Prompt Engineering Internship Assignment

# Company: FirstWork

FirstWork is transforming how HR technology empowers blue-collar and frontline workers. We equip businesses with advanced tools to rapidly hire, verify, and manage large-scale workforces. One of the core components of our platform is an AI-powered document processing system that operates effectively in low-resource environments to handle various document types—from identity checks to onboarding and payroll proofs—at scale.  
  
We’re currently building the next generation of prompt management infrastructure. As an intern, you’ll play a critical role in developing this system.

# Assignment Overview

## Objective

Build a prototype using Google Gemini (via free-tier API key) as the core LLM to extract structured data from the following datasets:  
  
- Driving Licenses – Easy  
- Shop Receipts – Medium  
- Resumes (CVs) – Hard  
  
You are encouraged to use open-source tools (OCR, validators, parsing utilities, etc.) to improve accuracy and reduce cost.

## Real-World Context

Assume FirstWork is onboarding a major logistics company. Your task is to build an initial version of a document understanding pipeline that:  
  
- Extracts structured data from unstructured documents  
- Manages noisy, inconsistent, and irrelevant inputs  
- Is cost-effective and easy to scale for future document types  
  
This prototype will assess both your prompt design and systems thinking abilities.

# Technical Details

## Input

Three folders of images or PDFs will be provided:  
  
- driving\_license/  
- shop\_receipts/  
- resumes/

## Output

Generate a JSON output for each document, extracting the following fields:  
  
1. Driving License:  
- Name  
- Date of Birth  
- License Number  
- Issuing State  
- Expiry Date  
  
2. Shop Receipt:  
- Merchant Name  
- Total Amount  
- Date of Purchase  
- List of Items (name, quantity, price)  
- Payment Method  
  
3. Resume:  
- Full Name  
- Email  
- Phone Number  
- Skills  
- Work Experience (company, role, dates)  
- Education (institution, degree, graduation year)  
  
\*Note: Be prepared to handle new fields introduced during evaluation.\*

# Implementation Requirements

- Use Google Gemini API (free tier) as the primary LLM.  
- Optional Tools: OCR (Tesseract, EasyOCR, Google Vision), Parsers, Validators.  
- Code should be modular, scalable, and robust against irrelevant or malformed documents.

# Evaluation Criteria

1. Production-Ready Code: Clean, modular, and scalable.  
2. Prompt Quality: Effectively handles edge cases and hallucinations.  
3. Extraction Accuracy: Correctness and completeness of outputs.  
4. Extensibility: Ease of incorporating new document types.  
5. Code Quality: Architecture, naming, readability, logging, error handling.  
6. Cost Efficiency: Minimize token and API usage.  
7. Performance: Fast processing per document.

# Bonus Points

- LLM self-validation or retry logic  
- CLI or GUI for testing new prompts or datasets  
- Multi-language support  
- Benchmarks against other LLMs or tools

# Submission Guidelines

Deadline: Submit within 2 days of receiving datasets.  
Format: GitHub repo link or ZIP folder  
  
Include:  
- README.md with setup and execution instructions  
- Sample output JSONs