

learn ai with **dany**

FREE AGENT GUIDE

# Job Change Play

When a customer's champion changes jobs, this agent writes the two plays you run at once: protect the account they left, and open the door at the company they just joined.

Comment JOBCHANGE on the video  
or go to [learnaiwithdany.com/agents/job-change-play](https://learnaiwithdany.com/agents/job-change-play)

2026-08-29

# What it does

The video's claim is that this agent "flags job changes and runs two plays at once: protect the account they left, and open the door at the company they just joined." The two-play part is true and confirmed by testing: give it a job change, and it writes real, specific outreach for both sides, not a generic template. One correction on the "flags" part: this agent does not watch LinkedIn, your CRM, or Sales Navigator for you. It assumes you already know about the job change (from LinkedIn, a CRM enrichment tool, or a Sales Navigator alert) and tells you exactly what to do once you know.

Here is what it actually produces. On the account your champion left, it tells you to find whoever inherited their responsibilities and reach out within 5 days with a usage summary and account history, explicitly not a pitch, just help them look good during the handover. On the company your champion just joined, it tells you to wait until day 14 to 21, deliberately not day 1, because the first two weeks are onboarding chaos and nobody is evaluating new tools yet. It then writes a real 7-day, 3-touch sequence (email, LinkedIn message, value-add email) and a CRM hygiene checklist so the old and new records both get updated correctly.

In testing, this was run on an invented scenario: a VP of RevOps champion who left one company for another, with a real account and renewal date attached. The output correctly produced both tracks, correctly applied the day 14-21 timing rule instead of jumping in on day 1, and correctly treated the departure as a senior-level move that gets immediate personal outreach rather than automated nurture. The core promise of the video holds up.

One more thing worth knowing before you install: the repo ships a small Python script that is supposed to check your finished output against a checklist. It does not work out of the box, it fails even against the repo's own blank template, because the template's own section headings do not match what the checker is looking for. Skip that script. The playbook itself is solid, its own QA tool just is not wired up correctly yet.

# Who it is for

This is for anyone running sales or customer success who has ever lost a renewal because the person who championed the deal quietly left, and nobody on the vendor side noticed until it was too late. It replaces the job of remembering to do two things at once when a job change happens: protect what you already have, and go after what just opened up.

# Before you start

- Claude Code (or another Agent Skills-compatible tool, the skill's own frontmatter also lists Codex, Cursor, Gemini CLI, and others) installed on your machine
- git installed, to clone the source repo
- A job change signal already in hand: this agent does not detect changes itself, you need to already know a champion moved (LinkedIn, your CRM's job-change alert, Sales Navigator, or similar)
- No API key or paid account needed, it is pure markdown and logic, no LLM calls of its own
- Install took about 5 minutes in testing using the manual copy method below (the repo's own documented one-line install command failed in testing, see Troubleshooting)

# Install

## Step 1: Clone the source repo

This downloads the full skills library onto your machine. You only need one folder out of it.

```
git clone https://github.com/LeadMagic/gtm-skills.git
```

## Step 2: Copy just the Job Change Play skill into your project

This creates the folder Claude Code looks in for skills, then copies only the one you need into it.

```
mkdir -p .claude/skills  
cp -r gtm-skills/skills/sales-plays/job-change-play .claude/skills/job-change-play
```

## Step 3: Load the skill in Claude Code

Open Claude Code in this project and ask it to use the skill. This is the step that actually runs the play.

```
claude
```

Then, inside Claude Code, describe your job change situation and ask it to run the job-change-play skill.

## Run it

Sample input used in testing (invented company and people, not real):

"Our champion at Brightwell Logistics, Jordan Ellis (VP of RevOps), just left. LinkedIn shows Jordan started a new role last week as VP of RevOps at Northgate Freight Corp, which is not currently a customer. Brightwell is a \$42k/year account, renewal is in 4 months. Build the job change play for this: protect the Brightwell account and open the door at Northgate."

Abbreviated real output:

```
Track 1, Brightwell (old company): identify who inherited Jordan's role,  
reach out within 5 days with a usage summary and account history. No pitch.  
  
Track 2, Northgate (new company): wait until Day 14-21, not Day 1.  
Draft email bridges Jordan's result at Brightwell to a likely problem at Northgate.  
  
7-day sequence: Day 1 email, Day 3 LinkedIn message, Day 7 value-add email,  
Day 14+ falls back to standard nurture if no reply.  
  
CRM hygiene: update Jordan's record, create new opportunity at Northgate,  
flag Brightwell for champion-replacement follow-up.  
  
Quality check: 6 of 6 items satisfied, seniority correctly tiered  
(VP-level move gets immediate personal outreach on both tracks).
```

## Troubleshooting

The documented `gh skill install` command fails with "unknown command skill for gh". This is a real bug found in testing: the installer only falls back to its own manual copy method when the `gh` CLI is completely missing, not when `gh` is present but its `skill` extension is not installed. Skip the one-liner entirely and use the manual copy method in Install above, it produces an identical result and was confirmed working.

**scripts/check-output.py reports missing sections even on a correct deliverable.** This is a real bug in the repo, not something you did wrong. The bundled template's own headings do not match what the checker looks for, so it fails even against the repo's blank template. Do not use this script to judge your output, judge it against the SKILL.md Quality Check list instead.

**Claude Code does not seem to pick up the skill.** Confirm the folder is exactly at `.claude/skills/job-change-play/` in the project you opened Claude Code from, and that `SKILL.md` is directly inside it.

**The output feels generic instead of specific to your accounts.** Give it the actual names, account value, and renewal date in your prompt, the way the sample task above does. The skill fills in real specifics when you give it real specifics.

## Credit

Built by LeadMagic (<https://github.com/LeadMagic/gtm-skills>), MIT license, 53 stars at test time. Two corrections to how it is described: the skill does not detect or "flag" job changes on its own, it assumes you already have that signal, and the bundled `scripts/check-output.py` QA script does not work out of the box, it fails against the repo's own template.

The full write-up, with the PDF and every guide, lives free at [learnaiwithdany.com](https://learnaiwithdany.com). Comment JOBCHANGE on the video to get this link sent to you directly.

RECAP

## Job Change Play

When a customer's champion changes jobs, this agent writes the two plays you run at once: protect the account they left, and open the door at the company they just joined.

Built by LeadMagic · MIT license · [learnaiwithdany.com/agents/job-change-play](https://learnaiwithdany.com/agents/job-change-play)

Some links in this guide may be affiliate links. If you sign up through them I may earn a commission at no extra cost to you.