Deployment Details – Regulatory Monitoring Prototype

# 1. Development Environment

| **Component** | **Details** |
| --- | --- |
| **IDE** | [Cursor IDE](https://cursor.sh/) (VS Code-based, free) |
| **Package Manager** | [uv](https://github.com/astral-sh/uv) – fast Python package manager |
| **Python Version** | Python 3.10+ |
| **Version Control** | GitHub Repository → <https://github.com/psxom3/genai_regwatch> |

# 2. Core Components & Libraries

| **Function** | **Libraries/Tools** | **Description** |
| --- | --- | --- |
| **Scraping** | requests, beautifulsoup4, playwright | Extract circulars from RBI & NHB. |
| **File Handling** | PyMuPDF, pandas, python-docx | Extracts text from PDFs, Excel, CSV, DOCX. |
| **Database** | PostgreSQL, psycopg2-binary | Stores metadata, summaries, and actions. |
| **GenAI Processing** | Ollama (local LLM like llama3.2:3b or mistral:7b-instruct) | Performs summarization & compliance action extraction. |
| **Notifications** | smtplib (via Gmail SMTP) | Sends new circular alerts to your inbox. |
| **Web Dashboard** | Gradio | Simple browser-based dashboard for updates, summaries, and actions. |
| **Configuration** | python-dotenv, .env | Stores secrets, credentials, and environment paths. |
| **Utilities** | hashlib, datetime, json | File hashing, date handling, and serialization. |

# 3. Installation Steps (Local Deployment)

#### 1: Clone the Repository

git clone https://github.com/psxom3/genai\_regwatch

cd genai\_regwatch

#### 2: Install uv (if not installed)

pip install uv

#### 3: Install Dependencies

uv pip install -r requirements.txt

# 4. ****Environment Setup (.env Configuration)****

Make sure your .env file (already provided) looks like this:

# PostgreSQL Database

DB\_NAME=regdb

DB\_USER=postgres

DB\_PASS=password

DB\_HOST=localhost

DB\_PORT=5432

# Ollama Local Server

OLLAMA\_URL=http://localhost:11434

OLLAMA\_MODEL=llama3.2:3b

# SMTP (Gmail)

SMTP\_HOST=smtp.gmail.com

SMTP\_PORT=587

SMTP\_USER=mainkaromkar13@gmail.com

#SMTP\_PASS=your\_app\_password\_here

SMTP\_PASS=qgglzjdqwpednprw

ALERT\_EMAIL\_TO=mainkaromkar13@gmail.com

ALERT\_EMAIL\_FROM=mainkaromkar13@gmail.com

# Local storage path

LOCAL\_STORAGE=./storage/raw

# Parallel workers

MAX\_WORKERS=3

**Important:** For Gmail SMTP, you must use a Google App Password — not your login password.  
Follow: <https://myaccount.google.com/apppasswords>

# 5. ****Database Setup (PostgreSQL)****

#### Step 1: Install PostgreSQL

Download and install from <https://www.postgresql.org/download/>

During installation:

* Set username: postgres
* Set password: password
* Port: 5432

#### Step 2: Create Database

Open psql shell:

psql -U postgres

Then run:

CREATE DATABASE regdb;

\c regdb;

#### Step 3: Create Tables

Run the following commands inside psql:

CREATE TABLE reg\_updates (

id SERIAL PRIMARY KEY,

regulator VARCHAR(50),

title TEXT,

url TEXT,

pub\_date DATE,

hash VARCHAR(64) UNIQUE,

raw\_file\_path TEXT,

status VARCHAR(20),

inserted\_at TIMESTAMP

);

CREATE TABLE reg\_actions (

id SERIAL PRIMARY KEY,

update\_id INTEGER REFERENCES reg\_updates(id) ON DELETE CASCADE,

actions\_json JSONB,

created\_at TIMESTAMP

);

CREATE TABLE reg\_summaries (

id SERIAL PRIMARY KEY,

update\_id INTEGER REFERENCES reg\_updates(id) ON DELETE CASCADE,

exec\_summary TEXT,

created\_at TIMESTAMP

);

Verify tables:

\d+ reg\_updates;

\d+ reg\_actions;

\d+ reg\_summaries;

# 6. ****Ollama Setup (Local GenAI Engine)****

#### Step 1: Install Ollama

Download from <https://ollama.com/download> and install.

#### Step 2: Pull Model

Open Windows Powershell

ollama pull llama3.2:3b

(Or use mistral:7b-instruct for faster performance)

#### Step 3: Test

To check models: ollama list

ollama run llama3.2:3b

If you see a response, your model works!

#### Step 4: Run Ollama Server

Keep Ollama running in background:

ollama serve

Ensure it listens on <http://localhost:11434>.

# 7. ****Run Pipeline and**** Launch Gradio dashboard

In your **Cursor terminal** (or any terminal where the virtual environment is activated):

# Navigate to the project directory

cd genai\_regwatch

# Run the Gradio dashboard (this will open a local web interface)

uv run app.py