**Report on Multi-Agent Architecture for Industry and Use Case Analysis Using CrewAI**

**Project Overview**

This multi-agent architecture system is designed to perform in-depth market research and generate AI use cases for companies across various industries. The architecture uses CrewAI and the SerperDevTool to enable each agent to conduct industry research, generate use cases based on market standards, and collect relevant resources. The system comprises three agents, each assigned a specific role, equipped with tools, and linked to tasks that produce detailed outputs to support business decisions and operational improvements.

### Agent and Task Breakdown

#### 1. Industry Research Agent

* **Role**: Industry Researcher
* **Goal**: Research the industry and determine the company’s strategic focus and key offerings.
* **Backstory**: The Industry Research Agent is tasked with leveraging web tools to gather and synthesize information on the company’s industry segment. This involves identifying market position, primary offerings, and strategic goals.
* **Task**: Uses the SerperDevTool to conduct web-based research. The task requires a comprehensive report detailing the industry landscape, the company's products or services, and overarching goals.
* **Expected Output**: A report summarizing the company's industry, offerings, and strategic priorities.

This task serves as the foundation for the subsequent agents by establishing a contextual understanding of the industry, allowing future agents to align their outputs with real-world market conditions.

#### 2. Use Case Generation Agent

* **Role**: Use Case Generator
* **Goal**: Analyze industry trends, particularly in AI, ML, and automation, to suggest actionable AI/ML use cases.
* **Backstory**: The Use Case Generator interprets the industry analysis and identifies trends that align with AI/ML applications, focusing on potential enhancements to operational efficiency and customer satisfaction.
* **Task**: Uses the SerperDevTool for market analysis and then proposes relevant AI/ML use cases. Examples include predictive maintenance, customer sentiment analysis, and supply chain optimization.
* **Expected Output**: A list of use cases with detailed descriptions, indicating how each can benefit the company.

This agent plays a crucial role by translating industry insights into applicable AI-driven solutions that can address both operational and customer-oriented needs.

#### 3. Resource Collection Agent

* **Role**: Resource Collector
* **Goal**: Gather datasets from platforms like Kaggle, HuggingFace, and GitHub to support the proposed use cases.
* **Backstory**: The Resource Collector searches for open-source datasets and tools that align with the generated use cases, enabling seamless integration and implementation.
* **Task**: Scans online repositories to find relevant datasets and models for the identified AI/ML solutions.
* **Expected Output**: A list of resource links, presented as clickable URLs, to datasets and tools for the recommended use cases.

This agent’s task provides the necessary resources for developing and testing the proposed use cases, making it a critical step toward the operational deployment of AI/ML solutions.

### Crew Workflow

The Crew class orchestrates the execution of these agents and their tasks, ensuring a smooth handoff between tasks:

1. **Industry Research Task** is executed first to set the foundation for further analysis.
2. **Use Case Generation Task** leverages the findings from the research to propose actionable AI/ML use cases.
3. **Resource Collection Task** identifies relevant data resources, completing the multi-agent workflow.

### Key Benefits and Applications

This multi-agent architecture offers significant advantages:

* **Scalability**: Can be adapted to any industry and tailored to different business needs.
* **Efficiency**: Automates the process of gathering insights, generating actionable use cases, and collecting resources, saving time and effort.
* **Decision Support**: Provides businesses with structured insights into AI/ML applications, enabling data-driven decision-making.

Start

Crew Initialization

- Initializes agents and tasks

- Sets up workflow for each agent and task

Industry Research Task

Agent: Industry Researcher

Task: Conduct web-based research to understand industry and company’s strategic focus areas.

Tool: SerperDevTool Expected Output: Industry report (key offerings, segment, strategic goals)

Market Standards & Use Case Generation Task

Agent: Use Case Generator

Task: Analyze industry trends related to AI, ML, and automation generate relevant AI/ML use cases for the company.

Tool: SerperDevTool Expected Output: List of actionable AI/ML use cases

Resource Asset Collection Task

Agent: Resource Collector

Task: Search for datasets relevant to the proposed use cases o platforms such as Kaggle, HuggingFace, and GitHub.

Expected Output: Clickable list of resource links for the use cases

Final Output

-Consolidated report of industry research

-List of actionable AI/ML use cases - Resource links for datasets

END

**Integration with Streamlit**: Deploying this architecture in Streamlit would allow interactive visualization and real-time data input, enhancing user experience.

 **Start (User Interface)**: The Streamlit app initiates with a simple user interface setup.

 **Setup API Keys**: The code retrieves API keys for OpenAI and Serper. The OpenAI key is stored as the model to be used by the agents, while the Serper key is used to allow agents access to a web search API.

 **Initialize Agents & Tasks**: The function initialize\_agents\_and\_tasks() configures three main agents:

* **Industry Research Agent**: Conducts an initial search to gather information on the company’s industry and focus areas.
* **Use Case Generation Agent**: Analyzes AI/ML trends within the specified industry to suggest relevant use cases.
* **Resource Asset Collection Agent**: Collects resources, like datasets or tools, that support the generated use cases.

 **User Inputs**: The user provides the company name and selects the industry from a dropdown menu.

 **Button "Generate Use Cases"**: When clicked, the crew.kickoff() method is called, running all agent tasks in the order defined by initialize\_agents\_and\_tasks().

 **Crew Kickoff (Execute Agent Tasks)**: The multi-agent process begins, each agent executing its task based on the company\_name and industry\_type input values. Results from each agent are stored.

 **Results Display**:

* **Industry Research**: Displays the industry insights from the Industry Research Agent.
* **Generated Use Cases**: Lists relevant AI/ML use cases provided by the Use Case Generation Agent.
* **Resource Assets**: Displays clickable resource links collected by the Resource Collection Agent.

 **End**: Once all results are displayed, the workflow concludes.

Results Display

- Industry Research

- Generated Use Cases

- Resource Assets

Crew Kickoff

(Execute Agent Tasks)

Button: "Generate Use Cases"

User Inputs

(Company Name, Industry)

Initialize Agents & Tasks

Start

END

### **Conclusion**

The proposed multi-agent architecture built with CrewAI streamlines the process of market research, use case generation, and resource collection for companies aiming to leverage AI/ML technologies. By combining industry insights with actionable AI use cases and relevant datasets, this architecture serves as a valuable tool for companies across sectors seeking to enhance their operations and customer engagement through AI-driven innovation.