Wasserstoff – AI Software Intern Task

Role: AI Intern (Generative AI) – 6 Month Full-Time Internship
Location: Remote/Onsite
Company: Wasserstoff
Contact: Divyansh Sharma – divyansh.sharma@thewasserstoff.com

---

Overview

Welcome to Wasserstoff!

As an AI Intern for 6 months (full-time), you will engage in research-driven development of Generative AI applications. The internship emphasizes both academic research and hands-on implementation, contributing to real product development, exploring research papers, and building internal tools.

Internship Qualification Project: Document Research & Theme Identification Chatbot

This is not a Basic RAG, you need to dive deep and create this system.

Objective

Build a chatbot that can:

- **Ingest 75+ documents** (PDFs, scans, text, etc.)
- **Let users ask questions** about the content
- **Extract answers** from each document with accurate **citations** (page, paragraph, or sentence)

- **Identify and summarize common themes** across all documents for **each query**
- Present everything in a simple, easy-to-use web interface

Key Requirements

1. Document Upload & Knowledge Base

- Users must be able to upload at least 75 documents (PDF, image scans, text, etc.)
- Extract text from documents (including OCR for scanned images)
- Store documents and extracted text in a searchable database

2. Query Processing & Document Management

- Create a UI for:
    - Viewing uploaded documents
    - Submitting questions in natural language
- For each user question:
    - Search **each document** for relevant answers
    - Return each answer **with clear citations** (e.g., DocID, Page, Paragraph)
- **Display all individual document answers in a table format (see example below)**

3. Theme Identification & Synthesis

- After querying, analyze all document responses together

- **Identify main themes** (there may be more than one) **per query**
- Give a **synthesized, chat-style answer** summarizing each theme, with supporting document citations.

Extra Credit (Optional, for higher score)

- Give paragraph/sentence-level citations (not just document/page)
- Visual interface for mapping citations (e.g., clickable links from theme → document)
- Filtering: allow users to sort/filter by date, author, document type, or relevance
- Let users include/exclude specific documents from search

Technical Guidelines

- **AI Models:** Use any major LLM (OpenAI GPT, Gemini, Groq/LLAMA, etc.)

  Tip: OpenAI and Gemini provide free credits. Groq is free for LLAMA models.

- **Vector Search:** Use Qdrant, ChromaDB, or FAISS for efficient semantic search
- **OCR:** Use Tesseract or PaddleOCR for scanned PDFs/images
- **Backend:** Python (FastAPI or Flask recommended)
- **Frontend:** Any modern web framework (Streamlit, React, etc. — simple UI is fine)

- **Deployment:** Use free cloud platforms (see list below)
- **Version Control:** Use Git for your code, with clean commits and comments
  You can use any kind of service or framework you prefer, there is no need to ask for confirmations, the execution of the task is on you.

Deliverables

1. **Web-based chatbot** (user uploads docs, asks questions, gets answers & themes)
2. **Well-documented source code** (comments, modular structure, README)
3. **Brief report** (explain your approach, tech choices, and methodology)
4. **Demo video or presentation** (showcase main features and workflow)

Evaluation Criteria

- **Functionality:** Does the system work as described? Are answers and themes accurate, with proper citations?
- **Code Quality:** Clean, modular, readable code. Proper use of comments and Git.
- **User Experience:** Simple, clear, and intuitive UI.
- **Documentation:** Clear README, brief report, and video demo.
- **Error Handling:** Graceful handling of upload errors, OCR issues, model failures, etc.
- **System Design:** Is the design scalable? Can it be extended to more documents/users?

How Results Should Look

Individual document answers (tabular):

Document ID	Extracted Answer	Citation
DOC001	The order states that the fine was imposed under section 15…	Page 4, Para 2
DOC002	Tribunal observed delay in disclosure violated Clause 49…	Page 2, Para 1

Synthesized (theme) answer (chat format):

Theme 1 – Regulatory Non-Compliance:

DOC001, DOC002: Highlight regulatory non-compliance with SEBI Act and LODR.

Theme 2 – Penalty Justification:

DOC001: Explicit justification of penalties under statutory frameworks.

Recommended Project Structure

chatbot_theme_identifier/

├── backend/

│ ├── app/

│ │ ├── api/

│ │ ├── core/

│ │ ├── models/

│ │ ├── services/

│ │ ├── main.py

│ │ └── config.py

│ ├── data/

│ ├── Dockerfile

│ └── requirements.txt

├── docs/

├── tests/

├── demo/

└── README.md

Dataset

Use any set of related documents (e.g., legal cases, technical reports, policy docs, research articles, business files).
If you need a dataset suggestion, just ask!

Deployment – Free Options

- Render
- Railway
- Replit
- Hugging Face Spaces
- Vercel

Questions or Submissions?

Reach out to Divyansh Sharma (divyansh.sharma@thewasserstoff.com) with any questions, for dataset help, or to submit your final project.

Checklist to track all components of the task

☑ Upload and process 75+ documents (PDF/text/image with OCR)
☑ Store & manage documents for fast search
☑ User can ask questions in natural language

☑ Extract and cite answers from each document
☑ Identify common themes and give synthesized, cited summary and generate a consolidated answer.
☑ Clean, well-commented code and project README
☑ Demo video or presentation

We look forward to seeing your creativity, research skills, and coding craftsmanship in action!