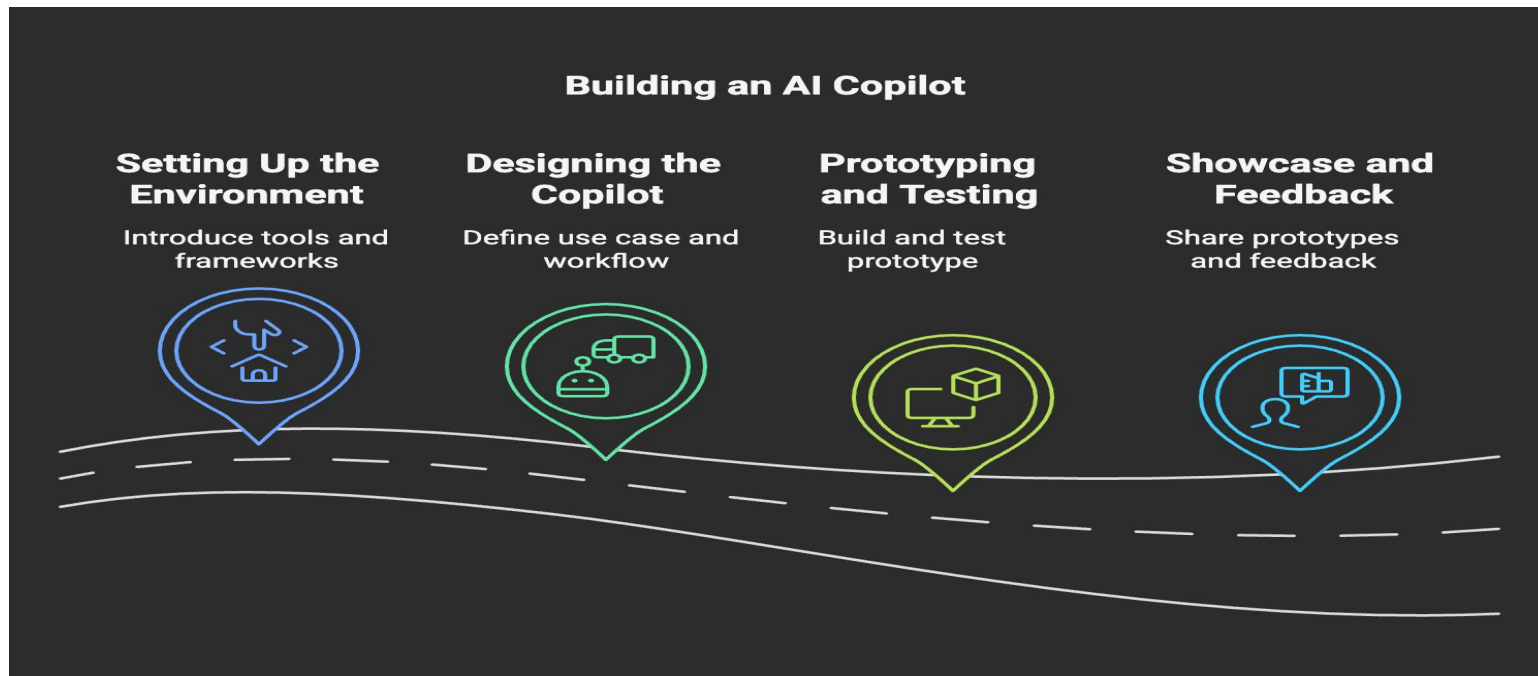# Hands-On Session:

Building an AI Copilot

Agenda

Tools and Libraries

Agents Framework - llama - index (other frameworks - langchain, autogen, crewai .etc)

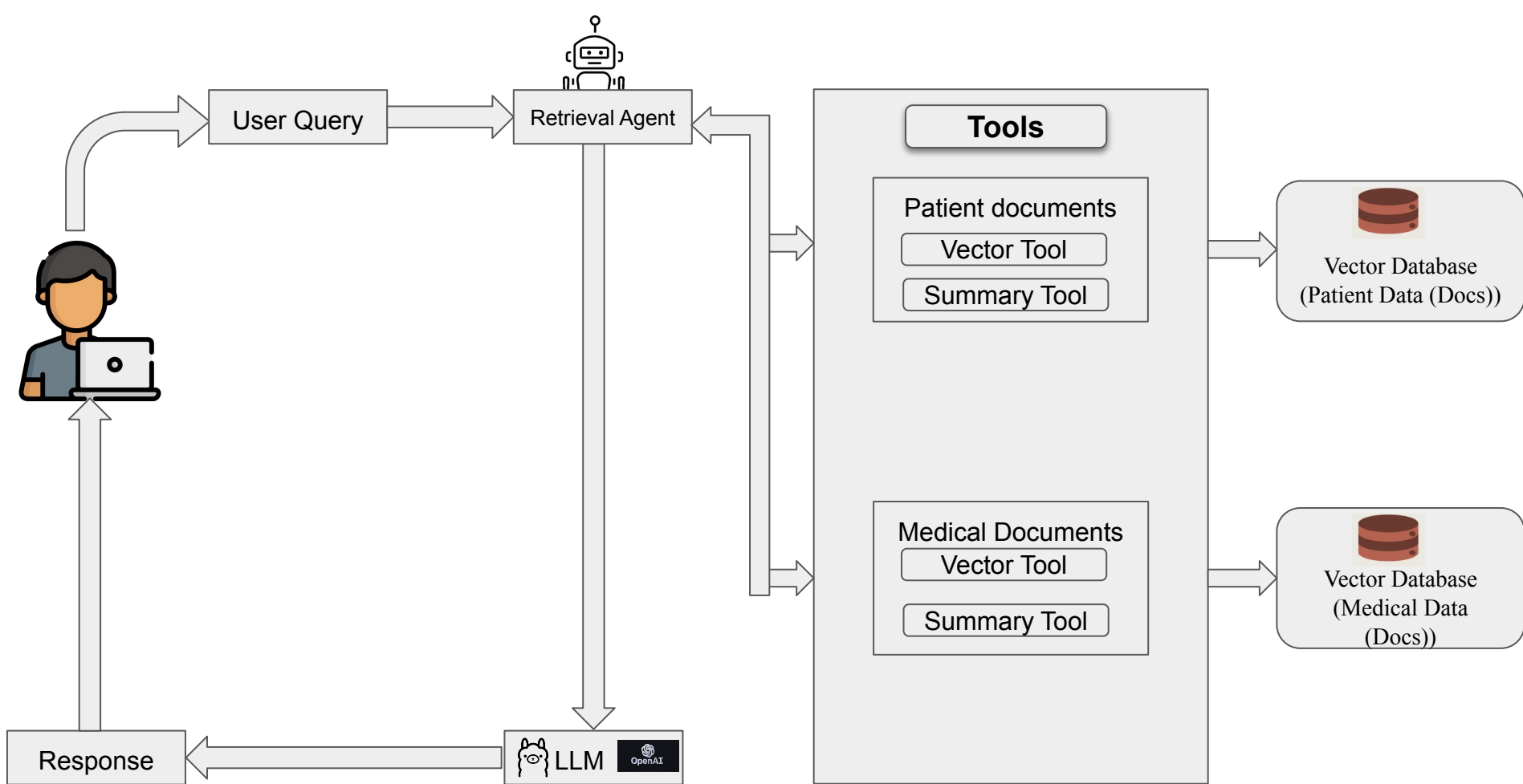**LLM Model** - AzureOpenai

Document parsing - docling

UI - streamlit

Setting Up the Environment

Code path - **https://github.com/cazelabs/workshop.git**

Create new environment **-** conda create -n aiDevCon python=3.11

Activate environment - conda activate aiDevCon

Install Package - pip install -r requirements.txt

Building Multi-Document Agents

1. Build a document agent for each document

2. Define the top-level parent agent with an object index.

1. Build a document agent for each document

- Define both a vector index (for semantic search) and summary index (for summarization) for each document.

- The two query engines are then converted into tools that are passed to an OpenAI function calling agent.

- This document agent can dynamically choose to perform semantic search or summarization within a given document.

2. Define the top-level parent agent with an object index.

- We build a top-level agent that can orchestrate across the different document agents to answer any user query.

- This agent takes in all document agents as tools. This specific agent performs tool retrieval before tool use

Execute

Command - **streamlit run app.py**