Vizuara AI Agents Bootcamp Day 4

An overview of the types of agentic frameworks

VIZUARA AI
JUN 26, 2025

In this post, we'll break down the key takeaways from **Day 4 of the Vizuara AI Age** **Bootcamp**, focusing on "**Agentic Frameworks**" and how the world is organizing the into Code, Low-code, and No-code tools.

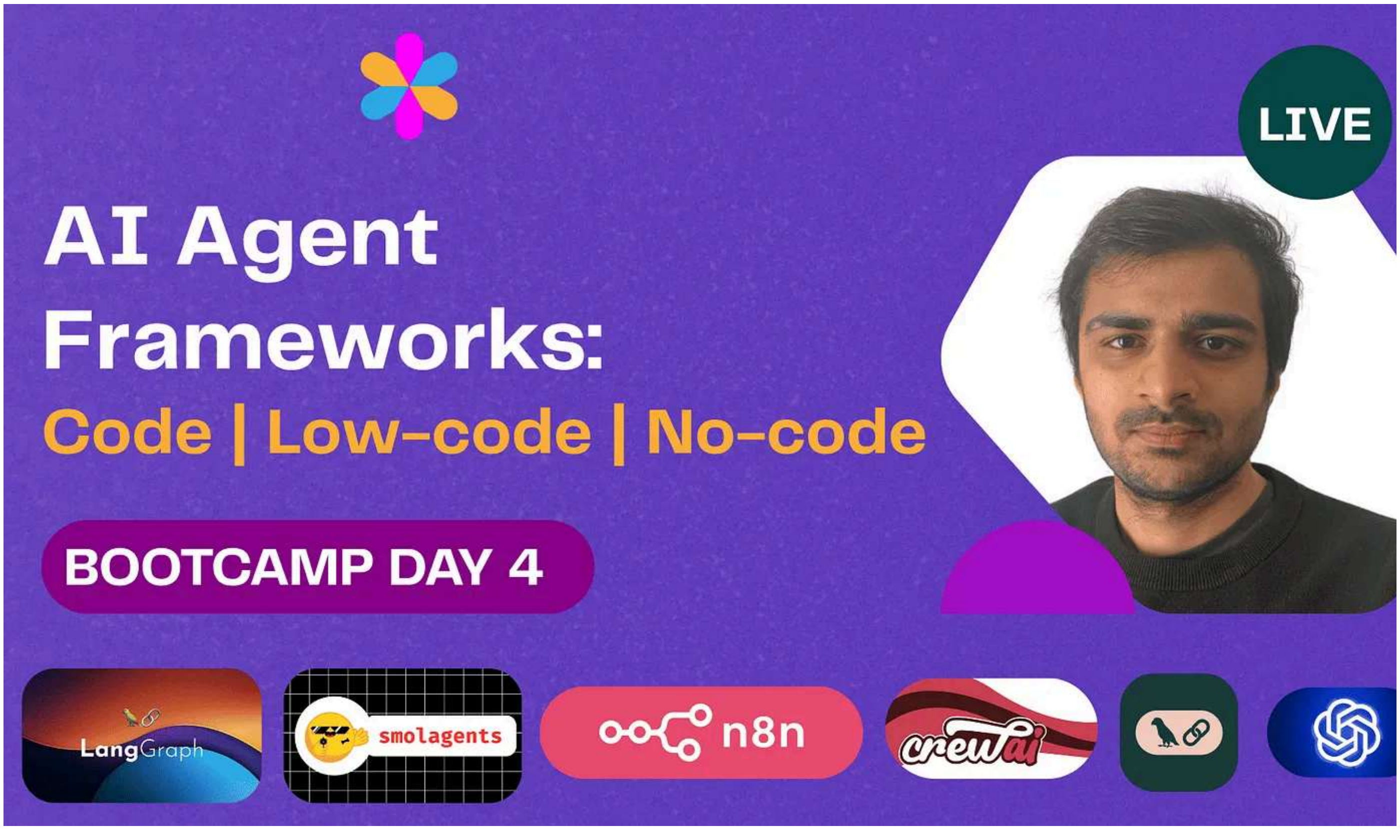
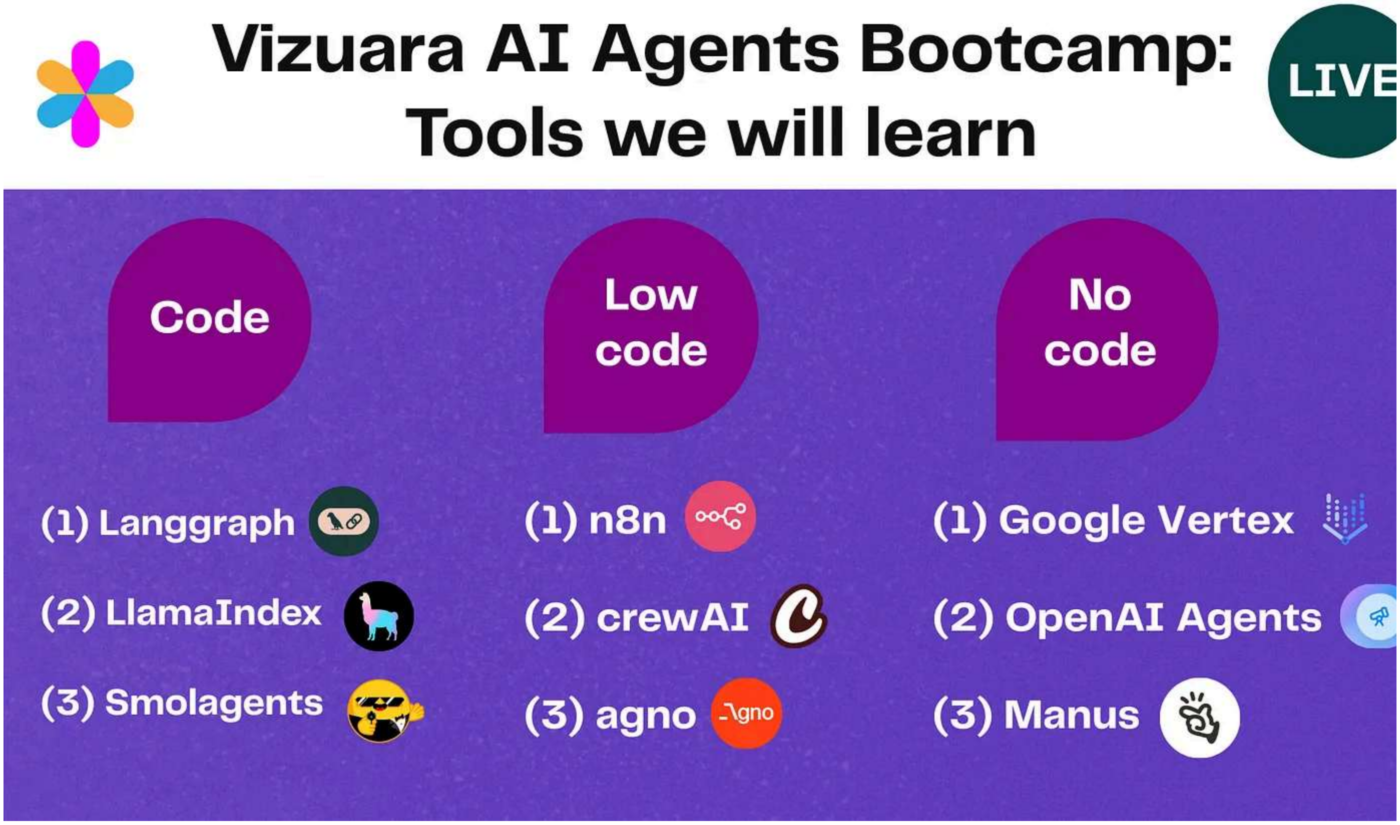


Table of contents

(1) Overview figure: Code, Low-Code, and No-Code Ager



Vizuara AI Agents Bootcamp: Tools we will learn

LIVE

Code

(1) Langgraph

(2) LlamaIndex

(3) Smolagents

Low code

(1) n8n

(2) crewAI

(3) agno

No code

(1) Google Vertex

(2) OpenAI Agents

(3) Manus

(2) Code Frameworks

- **LangChain:** The OG of agent frameworks – an open-source library that makes easier to connect LLMs with data and APIs in a chain or agent loop. LangChai provides modular components (prompts, memory, tool interfaces, etc.) to build everything from chatbots to autonomous agents.

Launched in late 2022, it **rose in popularity**, becoming *the fastest-growing open-source project on GitHub by mid-2023*. Think of LangChain as the general-purpos toolkit for LLM apps and agents, used by thousands of developers.

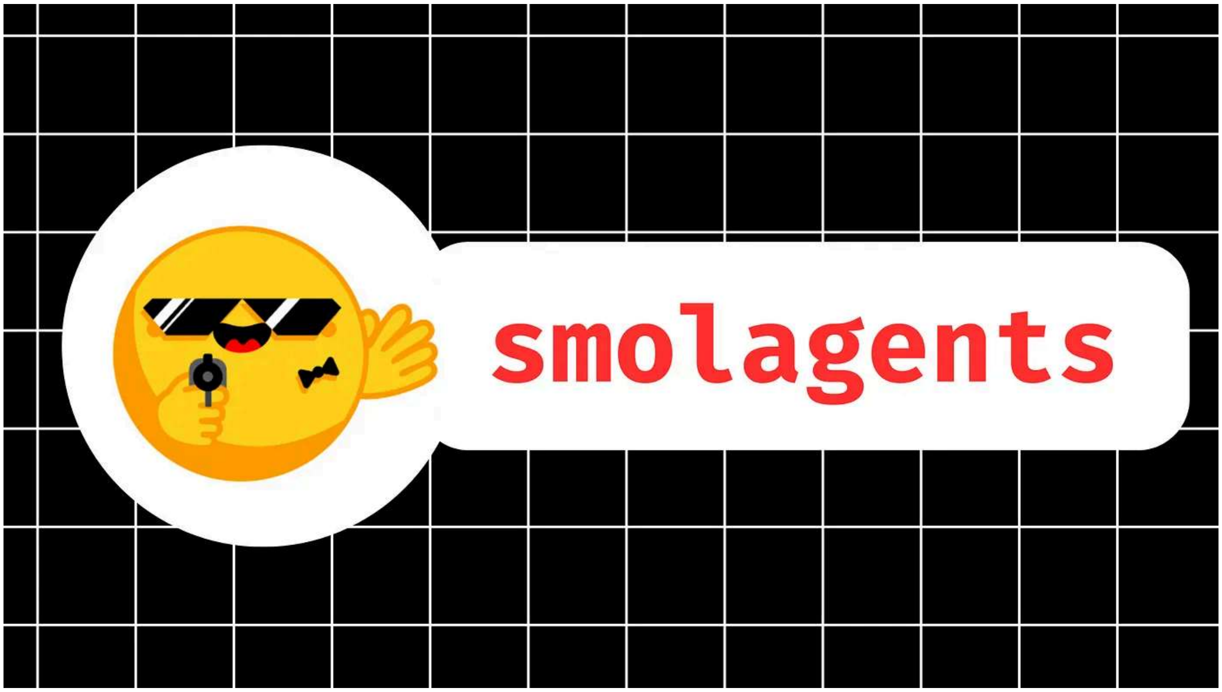
- **LangGraph:** A newer framework from the LangChain team, LangGraph takes a *graph-based* approach to agent orchestration. Instead of a linear chain of calls, y define a graph of possible steps and decision points. This allows for **complex workflows with conditional branching** – ideal when an agent might have to handle many possible sub-tasks.

LangGraph is considered more low-level and **controllable** than LangChain's bu in agents, giving developers the ability to chart out exactly how an AI should navigate a task (like a map with multiple routes). It's great when you need that extra control over an agent's decision flow.

- **LlamaIndex:** Formerly known as GPT Index, LlamaIndex is *the go-to framework LLMs + your data*. It specializes in **retrieval-augmented generation** (**RAG**), letti you build agents and assistants that can ingest and query large external dataset documents.

For example, with LlamaIndex you can set up an agent that indexes a knowledg base and intelligently fetches facts to answer questions. It provides pre-built agents (like a question-answering agent) and tools to integrate with databases, vector stores, and APIs. If your agent needs to *know about* specific data (beyond what the base LLM knows), LlamaIndex is a powerful ally.

- **SmolAgents:** An aptly named framework by Hugging Face, **smolagents** aims to keep things simple and "small." It's a minimalist library that **enables powerful agents in just a few lines of code**. SmolAgents embraces a particular philosoph instead of having the LLM output actions in JSON or text, it lets the agent *writ and execute Python code* directly as its way to take actions.



This “code-as-actions” approach can be highly efficient – smolagents reports u to *30% fewer steps and API calls* by having the AI generate code to solve sub-task rather than verbose instructions. For instance, an agent that needs to do some math or web scraping can write a short Python snippet to do so, run it, and use result. SmolAgents is lightweight (the core logic is ~1000 lines of code) and **developer-friendly**, making it a good choice for quickly deploying an agent without too much overhead.

- **Autogen** (**Microsoft**)**:** Autogen is a framework from Microsoft that pushes the envelope on *multi-agent communication*. It models an AI application as a **conversation between multiple specialized agents**. You might have a “User Assistant” agent interacting with a “Dev Assistant” agent, for example, passing tasks back and forth. Autogen provides high-level patterns like group chats, hierarchical chats (one agent managing others), and even the ability to inject human feedback in the loop.

A standout feature is its support for **code execution** – Autogen agents can automatically run generated code in various environments (local shell, Docker, Jupyter) as part of their reasoning. *Use case example:* Researchers have used Autogen to have one agent generate code solutions which another agent critiqu and tests – a kind of AI pair-programming. This yields a multi-step reasoning process where the agents collectively hone in on a correct answer or working program. Autogen shines in scenarios that benefit from multiple perspectives o skills encapsulated in different agents.

(3) Low-Code Tools

- **LangFlow:** LangFlow is essentially *LangChain with a visual face*. It provides a we based **graphical UI to build LangChain flows** by dragging and dropping components. Instead of writing code to connect a prompt to an LLM to a tool, y draw it out as a flowchart.

This lowers the entry barrier for those who understand the concepts but aren't comfortable in code. You can configure prompts, select models, add memory, a see the "thought" process of agents in an interactive way. LangFlow doesn't introduce new capabilities beyond what LangChain offers – it's an abstraction convenience and rapid prototyping. For a developer, it's a quick way to experim with chain structures; for a non-dev, it's a way to *use LangChain without coding*.

- **CrewAI:** Also written as **crewAI**, this framework takes inspiration from how humans work in teams. It lets you set up multiple agents with different roles (a "crew") and define how they collaborate on a task. The interface is more high-l configuration than coding. For example, you might create a *Researcher agent* an *Writer agent*, where the Researcher can use web search tools and the Writer focuses on composing text. CrewAI provides abstractions for tasks, planning, a even memory types (short-term, long-term memory for agents) – but it wraps th in a user-friendly way.

It’s often described as *“developer-friendly”* because you don’t have to reinvent common patterns; much is handled under the hood. If LangChain/LangGraph a like raw toolkits, CrewAI is like a semi-finished assembly where you just plug i specifics of your use case. It’s particularly good for workflows that naturally br into sub-tasks – the framework helps coordinate the sub-agents for you.

- **n8n** (**AI Integrations**): **n8n** is a popular general automation tool (think open-source Zapier) that has embraced LLM agents. It offers a *no-code/low-code work editor* where you can drag nodes representing actions or services. Recently, n8n introduced **AI agent nodes** that allow an LLM to operate within those workflo For instance, you can have a trigger like “New email arrives” -> then an “LLM Agent” node that analyzes the email’s content and decides an action -> then branches to different outcomes.

Under the hood, n8n's agent node can maintain conversation context and call tools (like HTTP requests, database queries) as directed by the LLM. The beaut in the visual orchestration: you design the *surrounding* workflow (say, when to invoke the agent, what to do with its output) using n8n's UI, mixing and match with 3rd-party integrations.

Use case example: A support team could set up n8n such that whenever a ticket i created, an AI agent automatically gathers relevant info (pulling customer data API, knowledge base articles via search), then recommends a response. The agent's suggestions could be reviewed by a human, then an automated follow-u sent. All of this can be configured with n8n's nodes – no need to write a custom server for it. This way, **n8n acts as the conductor and the LLM agent is a talen soloist** playing within the orchestra of enterprise tools.

- **Agno: Agno** is an open-source framework that prides itself on being "**full-stacl for multi-agent systems**. It falls in the low-code realm because it offers a lot of built-in functionality (and even a playground UI) so you can create sophisticate agents with minimal glue code. Agno focuses on performance and clean design its creators boast about *blazing speed* (agents that initialize in microseconds, minimal memory overhead) and a **model-agnostic** approach supporting 20+ mo providers.

It’s a promising framework for those who want the power of multi-agent, multi modal systems without the pain of piecing together many libraries. (The trade- is Agno is evolving fast, so one has to keep up with its updates and maybe deal with less community support than LangChain.)

(4) No-Code Platforms

- **Manus:** Among the new wave of autonomous agent startups, **Manus AI** stands for its bold claim of being a *“general-purpose AI agent”*. Manus is like having an employee who can learn and do a wide variety of work tasks. It operates throug chat interface where you give it high-level goals, and Manus will **plan out and execute the necessary steps, across potentially hours or days, using a suite of tools**.
  Under the hood, Manus has its own “cognition” engine and can call APIs, run browsing, write to files, etc., all autonomously. The user doesn’t script those ste – you just say, for example, “Analyze these five research papers and draft a summary comparing their findings,” and Manus takes it from there. It might decide to search for additional references, create a to-do list, read each paper (using an LLM to extract key points), and then compile a report.

Manus can **operate continuously without supervision**, thanks to an internal feedback loop and memory to track progress. This is a step beyond typical ChatGPT usage where the AI responds once and stops. Manus will keep going until the task is done (or you intervene). It also **actively uses tools** – if it needs t calculate, it will execute code; if it needs information, it will perform web queri etc..

(5) Recommendations for Industry Professionals

Code-level agent frameworks :

(1) LangGraph – A graph-based orchestration platform built on top of LangChain (https://www.langchain.com/langgraph)

(2) LangChain – A popular framework for building LLM-powered agents and integrating tools

(https://www.langchain.com/)

(3) LlamaIndex - used for ingestion, indexing, and retrieval over data (https://www.llamaindex.ai/)

(4) SmolAgents – minimalist agents library for lightweight agenting (https://huggingface.co/docs/smolagents/en/index)

Low-code platforms:

(1) CrewAI – Collaboration-first multi-agent orchestration platform (https://www.crewai.com/)

(2) Langflow – Visual workflow designer for agents

(https://www.langflow.org/

(3) n8n – Open-source low-code workflow automation tool, with AI integrations

(https://n8n.io/)

No-code agent builders

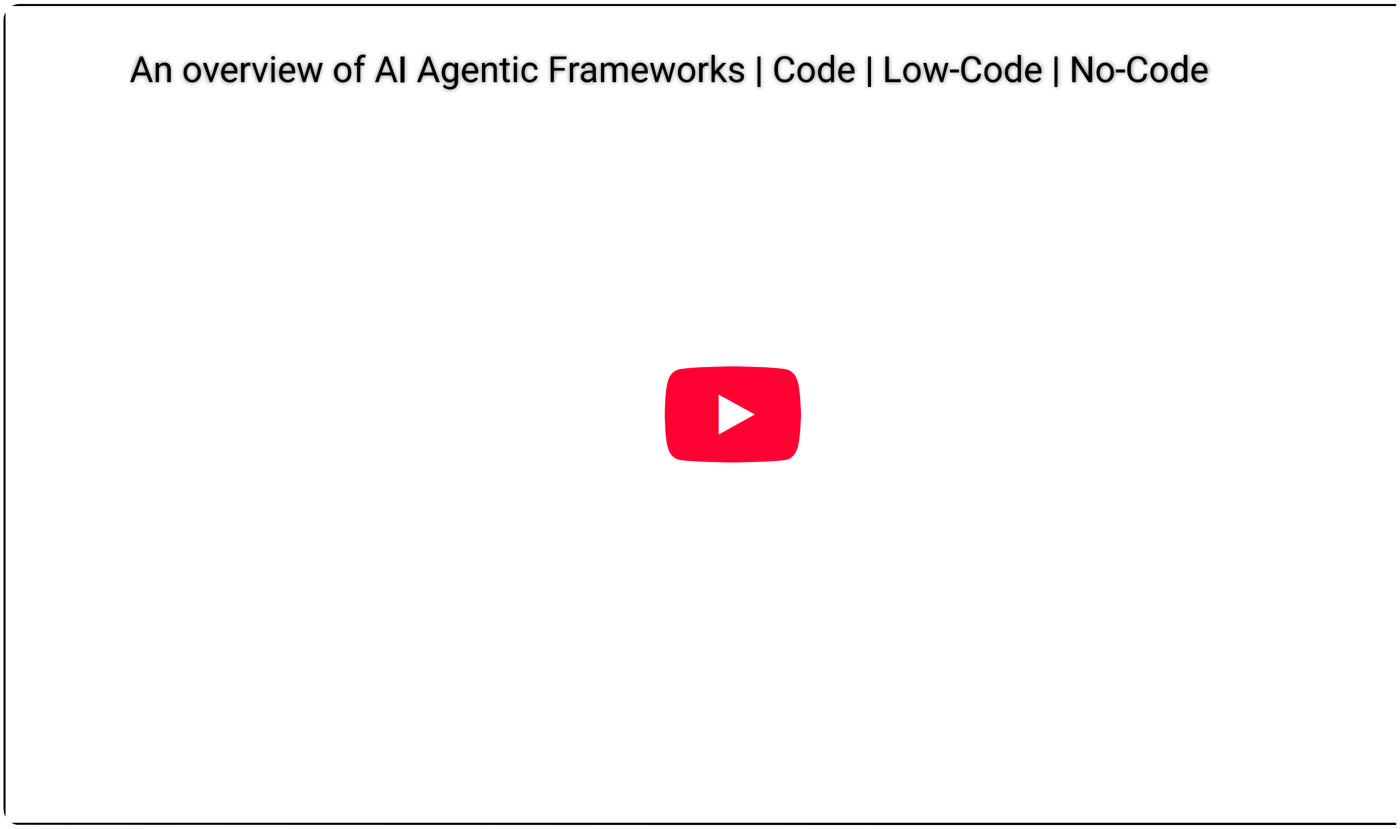
(1) Manus – Full-stack autonomous agent from China's Monica startup

(https://manus.im/)

(2) OpenAI Deep Research

(3) Gemini Deep Research

(6) Wrapping Up Day 4



An overview of AI Agentic Frameworks | Code | Low-Code | No-Code

A huge thank you to all the attendees of Day 4! 🎉

As we wrap up Day 4, it’s clear that **agentic frameworks** are empowering us to buil
AI that’s far more **active and autonomous** than ever before. The landscape is evolvi
rapidly – what’s cutting-edge today might be standard practice tomorrow.

6 Likes

Discussion about this post

Comments Restacks

Write a comment...