Agentic AI

Understanding Autonomy, Planning, Reasoning, and Context Awareness

What is Agentic AI?

Agentic AI is a form of Artificial Intelligence that can take a **goal or task** from a user and work toward completing it **autonomously**, with minimal human intervention. It can **plan, act, adapt, and collaborate** intelligently — seeking help only when necessary.
In short: Agentic AI represents a shift from *reactive generation (GenAI)* to *proactive goal pursuit.*

Example – Agentic AI in Hiring Process

Scenario: The user instructs, "Hire a backend engineer."
The Agent:

1. Creates a job description (JD)
2. Posts it across platforms (LinkedIn, AngelList)
3. Parses applications automatically
4. Schedules interviews
5. Requests human feedback when uncertain

Result: The agent autonomously executes the hiring workflow, while staying aligned with human preferences and company policies.

Key Characteristics of Agentic AI

- **Autonomous**
- **Goal-Oriented**
- **Planning Capability**
- **Reasoning Ability**
- **Adaptability**
- **Context Awareness**

Autonomy

Definition: Autonomy is the AI system's ability to make decisions and act independently to achieve goals — without step-by-step human input.

Facets of Autonomy

- **Execution Autonomy** – Performs multi-step tasks independently.
- **Decision Autonomy** – Chooses actions based on reasoning.

- **Tool Autonomy** – Uses external tools/APIs without explicit commands.

Controlling Autonomy

1. **Permission Scope:** Limit actions (e.g., can screen candidates but not reject).
2. **Human-in-the-Loop (HITL):** Insert checkpoints (e.g., approval before posting JD).
3. **Override Controls:** Pause or modify agent behavior anytime.
4. **Guardrails/Policies:** Define ethical or operational boundaries.

Risks of Unchecked Autonomy

- Sending incorrect job offers.
- Violating HR or anti-discrimination laws.
- Overspending on ads or resources.

Goal-Oriented Behavior

Definition: Agentic AI works persistently toward a defined goal, adapting actions as needed — unlike reactive systems that respond only to prompts.

- Goals serve as a **compass** for autonomy.
- Goals include **constraints and progress tracking.**
- Goals are stored in structured memory.

Example – Goal Representation in Memory

```
{
  "main_goal": "Hire a backend engineer",
  "constraints": {
    "experience": "2-4 years",
    "remote": true,
    "stack": ["Python", "Django", "Cloud"]
  },
  "status": "active",
  "created_at": "2025-06-27",
  "progress": {
    "JD_created": true,
    "posted_on": ["LinkedIn", "AngelList"],
    "applications_received": 8,
    "interviews_scheduled": 2
  }
}
```

Goals can also evolve dynamically as the environment or user requirements change.

Planning

Definition: Planning enables the agent to decompose a high-level goal into subgoals or a sequence of actions to efficiently reach the desired outcome.

Steps in Planning

1. **Generate candidate plans:**
    - Plan A: Post JD on LinkedIn, GitHub Jobs, AngelList
    - Plan B: Use referrals or hiring agencies
2. **Evaluate plans:** Efficiency, Tool availability, Cost, Risk, and Constraint alignment.
3. **Select plan using:** HITL input or programmed policies (e.g., “prefer low-cost options”).

Reasoning

Definition: Reasoning allows the agent to interpret data, draw logical conclusions, and make context-aware decisions during both planning and execution.

Reasoning During Planning

- Goal decomposition
- Tool selection
- Resource estimation

Reasoning During Execution

- Decision-making (e.g., select best candidates)
- HITL awareness (when to pause or ask for input)
- Error handling (recover from API or tool failures)

Adaptability

Definition: The ability to modify strategies and actions dynamically in response to unexpected conditions while maintaining alignment with the main goal.

Examples:

- Handling tool failures (Calendar API down)
- Adjusting based on feedback (few applicants)
- Adapting to goal changes (switch to freelancer hiring)

Context Awareness

Definition: Context awareness allows an agent to understand and use relevant information from its environment, memory, and user preferences to make better decisions.

Types of Context

1. The original goal
2. Progress and interaction history
3. Environment state (e.g., deadlines, applicant counts)
4. Tool responses (API outputs)
5. User preferences
6. Policies and guardrails

Memory Implementation

- **Short-term memory:** Current session/task state
- **Long-term memory:** Persistent goals, preferences, and context

Summary Insight

Agentic AI extends Generative AI by adding **autonomy, memory, reasoning, and context awareness**. It transitions AI from a reactive content generator to a proactive decision-maker capable of real-world action.