



FREE SELF-SERVICE INSTALLATION GUIDE

Job Search Dashboard

Build a private dashboard on your own computer that turns job-application emails into a clear, refreshable status view.

1	Extract	2	Run locally	3	Connect read-only email
---	---------	---	-------------	---	-------------------------

For macOS, Windows, and Linux - with guided setup for Codex or Claude

See the working demonstration at <https://human-workforce-job-search-demo.fly.dev>

Version 1.0 / August 2026

Contents

Start with Quick Start. Then use only the installation and email-provider sections that match your computer and mailbox.

Quick Start	4
Know the Files in Your ZIP	5
Choose Your Setup Path	6
Install on macOS	7
Install on Windows	8
Install on Linux	9
Use Codex	10
Use Claude	11
Choose Your Campaign Start Date	12
Connect Gmail	13
Yahoo Mail	14
Apple Mail and iCloud	14
Outlook and Other Mailboxes	15
Configure the Email Route	16

Preview and Confirm the First Import	17
Use the Dashboard Every Day	18
Start a New Job Search	19
Back Up and Restore	19
Update the Free Kit	20
Troubleshooting	21
Privacy and Security	22
Uninstall	23
Quick Reference	24

Quick Start

The shortest safe path from the downloaded ZIP to a working local dashboard.

What you are building

This kit builds a private job-search status dashboard that runs at localhost on your own computer. It reads job-related email signals from the same mailbox you use when applying, groups messages by employer and role, and shows what is open, rejected, interviewing, waiting for action, or stale.

Your dashboard database stays in the project folder. A local dashboard is not automatically an offline system: Codex, Claude, an email connector, or your mail provider may process selected information according to its own terms. Read Privacy and Security before connecting email.

The seven-step path

1. Download the ZIP and keep the original ZIP as a clean backup.
2. Extract it; do not work inside the compressed file.
3. Open START-HERE.pdf.
4. Copy reference-project to a new folder named My-Job-Search-Dashboard.
5. Follow the section for your operating system.
6. Set the first date of your current job search.
7. Preview the first email import before you confirm it.

Success looks like this: your browser opens <http://localhost:8010/dashboard.html>, the heading shows your campaign date, and the initial sample data contains only fictional companies. Email connection comes after that local test passes.

Before you begin

Use the same email address for job applications and status messages. Install a current Python 3 release. Have Codex or Claude available. Set aside about 30 minutes for the first setup and another 10-20 minutes to connect email.

Never paste a normal mailbox password, access token, app password, data.json, or a raw email into an AI chat or a support message. Secrets belong in an operating-system environment variable or the provider's approved sign-in flow.

Know the Files in Your ZIP

A simple map of what to open, copy, and leave alone.

Start here

START-HERE.pdf is the only manual and the file you are reading now. reference-project is the tested local application. prompts contains ready-to-paste instructions for Codex and Claude. setup contains small platform launch notes. support contains a safe diagnostic checklist and an email template.

MANIFEST.sha256 is an integrity list. It lets a technical helper confirm that files were not changed accidentally. sources.md links to official provider documentation. PRIVACY.md and LICENSE.txt state the boundaries of the free kit.

Make a working copy

Keep the downloaded ZIP unchanged. After extraction, duplicate reference-project and rename the duplicate

My-Job-Search-Dashboard. Work only in that duplicate. If an experiment goes badly, delete the working copy and begin again from the extracted reference project.

Do not replace the fictional data.json with your current dashboard file until the local sample opens successfully. Do not add the Human Workforce logo to the working application; the logo belongs to this guide and the public demonstration, not to your private tool.

Choose Your Setup Path

Pick one operating system, one assistant, and one email route.

Three choices

Operating system: macOS, Windows, or Linux. Assistant: Codex or Claude. Email route: a supported read-only connector first, or provider IMAP as a fallback. Write those three choices down before continuing so you do not mix commands from different sections.

Connector-first is easiest for most people because the provider handles sign-in and token storage. IMAP can be useful for Gmail, Yahoo, iCloud, or another provider, but it requires more careful credential handling. Outlook.com normally requires Modern Authentication, so use a supported connector rather than trying to force a password-only IMAP login.

Apple Mail is an app, not a provider

If you read mail in Apple Mail, identify the account behind it: iCloud, Gmail, Yahoo, Microsoft, or a work provider. Follow that provider's connection instructions. The dashboard reads the mailbox; it does not automate the Apple Mail desktop application.

Install on macOS

Use Finder and Terminal to start the local reference project.

Extract and verify Python

In Finder, double-click

HumanWorkforce-Job-Search-Dashboard-Free-Kit.zip. Move the extracted folder to Documents. Copy reference-project and rename the copy My-Job-Search-Dashboard.

Open Terminal from Applications > Utilities. Type `python3 --version` and press Return. A version number means Python is available. If the command is missing, install Python 3 from python.org, close Terminal, reopen it, and run the check again.

Start the dashboard

In Terminal type `cd` followed by one space, drag My-Job-Search-Dashboard from Finder into the Terminal window, and press Return. Then run: `sh scripts/start-mac-linux.sh`

Open <http://localhost:8010/dashboard.html>. Keep Terminal open while using the dashboard. To stop it, return to Terminal and press Control-C. If macOS asks whether Terminal may accept incoming connections, deny public-network access; this server is intended

for your computer only.

macOS checkpoint

The browser shows a local address, fictional sample companies, search and status filters, and a Refresh button. If the page does not open, run `python3 server.py 8010` from the same folder and read Troubleshooting before changing files.

Install on Windows

Use File Explorer and PowerShell; no administrator account is normally required.

Extract and verify Python

In File Explorer, right-click the ZIP, choose Extract All, and place the folder in Documents. Open it, copy `reference-project`, and rename the copy `My-Job-Search-Dashboard`. Do not launch files while they are still inside the ZIP preview.

Open PowerShell. Run `py --version`. If Windows cannot find `py`, install Python 3 from python.org and select Add Python to PATH during installation. Reopen PowerShell and repeat the check.

Start the dashboard

In File Explorer open `My-Job-Search-Dashboard`, click the address bar, copy the full folder path, then run `cd "PASTE-YOUR-FOLDER-PATH-HERE"` in PowerShell. Run: `powershell -ExecutionPolicy Bypass -File scripts/start-windows.ps1`

If your organization blocks scripts, run `py server.py 8010` instead. Open <http://localhost:8010/dashboard.html>. Keep PowerShell open. Press Control-C there to stop the dashboard.

Windows checkpoint

The address starts with localhost or 127.0.0.1, not a public web address. Windows Firewall should not need a public exception. If it asks, cancel and confirm that `server.py` binds only to 127.0.0.1.

Install on Linux

Use your distribution's Python 3 and a terminal.

Extract and verify

Extract the ZIP with your file manager into Documents or another user-owned directory. Copy `reference-project` to `My-Job-Search-Dashboard`. In a terminal run `python3 --version`. If missing, install Python 3 using your distribution's normal package manager and documentation.

Change into the working folder with `cd`, then run `sh scripts/start-mac-linux.sh`. Open <http://localhost:8010/dashboard.html>. Stop with Control-C. Do not run the project with `sudo`; it does not require elevated privileges.

Linux checkpoint

Use `ss -ltn` if you need to confirm the listener. It should show 127.0.0.1:8010, not 0.0.0.0:8010. Desktop distributions may open the browser automatically; headless machines are outside this self-service desktop guide.

Use Codex

Let Codex verify and configure the supplied project without inventing a new application.

Open the working folder

Open My-Job-Search-Dashboard as the workspace in the Codex app or Codex CLI. Open prompts/codex/01-setup.md from the extracted kit, copy its full contents, and paste it into the task. Tell Codex which operating system and email provider you chose.

The prompt instructs Codex to inspect the supplied code, run its Python and JavaScript tests, preserve local-only binding, and explain any failed check. Approve only changes inside your working project. Do not approve publishing, cloud deployment, mailbox sending, deletion, or broad filesystem access.

What Codex should report

Expect a Python test count, a JavaScript test count, the start command for your operating system, and the localhost link. For email setup, Codex should preview qualifying records before the first import. If it asks for a normal email password in chat or proposes storing a secret in data.json, stop and use Troubleshooting.

Use Claude

Use Claude Code with the same project boundaries and verification steps.

Open and prompt

Start Claude Code in My-Job-Search-Dashboard. Open `prompts/claude/01-setup.md` from the extracted kit and paste the complete prompt. State macOS, Windows, or Linux and name the provider behind your mailbox.

Allow Claude to read and edit the working project and run local tests. Review requests that reach outside the folder. The correct outcome is the supplied dashboard running locally, not a rewritten cloud service.

Connector work

If your Claude environment offers an authorized mail connector or MCP server, ask for read-only search and message-reading operations only. Reject sending, deleting, labeling, or account-management permissions. Use `prompts/claude/02-connector-refresh.md` for the supervised refresh design.

Choose Your Campaign Start Date

Keep a new search separate from job applications made years ago.

Pick the boundary

Choose the first calendar day on which you applied during the current job search. Use your local timezone. The date is inclusive: messages received on that day may qualify. Do not choose the mailbox creation date, your graduation date, or the date you installed this tool.

The initial scan uses this boundary. Later scans use the later of the campaign date and the last successful scan time, with controlled overlap and stable message-ID deduplication. That is what prevents old campaigns from appearing while protecting against messages arriving around a refresh boundary.

Confirm before importing

Open data.json only after making a backup. Verify `search_start_date` is an ISO date such as 2026-08-01 and `timezone` names your local zone. On the first refresh, the dashboard presents a preview. Review employer, role, date, subject summary, and proposed status; confirm only when the records belong to this campaign.

Connect Gmail

Prefer Google's authorized sign-in path; use IMAP credentials only when necessary.

Recommended route

Use a supported Gmail connector that clearly identifies read-only message search and read scopes. Sign in on Google's own authorization page. Verify the selected Google account is the same address used for applications. Decline Gmail send, modify, delete, contacts, calendar, and Drive permissions because the dashboard does not need them.

Ask the assistant to retrieve only likely job-application messages on or after the campaign boundary, return metadata and a short factual excerpt, and never save a full body or attachment in the project.

IMAP fallback

Gmail IMAP uses `imap.gmail.com` on port 993 with TLS. Google may require two-step verification and an app password for clients that cannot use modern sign-in. Store that generated value in the environment variable named by your local configuration, never in a file or prompt. A normal Google password is not an acceptable fallback.

After setup, run the preview. Revoke the app password or connector grant from your Google Account when you uninstall or stop using the tool. Provider settings change; use the official Google link in `sources.md` if a screen differs from this guide.

Yahoo Mail

Connect to the incoming mailbox without enabling outgoing mail.

Connection details

Prefer an authorized connector if available. Yahoo's published incoming IMAP service uses `export.imap.mail.yahoo.com` on port 993 with SSL. Depending on the account, Yahoo may require an app password. Generate it in Yahoo account security and place it only in the configured environment variable.

The project does not need SMTP, so do not configure `smtp.mail.yahoo.com`. Preview the first import and check ATS messages carefully: a sender such as Greenhouse or Workday is not necessarily the employer. The included matching logic uses employer, role, subject patterns, and thread context to reduce collisions.

Apple Mail and iCloud

First identify the provider; use iCloud settings only for an iCloud mailbox.

Apple Mail accounts

Open Mail > Settings > Accounts and identify whether the address is iCloud, Google, Yahoo, Microsoft, or another provider. Follow the matching section. Apple Mail itself is not the mailbox service and is not scanned directly by this kit.

iCloud Mail

For an @icloud.com, @me.com, or @mac.com mailbox, Apple's published incoming server is imap.mail.me.com on port 993 with SSL. Apple may require an app-specific password. Generate it in Apple Account security and use an environment variable. Do not configure the outgoing SMTP service.

If Advanced Data Protection, managed-device rules, or account policy prevents access, do not weaken account security. Use an approved connector or manually maintain the dashboard until the provider route is supported.

Outlook and Other Mailboxes

Modern authentication and provider policy take priority over generic instructions.

Outlook.com and Microsoft 365

Use a supported Microsoft connector with read-only Mail permissions. Microsoft services generally require OAuth2 or Modern Authentication. Do not enable legacy authentication or paste a Microsoft password into the project to force IMAP to work. Work or school tenants may require administrator approval.

Other providers

Find the provider's official incoming-mail documentation. You need the IMAP hostname, TLS port, username format, authentication method, and confirmation that read-only access is possible. Do not guess settings from an unrelated blog. Do not configure outgoing mail.

If the provider cannot offer a safe supported route, use the JSON bridge with a supervised assistant workflow or update records manually. A partially working insecure connection is not a successful installation.

Configure the Email Route

Use connector-produced excerpts or read-only IMAP without placing full messages in the project.

Connector bridge

The bridge accepts a small JSON list of normalized job events. Each event needs `message_id`, `received_at`, `sender`, `subject`, `excerpt`, `company`, and `position`; `thread_id` is optional. Excerpts are limited to 500 characters. A field named `body` or `full_body` is rejected. Delete the bridge file after a successful import if your workflow does not need it.

Configure `email_route` with type `bridge` and the relative path to the event file. Have Codex or Claude create only the schema values from relevant messages since the effective boundary. Never export a mailbox archive or attachments into the project.

Read-only IMAP

Configure type `imap`, the official hostname, port `993`, `INBOX` mailbox, username, and `password_env`. The `password_env` value is the name of an environment variable, not the secret itself. Start the server from a terminal session where that variable is set. The implementation selects messages, applies exact timestamp and job relevance filters, and discards candidates without a defensible employer and role match.

Preview and Confirm the First Import

The first scan is intentionally a two-step operation.

Review the preview

Click Refresh. When no successful scan exists, the server returns a preview instead of writing. The dashboard shows the qualifying records and asks for confirmation. Check that every item is job-related, on or after the campaign start, and assigned to the correct employer and role.

Ordinary personal mail must not appear. ATS messages should name the actual employer, not only Workday, Greenhouse, Lever, or another platform. If several roles at one employer collapse together, cancel and correct the extraction before importing.

Confirm or cancel

Confirm only after the preview is sound. The server then merges events, validates the state, creates a backup, atomically replaces data.json, and advances last_successful_scan_at only after success. Canceling changes nothing. A failed write must leave both the prior data and prior cursor available.

Use the Dashboard Every Day

Start, refresh, review, and stop in a few minutes.

Normal routine

Start the server with your platform command and open the localhost page. Click Refresh once. Wait for the success message and verify the last scan time. Search by company or role, filter by status, and open Needs Review items against the original email.

Applied/Open means active or acknowledged.

Interview/Screening covers recruiter screens and interviews.

Action Required includes assessments, scheduling, or documents. Offer and Rejected require explicit signals. Stale/No Update is calculated after a period without a meaningful event.

Needs Review records uncertainty rather than inventing certainty.

What refresh reads

Refresh considers new relevant messages since the effective boundary and a small safe overlap. Stable message IDs prevent duplication. It stores concise event facts, not attachments or full email bodies. The original email remains the authoritative record for deadlines, interview details, and offers.

Start a New Job Search

Archive the old campaign instead of mixing it with a future search.

Archive and reset

Stop the server. Run `python3 scripts/new_campaign.py YYYY-MM-DD` on macOS/Linux or `py scripts/new_campaign.py YYYY-MM-DD` on Windows, replacing the date with the first day of the new search. The script creates a uniquely named archive and resets active applications and the scan cursor.

Confirm the archive exists before restarting. Open the dashboard, verify the new campaign date and empty active list, then run a new preview. Never simulate a new campaign by simply moving the cursor forward; that leaves old applications active.

Back Up and Restore

Make every update and repair reversible.

Back up

Stop the server before a manual backup. Run `python3 scripts/backup.py` on macOS/Linux or `py scripts/backup.py` on Windows. Confirm the printed backup path exists and has a nonzero size. Copy important backups to an encrypted location you control.

Automatic refresh also creates a backup immediately before replacing active data. Keep several recent backups. A backup

contains job-search details and should be treated as private personal data.

Restore

Stop the server. Run the restore script with the exact backup path, review the confirmation, and restart. Verify campaign date, counts, and recent applications before refreshing. Do not restore an arbitrary JSON file or a backup created by a different application.

Update the Free Kit

Replace application files without overwriting your job-search history.

Safe update

Download a new kit from the Human Workforce demo page and keep the old ZIP. Stop your dashboard and create a backup. Extract the new kit separately. Ask Codex or Claude to compare the new reference project with your working project, preserving data.json, archives, local email route settings, and secret environment variables.

Run all included tests before the first refresh. Start the updated server and inspect the dashboard with existing data. If anything is wrong, stop it and restore the saved working folder. Never copy a new sample data.json over your active file.

Troubleshooting

Fix the common problem without exposing your mailbox or history.

Dashboard does not open

Confirm the terminal is still running and that you started it from My-Job-Search-Dashboard. Try the direct command for your platform. If port 8010 is in use, stop the other local program or choose another port and use the matching localhost address. Do not change the bind address from 127.0.0.1.

Refresh finds nothing

Confirm the same mailbox was used for applications. Check the campaign start date and timezone. Verify the connector account or IMAP hostname and environment-variable name. Search the mailbox manually for a known application acknowledgment after the boundary. Do not solve this by setting the date years earlier without reviewing the resulting preview.

Wrong company, role, or status

Compare the event with the original email. ATS sender domains can represent many employers. Keep separate roles at the same employer distinct. Correct the normalized employer/position or route the event to Needs Review. Status classification must follow explicit message facts, not optimistic interpretation.

Safe support

Use prompts/codex/03-diagnose.md or prompts/claude/03-diagnose.md. The diagnostic report must redact nested tokens, passwords, mailbox identities, message subjects, and local usernames. Use support/email-support-template.txt. Never attach data.json, an inbox export, raw email, screenshots containing identities, or credentials.

Privacy and Security

Understand what stays local and what may leave the computer.

Local data boundary

The web server binds to 127.0.0.1 and serves the dashboard only to your computer. data.json, archives, and backups remain in the working project unless you copy or sync them elsewhere. Browser pages from other sites cannot trigger refresh because the server validates local Host and Origin values and serializes refreshes with a lock.

Localhost protects network exposure; it does not encrypt the disk or prevent another user account with file access from reading the project. Use operating-system login security, disk encryption, backups you control, and a private non-shared folder.

Email and AI boundary

A connector or coding assistant may process selected email information in the provider's cloud. Review its current privacy terms and granted scopes. IMAP connects to the mail provider directly, but configuration and troubleshooting can still reveal data if you paste it into chat. Use short factual excerpts only and never persist full message bodies or attachments.

Least privilege

Grant read-only message search/read access, to the one mailbox used for applications, for as long as you use the tool. No feature requires sending, deleting, moving, labeling, contacts, calendar, or outgoing SMTP. Revoke connector grants and app passwords when no longer needed.

Uninstall

Disconnect access first, then remove the local files.

Remove safely

Stop the server. In the provider account, revoke the connector grant or delete the app-specific password. Remove any shell or PowerShell environment variable you created. Decide whether you need an encrypted archive for your records; remember that it contains personal job-search data.

Move My-Job-Search-Dashboard, extracted kit copies, and unwanted backups to Trash or Recycle Bin, then empty it when you are certain. The original job emails remain in the mailbox

because this tool never deletes or modifies them.

Quick Reference

Keep these checks near your working folder.

Start and stop

macOS/Linux: sh scripts/start-mac-linux.sh. Windows: powershell -ExecutionPolicy Bypass -File scripts/start-windows.ps1, or py server.py 8010. Open <http://localhost:8010/dashboard.html>. Stop with Control-C in the terminal.

Refresh checklist

Confirm campaign date. Click Refresh once. Review a first-import preview. Verify the successful scan timestamp. Review Needs Review against original mail. Never advance the cursor after a failed save.

Safety checklist

No normal password or token in project files. No full body or attachment in bridge files. No SMTP or mailbox-write scope. No public bind address. No data.json or raw email sent to support. Back up before updates and new campaigns.

Official references

Live demonstration:

<https://human-workforce-job-search-demo.fly.dev>. Human

Workforce demo catalog:

<https://human-workforce-consulting.vercel.app/demos>. Current

provider and assistant documentation links are collected in sources.md inside the kit. When a provider screen changes, the official source wins.

Live demo: <https://human-workforce-job-search-demo.fly.dev>

Demo catalog: <https://human-workforce-consulting.vercel.app/demos>