in Airtable only (Automation 6)
Voicemail Left	Update attempt status in Airtable only (Automation 6)
Wrong Number / Invalid	Flag for disqualification — exception log entry
Gatekeeper	Log and re-queue
Not Interested	Route to Closed Lost prep in GHL (Automation 5)
Not Qualified	Route to Closed Lost prep in GHL (Automation 5)
Interested — Send Info	Trigger follow-up workflow in GHL via n8n (Automation 4)
Call Back Scheduled	Auto-create task in Close (Automation 3)
Booked	Create booking in GHL, move pipeline, assign closer, lock SDR attribution (Automation 2)
Do Not Call	Permanent exclusion from all queues and reactivation engine

8. Airtable Reporting Views — Required

These ten views are the reporting deliverables for the operator and the rep teams. They are configured against the Airtable schema established in Phase 1 and populated by the sync flows established in Phase 3. Final view configuration is a Phase 6 deliverable.

View Name	What It Shows
Operator Daily Overview	Key metrics at a glance: lead volume, attempts, bookings, shows, wins — current period.
SDR Scoreboard	Per-rep: attempts, contacts, engagements, bookings, booking rate, connect rate, QA score.
Closer Scoreboard	Per-closer: appointments run, no-shows, wins, losses, close rate, revenue, revenue per booking.
Campaign / Source Performance	Leads, bookings, wins by source and campaign.
Booking Pipeline Reporting View	All booked appointments: status, assigned closer, revenue outcome.
Won / Lost Outcome View	All closed opportunities with outcome, loss reason, revenue, attribution.
Reactivation Performance View	Reactivation volume, booking rate from reactivated leads, cycle history.
Exception Queue	All open exceptions by severity — triage and resolution workflow.
Duplicate Review Queue	Records flagged as potential duplicates pending controlled review.
No-Show / Reschedule View	All no-shows with reschedule status.

9. Glossary & Naming Conventions

Use this section to decode any term, ID format, or convention used elsewhere in the document. If a term is used in the system but not in this glossary, please flag it for inclusion in the next document revision.

Field Naming Conventions

- All Airtable field names use snake_case (lower_case_with_underscores). This is enforced because n8n's field-mapping nodes compare strings exactly.
- All GHL custom field names also use snake_case to mirror Airtable. The pairing is what makes n8n's field map machine-readable.
- Pipeline stage names use Title Case with hyphens for compound words (e.g. 'No-Show', 'Booked', 'Reactivation'). These are CASE-SENSITIVE in the spec.

Key Terms

Term	Meaning
universal_lead_id	The single identifier that joins a lead's record across GHL, Close, and Airtable. Auto-assigned by GHL on contact creation, and propagated everywhere downstream.
Attribution Lock	A one-way irreversible state change. Once 'locked', the corresponding owner field (sdr_owner or closer_owner) cannot be edited by any system. Enforced at the n8n routing layer.
Ownership Violation	An attempted write by a non-owner system to an owned field. Logged to Exception_Logs with the attempted change and source system, never applied.
Cooling Period	The mandatory wait time between a lead being marked Lost and becoming eligible for reactivation. Duration is configurable — currently pending client confirmation.
Smart View	A saved filter in Close that defines a working queue for SDRs. SDRs work views top-to-bottom by priority order.
Disposition	The SDR's post-call classification of a call outcome. The single trigger for every Close-originated automation.
Sync Direction	A one-way data flow between two systems, brokered by n8n. There are exactly three: GHL→Close, Close→Airtable, GHL→Airtable. Reverse writes are blocked.
Exception Log	A row in the Airtable Exception_Logs table recording any failed write, validation rejection, duplicate, or ownership violation. The triage source for system health monitoring.

Status Conventions in This Document

-  Done — verified live in the production environment with a screenshot below.

-  Pending — scoped, specified, but not yet built or configured.
-  Action Required — needs explicit client input before work can proceed.
-  Architecture Lock — a non-negotiable rule that should never be modified without formal review.

End of Document

KRONK — Phase 1 Client Handover Document (Detailed Edition)

Prepared by Agentum AI | Internal Use Only | v3.0