

---

# Automated LinkedIn-to-Blog Publishing Playbook

Playbook Visual Link:  Automated Blogs.png

## How it works?

### Step 1: Identify top creators to monitor

Research the creators worth tracking in your domain.

- Build a list of top competitors and creators in your space.
- Collect each creator's LinkedIn profile URL.
- Prioritize creators whose posting style and topics fit your brand.

### Step 2: Set up monitoring in Jungler AI

Plug the creators into Jungler and route output to Notion.

- Paste each LinkedIn profile URL into Jungler AI.
- Configure a Jungler workflow that sends new posts to **Notion**.
- Jungler populates the Posts database automatically.

### Step 3: Create the LinkedIn Posts database

The raw feed where every monitored post lands.

- Required columns:
  - Creator Name
  - LinkedIn Post URL
  - Post Content
  - Posted On
  - Fetched On
- Every new row represents one post from one creator.

---

#### Step 4: Create the Blogs database

The structured workspace where posts become publishable blogs.

- Required columns:
  - Blog Title
  - Blog Body
  - Category
  - Source LinkedIn URL
  - Published
- Status field with three options:
  - In Review
  - Approved
  - Rejected

#### Step 5: Schedule the daily blog writer

Set up the job that transforms posts into blogs.

- Create a **Cowork scheduled task** that runs once per day.
- Connect **Notion MCP** so Claude can read and write to both databases.
- Instruct Claude to:
  - Read new posts from the Posts database.
  - Apply the brand voice guide.
  - Rewrite as a long-form blog, not a paraphrase.
  - Avoid vague or generic phrasing.

#### Step 6: Write blogs into the Blogs database

Claude creates one blog row per qualified post.

- Generate Blog Title and Blog Body.
- Assign a Category.
- Copy the LinkedIn Post URL into Source LinkedIn URL.
- Default Status to 'In Review'.

---

### Step 7: Review and approve

Decide which blogs ship.

- Open the Blogs database each morning.
- Read each blog for voice, accuracy, and originality.
- Toggle Status to Approved or Rejected.

### Step 8: Trigger the publish webhook

Fire the automation on approval.

- Notion automation watches Status for changes.
- Trigger condition: Status changes to Approved.
- Webhook payload includes Title, Body, Category, Source LinkedIn URL.

### Step 9: Publish to Webflow

Ship the blog to the live site.

- Webhook routes to Claude with **Notion MCP + Webflow MCP** connected.
- Claude creates a new Webflow CMS item with the blog fields.
- Publish the item live.