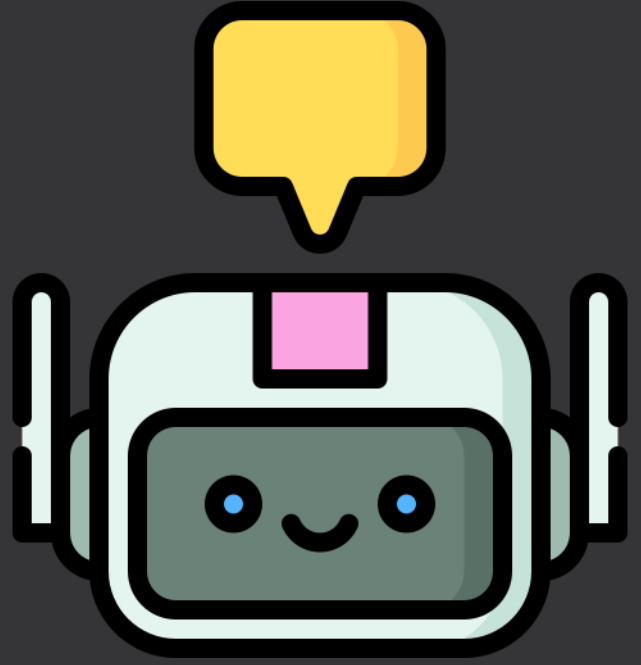


Agentic AI

Lecture 23 – HCCDA-AI

R.A: Imran Nawar

R.A: Assad Ullah Khan

30 Aug 2025

How You Can Engage with AI

1. Learn AI/LLMs and Build Them:

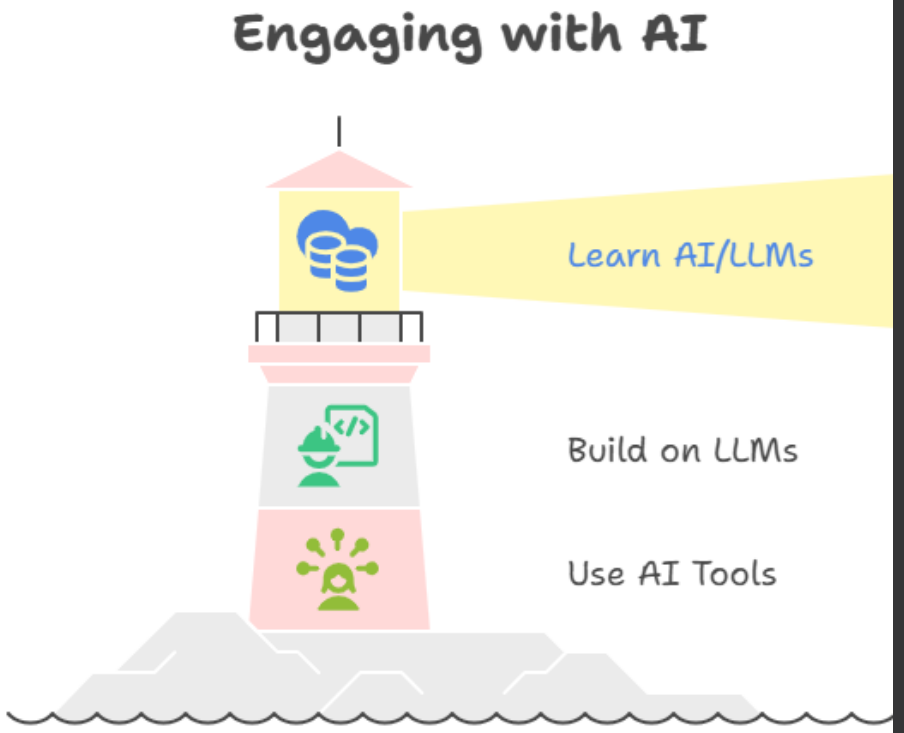
- Research and develop AI models (e.g., training you own language or vision models).
- Requires deep ML expertise and significant compute; typically a research track (often MSc/PhD or equivalent experience).
- Reaching this level requires deep study and expertise.

2. Build on Top of Existing LLMs:

- Use APIs/SDKs (OpenAI, Google Gemini, Mistral, Meta Llama) to create apps, agents, and workflows.
- Create practical applications such as customer support chatbots, resume-screening tools, or study-assistant apps (instead of building a model from scratch).

3. Use AI Tools Without Coding:

- Even if you can not build bots or work with APIs.
- Leverage no-code/low-code tools (ChatGPT, Claude, Gemini, Cursor, Notion AI, Zapier).
- Focus on productivity, prompt design, and integrating AI into daily work.



Generative AI vs Agentic AI

Generative AI (GenAI):

- **Reactive**: Responds when prompted.
- **Prompt → Output** (text, image, code, audio).
- Stops at generation unless the user continues the process.
- Useful for content creation, prototyping ideas, and boosting daily productivity.

Agentic AI:

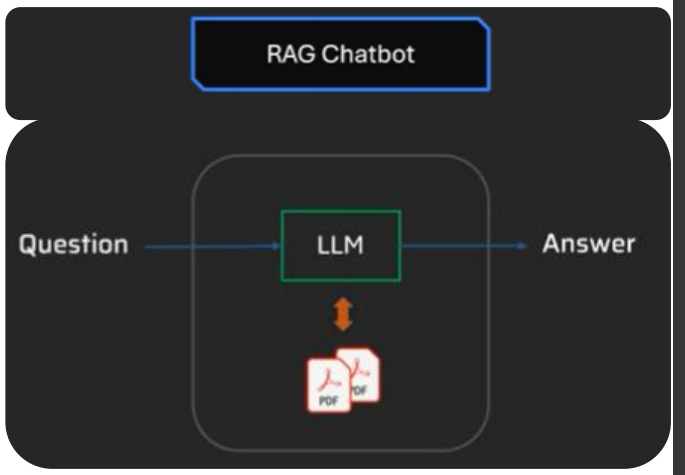
- **Proactive**: Takes actions to pursue a goal.
- **Loop**: Perceive → Decide → Execute → Learn.
- Handles multi-step processes and can coordinate tools or services.
- Goes beyond content creation → works like an independent problem-solver.

- **Shared Foundation:**
  - **LLMs** often serve as the reasoning/decision engine for chatbots and agentic systems.
  - Other model families extend capabilities (e.g., **diffusion models** for images, speech models for audio).
  - Real-world systems often **combine LLMs + vision/audio models + tool use** for richer functionality.

Evolving Chatbots: RAG → Tool-Augmented → Agentic

HR Assistant:

- Answers simple policy questions:
  "How many vacation days do we get per year?"
  "What is the policy on sick leave?"
- Policy data is available in PDFs.
- A **RAG-based chatbot** can retrieve answers from these PDFs.

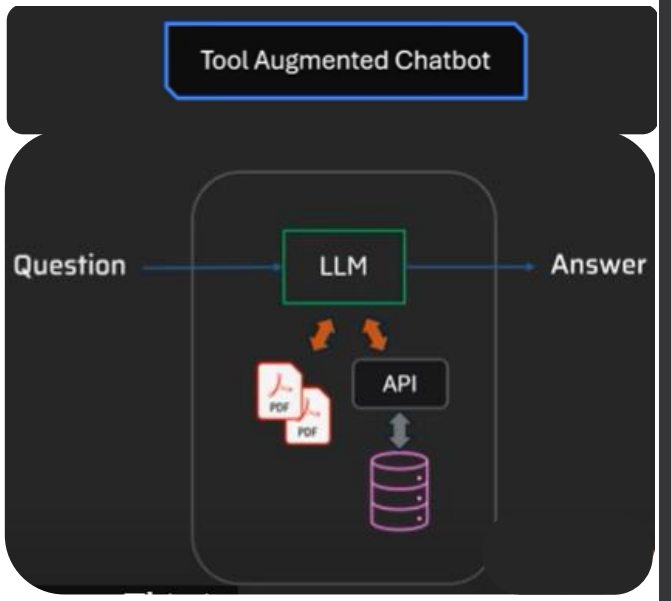


Is this an agentic AI? No

- Let's advance this chatbot a bit more:
  "How many leaves do I have left?"
- Still just retrieving/supporting answers.

Is this an agentic AI? No

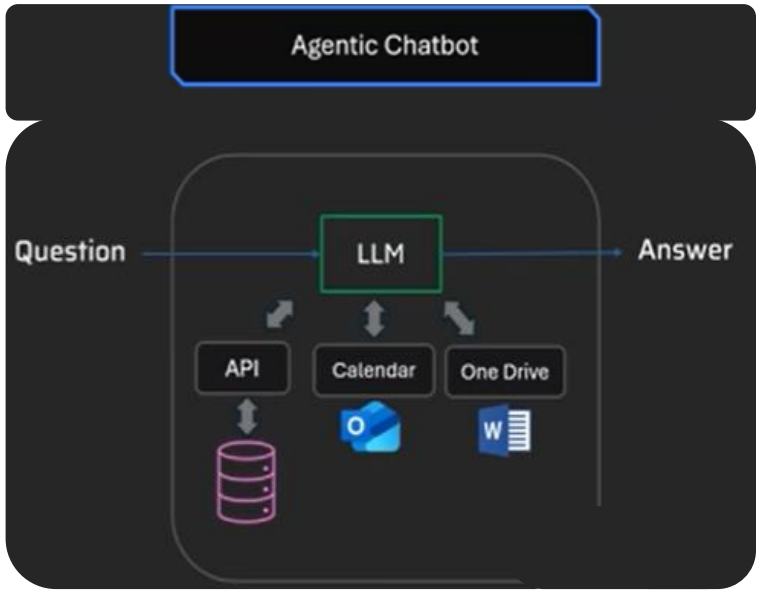
- **Tool-augmented chatbot** → combines an LLM with APIs (e.g., HR database) to fetch personalized info.

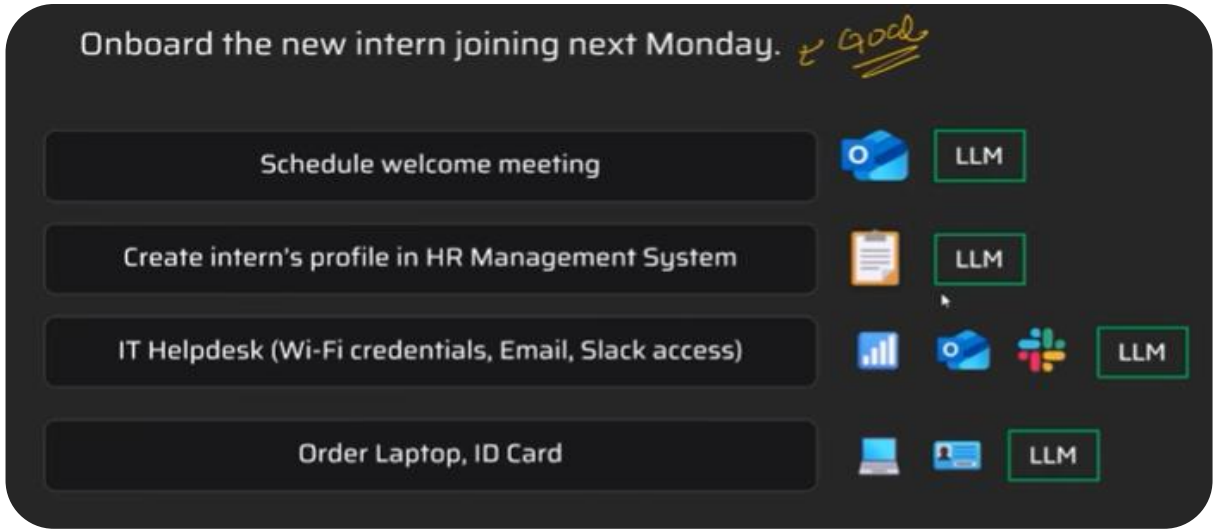


Evolving Chatbots: RAG → Tool-Augmented → Agentic

HR Assistant:

- Now we give it **a goal**, e.g.:
  "Prepare for Sarah's maternity leave."
- This requires:
  - Multi-step reasoning
  - Multi-turn planning
  - Taking actions with tools (not just instructions)
- **Example tasks:**
  - Onboard the new intern starting next Monday.
  - Update schedules and approvals automatically.

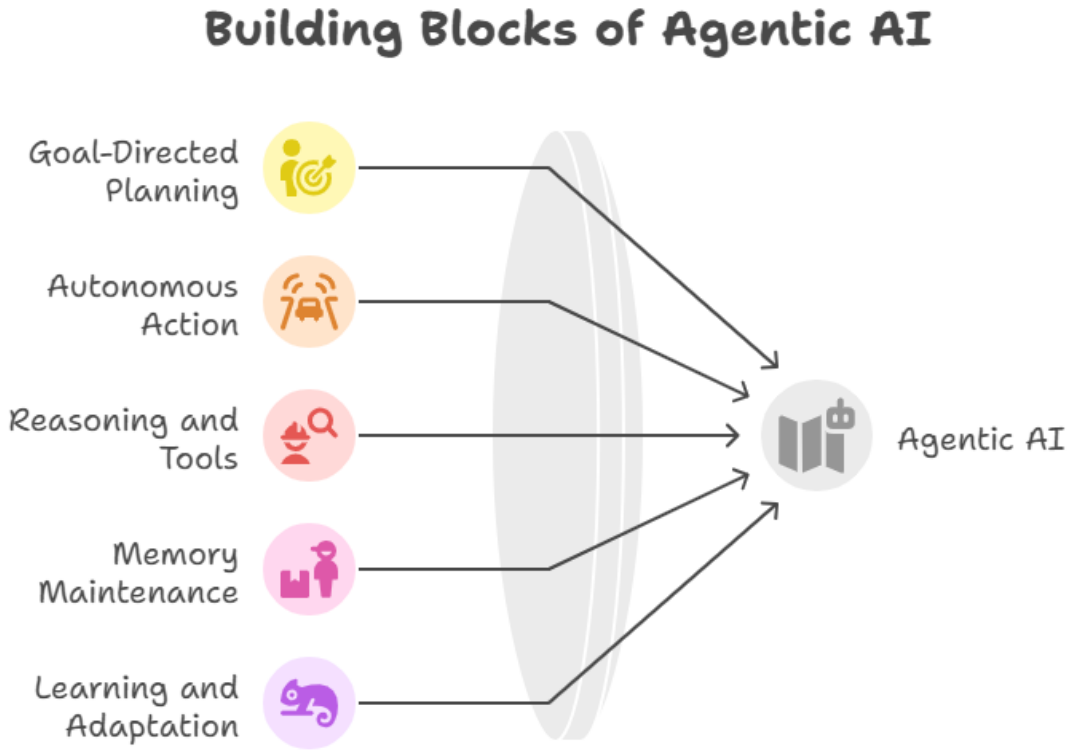




What Makes AI ‘Agentic’

An AI is agentic when it:

- Has goal-directed planning (breaks goals into steps).
- Acts autonomously (can initiate actions).
- Uses reasoning + tools (search, APIs, code execution).
- Maintains memory across interactions or sessions.
- Learns from outcomes and adjusts behavior.



Agentic AI combines planning, memory, and interaction into one loop.

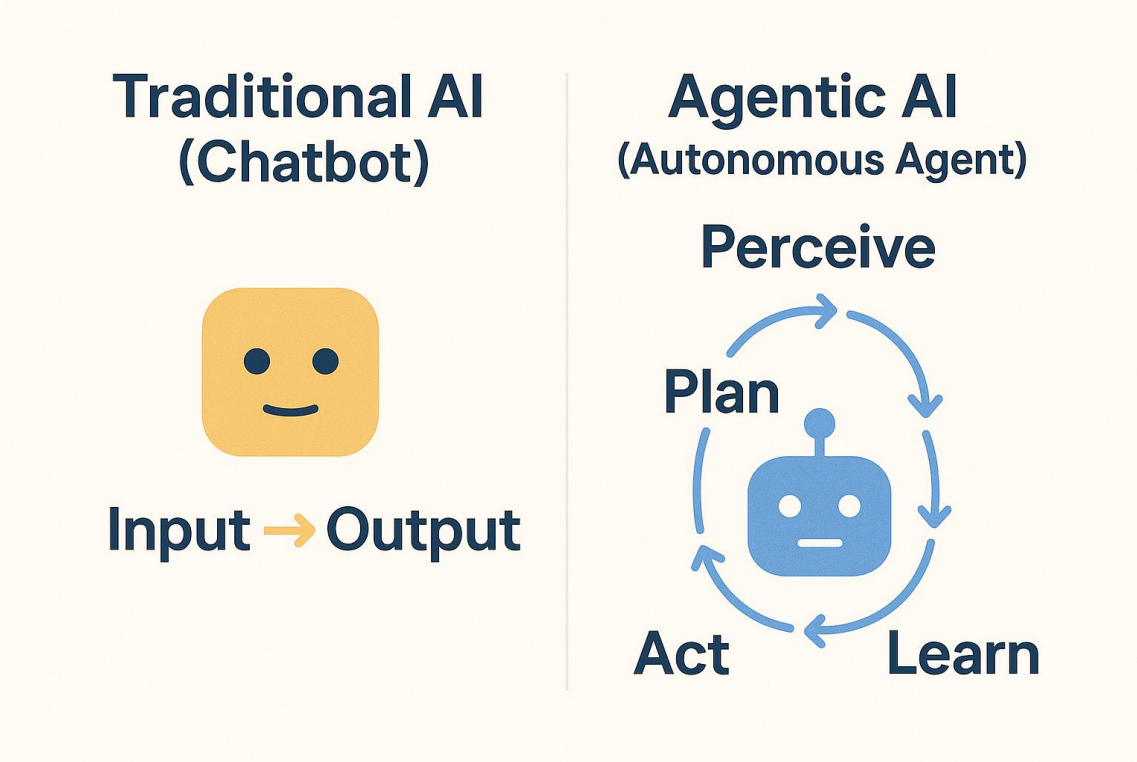
Why Do We Need Agentic AI?

Traditional AI models like ChatGPT and image generators are passive responders, they take one input and give one output.

However, many real-world tasks (like booking flights, writing reports, analyzing trends) require:

- Multi-step decision making
- Tool use (e.g., search, coding)
- Long-term memory
- Adaptation and autonomy

Thus, Agentic AI emerged to solve these limitations.



Agentic AI overcomes the limitations of prompt-response systems.

What is Agentic AI?

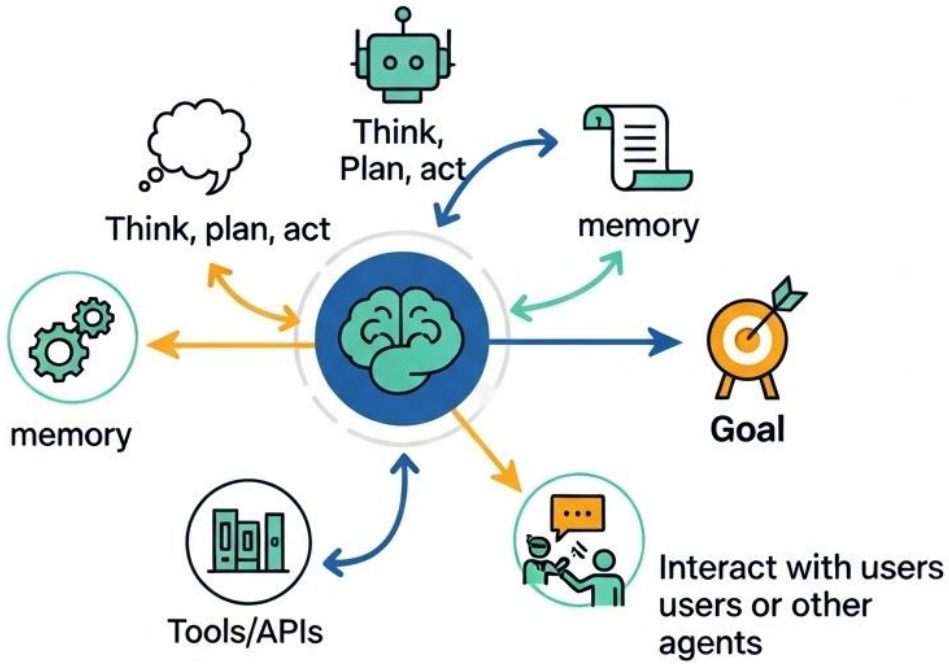
- AI system that can make decisions and take actions on its own to achieve a goal without being told exactly what to do at every step.

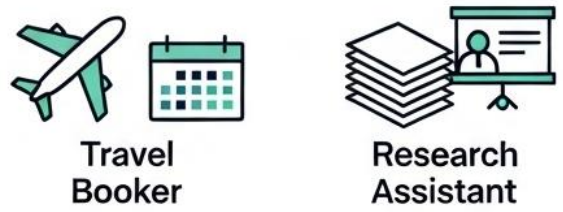
- Systems that think, plan, act and learn to achieve goals.

- These systems reason step-by-step, use external tools/APIs, maintain memory, and interact with users or other agents.

- Unlike traditional models (like GPT used as a chatbot), agentic systems can break down tasks, delegate, and iterate intelligently.

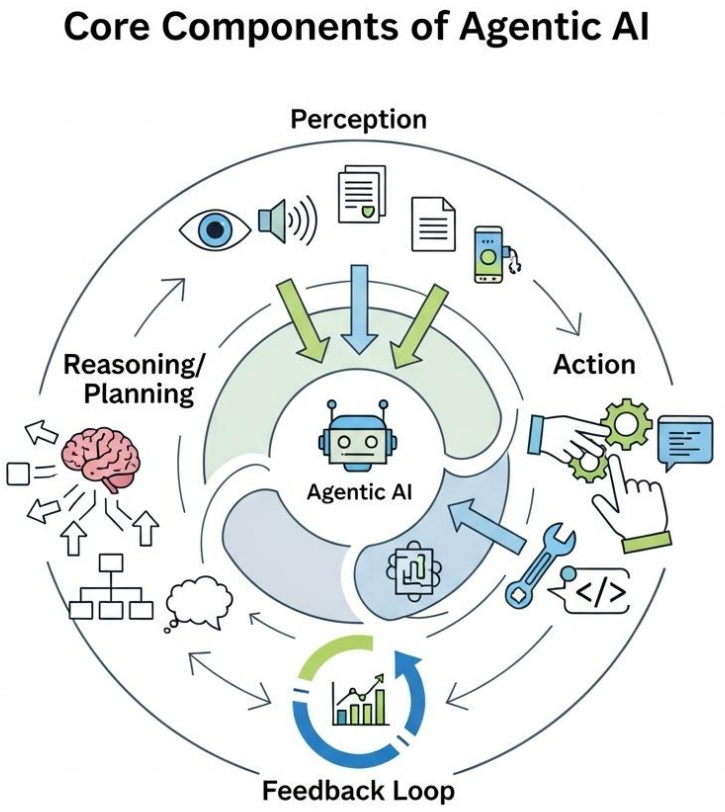
- **Example agents:** travel Booker (search flights, compare, book, update calendar), research assistant (find papers, summarize, create slides across sessions).





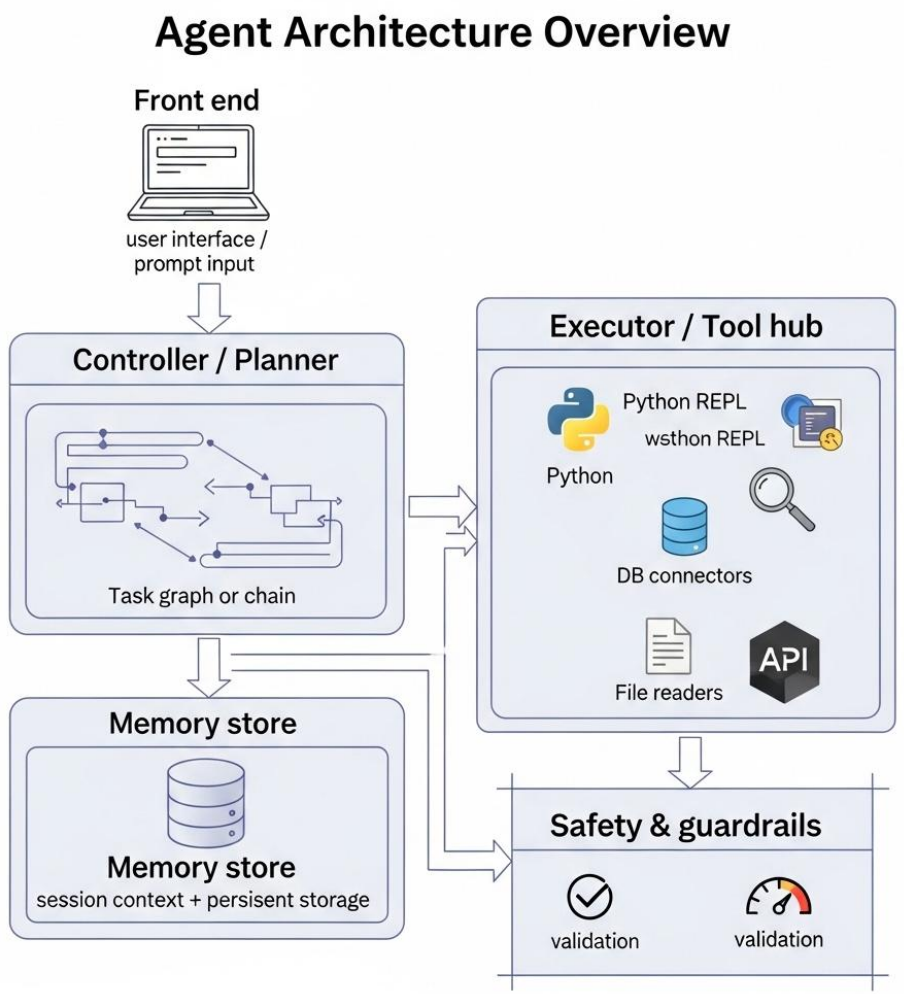
Core Components of Agentic AI

- **Perception:** Understands and processes input (text, image, audio, sensor).
- **Reasoning/Planning:** Create multi-step plans; decide next action.
- **Action:** Executes tasks via tools, APIs, environment interaction.
- **Feedback Loop:** Learns from outcomes for continuous improvement.



Agent Architecture Overview

- **Front end**: user interface / prompt input.
- **Planner (Controller)**: Turns the goal into a step-by-step plan (task list/graph) and decides the next action.
- **Executor (Tool Hub)**: Carries out actions via tools/APIs (Python code runner, web search, database queries, file readers, email/calendar).
- **Memory store**:
  - **Short-term:** current conversation/context.
  - **Long-term:** user preferences and history for continuity across sessions.
- **Safety & guardrails**: Validations before actions, rate limits, permissions/roles, human-in-the-loop approvals for risky steps, and logging/audit.



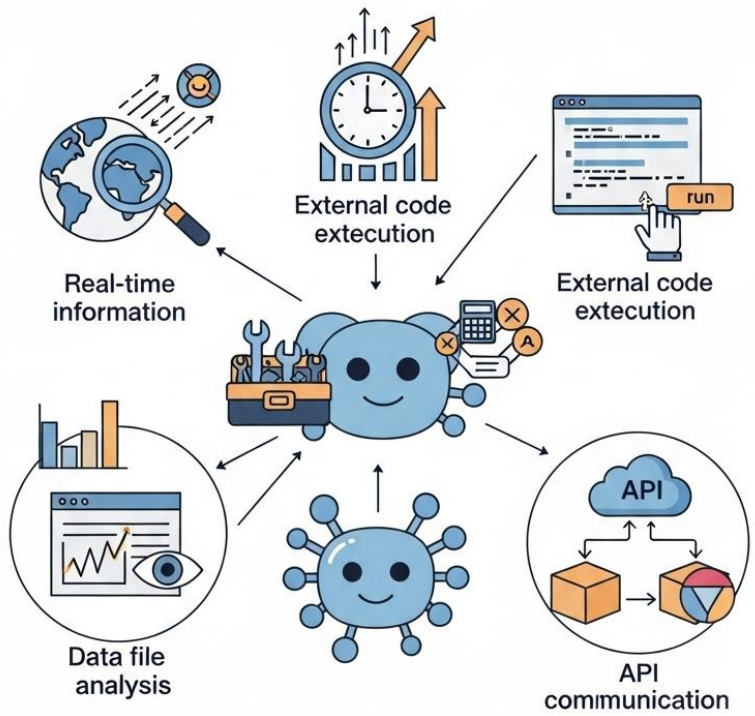
Tools in Agentic AI: Extend Agent Capabilities

- In Agentic AI, tools are external interfaces that agents can call upon to perform specific tasks just like humans use a calculator or search engine.
- They help agents act in the real world, not just generate language.

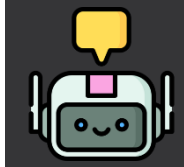
These tools enable:

- Real-time information access
- External code execution
- Data file analysis
- API communication

Using tools makes agents more **useful**, **accurate**, and **adaptive**.



Common Tools Used by Agentic AI Agents



Tool Name	Function	Example Use
Web Search	Retrieve live info from the web	Find latest product prices
API Connector	Call third-party or internal APIs	Get weather from OpenWeather API
File Reader / Parser	Read and parse user files (PDF, CSV, DOCX)	Summarize the content of a PDF.
Scheduler / Calendar API	Create events, reminders, invites	Add the project kickoff to Ali's calendar.
Browser/Automation	Automate web interactions (form fill, scraping)	Fill and submit the flight booking form.
Email / SMS API	Send notifications or confirmations	Send booking confirmation email to user.
Auth & Secret Manager	Manage credentials, tokens, and permissions	Retrieve API key securely before calling service

Frameworks: code vs no-code

Code-first (for developers):

- **LangChain**: Build chains, memory, and tool integrations, to create custom agents.
- **LangGraph**: Plan tasks as a graph for multi-step workflows (built on chains).
- **AutoGen (Microsoft)**: Coordinate multiple agents and delegate subtasks programmatically.

No-code / low-code:

- **n8n**: Visual workflow builder to connect APIs and trigger actions (no programming required).
- **CrewAI & team-agent platforms**: role-based multi-agent setups (e.g., researcher, writer, planner) for collaborative tasks.

When to use which?

- Use **Code-first** when you need full control, custom logic, or complex integrations.
- Use **No-code** when you need quick automation or non-developer setup.

Challenges & risks with Agentic AI

- **Hallucinations / wrong actions**: LLM may reason incorrectly or take unintended steps.
- **Tool integration complexity:** handling APIs, authentication, and edge cases is non-trivial.
- **Security & privacy:** agents with access to user data require strict controls.
- **Ethics & misuse**: autonomous actions can be harmful if unchecked.
- **Testing & observability**: difficult to anticipate and test all action paths.

Applications of Agentic AI

- **Personal Assistant**: Plan a day trip (flights, hotel, calendar updates).
- **Research Automation:** Find papers, extract methods, draft summaries.
- **Business Process:** Automate invoice processing, approvals reporting.
- **Health & Fitness Coach**: Track meals, suggest workouts, schedule reminders.
- **Robotics & Control Systems:** Agent issues commands to robots, adjusts plans.

Lab – Building a Research Agent

- Create an AI-powered assistant that searches the web and summarizes results into concise research notes.

- **Key Components:**
  - **LLM Core**: *Google Gemini 2.0 Flash* (reasoning and summarization)
  - **Tools**:
    - *SerpAPI* (live web search capabilities)
    - Custom research planner (step-by-step execution)
  - **Memory**: Short-term context retention for multi-step research
  - **Safety**: Content filtering and relevance validation
  - **Human Oversight**: User review and approval of final summaries

Steps in Lab:

- Set up environment & API keys
- Write agent code (planner + tools + memory)
- Run a sample query (e.g., *"Latest AI in healthcare"*)
- Inspect agent workflow (planning → search → summary)
- Discuss limitations & improvements

Thank You