JobHacki
OPPORTUNITY GUIDE

⚡ **Premium Action Guide**

Build and Sell an AI Resume Tool

Launch a workflow that scrapes roles, tailors resumes and cover letters with AI, stores clean Google Docs, and logs insights to a master Google Sheet—ready to email in minutes.

📌 **Updated for 2026** 🧾 **≈$0.02** Approx. compute cost per tailored resume (source-reported)

📣 **@jobhacki** · JobHacki Community

🚀 **Start Fast**
Clear first steps you can take this week.

📋 **Real Sources**
Built from people who actually did it.

💰 **Honest Numbers**
Source-reported pay, costs, and risks.

Get the Full Guide Vault →

🔒 Includes checklists, scripts & source-backed insights

YOU WILL LEARN

- ☑ What you'll build
- ☑ Step 1 — 🎯 Decide your product
- ☑ Step 3 — 🧱 Pick your build path
- ☑ Platforms & tools
- ☑ Step 2 — 🧭 Wireframe the flow
- ☑ Step 4 — 🔐 Connect Google Sheets/Docs

▸ MODULE 01

What you'll build

An end-to-end AI resume system: pull fresh jobs, parse a candidate's base resume, analyze job fit, and generate tailored resumes and cover letters. Store polished Google Docs and log key insights (strengths, gaps, fit score) in a Google Sheet—optionally email or schedule follow-ups automatically.

Choose your build path (Make.com, n8n, Python, or Antigravity) and plug in supporting tools (Apify for scraping, OpenAI/Claude for AI, Google Sheets/Docs for storage).

1. Scrape targeted roles and de-duplicate listings
2. AI-tailor resume + cover letter per job
3. Google Docs output + Sheets logging
4. Fit analysis (strengths/weaknesses/risks/rewards)
5. Optional email delivery and follow-ups

▸ MODULE 02

Platforms & tools

Lead & Outreach

2 free · 6 paid

LinkedIn · Upwork · Loom · Apollo · Cal.com · Smartlead · Fiverr · Instantly

Website Builders

0 free · 2 paid

Lovable · GoHighLevel

AI Models

0 free · 5 paid

Claude · ChatGPT · Gemini · OpenAI · Anthropic

Dev & Hosting `0 free · 2 paid`

GitHub · VS Code

Automation `1 free · 3 paid`

Make.com · n8n · Apify · Zapier

▸ MODULE 03

Step 1 — 🎯 Decide your product

Offer a simple promise: for each target job, produce a tailored resume, matching cover letter, and a fit analysis—logged and linkable. Add optional delivery (email) and scheduled follow-ups.

> **Tip**
> Avoid auto-applying to jobs from scrapers; it risks ToS violations and reputational harm. Focus on personalization and measurable evidence.

▸ MODULE 04

Step 2 — 🧭 Wireframe the flow

Sketch the pipeline: Inputs (job query + base resume) → Scrape jobs → Parse resume → AI analysis (fit) → AI tailoring → Store Docs + log to Sheets → Email/follow-ups. Use a simple diagram to align steps and data fields before building.

HELPFUL

Excalidraw — Quick wireframes for triggers and data flow ↗

> **Tip**
>
> Building without a wireframe wastes time. Map triggers, data sources, transformations, and AI outputs first.

▸ MODULE 05

Step 3 — 🧰 Pick your build path

Choose one primary platform to orchestrate everything. You can always port later.

Platform	What to do in this step
Make.com	Create a new folder. Plan two scenarios: (1) Scrape roles (Apify). (2) On new dataset items → parse resume → AI tailor → write Google Docs → log to Sheets → (optional) Gmail.
n8n	Create a workflow: Webhook/Timer → Apify Actor → Switch (file type) → Info Extractor → OpenAI (fit JSON) → OpenAI (resume/letter) → Google Docs → Google Sheets → (optional) Gmail.
Python	Set up venv or Colab; pip install google-api-python-client, openai, apify-client. Scaffold scripts for scraping, parsing, AI calls, and Google APIs.
Antigravity	Install and open. Start a new chat. In planning mode, describe the pipeline; let the agent draft a build plan before execution.

▸ MODULE 06

Step 4 — 🔐 Connect Google Sheets/Docs

Authorize access to Google Sheets and Docs so your workflow can store outputs and log results.

Platform	What to do in this step
n8n	Create Google Sheets credentials. In Google Cloud Console: new project 'job finder' → enable Sheets API → OAuth consent 'job finder' (use your email) → create OAuth Client (Web) → add your n8n redirect URI → copy Client ID/Secret into n8n → publish and sign in.
Make.com	Add Google Sheets and Google Docs connections via built-in OAuth when you first drop the modules. Grant requested scopes.
Python	Create OAuth Client in Google Cloud Console. Use google-api-python-client to run OAuth flow, storing token.json. Enable Sheets and Docs APIs.
Antigravity	If orchestrating n8n/Make via agents, complete their Google auth steps first; the agent will call those connections.

DOCS

Google Cloud Console — Create OAuth credentials and enable APIs ↗

Tip

Google's OAuth warning screen is normal—use Advanced → continue. Use the correct nested Google Drive file ID when writing/reading Docs.

▸ MODULE 07

Step 5 — 🔌 (Optional) Link Antigravity to n8n

Let an agent help plan and manage your n8n automations from one place.

- Download and install Antigravity; sign in with Google.
- Open a new chat; ensure planning mode is enabled.

- Install n8n skills: paste the GitHub repo URL when prompted.
- In n8n, create an API key (no expiration) and paste it into Antigravity's JSON config.
- Modify the local JSON path to your n8n MCP index.js if using a local dev setup.
- Enable auto-execution for smoother agent commands.

▸ MODULE 08

Step 6 — 🤖 Add OpenAI/Claude access

Platform	What to do in this step
n8n	Open the OpenAI node → create new credentials → paste your OpenAI API key (after adding a payment method) → save.
Make.com	Insert an OpenAI module → add connection → paste API key → pick model (e.g., GPT-4).
Python	Set OPENAI_API_KEY in environment variables. Use JSON output mode for structured replies.
Antigravity	Store model API keys in agent settings so subflows can call them.

Tip

Faster small models (e.g., GPT-4o mini) are cheaper but weaker; use stronger models for resume tailoring and fit analysis.

▸ MODULE 09

Step 7 — 🔎 Source jobs with Apify

Use Apify's Indeed bulk scraper to fetch job listings that match your search. Trigger tailoring only for fresh items.

• In Apify, get your API key and note Actor/Task names.

• Set search parameters (e.g., job title 'data engineer', location 'San Francisco, United States', max items 10).

• In Make.com: scenario 1 runs the Indeed actor; scenario 2 watches for completion, then processes new dataset items.

• In n8n: add an Apify node, paste your API key, and poll/watch for new dataset items.

• Understand Apify basics: Actors run, Tasks configure runs, Datasets store results.

• Prefer 'watch actor runs' triggers to avoid wasteful polling.

> **Tip**
> Do not auto-apply from scrapers. Some listings are stale or misclassified—manually qualify before sending outreach.

▸ MODULE 10

Step 8 — 📄 Create your data schema

Standardize your Google Sheet so each processed job has a row with links and analytics you can search and filter easily.

• Sheet columns: Position Name, Job Type, Company Location, Description, URL, Customized Resume, Date Created, Posted At, Scraped At.

• Also map fields for candidate details and AI analysis: date/time, resume link, first name, last name, email, strengths, weaknesses, risk factor, reward factor, overall fit, justification.

• Store source post URLs to adjust scrape targets without redeploying.

▸ MODULE 11

Step 9 — 🧰 Parse resume files robustly

Support DOCX, PDF, and TXT resumes. Normalize to clean text before AI steps, and extract key candidate fields early.

• Add a Switch by MIME type; use the appropriate extraction for each.

• Run an Information Extractor to pull first name, last name, and email (mark all required).

• Keep standardized text for downstream prompts.

> **Tip**
> Do not overload the AI agent—use a dedicated extractor node for names/emails; reserve the model for analysis and tailoring.

▸ MODULE 12

Step 10 — 🧠 Analyze job fit with structured AI

Score each role by strengths, weaknesses, risk, reward, and an overall fit rating with justification. Save each field to your Sheet.

• System: “Analyze this resume against this job description. Return strengths, weaknesses, risk, reward, overall fit rating (0–100) and a short justification.”

• User: include clean resume text and the job description in JSON.

• Use a structured output parser (JSON schema) so each field is reliably separated and mappable to Sheets.

> **Tip**
>
> Feed prompts with JSON and request JSON replies—treat the model like an API for predictable parsing.

▸ MODULE 13

Step 11 —  Tailor the resume content

Customize the resume and cover letter with the right keywords and evidence. Output in markdown (ATX) for easy formatting into Google Docs.

PROMPT — RESUME TAILORING (TECH ROLES)

```
System: You tailor resumes and concise cover letters for [TECH ROLE]. Optimize
for ATS keywords and clarity.
User: Return two markdown sections:
# Tailored Resume
Rewrite using [TOP 6 KEYWORDS FROM JOB], emphasize measurable results, project
links if present. Keep to one page.
# Cover Letter
Short 120–160 words, mention [COMPANY], align 3 bullets to their stack:
[STACK], [CLOUD], [DATA/SECURITY].
Inputs:
- Base Resume: [RESUME_TEXT]
- Job Posting: [JOB_DESCRIPTION]
- Role: [ROLE_TITLE]
Constraints:
- Bullet points start with strong verbs
- Prefer numbers/metrics when available
- No generic claims; cite specific tools
- Output markdown ATX only
```

PROMPT — RESUME TAILORING (CUSTOMER SUPPORT ROLES)

```
System: You tailor support resumes and cover letters.
User: Produce:
# Tailored Resume
Emphasize ticket volumes, CSAT, first-response/resolve times, tools
(Zendesk/Intercom), HIPAA if relevant.
# Cover Letter
100–140 words, empathetic tone, 1 example staying calm under pressure,
alignment with [COMPANY] values.
Inputs: [RESUME_TEXT], [JOB_DESCRIPTION]
Constraints: Use keywords like member support, customer service, communication,
organization, problem-solving. Output markdown ATX.
```

PROMPT — RESUME TAILORING (MARKETING ROLES)

```
System: You tailor marketing resumes and letters.
User: Return:
# Tailored Resume
Highlight growth metrics, channel expertise (paid social, SEO, email), notable
campaigns. Front-load best 100 characters.
# Cover Letter
List 3 bullets mapping portfolio items to [COMPANY] goals.
Inputs: [RESUME_TEXT], [JOB_DESCRIPTION]
Constraints: Use role keywords; avoid fluff; markdown ATX only.
```

Tip

Don’t have AI generate entire documents unbounded—use stable templates and let AI fill scoped sections for predictable quality.

▸ **MODULE 14**

Step 12 — 🧩 Assemble Docs and log to Sheets

Create a Google Doc per tailored resume and cover letter; write links plus analysis fields into your Sheet for tracking.

• Write tailored markdown to Google Docs (convert headings and bullets cleanly).

• Add a Google Sheets row per job with: job details, doc links, strengths/weaknesses, risk/reward, fit score, justification.

▸ MODULE 15

Step 13 — ✉️ (Optional) Email delivery

Auto-email the candidate a review link or send a concise hiring-manager email with the tailored resume link.

TEMPLATE — HIRING MANAGER EMAIL

```
Subject: Quick intro for [ROLE] at [COMPANY]

Hi [Name],

I wanted to personally pass along my resume for the [ROLE] you have open at
[Company]. I'd love the chance to share how my background could help the team
and why I'm excited about the role.

Best,
[Your Name]
[Resume/Doc Link]
```

TEMPLATE — INFORMATIONAL DM

```
Subject: [Personalized hook for [Name]]

Hey [Name],

I’m exploring opportunities on the [TEAM] at [Company]. Would you be open to
sharing what the culture is like and how your team approaches [AREA]?

Thanks,
[Your Name]
```

> **Tip**
> Cold LinkedIn messages often get buried; email first. Use short, punchy copy and include your tailored Doc link (avoid attachments).

▸ MODULE 16

Step 14 — ⏱️ Follow-ups, delays, retries

Schedule polite follow-ups without keeping long sleeps inside your automations. Queue and retry robustly to survive rate limits.

• Use Hookdeck to delay webhooks longer than platform limits; test by POSTing sample payloads.

• Add a Break module for backoff and retries; use exponential intervals rather than linear.

• Use an Ignore/Continue-on-error step where safe so one bad item doesn’t halt the run.

QUEUE & DELAY

Hookdeck Docs — Delay, queue, and transform events ↗

▸ MODULE 17

Step 15 — 🧪 Test end-to-end

Run the pipeline with sample inputs: location, job title, and a resume upload. Verify Docs and Sheet entries, then email outputs.

• Test DOCX, PDF, and TXT paths via your Switch node.

• If something errors, ask your Antigravity agent to inspect recent executions and auto-fix the workflow.

• Confirm the Sheet shows job title, description, company, posted date, and links to personalized cover/resume.

> **Tip**
> Enable storing incomplete executions; it makes manual retries and break-module backoffs effective.

▸ MODULE 18

Step 16 — 🗃️ Version control prompts & flows

Treat your prompts and blueprints like code. Use Git to snapshot changes and recover fast if a prompt regressions hurts output quality.

• Install Git; connect your GitHub account.

• Commit flow diagrams, prompt files, and environment templates.

• Use a daily agent script to summarize work and commit at end of day.

REPOS

GitHub	Host and version your project ↗

▸ MODULE 19

Step 17 — 🧩 Optional: Upwork proposal generator

Add a side feature for freelancers: paste a job description, generate application copy (JSON), a Google Doc proposal, and a Mermaid system diagram automatically.

- Create three subflows: application copy, Google Doc proposal, Mermaid diagram.
- Configure the agent to generate all three assets from one input job post.

PROMPT — UPWORK APPLICATION WRITER (JSON)

```
System prompt: You are a helpful intelligent Upwork application writer.

User prompt: Your task is to take as input an Upwork job description and return as output a customized proposal.

Template example: Hi, I do [thing] all the time. I'm so confident I'm the right fit for you that I just created a workflow diagram plus a demo of your {{job description}} in no code. [Link]

About me: I'm a relevant job description that has done cool relevant things of note, other cool tie-ins. Very neat. Happy to do this for you anytime, just respond to this proposal else I don't get a chat window. Thank you.

Rules and formatting: Output in JSON format with keys for each section, use new line delimited bullet points, avoid emojis and flowery language, use first-person language.
```

PROMPT — GOOGLE DOC PROPOSAL

```
You are a helpful intelligent proposal writer.
I'm an automation specialist applying to jobs on freelance platforms.
Your task is to take as input an Upwork job description and return as output
JavaScript object notation for a customized proposal which I'll upload to
Google Docs.

High performing proposals are typically templated as follows:
- Title of system
- Brief explanation of system
- Hi, as mentioned, I'm so confident I'm the right fit for this that I went
ahead and created a proposal for you including a step-by-step of how I do it.
- I've done the below many times and working with specific part of their
request is actually one of my favorite parts of automation.
- Here is how we'll build all of this stuff: left to right flow with arrows.
- A little bit about me: bullet points.

Rules:
- Write in a casual Spartan tone of voice.
- Don't use emojis or flowery language.
- If there's a name included somewhere in the Upwork job description, add it
for personalization.
- Return step-by-step bullet points.
- Delimit each bullet point with a backslash and include a hyphen.
- Prefer not to mention social proof that includes money and numbers in about
me bullet points.
- Use first-person language.
```

PROMPT — MERMAID DIAGRAM

```
Your task is to take as input an Upwork job description and return as output a
Mermaid diagram that I can visualize using a subsequent Mermaid live editor.

Example output:
graph TD
A[Receive email from Facebook lead ads] --> B[Add to CRM]
B --> C[Send customized SMS]

Rules:
- Only output flowcharts, no sequence diagrams, no Gantt charts.
- Do not output any accessory formatting information like backticks.
- Your first character should be 'g' (for graph TD).
```

VIEWER

Mermaid Live Editor — Paste the diagram to preview ↗

▸ MODULE 20

Step 18 — 📹 Optional: 60–120s Loom intro

Let candidates add a short Loom link that humanizes their application. Embed the link into the Sheet and cover letter for quick access.

- Script: who they are, a key result, why this role.
- Record while scrolling their LinkedIn profile or portfolio so it feels alive.

▸ MODULE 21

Step 19 — 🚀 Launch and sell

Publish a clean one-page site, front-load social proof in 100 characters, and pitch succinctly. Sell to jobseekers directly or to bootcamps/staffing firms as a white-label service.

TEMPLATE — SHORT PITCH (B2B)

```
Subject: Build an AI Agent to Scale Your Business Without Hiring

You as a solo founder or small team want to scale but can’t because hiring more
people requires capital and you’re stuck at a ceiling. AI agents and AI
automation can help you break free and grow past these limits. Here’s how it
works...
```

Tip

Using your own name as the brand can help trust. Use multiple channels (email, LinkedIn, Instagram, Twitter) to increase reply odds.

▸ MODULE 22

Mistakes to avoid

⚠️

UNBOUNDED AI DRAFTING

Don’t let AI write entire documents without templates; scope outputs for predictable quality.

⚖️

AUTO-APPLY SPAM

Avoid automatically applying to jobs from scrapers—violates terms and hurts your brand.

🧱

NO WIREFRAME

Skipping a process map leads to rework. Diagram triggers, data, AI steps first.

🔑

GOOGLE OAUTH SNAGS

Use correct redirect URIs and file IDs; the warning screen is expected—proceed via Advanced.

🧠

OVERLOADED AGENT

Separate field extraction from analysis; use a structured output parser for clean JSON.

⛔

DIY LINKEDIN SCRAPER

LinkedIn anti-scraping is tough; prefer marketplace scrapers and official APIs where possible.

🧪

NO ERROR STRATEGY

Without incomplete-execution storage, break/ignore modules and retries won't help much.

🧩

WEAK TAILORING

Avoid vague bullets. Use job keywords and measurable results or you'll be invisible.

📨

SINGLE-CHANNEL OUTREACH

Messages on LinkedIn get buried. Email first; keep it short with a direct Doc link.

🔐

SECURITY BLIND SPOTS

Handle PII carefully; prefer compliant platforms and least-privilege credentials.

▸ MODULE 23

Income Forecast

$1,500–$2,000/mo

Example AI automation retainer per client (source-reported)

$24–$30/hr

Comparable market rate for customer support work (source-reported)

$20–$22/hr

Comparable enrollment specialist rate (source-reported)

▸ MODULE 24

Resources

RESOURCES

Google Cloud Console	Create OAuth credentials; enable Sheets/Docs APIs ↗
n8n Docs	Credentials and node configuration ↗
Make.com Help Center	Modules, connections, and scenarios ↗
Apify Indeed Scraper	Bulk job scraping actor ↗
Hookdeck Docs	Delays, retries, and event queues ↗
Mermaid Live Editor	Visualize flow diagrams ↗
WebSpaceKit — Agent Templates	Job searching AI agent template ↗
Google Colab	Cloud Python notebooks ↗
GitHub	Version control for flows and prompts ↗
Indeed Job Search	Cross-check scraped listings ↗

TABLE OF CONTENTS

01 What you'll build

02 Platforms & tools

03 Step 1 — 🎯 Decide your product

04 Step 2 — 🧭 Wireframe the flow

05 Step 3 — 🥾 Pick your build path

06 Step 4 — 🔐 Connect Google Sheets/Docs

07 Step 5 — 🔌 (Optional) Link Antigravity to n8n

08 Step 6 — 🤖 Add OpenAI/Claude access

09 Step 7 — 🔎 Source jobs with Apify

10 Step 8 — 🗂️ Create your data schema

11 Step 9 — 🧰 Parse resume files robustly

12 Step 10 — 🧠 Analyze job fit with structured AI

13 Step 11 — 📝 Tailor the resume content

14 Step 12 — 🧩 Assemble Docs and log to Sheets

15 Step 13 — ✉️ (Optional) Email delivery

16 Step 14 — ⏱️ Follow-ups, delays, retries

17 Step 15 — 🧪 Test end-to-end

18 Step 16 — 🗃️ Version control prompts & flows

19 Step 17 — 🧩 Optional: Upwork proposal generator

20 Step 18 — 🎥 Optional: 60–120s Loom intro

21 Step 19 — 🚀 Launch and sell

22 Mistakes to avoid

23 Income Forecast

24 Resources

THE JOBHACKI ARSENAL

What All-Access members get

All 7 JobHacki tools on one membership.

Hack your career.

AI Auto Apply

Swipe real jobs. We get each application ready; you check it and press submit.

Job Directory

Live jobs pulled straight from company hiring systems.

Resume Builder

A role-ready resume in minutes.

Resume Match

A 0-100 match score against live roles.

Interview Role Play

Free practice tests with an account; voice mock interview after the trial.

Playbook Directory

Step-by-step income and job playbooks.

Tool Directory

Tools, prompts, GitHub repos and GPTs, by category.

Create your free account

Get All-Access — see price and terms

See the community & what's inside

A free account opens the job hubs, playbook library, Tool Directory and Interview Role Play's practice tests.

Learn More →

Created by JobHacki · @jobhacki · JobHacki Community · Version 2026

Income figures in this guide are source-reported or estimates from real creators and practitioners — not guarantees. Income depends on location, skill, speed, and demand. Verify pay rates and offers before applying or buying.