Internship Assignment

Building a CCR(California Code of Regulations) Compliance Agent

1. Overview and Intent

This assignment is part of the selection process for an **Engineering Internship** at a consulting firm which specialises in state and local government agencies. Reach out to aravind.karanam@gmail.com or 8688743302 on whatsapp for any clarifications in scope.

The goal of this project is **NOT** to test whether you already know web crawling, vector databases, or regulatory law.

Instead, this assignment is designed to evaluate:

1. **Communication skills**
   - How clearly you explain your thinking, approach, and trade-offs
   - How well you document decisions, assumptions, and unknowns
2. **Ability to leverage modern LLMs**
   - Effective use of tools like **ChatGPT, Claude, or similar LLMs**
   - Using LLMs to reason, plan, debug, refactor, and generate structure
   - Combining LLMs with existing libraries rather than reinventing everything
3. **Fearlessness in approaching non-trivial problems**
   - This is a deliberately large, ambiguous, real-world problem
   - We care more about *how* you break it down than about perfection
4. **Engineering judgment**
   - Making reasonable assumptions
   - Knowing when to simplify
   - Instrumenting systems instead of guessing
   - Showing ownership over an end-to-end problem

You are **strongly encouraged** to:

- Use LLMs extensively
- Ask questions early and often
- Document partial progress and dead ends
- Be explicit about trade-offs and limitations

You are **not expected** to:

- Know the California Code of Regulations (CCR) beforehand
- Achieve perfect coverage on the first attempt
- Avoid external tools, libraries, or AI assistance

If anything is unclear—technical, product-related, or scope-related—you should **reach out directly for clarification**.
Asking good questions is a positive signal.

---

2. Internship Details

- **Role:** Engineering Intern
- **Duration:** 3 months
- **Stipend: ₹15,000 per month**
- **Location:** Remote

Conversion to Full-Time

High-performing interns may receive a **full-time offer** at the end of the internship based on:

- Technical progress
- Ownership and initiative
- Communication quality
- Ability to work with ambiguity
- Effective use of AI tooling

Performance will be evaluated **holistically**, not just on code output.

---

3. Assignment Overview

Problem Statement

Build a system that can:

1. Crawl the **California Code of Regulations (CCR)** from `https://govt.westlaw.com/calregs`
2. Extract **every law section** as clean **Markdown**
3. Organize the data into the **canonical CCR hierarchy**
4. Load the data into any **vector database (with a free tier)**

5. Build an **AI agent** that advises facility operators (e.g. restaurants, movie theaters, farms etc) on **which CCR sections apply to them**, with citations

---

4. Key Technical Challenge (Very Important)

The **core difficulty** of this assignment is:

> **Reliably extracting *every single CCR section***
> not just "most pages" or a small subset.

You are expected to:

- Design a crawling strategy that prioritizes **completeness**
- Prove and validate coverage
- Track failures and retries
- Be explicit about what was missed (if anything)

A partially correct but well-instrumented system is **better** than a silent, incomplete one.

---

5. Canonical Data Structure Expectations

You must organize CCR data into a **canonical hierarchy**.
At minimum, your extracted data should support:

- `title_number` (e.g., 17)
- `title_name`
- `division` (number / name, if present)
- `chapter`
- `subchapter` / `article` (if present)
- `section_number` (e.g., 1234)
- `section_heading`
- `citation` (e.g., `17 CCR § 1234`)
- `breadcrumb_path`
- `source_url`
- `content_markdown`
- `retrieved_at`

Not all sections contain every level—your schema must handle missing levels gracefully.

---

6. Crawling Requirements (Using Crawl4AI)

You must use **Crawl4AI** to perform the crawl.

Minimum expectations:

- Controlled concurrency (avoid hammering the site)
- Retry logic with exponential backoff
- URL normalization and deduplication
- Persistent checkpoints (resume after crashes)
- Clear separation between:
  - URL discovery
  - Section content extraction

Your crawler should output structured data (JSON / JSONL) that can be indexed later.

---

7. Vector Database Requirements

Choose **any vector database with a free tier**, such as:

- Supabase + pgvector
- Qdrant Cloud
- Pinecone
- Weaviate
- Chroma (local acceptable for prototype)

Minimum requirements:

- Semantic search
- Metadata filtering (by title, section, etc.)
- Idempotent upserts (safe re-indexing)
- Stored citations and source URLs

You must justify your choice briefly.

---

8. Compliance Agent Requirements

Build an AI agent that can answer questions like:

- “What CCR sections apply to a restaurant in California?”
- “What regulations should a movie theater operator be aware of?”
- “What laws apply to farms or agricultural facilities?”

Agent expectations:

- Uses retrieval (RAG), not hallucination
- Returns **specific CCR section citations**
- Explains *why* each section applies
- Asks follow-up questions if information is insufficient
- Includes a clear “not legal advice” disclaimer

Output quality matters more than UI polish.

---

9. Expected Deliverables

Submit a repository on github, which should contain:

Code

- Crawling scripts
- Indexing pipeline
- Agent implementation (CLI or small API)

Data

- Discovered section URLs
- Extracted CCR sections (structured)
- Coverage / completeness report

Documentation

- `README.md` with:
  - Setup instructions
  - How to run each stage
  - Design decisions
  - Known limitations
  - What you would improve next

---

10. Evaluation Criteria

You will be evaluated on:

1. **Coverage & correctness** (most important)
2. **Clarity of communication**
3. **Use of LLMs as leverage**
4. **Problem decomposition**
5. **Engineering hygiene**
    - Logging
    - Idempotency
    - Error handling
6. **Honesty about limitations**

Perfection is not required.
Clear thinking and ownership are.

---

11. Communication & Support

If you are blocked, unsure, or need clarification at any point:

- **Reach out directly**
- Ask questions
- Share partial progress

Silence is worse than asking.

---

12. Final Notes

This assignment mirrors the kind of work you would do here:

- Open-ended
- Real-world
- Ambiguous
- High leverage with AI tools

We are less interested in *how much you know today*
and more interested in *how you learn, think, and communicate*.

Good luck.