

Behind the Workflow

How I turn operational work into systems people can actually follow.

Three systems I've built and run for coaching, consulting, and service businesses: a task **command center**, a Kajabi **onboarding automation**, and action-based **email journeys**. Each is shown with real, redacted evidence and the reasoning behind how it's structured. The point isn't the software — it's the decisions.

9

OPERATIONAL VIEWS

3

AUTOMATION PLATFORMS

1

OPERATING LOGIC

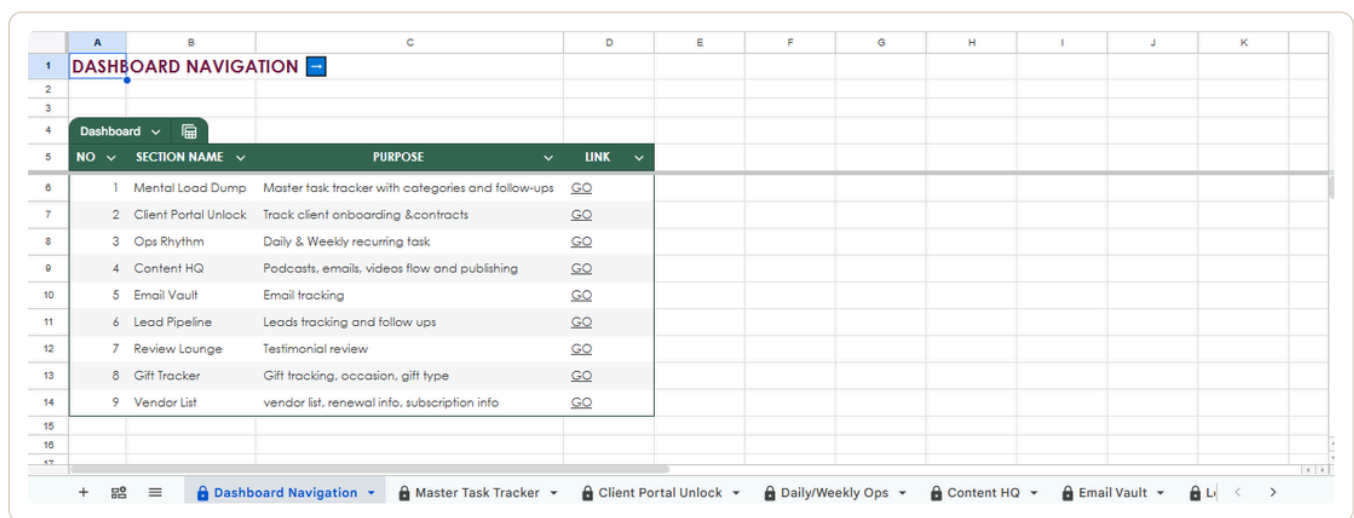
From scattered tasks to an operations command center

THE PROBLEM

The challenge wasn't tracking tasks — it was that different work needs different information. A client onboarding task doesn't need the same fields as a newsletter, a lead follow-up, or a recurring operational task.

Instead of forcing everything into one oversized tracker, I built a central command center with dedicated operational views.

THE ARCHITECTURE



NO	SECTION NAME	PURPOSE	LINK
1	Mental Load Dump	Master task tracker with categories and follow-ups	GO
2	Client Portal Unlock	Track client onboarding & contracts	GO
3	Ops Rhythm	Daily & Weekly recurring task	GO
4	Content HQ	Podcasts, emails, videos flow and publishing	GO
5	Email Vault	Email tracking	GO
6	Lead Pipeline	Leads tracking and follow ups	GO
7	Review Lounge	Testimonial review	GO
8	Gift Tracker	Gift tracking, occasion, gift type	GO
9	Vendor List	vendor list, renewal info, subscription info	GO

Command center navigation. One entry point routing to nine dedicated operational areas, each with its own tracker.

01	Mental Load Dump	Master task capture
02	Client Portal Unlock	Onboarding & contracts
03	Ops Rhythm	Daily & weekly recurring work
04	Content HQ	Publishing workflow
05	Email Vault	Email tracking
06	Lead Pipeline	Opportunities & follow-up
07	Review Lounge	Testimonials
08	Gift Tracker	Occasions & gift type
09	Vendor List	Renewals & subscriptions

WHY I STRUCTURED IT THIS WAY

One entry point, separate workflows. The dashboard is the control center; each operational area keeps its own tracker. The system stays easy to scan without losing the fields a specialized workflow actually needs.

THE TASK LOGIC

	A	B	C	D	E	F	G	H
1		MENTAL LOAD DUMP 🧠						
2			Mental Load: 13 tasks this week, 4 high, 0 to do related — clarity unlocked.					
3								
4								
5		Task Tracker						
6		TASK NAME	CATEGORY	STATUS	PRIORITY	CHANNEL	DUE DATE	STATUS TRIGGER
7	1	Login to gmail	Backend	Done	High	Email	14/07/2025	Overdue
8	2	Update email footer to	Backend	To-do	Medium	Email	14/07/2025	Overdue
9	3	Research gift platform for postcards in envelopes	Content	In progress	High	Website	14/07/2025	Overdue
10	4	Send Review Google Form to July clients who are finishing	Review	To-do	High	Email	14/07/2025	Overdue
11	5	Confirm login access: Kajabi, KIT	Backend	Waiting for approval	Medium	Email	14/07/2025	Overdue
12	6	Join Telegram group & change avatar	Backend	Waiting for approval	Medium	Telegram	14/07/2025	Overdue
13	7	Join Slack	Backend	Done	Medium	Slack	14/07/2025	Overdue
14	8	Replace old excel to new dashboard	Strategy	Waiting for approval	Medium	Google Drive	18/07/2025	Overdue
15	9	Organize canva folder and brand asset	Admin	To-do	Medium	Canva	18/07/2025	Overdue
16	10	Add contract end dates for renewal tracking	Admin	To-do	High	Google Drive	16/07/2025	Overdue

+ Dashboard Navigation Master Task Tracker Client Portal Unlock Daily/Weekly Ops Content HQ Email Vault L < >

Master task tracker. Task → Category → Status → Priority → Channel → Due date → Overdue trigger.

Each field earns its place. **Category** groups work by operational area. **Status** shows where a task sits; **Priority** is kept separate, because a task can be waiting for approval *and* still be high priority. **Channel** shows where the work actually happens — email, Slack, Drive, Telegram. **Due date + overdue trigger** make what's late visible instead of relying on memory.

THE SUMMARY LAYER

“Mental Load: 13 tasks this week, 6 high, 0 to-do related.” I added a summary line so no one has to read every row to understand the workload — volume and priority surface at a glance.

SAME PROBLEM, DIFFERENT TEAM

VA Command Center - Beby

Edited 1y agoShare

To Do ListBoardAllLoad per person

New

2

Task Name	Status	Assignee	Due	Priority	Category	To I
Schedule Podcast Ep 12	In Progress	Beby Karina	July 8, 2025	High	Podcast	
Follow up on contract for a new client	In Progress	Beby Karina	July 9, 2025	High	Client Comms	

+ New task

COMPLETE 0/2

2...+

Task Name	Status	Assignee	Due	Priority	Category	To I
Plan Live Q&A session on August	In Progress	Beby Karina	July 14, 2025	High	Client CommsPodcast	
Ideas for new lead magnet	In Progress	Beby Karina	July 15, 2025	High	Podcast	

+ New task

COMPLETE 0/2

Team-facing version in Notion. Task → Status → Assignee → Due → Priority → Category.

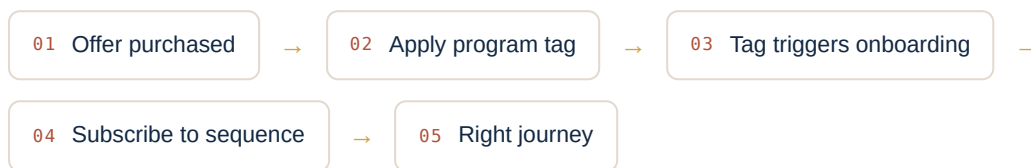
The tool changes; the operating logic doesn't. A team-facing version drops to a simpler structure — the goal isn't to capture every possible data point, it's to make ownership and the next action obvious to everyone involved. I choose the structure for the use case, not the other way around.

Turning a purchase into a client onboarding journey

THE PROBLEM

Different programs require different onboarding journeys. Handled manually, access and communication get delayed or sent to the wrong person.

THE LOGIC



THE EVIDENCE

Edit Automation

Created October 14, 2025 10:10 AM (CEST)

WHEN

Offer is purchased

1:1 Program

THEN

Add a tag

1-13 months

Add Condition

Step 1 – identify. WHEN offer purchased (1:1 Program) → THEN add tag (1-13 months).

Client and program names redacted – the automation logic is unchanged.

Edit Automation

Created October 06, 2025 05:34 PM (CEST)

WHEN

Tag is added

ProgramRoom2025

THEN

Subscribe to an email sequence

Program Room Onboarding (6-Month Program)

Add Condition

Step 2 – communicate. WHEN tag added (ProgramRoom2025) → THEN subscribe to sequence (6-month onboarding).

WHY I STRUCTURED IT THIS WAY

I separated purchase identification from onboarding communication. The purchase records what the client bought; the tag identifies the client/program relationship; the tag then triggers the right sequence.

When offers, programs, or sequences change, the workflow stays easy to maintain.

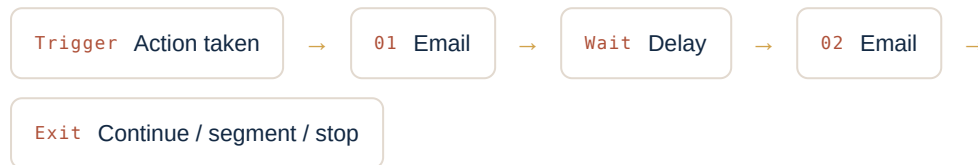
03

EMAIL AUTOMATION · GOHIGHLEVEL + KIT

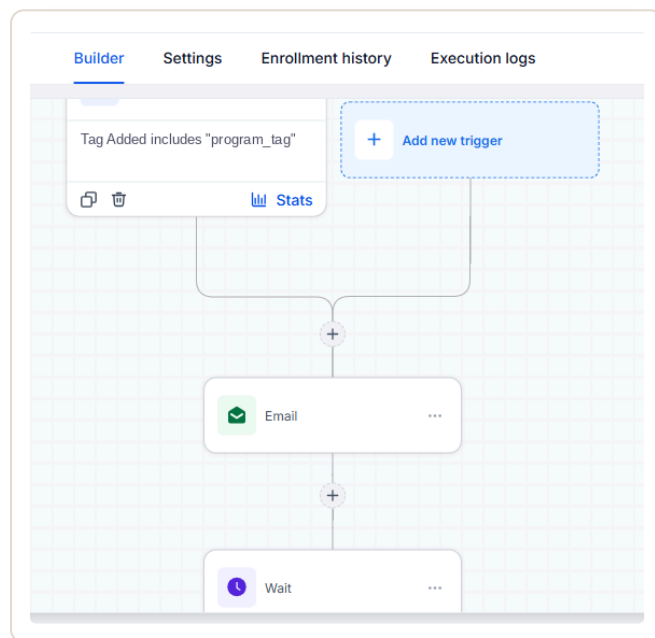
Designing email journeys around actions, not just dates

THE APPROACH

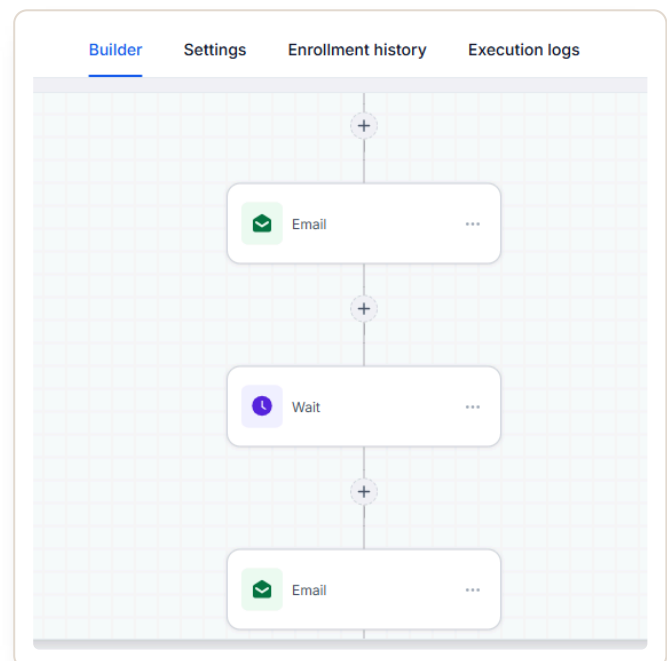
I map automated email journeys around four things: the action that starts them, what the recipient needs next, the right timing, and what should happen if their status changes.



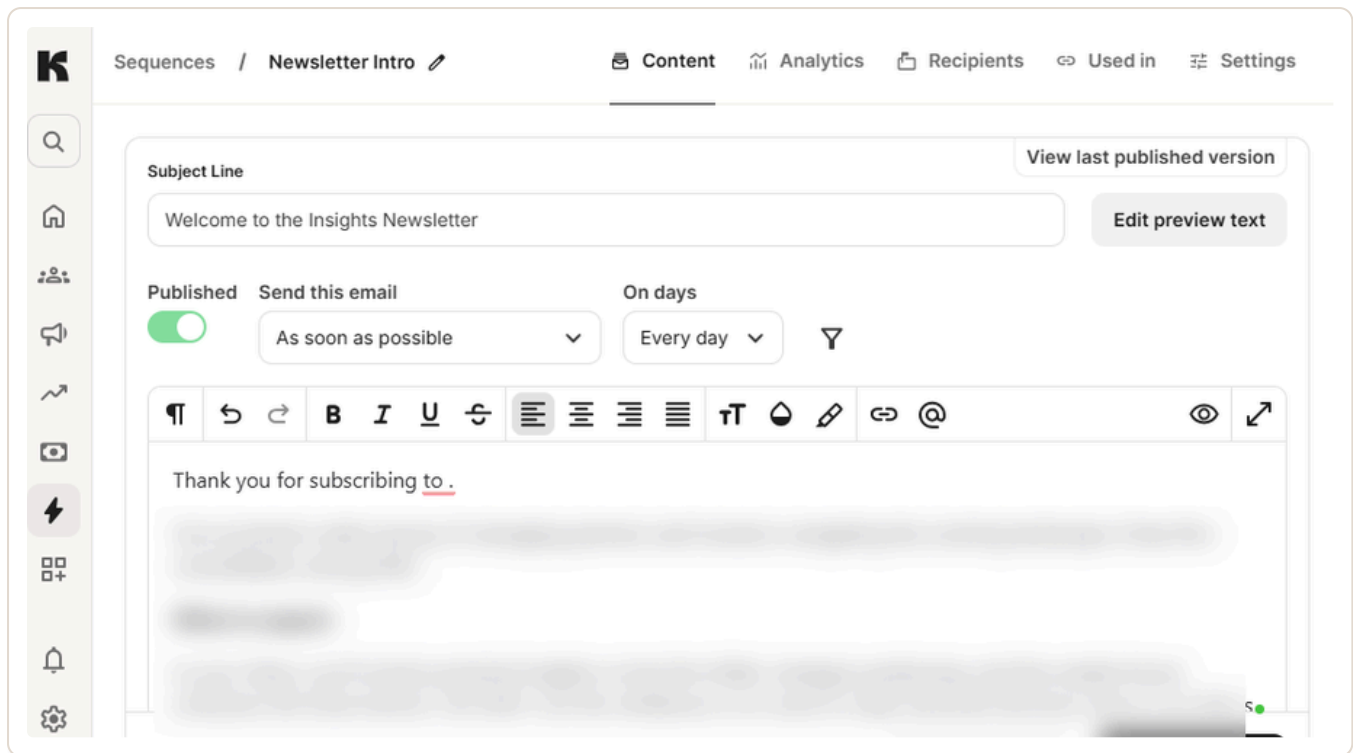
THE EVIDENCE



Action-based trigger. A tag added by the contact's behavior starts the journey.



Email → Wait → Email. Timed steps, not a single blast.



Sequence email built in Kit. The implementation layer – client copy hidden.

WHY IT MATTERS

The goal isn't simply to automate emails. It's to prevent irrelevant communication going out — while making sure important follow-up doesn't depend on someone remembering to send it.

My portfolio shows what I've worked on. This shows why I made the operational decisions the way I did.

Beby Karina

KAJABI · KIT · GOHIGHLEVEL