

# Capstone Project

## Multi-Agent Market Research and GTM Planning (n8n, MCP, and CrewAI)

### Overview

In this project, you will design and implement a multi-agent workflow system that automates market research and go-to-market (GTM) planning. The systems are implemented in **n8n** and **CrewAI** separately, but with similar functionality. The solution integrates four agents:

- Head Planner (orchestrator and documenter)
- Research Agent (finder)
- Analyst Agent (sense-maker)
- Strategy Agent (planner)

The workflow must orchestrate desk research, competitor analysis, and GTM plan drafting before exporting structured strategy documents to Google Docs.

### Instructions

- Review the lessons and supporting materials on n8n workflows, MCP, and CrewAI
- Set up the required environment on the Ubuntu VM, including Node.js, n8n, and Python for CrewAI
- Execute the following steps:
  - Build the multi-agent orchestration with a Head Planner and three specialist agents with tools
  - Start the MCP server
  - Configure and connect the n8n workflow with the MCP
  - Implement CrewAI with defined agents, flows, and tools (MCP, SerpAPI)
- Test and debug each component individually (MCP tools with curl, n8n nodes with Execute Node, and CrewAI tasks end-to-end)
- Document the architecture, configuration steps, test runs, and error resolutions
- Submit the following:
  - Exported n8n workflow JSON file
  - CrewAI project files (UV structure)
  - Sample Google Doc output
  - CrewAI chatbot screenshots

- Documentation (README file) covering the architecture, setup, and testing notes

## Situation

Product teams often spend days manually collecting sources, analyzing competitors, and drafting GTM strategy documents. This process is:

- **Slow:** Causes delays in responsiveness to market shifts
- **Error-prone:** Leads to missed insights due to manual effort
- **Inconsistent:** Results in a lack of reproducibility across research cycles

To solve these problems, a project requires a multi-agent planner with automation and observability (delivered in two implementations (n8n and CrewAI) for comparison).

## Key Performance Indicators (KPIs):

Your system will be considered successful if it meets these criteria:

- **Coverage:** ≥90% of research questions answered with linked sources
- **Source quality:** ≥80% citations from top-tier or primary sources, 0% broken links
- **Latency:** <15 minutes from project brief to drafted GTM document
- **Strategy quality:** Human rubric score ≥4/5 (clarity, feasibility, differentiation)
- **Reproducibility:** ≥80% consistent facts across multiple runs
- **Cost efficiency:** Cloud/API spend per run within budget cap

## Tasks

- Build a four-agent system (Head Planner, Research Agent, Analyst Agent, Strategy Agent)
- Collect research evidence with proper citations in JSON format
- Generate competitor tables, pricing matrices, and qualitative synthesis (SWOT/4P/7P)
- Draft a structured GTM plan (ICPs, value proposition, messaging, channels, and a launch plan)
- Export the plan to Google Docs, with PDF option
- Implement logging, retries, and cost/latency tracking
- Compare results of n8n vs. CrewAI implementation

## Actions:

### 1. Implementing multi-agent workflow with n8n

#### 1.1 Set up the environment

- Install Node.js and n8n on Ubuntu VM
- Start MCP server for research capabilities

#### 1.2 Build the workflow

- Configure nodes: Trigger → Head Planner → Research Agent → Analyst Agent → Strategy Agent → Docs Writer
- Integrate MCP tools for research and SerpAPI for queries
- Enable logging, retries, and cost tracking

#### 1.3 Test and export

- Test and execute each node
- Debug errors and validate outputs
- Export the workflow as a JSON file

### 2. Implementing multi-agent workflow with CrewAI

#### 2.1 Set up the environment

- Install Python and CrewAI framework
- Start MCP server for research tools

#### 2.2 Build the workflow

- Define the four agents with clear roles and tools
- Chain tasks with CrewAI Flows for hand-offs
- Retain interim outputs (tables, markdown)
- Integrate MCP tools and SerpAPI for research

#### 2.3 Test and export

- Run end-to-end workflow
- Capture logs and debugging notes
- Export chatbot demo screenshots
- Save project files in UV structure

## Testing

For both implementations, perform:

- **Unit tests:** Mock inputs/outputs for tools (SerpAPI, MCP, Docs)
- **Scenario tests:** Fixed briefs with golden expected outputs
- **Human review:** Evaluate strategy quality with rubric

- **Comparison:** Measure cost, latency, and reliability across n8n and CrewAI

## Risks and Mitigations

- **Source volatility:** Mitigate by caching results, storing page snapshots, and including timestamps
- **API rate limits:** Mitigate with exponential backoff, batching requests, or using multiple keys (if permitted)
- **Hallucinations:** Mitigate by mandating evidence IDs for each fact and flagging any uncited claims
- **Formatting drift:** Mitigate by using stable Google Docs templates and running post-write validation
- **Cost overruns:** Mitigate by enforcing token limits, summarizing early, and applying budget caps

## Result

By the end of this project, you will deliver a working multi-agent system for market research and GTM planning, implemented in both n8n and CrewAI, along with complete documentation and artifacts.

- Exported n8n workflow JSON file
- CrewAI project files (UV structure)
- Sample Google Docs output of GTM plan
- CrewAI chatbot screenshots
- Documentation (README) with architecture, setup steps, test notes, and comparison results