# Weekly Progress Report: Mitesh Singh

Project: MarketMind: An Open-Source Framework for AI-Driven Marketing Secondary Research

Reporting Period: June 23, 2025 – June 30, 2025

### 1. Executive Summary

This reporting period marks the foundational phase of the MarketMind project. The primary focus has been on translating the theoretical architecture outlined in the project's whitepaper into a tangible, functional codebase. Key achievements include the design and implementation of a multi-agent AI system using the crewai framework, the definition of specialized agent roles and corresponding research tasks, and the development of custom tooling for real-time web data acquisition. This initial sprint has successfully established the core architectural backbone of the application, paving the way for the integration of orchestration logic and a user-facing interface in the next development cycle.

### 2. Detailed Summary of Work Completed

The work accomplished over these two weeks was centered on building the modular components that will collectively form the AI research crew. This involved a deep dive into agent-based system design and the practical implementation of each component.

#### 2.1. Architectural Design & Framework Implementation

In alignment with the project's mission to democratize marketing research, I architected the system as a collaborative multi-agent crew. This approach, powered by the crewai library, allows for the simulation of a human research team, where different specialists handle distinct parts of the analysis. This modularity is crucial for creating a robust and extensible framework.

* **File:** agents.py
* **Contribution:** I designed and implemented the five core agents that constitute the "mind" of MarketMind. Each agent is meticulously crafted with a specific role, goal, and backstory to guide its reasoning process and ensure it performs its function effectively within the larger team.
  + **Strategy Consultant:** This agent acts as the project manager, initiating the research by breaking down the user's high-level goal into a series of strategic, probing questions.
  + **Competitor Analyst:** Equipped with web-scraping tools, this agent is responsible for identifying and analyzing key competitors, creating a detailed report on their strategies, strengths, and weaknesses.
  + **Customer Persona Analyst:** This agent focuses on the human element, researching and developing detailed user personas to understand the target audience's needs, motivations, and pain points.
  + **Devil's Advocate:** To mitigate confirmation bias and ensure analytical rigor, this agent's sole purpose is to critique the findings of the other agents, identifying potential risks, gaps, and flawed assumptions.
  + **Lead Strategy Synthesizer:** The final agent in the chain, responsible for consolidating all prior analyses and critiques into a single, cohesive, and comprehensive market research report.

#### 2.2. Task Definition and Workflow Structuring

To ensure a logical and repeatable research process, I defined a series of tasks that correspond to each agent's role. This creates a structured workflow from initial planning to final synthesis.

* **File:** tasks.py
* **Contribution:** I implemented the task classes, which serve as the instructions for the agents. Each Task object has a verbose description of what needs to be done and a clearly defined expected\_output. This structure is critical for ensuring that the output of one agent's task can be seamlessly used as the input for the next. The tasks are designed to produce individual Markdown files, creating a transparent and auditable trail of the research process.

#### 2.3. Custom Tool Development for Data Acquisition

A core requirement for effective secondary research is access to real-time information from the web. To facilitate this, I developed a custom tool for the agents.

* **File:** custom\_search\_tool.py
* **Contribution:** I engineered a WebSearchTool by creating a wrapper around the FirecrawlApp API. This tool provides the agents with the crucial ability to perform targeted web searches. It is more than just a simple search; it's an abstracted capability that allows an AI agent to query the internet for specific information, such as competitor product reviews or articles discussing market trends. This tool is the primary data-gathering mechanism for the Competitor Analyst and Customer Persona Analyst.

#### 2.4. Environment and Dependency Management

* **File:** requirements.txt
* **Contribution:** I established the formal list of project dependencies. This ensures that any current or future contributor can set up an identical development environment with a single command, which is a best practice for collaborative and open-source projects.

### 3. Challenges Encountered & Solutions

* **Challenge:** The most significant technical challenge was designing an effective method for inter-agent communication and data handoff. In a sequential process, the quality of the final report is entirely dependent on each agent successfully building upon the work of its predecessors. The Devil's Advocate, for example, is useless if it cannot access the reports from the Competitor Analyst and Customer Persona Analyst.
* **Solution:** I resolved this by leveraging the context parameter within the crewai task definition. By explicitly passing the antecedent tasks as context to the dependent tasks (e.g., risk\_critique\_task receives the analysis tasks as context), I established a robust data pipeline. This ensures that each agent has the complete and necessary information before beginning its work, effectively simulating a real-world workflow where team members review each other's documents.

### 4. Goals for the Next Reporting Period (July 1 - July 14, 2025)

The next two weeks will be focused on integrating these foundational components into a fully operational application.

1. **Implement Orchestration Logic (main.py):** The immediate next step is to write the main script that will instantiate the agents and tasks, assemble them into a Crew, and kick off the sequential execution process. This will be the engine that drives the entire analysis.
2. **Develop User Interface (app.py):** I will build a user-friendly front-end using Streamlit. This supports the project's core mission of democratization by providing an accessible interface that allows non-technical users to leverage this powerful AI tool.
3. **Integrate Frontend and Backend:** I will connect the Streamlit UI to the crewai backend, allowing user inputs to trigger the analysis. This will involve passing data from the web form to the backend script as environment variables.
4. **Implement Final Reporting:** The final step will be to ensure the comprehensive markdown report generated by the Lead Strategy Synthesizer is saved to the filesystem and then made available for viewing and download within the Streamlit application.

### 5. Hours Contributed During This Period

* **Total:** 20 hours