Team Details

- ➔ Project Name - Aegis Workspace
- ➔ Team name - TechoRiser
- ➔ Team leader name - Vijaya Chandra Reddy Poluri
- ➔ Problem Statement - The Local AI Performance Bottleneck

Brief about the idea

- Aegis Workspace is an air-gapped, on-device Retrieval-Augmented Generation (RAG) productivity agent made for businesses.
- It safely indexes local documents like PDFs, Word files, and emails. It runs advanced Large Language Models (LLMs) entirely offline. By using only AMD Ryzen AI hardware, it ensures complete corporate data privacy while providing cloud-level AI productivity to workers.

Opportunities

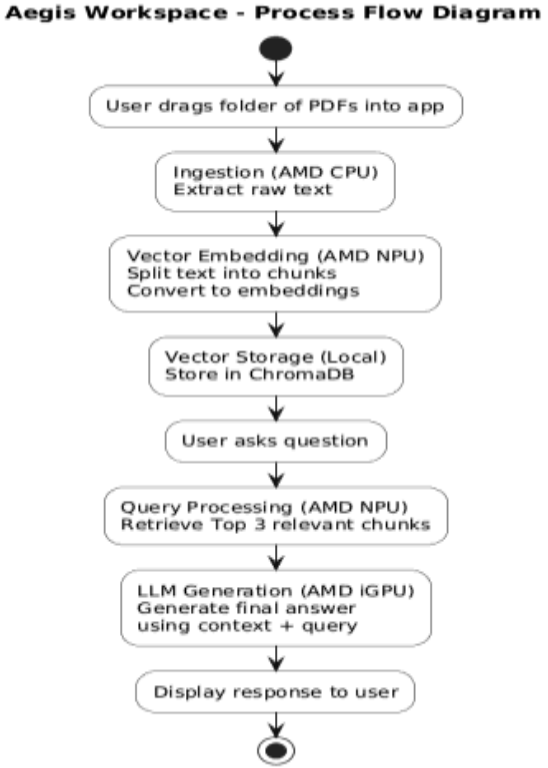
- ➔ How different is it from any of the other existing ideas?
- ➔ How will it be able to solve the problem?
- ➔ USP of the proposed solution

- **Opportunities:** Aegis Workspace provides secure, offline AI-powered document intelligence for the legal, healthcare, and corporate R&D sectors without risking sensitive data exposure.
- **Differentiation:** Unlike cloud RAG APIs, which can lead to data leaks and ongoing expenses, or CPU-based local AI, which can be slow and inefficient, Aegis Workspace uses AMD heterogeneous computing for secure and optimized performance.
- **Problem Solving Approach:** By using ONNX Runtime GenAI and supporting 16K token contexts in AMD Ryzen AI 1.7, it enables users to securely analyze and “chat” with large enterprise documents completely offline.
- **USP:** This air-gapped enterprise RAG solution offers cloud-level AI performance with no risk of data exfiltration and no recurring cloud costs, powered exclusively by AMD silicon.

List of features offered by the solution

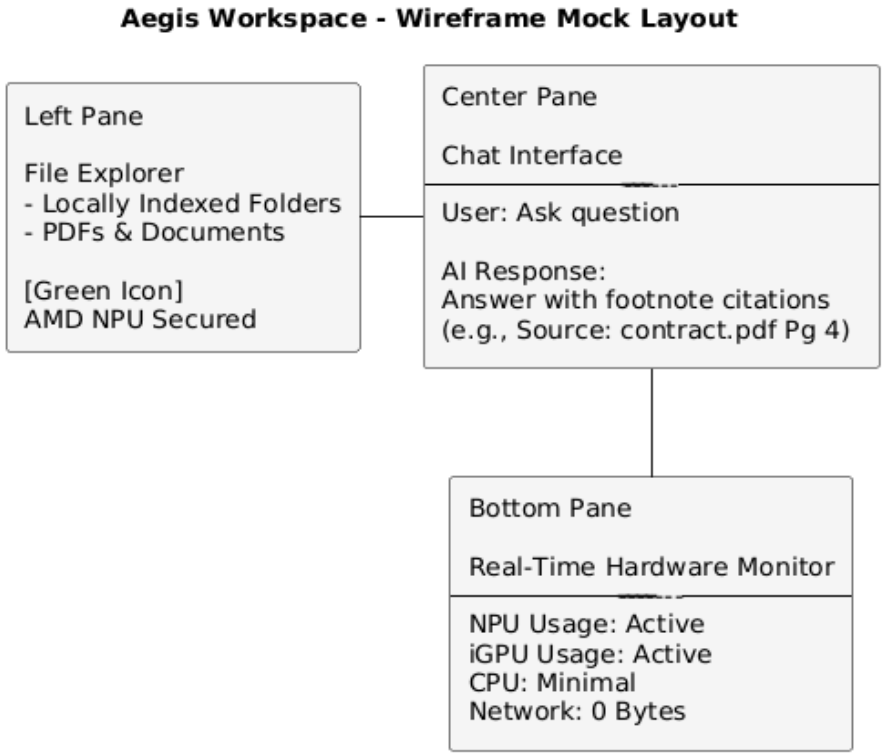
- **Air-Gapped Document Chat:** Chat with sensitive PDFs and text files completely offline.
- **Heterogeneous Load Balancing:** Background document indexing runs silently on the low-power NPU, which helps preserve laptop battery life.
- **Extended Context Analysis:** Uses 16K context support for deep document reading.
- **Hardware-Accelerated Token Generation:** Uses the DirectML Execution Provider on the iGPU for fast text generation.
- **Local Semantic Search:** Find exact paragraphs instantly across thousands of offline documents.

Process flow diagram



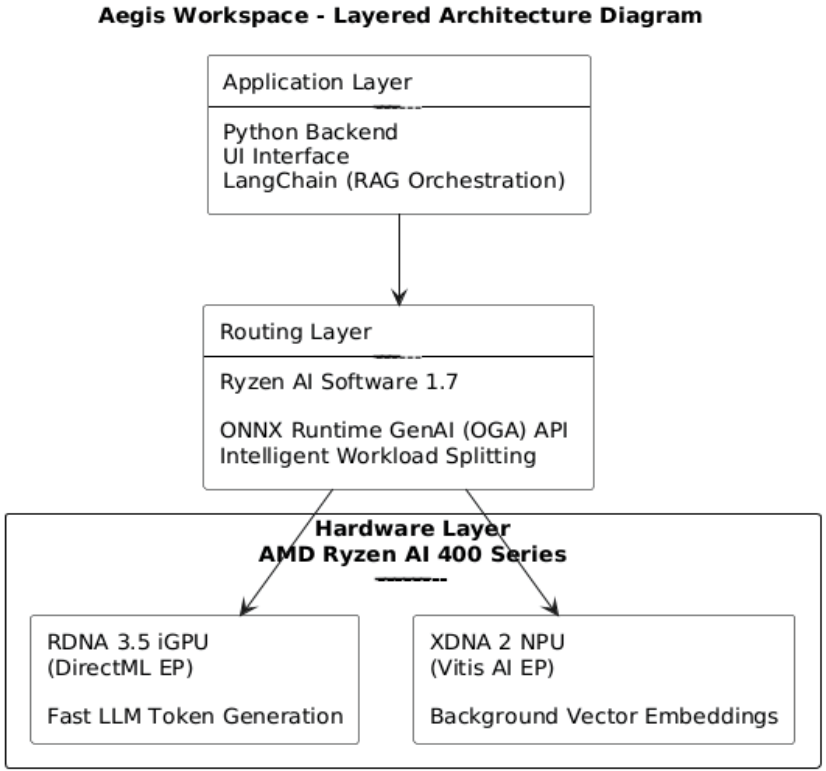
- **Ingestion (AMD CPU):** The user drags a folder of PDFs into the application. The CPU extracts raw text from the documents.
- **Vector Embedding (AMD NPU):** The text is split into smaller chunks. The NPU converts each chunk into vector embeddings using a lightweight model.
- **Vector Storage (Local Database):** All generated vectors are stored securely in a local ChromaDB database.
- **User Query (AMD NPU):** The user asks a question. The NPU retrieves the top 3 most relevant document chunks.
- **LLM Generation (AMD iGPU):** The query and retrieved context are passed to the local LLM. The iGPU generates the final response quickly and efficiently.
- **Final Output:** The system displays the answer to the user, completely offline and secure.

Wireframes/Mock diagrams of the proposed solution



- **Left Pane:** A file explorer displays locally indexed folders with a green "AMD NPU Secured" icon.
- **Center Pane:** A chat interface allows the user to query the data. Responses include footnote citations that link to the local files.
- **Bottom Pane:** A real-time hardware monitor shows that the NPU/iGPU are managing the workload while network usage stays at 0 bytes.

Architecture diagram of the proposed solution



- **Application Layer:** Python + UI + LangChain manage RAG orchestration.
- **Routing Layer:** Powered by Ryzen AI Software 1.7, where ONNX Runtime GenAI intelligently splits workloads.
- **Hardware Layer (AMD Ryzen AI 400 Series):** XDNA 2 NPU, vector embeddings RDNA 3.5 iGPU, LLM token generation
- **Outcome:** Optimized, secure, hybrid AI execution fully on-device.

Technologies to be used in the solution

- **Model Optimization:** AMD Vitis AI Quantizer, AMD Quark toolkit (for INT4/INT8 compression).
- **Runtime:** ONNX Runtime, ONNX Runtime GenAI (OGA) API.
- **Machine Learning:** PyTorch (trained/fine-tuned using ROCm 7.2).
- **RAG Framework:** LangChain, ChromaDB (Local Vector Store).
- **Models:** Quantized Llama-3.2 1B or DeepSeek-R1-Distill.

Usage of AMD Products/Solutions

- **Deployment Target:** AMD Ryzen AI 400 Series or Ryzen AI Max+ (Strix Halo) platforms.
- **Software Stack:** We rely heavily on the newly released Ryzen AI Software 1.7. This version supports a 16K LLM context length and offers OGA hybrid execution modes.
- **Training:** We suggest using AMD ROCm 7.2 on AMD Instinct MI300X or Radeon RX 7900 XTX GPUs for offline fine-tuning of base models.

Estimated implementation cost

- **Software/Licensing:** $0. The entire stack, including ROCm, Ryzen AI SDK, ONNX, and local models, is open-source.
- **Hardware (Development):** Around $1,500 to $2,500 for an AMD Ryzen AI 400 Series laptop.
- **Cloud API Costs:** $0 permanently. This ensures infinite scalability for the enterprise without subscription fees.

Prototype Assets

- ➔ GitHub Public Repository Link

GitHub Public Repository Link: https://github.com/p-vijaya-chandra/Aegis-RyzenAI-Workspace


AMD
Slingshot
HUMAN IMAGINATION
BUILT WITH AI
Powered by H2S
Thank you!