CASE STUDY

SmartHR

AI-Assisted Resume Shortlisting

From a crowded HR inbox to a ranked, explained and HR-approved shortlist

PREPARED BY **Ameya Gadhave** | Kiaratech Solutions Pvt Ltd

01 Project Overview

HR teams receive around 1,000 resumes by email every day, and each one must be downloaded and read before shortlisting can begin. **SmartHR** collects resumes automatically, compares them with each job's requirements by meaning, and gives HR a ranked shortlist in which every match is backed by a real line from the resume.

02 The Challenge

Business challenges

- High volume, often arriving in bursts.
- Every resume downloaded and read by hand.
- Judgement varied from one reader to another.

Technical challenges

- Keyword search misses differently worded skills.
- AI can claim skills not in the resume.
- Private data and a server with no GPU.

03 Our Approach

We traced a resume's journey from inbox to email, then separated machine work from human decisions:

- **HR stays in control** — HR approves each job checklist and every candidate before contact.
- **Evidence over opinion** — the AI must quote the exact resume line, and software checks it is real.
- **Fair by design** — names, photos and personal details are hidden from the AI.
- **Private by default** — all AI runs on the organisation's own server.

04 The Solution

An eight-step pipeline: steps 1 to 7 run automatically, and in step 8 HR reviews the evidence on one screen and decides. Approved emails are sent exactly once, with a 10-minute undo.

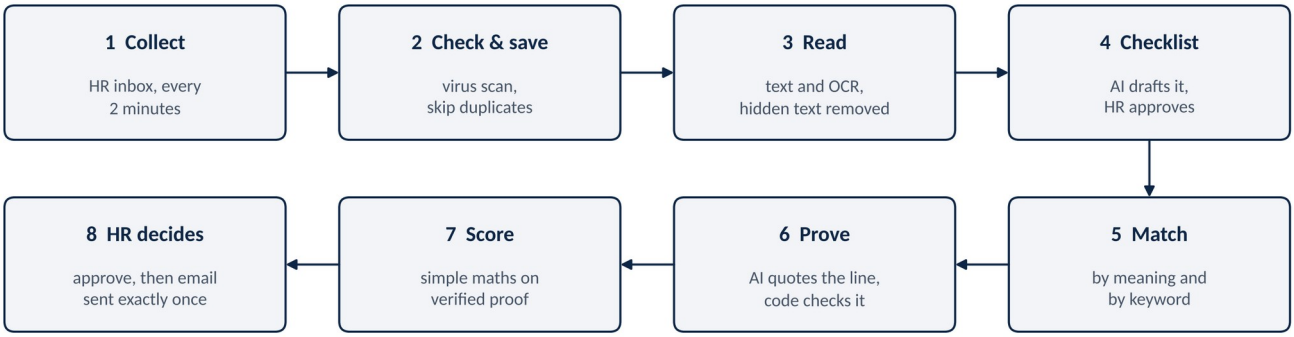


Figure 1 · How a resume moves through SmartHR

05 Results and Impact

Area	Before	With SmartHR
Time	Every resume downloaded and read by hand	Collected and read automatically, every 2 minutes
Matching	Keyword skimming misses good candidates	Skills matched by meaning, with evidence for each match
Consistency	Varies by reader and time of day	Same checklist and the same result every time
Privacy and control	Manual emails; personal details visible	AI never sees personal details; no email without HR approval

Results reflect the outcomes SmartHR was designed to deliver.

06 Technology Stack

Area	Technologies
Application	React, TypeScript, Python, FastAPI, Celery with Redis
Artificial intelligence (local)	Ollama with Qwen2.5-7B, bge embedding and reranking models
Documents and data	PyMuPDF, Tesseract OCR, ClamAV, MongoDB Atlas Vector Search
Microsoft 365	Microsoft Graph API, Microsoft Entra ID sign-in

> **Key takeaway:** Let the machine do the reading. Let people make the decisions.