Agent Frameworks
Anthropic Agent SDK / Pydantic AI / LangChain

Anthropic Agent SDK

Built for autonomous agents, not just orchestration
Designed around execution loops where agents can plan, act, verify, and iterate with minimal human intervention.

Native support for Model Context Protocol (MCP)
First-class integration with MCP enables secure, standardized access to external tools, services, and data sources at runtime.

First-class sub-agent delegation
Agents can spawn specialized sub-agents to parallelize work, delegate research, or isolate risky actions.

Deeply optimized for Claude models and long-context reasoning
Leverages Claude's strengths in long-horizon tasks, reasoning, and context compaction for sustained agent behavior.

Business Problem

When a critical security vulnerability is disclosed, engineering and security teams scramble to assess impact, patch systems, and coordinate fixes across repositories and environments—often through manual, time-consuming processes. The business needs an autonomous but controlled solution that can analyze exposure, execute remediation steps, and adapt as new information emerges, while still keeping humans in the loop for high-risk decisions

Pydantic AI

Built for structured, reliable agents (not free-form autonomy)
Designed for workflows where agent behavior must be predictable and repeatable, making it well suited for production systems.

Schema-first, type-safe outputs with Pydantic validation
Agent inputs and responses are validated against explicit schemas, reducing formatting errors and making outputs safe for automation.

Explicit tools as skills with validated inputs and outputs
Tools are defined as typed Python functions, which makes them easier to test, govern, and debug.

Model-agnostic support across major LLM providers
Works with OpenAI, Anthropic, Gemini, and others through adapters, allowing model changes without rewriting workflows.

Business Problem

When teams turn unstructured requests (emails, tickets, meeting notes) into operational work—like updating Salesforce records, generating implementation plans, or producing release checklists—errors creep in because the output format is inconsistent and hard to validate. The business needs a system that reliably turns messy inputs into structured, machine-ready outputs (with clear fields, statuses, and next steps) so automation can run safely and humans only review exceptions.

LangChain

Rich tool and integration ecosystem
Provides a large library of built-in tools and connectors (APIs, databases, vector stores, search) that agents can dynamically select at runtime.

Support for agent routing and multi-agent workflows
Enables agents to delegate tasks, route decisions, and coordinate across multiple agents using executors or graph-based flows.

Broad, multi-language LLM support
Works with most major LLM providers and supports both Python and JavaScript, making it easy to integrate into existing stacks.

First-class memory and retrieval capabilities
Includes built-in conversation memory, summarization, and vector-based retrieval to support long-running, context-aware agents.

Business Problem

When employees or customers need answers that span internal documentation, systems, and operational tools, teams often rely on manual searches or rigid workflows that don't adapt well to complex requests. The business needs a flexible agent that can retrieve knowledge from multiple sources, reason through multi-step problems, and take follow-up actions such as opening tickets or updating systems. This solution must support rich context, integrate with many existing tools, and evolve quickly as workflows and data sources change.

Quick rule of thumb

Pydantic AI: "Return a correct, typed artifact I can trust."

LangChain: "Use lots of tools + knowledge sources in a structured workflow."

Anthropic Agent SDK: "Run autonomously, delegate, and execute in loops."

Anthropic Agent SDK

Built for autonomous agents, not just orchestration
Designed around execution loops where agents can plan, act, verify, and iterate with minimal human intervention.

Native support for Model Context Protocol (MCP)
First-class integration with MCP enables secure, standardized access to external tools, services, and data sources at runtime.

First-class sub-agent delegation
Agents can spawn specialized sub-agents to parallelize work, delegate research, or isolate risky actions.

Deeply optimized for Claude models and long-context reasoning
Leverages Claude's strengths in long-horizon tasks, reasoning, and context compaction for sustained agent behavior.

Business Problem

When a critical security vulnerability is disclosed, engineering and security teams scramble to assess impact, patch systems, and coordinate fixes across repositories and environments—often through manual, time-consuming processes. The business needs an autonomous but controlled solution that can analyze exposure, execute remediation steps, and adapt as new information emerges, while still keeping humans in the loop for high-risk decisions

Pydantic AI

Built for structured, reliable agents (not free-form autonomy)
Designed for workflows where agent behavior must be predictable and repeatable, making it well suited for production systems.

Schema-first, type-safe outputs with Pydantic validation
Agent inputs and responses are validated against explicit schemas, reducing formatting errors and making outputs safe for automation.

Explicit tools as skills with validated inputs and outputs
Tools are defined as typed Python functions, which makes them easier to test, govern, and debug.

Model-agnostic support across major LLM providers
Works with OpenAI, Anthropic, Gemini, and others through adapters, allowing model changes without rewriting workflows.

Business Problem

When teams turn unstructured requests (emails, tickets, meeting notes) into operational work—like updating Salesforce records, generating implementation plans, or producing release checklists—errors creep in because the output format is inconsistent and hard to validate. The business needs a system that reliably turns messy inputs into structured, machine-ready outputs (with clear fields, statuses, and next steps) so automation can run safely and humans only review exceptions.

LangChain

Rich tool and integration ecosystem
Provides a large library of built-in tools and connectors (APIs, databases, vector stores, search) that agents can dynamically select at runtime.

Support for agent routing and multi-agent workflows
Enables agents to delegate tasks, route decisions, and coordinate across multiple agents using executors or graph-based flows.

Broad, multi-language LLM support
Works with most major LLM providers and supports both Python and JavaScript, making it easy to integrate into existing stacks.

First-class memory and retrieval capabilities
Includes built-in conversation memory, summarization, and vector-based retrieval to support long-running, context-aware agents.

Business Problem

When employees or customers need answers that span internal documentation, systems, and operational tools, teams often rely on manual searches or rigid workflows that don't adapt well to complex requests. The business needs a flexible agent that can retrieve knowledge from multiple sources, reason through multi-step problems, and take follow-up actions such as opening tickets or updating systems. This solution must support rich context, integrate with many existing tools, and evolve quickly as workflows and data sources change.

Quick rule of thumb

Pydantic AI: "Return a correct, typed artifact I can trust."

LangChain: "Use lots of tools + knowledge sources in a structured workflow."

Anthropic Agent SDK: "Run autonomously, delegate, and execute in loops."