Setup Guide: RAG in Production Workshop

This guide covers everything needed to set up and run the project from scratch.

Prerequisites

Requirement	Minimum Version	Notes
Python	3.10+	Uses modern type hints (`list[...]`, `X | Y`)
Ollama	latest	Local LLM inference server
pip or **conda**	-	For dependency management
Git	-	To clone the repository

1. Clone the Repository

```
git clone <repo-url>
cd prod-rag
```

2. Create a Python Environment

Choose **one** of the following options:

Option A: venv (recommended)

```
python -m venv rag-workshop

# Windows (PowerShell)
Set-ExecutionPolicy -Scope CurrentUser -ExecutionPolicy RemoteSigned
rag-workshop\Scripts\activate

# macOS / Linux
source rag-workshop/bin/activate
```

Option B: Conda

```
conda create -n rag-workshop python=3.10 -y
conda activate rag-workshop
```

Option C: Hatch

```
pip install hatch
hatch env create
hatch shell
```

3. Install Python Dependencies

```
pip install -r requirements.txt
```

This installs the following packages:

Package	Purpose
`langchain`	LLM orchestration framework
`langchain-core`	Core abstractions (Document, messages, etc.)
`langchain-ollama`	Ollama integration for LLM and embeddings
`langchain-community`	Community integrations (FAISS, BM25 retriever)
`langgraph`	Graph-based workflow orchestration
`faiss-cpu`	FAISS vector store for similarity search
`beautifulsoup4`	HTML parsing for web scraping
`requests`	HTTP client for fetching course pages
`ragas`	Automated RAG evaluation (LLM-as-judge)
`chainlit`	Web chat UI for the application
`python-dotenv`	Load environment variables from `.env`
`openpyxl`	Read/write Excel files (`qa_dataset.xlsx`)
`pandas`	Data manipulation for evaluation results
`matplotlib`	Charts and visualizations
`numpy`	Numerical operations
`pydantic`	Input/output validation in guardrails
`rank-bm25`	BM25 keyword-based retrieval for hybrid search

---

4. Install and Configure Ollama

4.1 Install Ollama

Download and install from ollama.com/download.

4.2 Start the Ollama Server

Open a **separate terminal** and keep it running:

```
ollama serve
```

By default Ollama listens on `http://localhost:11434`.

4.3 Pull Required Models

Three models are required. Pull each one (this downloads the model weights):

```
# Main LLM for answer generation (~2 GB)
ollama pull llama3.2

# Embedding model for vector search (~274 MB)
ollama pull nomic-embed-text

# Content safety classifier for guardrails (~2.5 GB)
ollama pull llama-guard3
```

4.4 Verify Models Are Available

```
ollama list
```

You should see `llama3.2`, `nomic-embed-text`, and `llama-guard3` in the output.

---

5. Configure Environment Variables

Copy the example file and edit if needed:

```
# Windows
copy .env.example .env

# macOS / Linux
cp .env.example .env
```

`.env` contents

```
# Ollama server URL (change only if Ollama runs on a different host/port)
OLLAMA_BASE_URL=http://localhost:11434
```

If Ollama is running locally on the default port, the `.env` file works as-is. If you run Ollama on a remote machine or custom port, update `OLLAMA_BASE_URL` accordingly.