What is Agentic AI?

Agentic AI is a type of AI that can take up a task or goal from a user and then work toward completing it on its own, with minimal human guidance.
It plans, takes action, adapts to changes, and seeks help only when necessary.

Example

26 June 2025 18:00

scenario → hr recruitment

Agentic AI chatbot

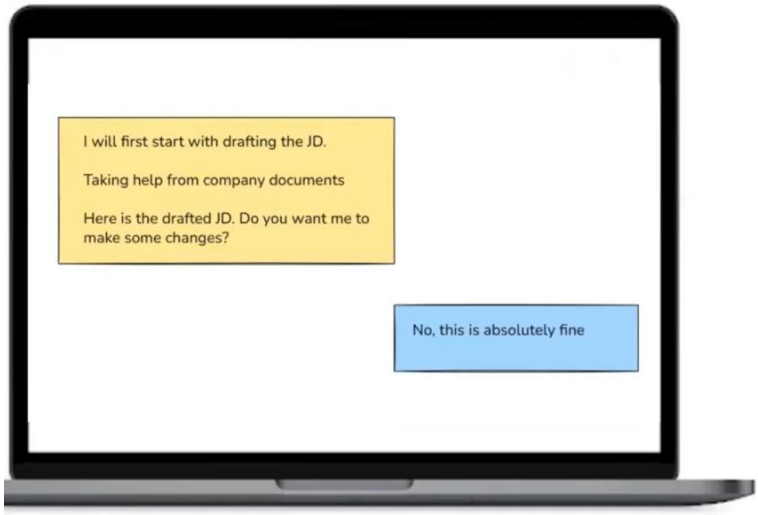
I want to hire a backend engineer.

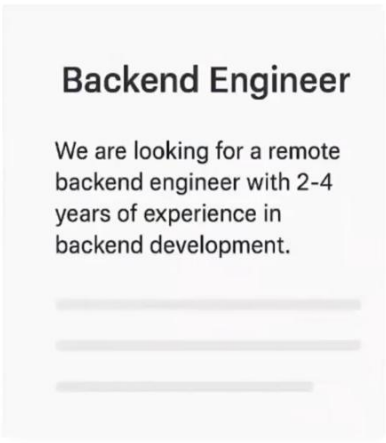
✅ Goal: Hire a remote backend engineer (2-4 yiers expecerrience).

Proposed plan:

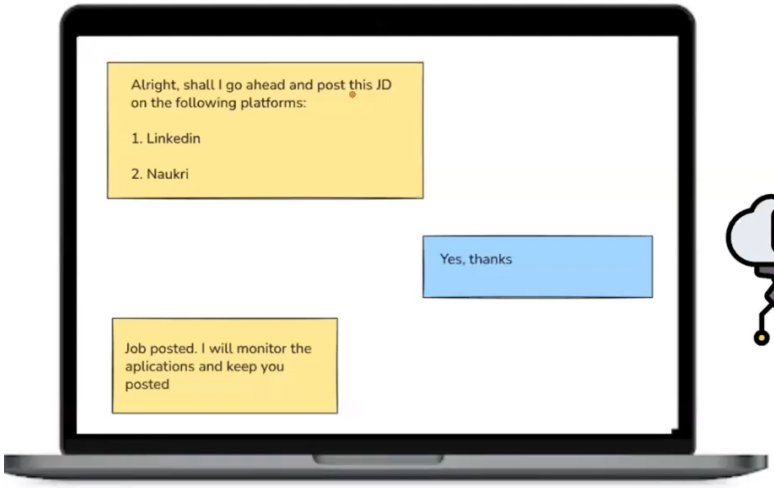
1. Draft JD → post on best platforms
2. Monitor pipeline → adjust strategy if needed
3. Screen candidates → schedule interviews
4. Draft + send offer → start onboarding

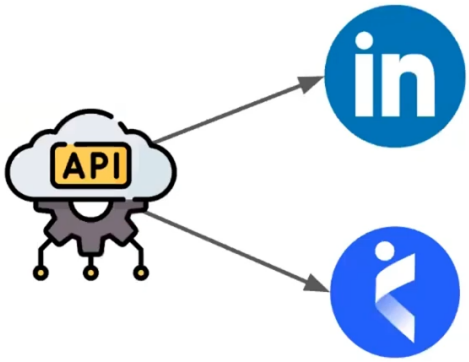
1. Drafting JD



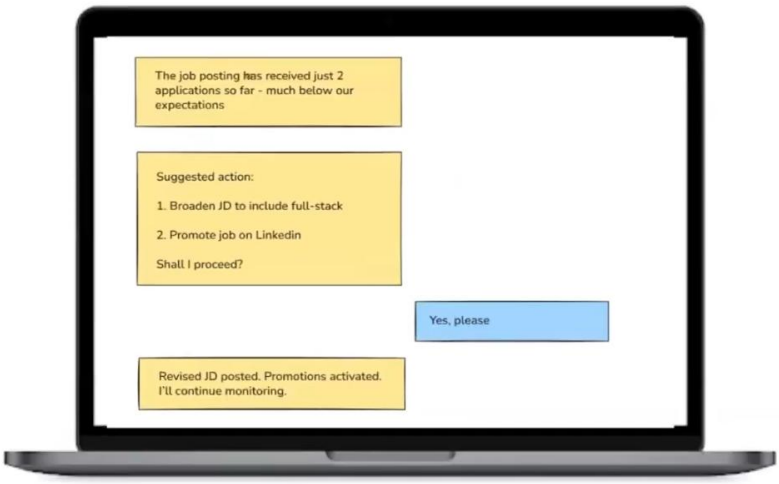


2. Posting JD





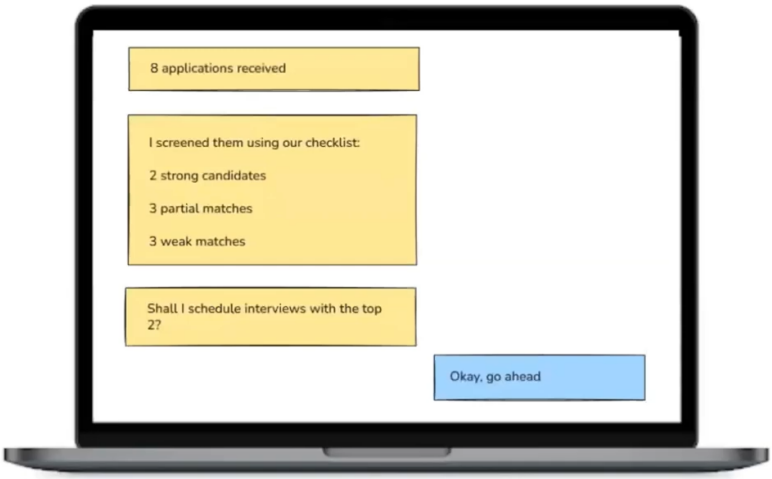
3. Revising the JD

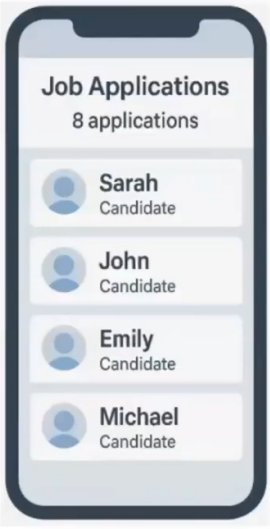


Revised JD

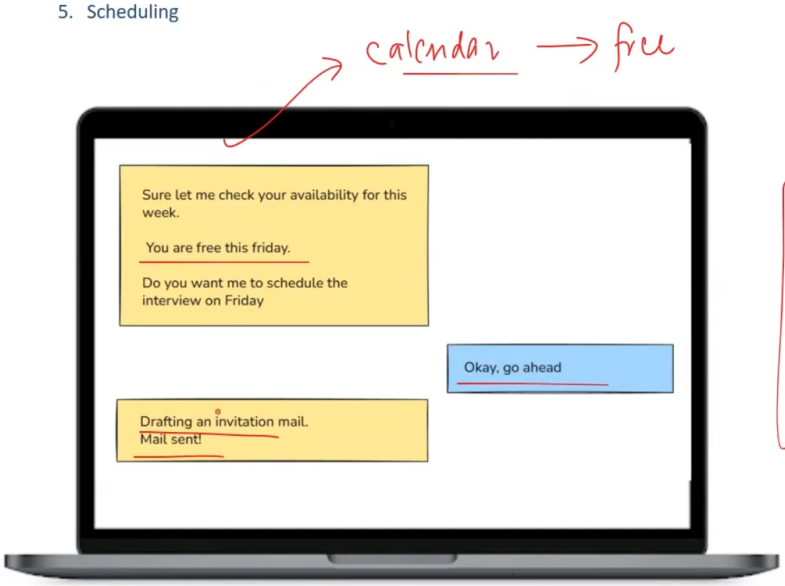
Full-stack candidates are now allowed.

4. Shortlisting





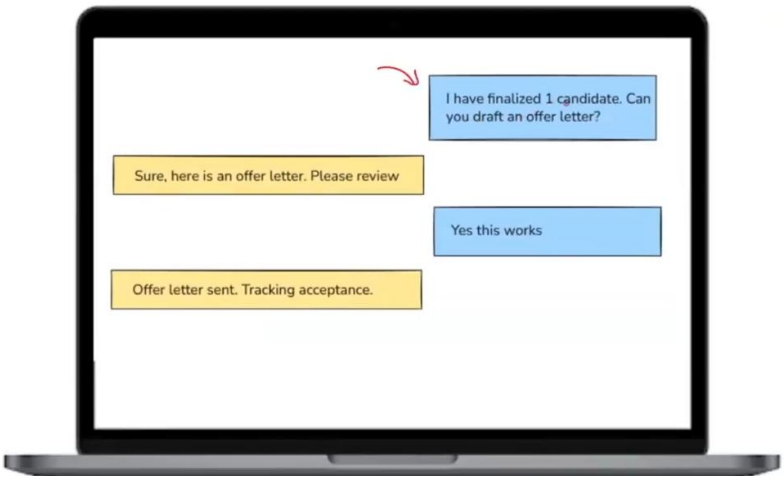
5. Scheduling



Hi [Candidate],

We'd like to schedule a 45-minute interview for the Backend Engineer role. Please share your availability

7. Sending offer letter

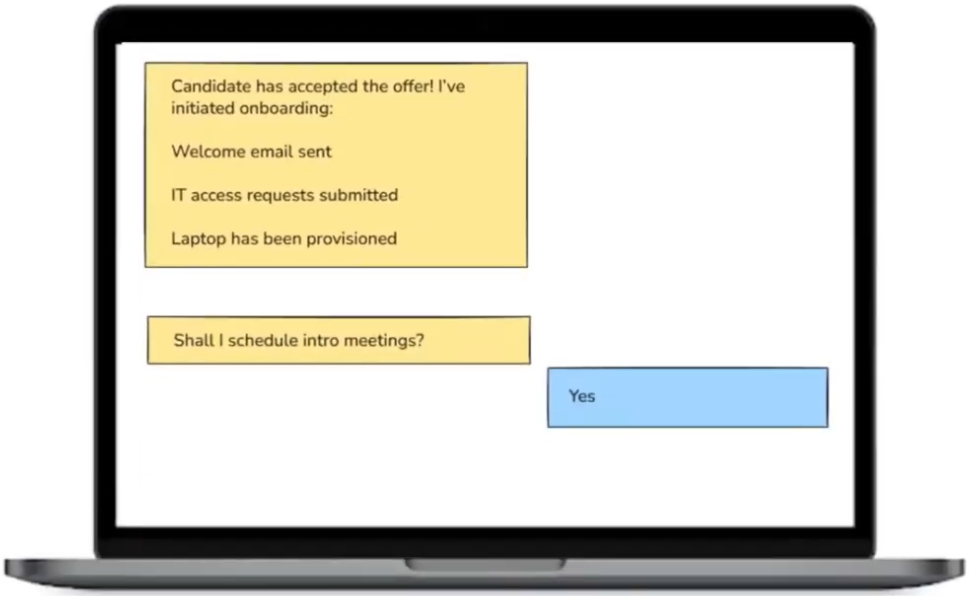


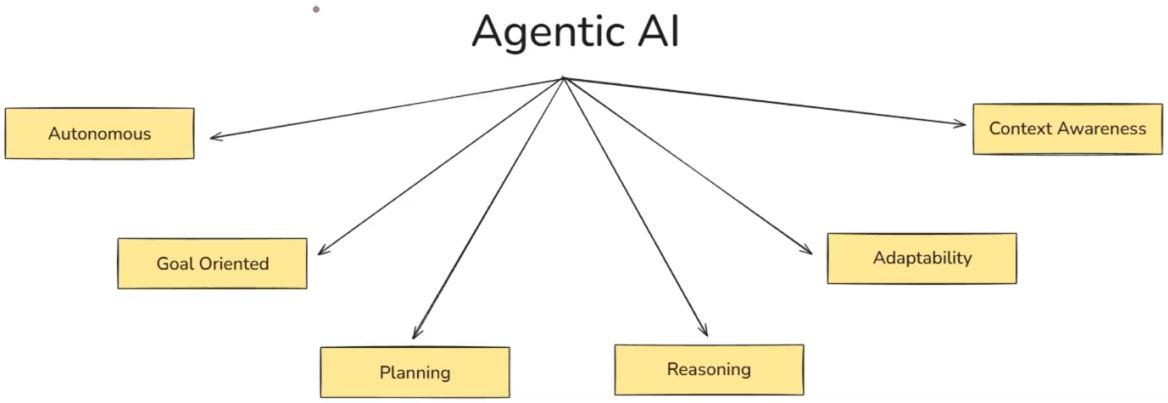
Dear [Candidate],

We are pleased to offer you the position of Backend Engineer.

Please let us know if you accept.

8. Onboarding





Autonomy

Autonomy refers to the AI system's ability to make decisions and take actions on its own to achieve a given goal, without needing step-by-step human instructions.

1. Our AI recruiter is autonomous
2. It's proactive
3. Autonomy in multiple facets
    a. Execution
    b. Decision making
    c. Tool usage
4. Autonomy can be controlled
    a. **Permission Scope** - Limit what tools or actions the agent can perform independently. (Can screen candidates, but needs approval before rejecting anyone.)
    b. **Human-in-the-Loop (HITL)** - Insert checkpoints where human approval is required before continuing. (Can I post this JD)
    c. **Override Controls** - Allow users to stop, pause, or change the agent's behaviour at any time. (pause screening command to halt resume processing.)
    d. **Guardrails / Policies** - Define hard rules or ethical boundaries the agent must follow. (Never schedule interviews on weekends)
5. Autonomy can be dangerous
    a. The application autonomously sends out job offers with incorrect salaries or terms.
    b. The application shortlists candidates by age or nationality, violating anti-discrimination laws.
    c. The applications spending extra on LinkedIn ads.

Goal Oriented

Being goal-oriented means that the AI system operates with a persistent objective in mind and continuously directs its actions to achieve that objective, rather than just responding to isolated prompts.

1. Goals acts as a compass for Autonomy
2. Goals can come with constraints
3. Goals are stored in core 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
  }
}
```

4. Goals can be altered

Planning

Planning is the agent's ability to break down a high-level goal into a structured sequence of actions or subgoals and decide the best path to achieve the desired outcome.

Step 1: Generating multiple candidate plans

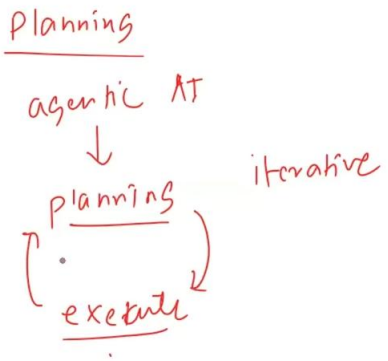
- **Plan A**: Post JD on LinkedIn, GitHub Jobs, AngelList
- **Plan B**: Use internal referrals and hiring agencies

Step 2: Evaluate each plan

- **Efficiency** (Which is faster?)
- **Tool Availability** (Which tools are available)
- **Cost** (Does it require premium tools?)
- **Risk** (Will it fail if we get no applicants?)
- **Alignment** with constraints (remote-only? budget?)

Step 3: Select the best plan with the help of:

- **Human-in-the-loop** input (e.g., "Which of these options do you prefer?")
- A pre-programmed **policy** (e.g., "Favor low-cost channels first")



Reasoning

Reasoning is the cognitive process through which an agentic ai system interprets information, draws conclusions, and makes decisions — both while planning ahead and while executing actions in real time.

Reasoning During Planning:

1. **Goal decomposition** - Break down abstract goals into concrete steps
2. **Tool selection** - Decide which tools will be needed for which steps
3. **Resource estimation** - Estimate time, dependencies, risks

Reasoning During Execution:

1. **Decision-making** - Choosing between options (3 candidates match → schedule 2 best, reject 1)
2. **HITL handling** - Knowing when to pause and ask for help (Unsure about salary range)
3. **Error handling** - Interpreting tool/API failures and recovering

Adaptability

Adaptability is the agent's ability to modify its plans, strategies, or actions in response to unexpected conditions — all while staying aligned with the goal.

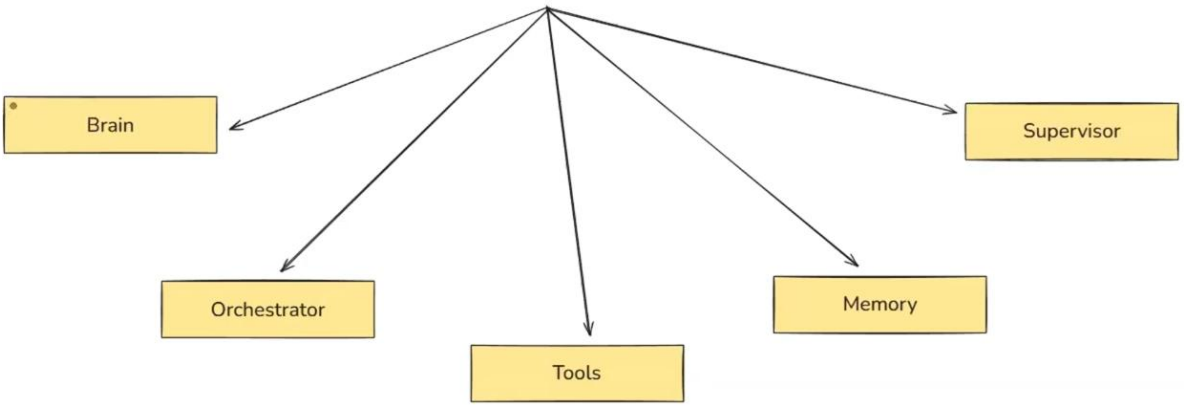
1. Failures (Calendar API)
2. External feedback (Less no of applications)
3. Changing goals (Hiring a freelancer)

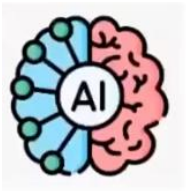
Context Awareness

Context awareness is the agent's ability to understand, retain, and utilize relevant information from the ongoing task, past interactions, user preferences, and environmental cues to make better decisions throughout a multi-step process.

1. Types of context
    a. The original goal
    b. Progress till now + Interaction history (Job description was finalized and posted to LinkedIn & GitHub Jobs)
    c. Environment state (Number of applicants so far = 8 or LinkedIn promotion ends in 2 days)
    d. Tool responses (Resume parser → "Candidate B has 3 years Django + AWS experience or Calendar API → "No conflicts at 2 PM Wednesday)
    e. User specific preferences (Prefers remote-first candidates or Likes receiving interview questions in a Google Doc)
    f. Policy or Guardrails(Do not send offer without explicit user approval or Never use platforms that require paid ads unless approved)
2. Context awareness is implemented through memory
3. Short term memory
4. Long term memory

Components





Brain

Goal Interpretation	Understands user instructions and translates them into objectives.
Planning	Breaks down high-level goals into subgoals and ordered steps.
Reasoning	Makes decisions, resolves ambiguity, and evaluates trade-offs.
Tool Selection	Chooses which tool(s) to use at a given step.
Communication	Generates natural language outputs for humans or other agents.

Orchestrator

Task Sequencing	Determines the order of actions (step 1 → step 2 → ...).
Conditional Routing	Directs flow based on context (e.g., if failure, retry or escalate).
Retry Logic	Handles failed tool calls or reasoning attempts with backoff.
Looping & Iteration	Repeats steps (e.g., keep checking job apps until 10 are received).
Delegation	Decides whether to hand off work to tools, LLM, or human.

Tools

External Actions	Perform API calls (e.g., post a job, send an email, trigger onboarding).
Knowledge Base Access	Retrieve factual or domain-specific information using RAG or search tools to ground responses.

Memory

Short-Term Memory	Maintains the active session's context — recent user messages, tool calls, and immediate decisions.
Long-Term Memory	Persists high-level goals, past interactions, user preferences, and decisions across sessions.
State Tracking	Monitors progress: what's completed, what's pending (e.g., "JD posted", "Offer sent").

Supervisor

↓

HITL

Approval Requests (HITL)	Agent checks with human before high-risk actions (e.g., sending offers).
Guardrails Enforcement	Blocks unsafe or non-compliant behavior.
Edge Case Escalation	Alerts humans when uncertainty/conflict arises.

Agent + human