✅ Lab: Create a GenAI Architecture Blueprint + Executive Pitch

---

🧩 Objective

By the end of this lab, you'll:

- Design a complete architecture blueprint for a real GenAI use case.
- Create a clear execution plan covering timeline, roles, and risks.
- Deliver a mock executive pitch focused on value, ROI, and strategic alignment.

🧠 Step 1: Choose Your Use Case

Pick a **realistic business use case** that leverages GenAI + knowledge graphs.

Examples:

- Internal legal document summarization tool
- AI-powered customer service assistant
- Research copilot for healthcare or biotech
- Compliance monitoring assistant
- Intelligent contract analyzer

🟨 **Tip**: Choose a domain you're familiar with or one relevant to your organization.

🗺️ Step 2: Define the Problem & Context

Write a 3–5 sentence **problem statement**. Identify:

- The **business challenge**
- Who the **end users** are
- What data or knowledge is needed
- Why traditional solutions fall short

🟨 Example:

> "Legal teams spend over 20 hours per week reviewing contracts for clause risks. Manual review is time-consuming and error-prone. We aim to build a GenAI-powered contract analyzer that highlights key risk clauses using ontology-based context and retrieval-augmented generation."

🧱 Step 3: Draft the Architecture Blueprint

Use a diagramming tool (draw.io, Lucidchart, Whimsical, etc.) to map:

🔷 Core Components:

Component: LLM
Example: OpenAI GPT-4 / Anthropic Claude

Component: Retrieval
Example: FAISS / Pinecone + RDF Knowledge Graph

Component: Agents
Example: CrewAI or LangGraph (retriever, planner, summarizer)

Component: Data
Example: CSVs, PDFs, web pages, databases

Component: Ontology
Example: Protégé-based OWL schema

Component: Frontend
Example: Streamlit / Web app

Component: Backend
Example: FastAPI / Flask REST services

Component: Deployment
Example: AWS Fargate, Azure Container Apps, or K8s

Include:

- Flow of data and prompts
- API endpoints
- Where vector search + SPARQL fit
- How context is passed to the LLM
- Agent orchestration (single/multi-agent)

📌 **Label each component clearly**.

🗓️ Step 4: Build an Execution Plan

Create a **1-page table or timeline** covering:

Phase 1 Milestone: Define ontology + ingest data
Team Involved: Ontologist + Data Engineer
Time Estimate: 1 week

Phase 2 Milestone: Set up vector + graph retrieval
Team Involved: ML Engineer
Time Estimate: 1 week

Phase 3 Milestone: Build multi-agent logic
Team Involved: Architect + ML team
Time Estimate: 2 weeks

Phase 4 Milestone: Deploy & monitor
DevOps + MLOps
Time Estimate: 1 week

Also include:

- Risks & mitigation (e.g., hallucination → fallback logic)
- Evaluation plan (success criteria)

🎤 Step 5: Prepare the Executive Pitch

Craft a short 5-slide deck or 3-minute spoken pitch using the following structure:

📊 Pitch Slide Framework:

1. **Problem**: What's broken today?
2. **Vision**: What does the solution look like?
3. **Solution Architecture**: Simplified diagram and how it works
4. **Value & ROI**: Time saved, accuracy improved, cost reduced
5. **Timeline & Next Steps**

🟨 Use visuals, analogies, and metrics—not technical jargon.

🧪 Step 6: Present or Record

Do one of the following:

- Record a video walkthrough of your pitch (Loom, OBS, etc.)
- Present live to peers/mentors/faculty
- Share your blueprint and execution plan with an expert for feedback

---

🎯 Success Criteria

You've completed the lab when you can:

- Clearly communicate the **business impact** of your system
- Walk through the **technical architecture** with confidence
- Show a realistic **execution plan**
- Defend trade-offs, risks, and ROI with clarity
