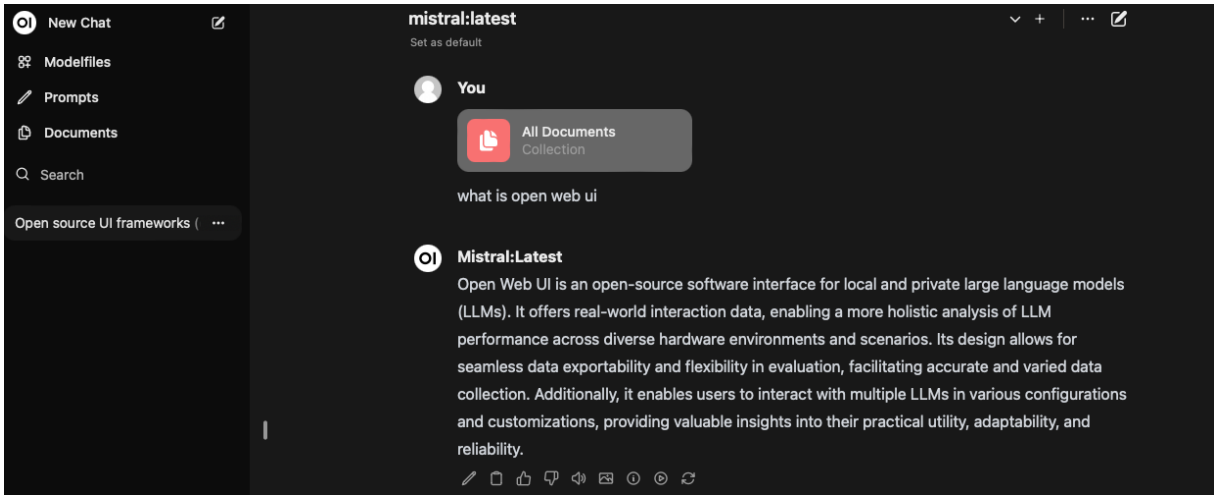
Veolia AI backbone as understood from 17/01/2025 meeting

1. Components:

UI: it seems to be based on OpenWebUI (open-webui/open-webui: User-friendly AI Interface (Supports Ollama, OpenAI API, ...))



Front Features:

a) Chat history (in different chat)
b) Attachments: any kind of doc parsed (pdf / ppt / excel / word etc)
c) Multimodal (we don't need): can generate images and text. OpenWebUI supports it (OpenAI's DALL-E or ConfyUI API).
d) Ability to choose the underlying LLM model (as for OpenWebUi).

Back architecture and features:

The backend is a Langchain server with the following features activated:

a) LangGraph
LangGraph

LangGraph is mostly a tool to organize workjflow routes on a Python Backend. Learn the basics. However, it is a very limited backend tool, and will probably create technical debt.

This could also be replaced by a more robust production worfklow tool for Python such as Apache Airflow.

Details of the LLM Airflow integration:
Copy of Building and deploying LLM applications with Apache Airflow | Airflow Summit 2023

Or a .NET backend as well.

b) The documents parsers.

The documents parser seem super accurate, I suspect they connect to the best open source tools for this which are:

LlamaIndex - Build Knowledge Assistants over your Enterprise Data

Quivr - Open source chat-powered second brains (a French YC startup that built the best PDF parser for RAG injection, mostly their main contribution).

c) API

They use Langchain tools and functions: Tool calling agent | 🦜🔗 LangChain

Interacting with APIs | 🦜🔗 LangChain

d) We didn't talk about production monitoring, but since they are fully compliant with the Langchain ecosystem recommendations I suspect they use LangSmith for produciton logs moniitoring, including past conversations.

e) The Vector DB: Being on serverless AWS cloud mainly, probably the default OpenSearch VectorDB. Note that they developed internally the possibility to switch the VectorDB manually, which was not natively possible in OpenWebUI.