

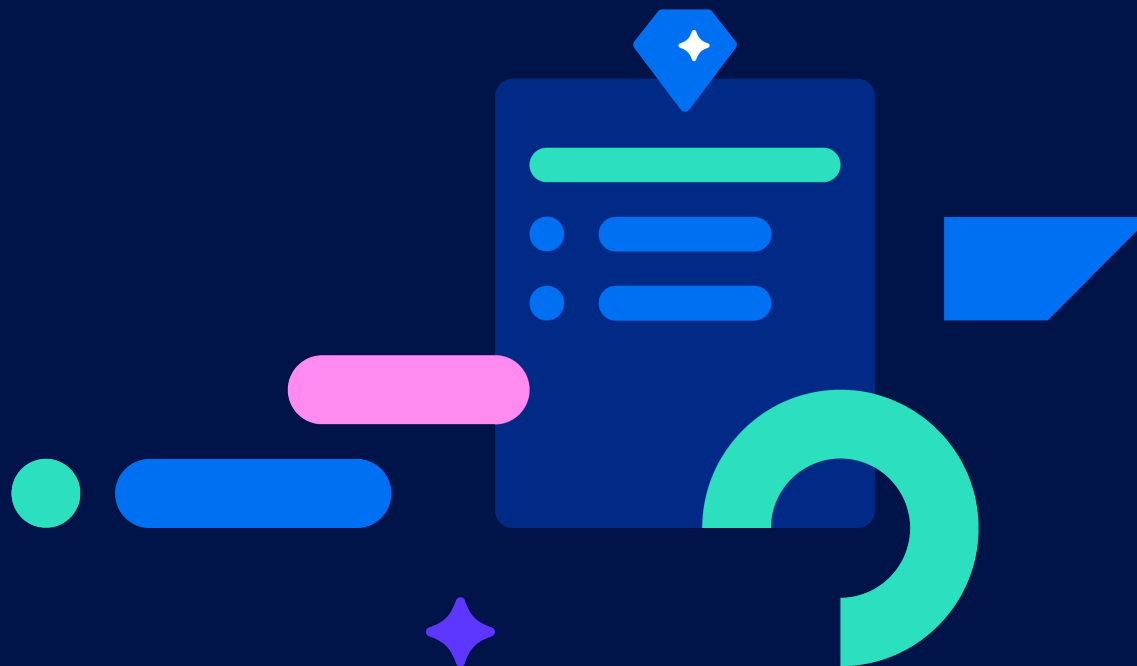
Devtoberfest by
SAP TechEd

Connect to SAP Generative AI Hub with LiteLLM



Karim Mohraz, SAP
October 2025

SAP TechEd

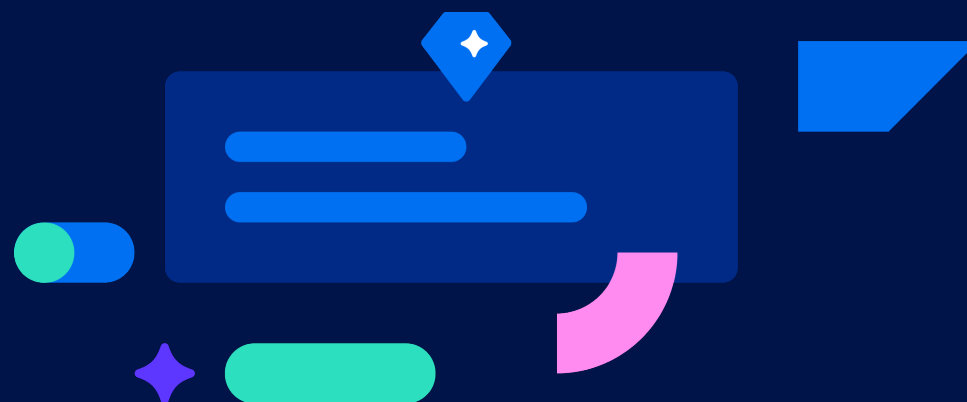


Agenda

- SAP Generative AI Hub & LiteLLM
- CrewAI
- Put Agents to Work
- Demo



SAP Generative AI Hub & LiteLLM



SAP Gen AI Hub & LiteLLM

LiteLLM:

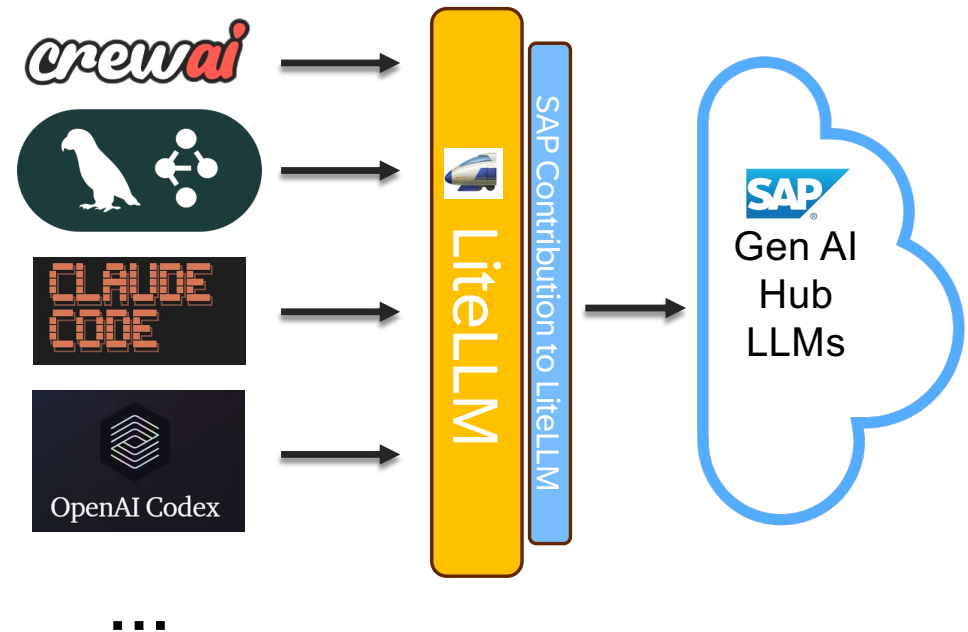
- Call 100+ LLMs in the **OpenAI** format

Agentic libraries and frameworks that support LiteLLM

- CrewAI
- LangGraph/LangChain
- Claude Code
- OpenAI Codex
- ...

SAP Generative AI Hub:

- Unified access to LLMs from major providers (ChatGPT, Claude, Gemini, Mistral, Llama)
- Pay as you go
- To be announced at TechEd:
[SAP contribution to LiteLLM](#)



CrewAI

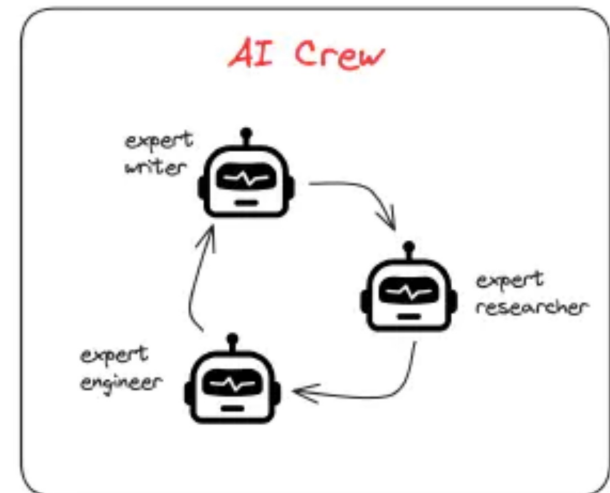


CrewAI

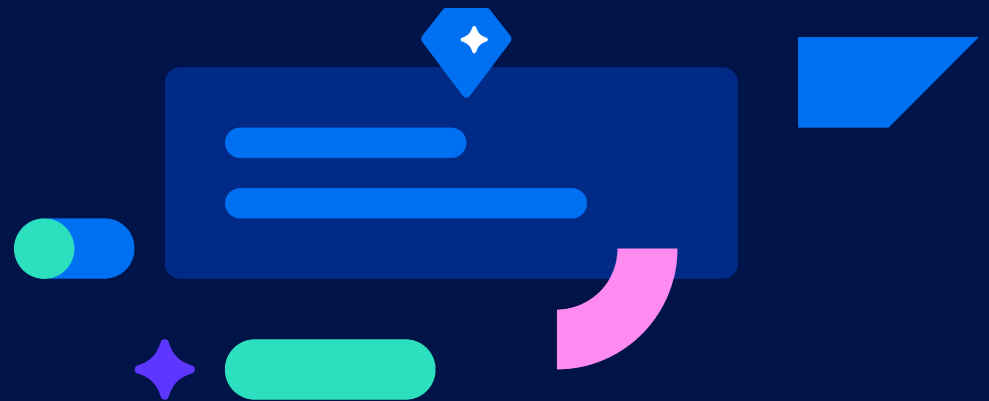
CrewAI is an *open-source* **multi-agent** orchestration framework.

This Python-based framework

- orchestrates role-playing autonomous AI agents
- that work together as a cohesive assembly or “expert crew” (simulating human-like teamwork)
- to complete complex tasks.





Put Agents to Work



Build the Expert Team

Tools:

- Generation Tool (expand/transform a topic) 
- Paraphrase Tool (summarize) 



Researcher (AI Expert)



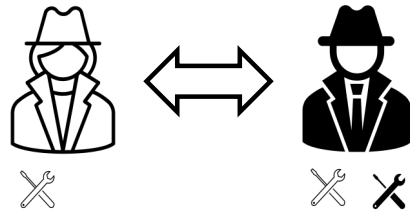
Developer Advocate (Writer)
receives Researcher output & writes a technical blog.



Result of Agent Collaboration Loop

Task:

Describe business use-cases that were *successfully* **solved** using agents.



Quotes from Researcher agent and final blog:

“The “Centaur” model (Human + AI), where the agent augments rather than replaces the human, often provides the optimal solution.”

*“Take, for example, [Klarna's AI system](#), which autonomously manages customer queries end-to-end, dramatically slashing resolution times while handling the equivalent workload of **hundreds** of human agents.”*

*“Similarly, in IT operations, these agents **monitor** and **optimize** infrastructure performance, preemptively managing resources like server loads to minimize downtime.”*

Demo



Step 1: Research Agent

```
My plan is to break down the research into several steps:
1. Define AI agents in a business context.
2. Identify the purpose and common business use-cases.
3. Understand the underlying architecture and inner workings.
4. Explore the strengths and limitations.
5. List popular tools and libraries for building agents.
6. Find specific real-world examples of successful adoption.
7. Uncover interesting or lesser-known facts.
8. Synthesize all the information into a final, structured answer.
I will start by getting a foundational understanding of AI agents and their purpose in business. Action:
generation_tool
```

Result:

Common Business Use-Cases and Successfully Automated Tasks

AI agents excel at tasks that are **rule-based**, **data-intensive**, and **repetitive**. Here are specific examples across key business functions.

1. Customer Support and Service

This is one of the most mature and successful domains for AI agents. The goal is to provide instant, consistent, and round-the-clock support.

Successfully Automated Tasks:

- Answering Frequently Asked Questions (FAQs):** Agents can instantly answer common questions like "What are your business hours?", "What is your return policy?", or "How do I reset my password?"
- Order and Status Tracking:** Customers can ask an agent...

Research Summary

```
### Real-World Examples of AI Agent Adoption

#### 1. Customer Service & E-commerce
* **Company:** Klarna (Global payments and shopping service)
* **The Problem:** Klarna handles millions of customer interactions monthly, ranging from simple queries about payments to more complex issues regarding returns and refunds. Scaling a human customer service team to provide 24/7, multilingual support is incredibly expensive, difficult to manage, and can lead to inconsistent service quality and long wait times for customers.
* **How the AI Agent Solved It:** In 2024, Klarna announced a significant deployment of a custom-built AI agent powered by OpenAI. This agent is not just a simple chatbot; it functions as a first-line customer service representative. It can:
    * Understand and respond to customer queries in multiple languages.
    * Access real-time customer data (securely) to provide personalized updates on payments, balances, and purchase history.
    * Independently manage the entire resolution process for tasks like refunds and return inquiries.
    * Escalate complex or sensitive cases to a human agent with a full summary of the interaction so far.

    **The result was staggering:** The AI agent is now handling the workload equivalent to 700 full-time human agents, has managed over 2.3 million conversations, and maintains a customer satisfaction score on par with its human counterparts, all while reducing resolution time from 11 minutes to under 2 minutes for most queries.

#### 2. Insurance Claims Processing
* **Company:** Lemonade (Insurtech company)
* **The Problem:** The traditional insurance claims process is notoriously slow, bureaucratic, and adversarial. It involves extensive paperwork, multiple handoffs between adjusters and approvers, and long wait times.

    I will start by getting a foundational understanding of AI agents and their purpose in business. Action:
    operation tool
    Result:
```

Summary

SAP LiteLLM contribution is a powerful new feature that

- enables seamless integration between SAP's Gen AI Hub and
- any open-source library or framework requiring access to large language models (LLMs).

What this means for you is unprecedented flexibility and choice in how you leverage AI within your SAP environment. By connecting through the LiteLLM library, you can tap into the latest advancements in generative AI—without being locked into a single vendor or proprietary solution.

How you'll benefit:

- **Open Ecosystem:** Easily connect SAP solutions to a wide range of open-source AI tools, expanding your innovation potential.
- **Cost Efficiency:** Reduce licensing and integration costs by utilizing open-source frameworks and libraries.
- **Future-Proofing:** Stay ahead of the curve by quickly adopting new AI technologies as they emerge, thanks to the adaptable LiteLLM interface.