

An outbound campaign, built for Ewake - not a template.

A real ICP , a real prospect list, real sequence copy, and an A/B testing plan , built the way I'd run week one on the job, using what Ewake actually sells and who actually feels the pain.

PREPARED BY

Maahir Bhatia
MIB Candidate, IESEG
School of Management
Lille, France

SUBMITTED FOR

Ewake - Business Development
Intern
Outbound & Growth
Paris, Station F

A NOTE BEFORE THE BUILD

Why I built it this way

The brief said: send something you've built, a campaign, an automation, a repo. So instead of writing about how I'd approach outbound, I actually did it. What follows is a full outbound campaign for Awake, built from scratch: I researched what the product actually does (an AI agent that watches production systems and helps engineering teams catch and resolve incidents faster), figured out who realistically feels that pain, found real companies showing that pain right now, and wrote the sequences I'd actually send them.

One honest caveat up front, because I'd rather be straight about it than pretend otherwise: I built this with public research like job postings, company sites, and news coverage and not with paid tools like Clay, Apollo or Sales Navigator, since I don't have seat access yet. Everything here is directionally real and immediately actionable; the exact contact names in a live campaign would take another twenty minutes per company on LinkedIn to lock down. I'd rather hand you something honest and rebuildable than something padded to look finished.

What's inside

01	Understanding Awake	What the product does, and who actually buys it
02	The ICP	Who I'm targeting, and why
03	The Prospect List	8 real companies showing live signal, researched today
04	LinkedIn Sequence	Connection note + two follow-ups, with an A/B variant
05	Email Sequence	Three-touch cold email, with an A/B subject line test
06	The Testing Plan	What I'd measure, and how I'd iterate weekly
07	Closing Note	Why I want this specific role

01 · CONTEXT

Understanding Ewake

Ewake builds an AI agent that sits inside a company's production environment and watches it around the clock. When something breaks, a failed deploy, a spike in errors, an alert firing at 2am – the agent investigates before an engineer has to dig through logs manually, and surfaces what actually happened and why. Founded in 2024 by Pooné Mokari and Omid Gosha, both ex-Criteo engineers, the company raised a €2M pre-seed round in late 2025 led by Connect Ventures, and is now scaling its Paris-based team out of Station F.

The clearest real-world proof point I found is their public case study with Booksy: Booksy's SRE team was getting pulled out of deep work constantly to answer developer questions like "why did my pipeline fail?" Ewake was integrated directly into their team chat to answer those questions automatically, freeing the SREs to focus on the reliability work that actually needed a human. Their stated goals were lower alert fatigue, faster incident response, and a lower Mean Time to Recovery (MTTR).

The one-line pitch I'd lead outbound with: "Ewake is an AI teammate that catches and triages production incidents before your engineers have to – so your on-call team stops burning hours on noise and starts spending them on the fixes that matter."

What this means for outbound

The buyer isn't "tech companies" broadly, it's teams that already have production complexity serious enough to hurt: an on-call rotation, an observability stack (Datadog, PagerDuty, Grafana), and engineers who lose real hours to alert noise and incident triage. That pain is loudest in two moments: right after a public outage, and right when a company is scaling its engineering org fast enough that reliability work starts falling behind. Both are visible from the outside if you know where to look, which is exactly how I built the list on the next page.

02 · STRATEGY

The ICP

I kept this narrow on purpose. A vague ICP ("tech companies in Europe") produces a list that looks big but converts nowhere, because the message can't be specific enough to land. A narrow ICP lets every line of the outreach be true and specific to the reader.

Company stage	Series A-B, roughly 50-500 employees – big enough to have real production traffic and an on-call rotation, small enough that one bad outage still hurts.
Signal we're watching for	Actively hiring for SRE, DevOps, or Platform Engineer roles right now. That's a public admission the team is understaffed for the reliability load it's carrying.
Tech context	Already running an observability/incident stack - Datadog, PagerDuty, Grafana, or similar. This means the pain is instrumented, not hypothetical.
Buyer persona	VP Engineering, Head of Platform / SRE, or CTO at smaller companies, whoever owns the on-call pager, budget, and the 2am phone calls.
Geography	France and wider Europe first, same time zone, same language advantage for a Paris-based founder-led sales motion, and the easiest to build genuine trust with quickly.

One thing I deliberately left out: company logos or brand prestige. It's tempting to build a list of recognizable names, but a company hiring its first SRE while running a lean stack is a better fit– and a warmer conversation than a company where reliability is already a mature, well-staffed function.

03 · THE LIST

Eight real companies, researched today

Every company below has a live SRE / DevOps / Platform posting as of this week – meaning the pain isn't theoretical, it's the reason the req exists. I pulled these directly from live job boards rather than a static company database, because a stale list is worse than no list.

Owkin

AI / Biotech SaaS

Senior SRE (remote)

Growth-stage, research-heavy production stack - reliability now has to scale alongside the science.

Doctolib

HealthTech SaaS

Senior SRE

Healthcare uptime is non-negotiable; any minute of downtime has real patient-facing consequences.

Doctrine

LegalTech SaaS

SRE - GCP

Cloud-native legal platform, actively hiring specifically for GCP reliability depth.

Lifen

HealthTech SaaS

SRE (SaaS transformation project)

Explicitly mid-transition to SaaS architecture - exactly the moment reliability debt gets exposed.

BlaBlaCar

Consumer Mobility Platform

SRE (2 live postings)

Large-scale consumer traffic, two concurrent SRE reqs - a team stretched thin right now.

Genesis

Tech / Hybrid

SRE (hybrid)

Actively expanding infra hiring - a team building out reliability practice from a smaller base.

PYSAE

Tech, hybrid

DevOps / SRE (H/F)

Smaller team, hybrid setup - a leaner org where one AI teammate has outsized impact per engineer.

Air Apps

Consumer App Platform

SRE (remote)

App-scale production stakes with a fully remote engineering org - alerting discipline matters more, not less.

Next step on a live campaign: 15-20 minutes per company on LinkedIn to confirm the named buyer (VP Engineering / Head of Platform / CTO) and pull a verified email pattern via Hunter.io's free tier. I've built the message architecture below so it drops in cleanly once those names are locked.

04 · OUTREACH

LinkedIn sequence

Three touches: a connection note, then two short follow-ups spaced a few days apart. I wrote two versions of the opening touch to A/B test – same target, same offer, different hook – because that's the one line that decides whether the rest gets read at all.

Touch 1 - Connection request (300 char limit)

Variant A

VARIANT A

Hypothesis: leading with the cost/consequence of downtime lands harder with a VP/CTO who owns the budget conversation.

Hi {FirstName} – saw the SRE opening at {Company}. I work with Ewake, an AI teammate that catches and triages production incidents before they eat your team's night. Would love to connect and hear how you're handling on-call load right now.

Variant B

VARIANT B

Hypothesis: leading with the human cost (burnout, noise) lands harder with someone closer to the day-to-day on-call rotation.

Hi {FirstName} – noticed {Company} is growing the SRE team. We're building Ewake, an AI agent that filters alert noise so engineers stop losing sleep to false alarms. Open to connecting?

Touch 2 - First follow-up (2–3 days after connection accepted)

Follow-up 1

Goal: earn the reply, not the meeting yet. Reference something real and specific to the company.

Thanks for connecting, {FirstName}. Quick one – with the SRE hire still open, how's the team currently splitting time between building and firefighting? We just helped a team like Booksy cut the constant "why did this fail" interruptions their SREs were fielding – curious if that's a familiar pattern at {Company} too.

Touch 3 - Second follow-up (4–5 days later, only if no reply)

Follow-up 2

Goal: low-friction close. Give an easy yes/no, not a big ask.

No worries if it's not a priority right now, {FirstName} – happy to send a 2-minute Loom instead of asking for time. Either way, if alert fatigue or MTTR ever becomes a headache for the team, I'm around.

05 · OUTREACH

Email sequence

Same three-touch structure, built for inboxes rather than LinkedIn. I A/B tested the subject line on email 1 specifically, since subject lines carry almost the entire open-rate decision before anyone reads a word of the body.

Email 1 - Cold open

Variant A

VARIANT A

Hypothesis: naming the pain (on-call load) directly in the subject line drives higher opens from time-pressed engineering leaders.

Subject A: Quick question about {Company}'s on-call load

Hi {FirstName},

Saw {Company} is hiring for the SRE team – usually means the current team's stretched between building and firefighting.

We built Ewake, an AI teammate that watches your production systems and investigates incidents before an engineer has to dig through logs. One team we work with, Booksy, used it to take alert-triage load off their SREs entirely, freeing them to focus on the reliability work that actually needed a human.

Worth 15 minutes to see if it'd help {Company} the same way?

Maahir

Variant B

VARIANT B

Hypothesis: leading with a proof point (a named result) builds more credibility than naming the pain outright.

Subject B: How Booksy cut MTTR without adding headcount

Hi {FirstName},

Noticed the open SRE role at {Company} - reliability work tends to outgrow the team just before a hire like that lands.

We built Ewake: an AI agent that sits inside your production environment, catches incidents early, and investigates root cause before your team even opens a dashboard. Booksy used it to cut down the constant "why did this fail" interruptions pulling their SREs out of deep work.

If that sounds familiar at {Company}, open to a quick call this week?

Maahir

Email 2 - Value-add follow-up (day 3)

Follow-up

Goal: give value even if they never take the meeting - keeps the door open.

Subject: One more thing on incident response

Hi {FirstName},

Following up in case this got buried – thought this might be useful either way: most teams we talk to underestimate how much engineering time goes into triage versus actual fixes until they measure it. If it's helpful, happy to share the breakdown we use with prospects, no call required.

Maahir

Email 3 - Breakup email (day 7)

Breakup

Goal: the classic breakup email - often the highest-reply-rate email in any sequence, because it removes pressure.

Subject: Should I close the loop on this?

Hi {FirstName},

I'll assume the timing's off for now – I'll stop here. If reliability or on-call load becomes more of a priority down the line, my inbox's open.

Maahir

06 · MEASUREMENT

The testing plan

Matches the cadence in the job post almost exactly – new hypothesis Monday, campaign live Wednesday, results reviewed Friday. Here's what I'd actually track and how I'd decide what to change.

Connection / open rate	Tells me if the hook (subject line or connection note) is landing before anyone reads the offer.
Reply rate	The real signal of message quality – did the value prop land, not just the subject line.
Meetings booked	The metric that actually matters to the business - everything above is a leading indicator.
Time-to-reply	Fast replies usually mean high relevance; slow trickles suggest the list, not the message, needs work.

Weekly loop

- Monday** Pick one variable to test (this week: subject line framing - pain-led vs. proof-led). Draft the hypothesis before writing copy, not after.
- Wednesday** Launch to a fresh batch of the list, split evenly between variants. Keep everything else - send time, list quality, follow-up cadence – identical.
- Friday** Pull open/reply/meeting numbers, declare a winner only if the sample's big enough to trust, and fold the winning pattern into next week's default before testing the next variable.

One rule I'd hold myself to: never change two variables in the same test. It's tempting to rewrite the whole message when reply rates are low, but then you never actually learn what fixed it.

07 · CLOSING

Why this role, specifically

I could've sent over a case study from my last internship and called it a day. I chose to build this instead because the posting was clear about what you actually want to see – someone who thinks in workflows, ships fast, and treats a cold list like a hypothesis to test rather than a task to finish.

I don't have a formal SDR title on my CV yet. What I do have is a habit of running things end to end and measuring what happens: building and A/B-testing marketing funnels that lifted CTR by 18%, cleaning up affiliate operations across five partner channels to cut campaign lag by 25%, and – closer to this kind of cold, structured research – independently validating 25,000+ orders for a quick-commerce startup and getting a founder on a call to pressure-test what I'd found. That last one is basically this exercise, just for a different product.

I'd want this internship because it's the fastest way I know to learn founder-led sales properly – watching how a deal actually closes, not just how a campaign gets sent. Happy to walk through any part of this build, defend the ICP choices, or start refining real send copy whenever it's useful.

Maahir Bhatia

bhatiamahir7@gmail.com · +33 754 043 486 · Lille, France