# **📘 Project Report: AI-Powered Job Recommendation System (n8n + Gemini)**

## **1. Problem Explanation**

Finding relevant jobs online is time-consuming. Job seekers must filter hundreds of postings, compare them with their own profile, and then craft personalized applications.  
 This project automates the entire process by:

* Scraping jobs from Indeed,
* Filtering & summarizing job descriptions,
* Matching them against a user profile with Gemini AI,
* Generating tailored cover letters,
* Saving results to Google Sheets,
* And sending an email with recommendations every day at 10 AM.

## **2. n8n Workflow Breakdown (Node-by-Node)**

**Node 1 – Schedule Trigger**

* Runs daily at **10 AM**.
* Ensures automation is consistent without manual input.

**Node 2 – HTTP Request**

* Calls **Indeed API via RapidAPI**.
* Retrieves latest job postings based on keywords & location.

**Node 3 – Code**

* Extracts the data array from API response.
* Converts it so that each job is treated as an individual item in the workflow.

**Node 4 – Gemini (Message a Model)**

* Input: Raw job description.
* Output: Summarized and cleaned job description.
* Also filters jobs (e.g., location India, experience ≤ 2 years).

**Node 5 – Code**

* Parses Gemini’s JSON output into structured job objects.
* Makes it easier for downstream nodes to handle.

**Node 6 – Gemini (Message a Model)**

* Matches **summarized JD** with the **user’s profile**.
* Assigns a **relevance score (0–100)** to each job.

**Node 7 – Code**

* Parses the Gemini scoring output (JSON string) into an array.
* Ensures scores align with the correct job entries.

**Node 8 – Gemini (Message a Model)**

* Generates a **personalized cover letter** for each job.
* Uses job details + user profile as input.

**Node 9 – Code**

* Parses cover letter JSON outputs into clean text format.

**Node 10 – Append or Update Row in Google Sheets**

* Stores the final processed data:  
  + Job Title, Company, Location, Summary, Score, Cover Letter.
* Creates a structured database of opportunities.

**Node 11 – Gmail: Send Message**

* Sends an automated email with top recommended jobs.
* Includes scores, summaries, and links for quick application.

## **3. API & Prompt Usage**

* **Indeed Jobs API (RapidAPI)** → fetches live job postings.
* **Gemini API** → three different use cases:  
  1. **Summarization & Filtering** (Node 4)
  2. **Job Matching & Scoring** (Node 6)
  3. **Cover Letter Generation** (Node 8)
* **Google Sheets API** → stores structured job matches.
* **Gmail API** → delivers recommendations directly to inbox.

## **4. Challenges & Solutions**

**🔹 Challenge 1: API Rate Limits**

* RapidAPI and Gemini have rate limits.
* ✅ Solution: Batched requests + retry/delay in n8n settings.

**🔹 Challenge 2: Unstructured Job Data**

* Job descriptions vary a lot (some missing experience/location).
* ✅ Solution: Gemini prompt ensures fallback (“Not specified”).

**🔹 Challenge 3: JSON Parsing Errors**

* Gemini sometimes returns extra text instead of pure JSON.
* ✅ Solution: Code nodes clean & parse outputs before next step.

## **5. Summary of Learnings**

* Gained hands-on experience building **end-to-end automation** in n8n.
* Learned how to integrate **Gemini LLM** for summarization, scoring, and text generation.
* Improved skills in **API integration** (RapidAPI, Google Sheets, Gmail).
* Understood how to handle **real-world data challenges** (rate limits, parsing, missing fields).
* Built a **production-ready workflow** that saves hours of manual effort.