

# Solution Design Document for Multi-Agent Competitor Analysis System

---

## 1. Introduction

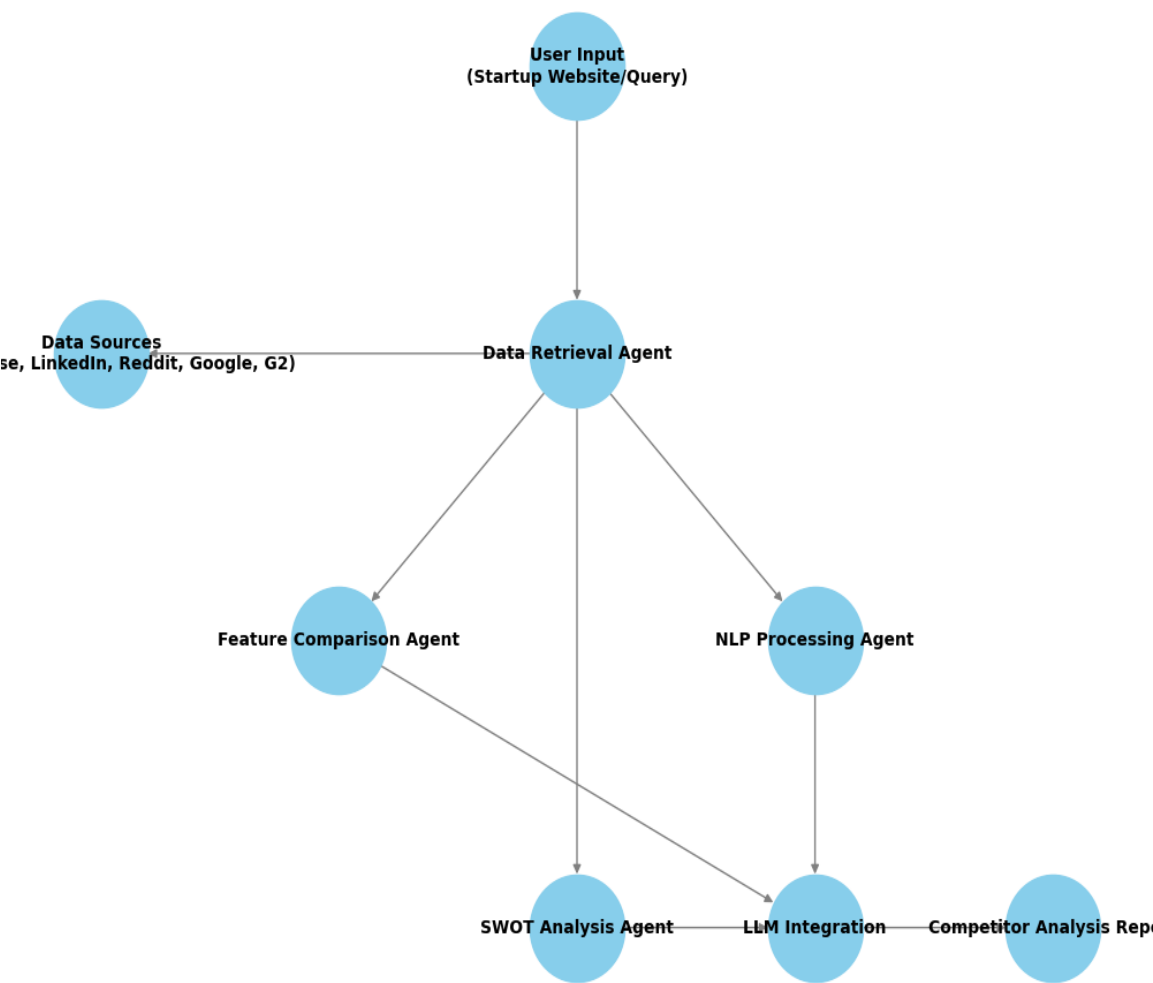
The objective of this project is to develop a multi-agent system that autonomously performs competitor analysis. The system integrates various data sources to gather insights about a given startup or product, processes this data using NLP techniques, and generates a comprehensive competitor analysis report. The system is powered by a Large Language Model (LLM) that synthesizes the data into actionable insights.

## 2. System Architecture

The architecture of the multi-agent competitor analysis system is modular, consisting of multiple agents working in tandem to gather data, process it, and generate insights. Each agent serves a distinct role, as follows:

1. User Input: Provides the input (startup website/product query) for analysis.
2. Data Retrieval Agent: Collects data from external sources such as Crunchbase, LinkedIn, Reddit, Google, and G2.
3. NLP Processing Agent: Processes textual data from the Data Retrieval Agent to extract meaningful insights.
4. Feature Comparison Agent: Compares features of the input with competitors' offerings.
5. SWOT Analysis Agent: Generates SWOT analyses based on the data retrieved and processed.
6. LLM Integration: Synthesizes all data into a coherent report, highlighting key differentiators, strengths, weaknesses, opportunities, and threats.
7. Competitor Analysis Report: The final deliverable generated by the LLM, presenting a comprehensive competitor analysis.

The system architecture is visually represented in the following diagram:



### 3. Justification for LLM, Frameworks, and Data Handling

The system leverages a Large Language Model (LLM), such as GPT-3 or GPT-4, to synthesize and process large amounts of data. LLMs are ideal for generating human-like insights and reports based on structured and unstructured data, which is essential for competitor analysis. They are capable of understanding complex relationships within the data and can generate meaningful summaries, SWOT analyses, and recommendations.

The following frameworks and libraries are recommended for the system implementation:

- TensorFlow / PyTorch : For any additional machine learning tasks.
  - HuggingFace Transformers : For utilizing pre-trained LLMs.
  - SpaCy : For Natural Language Processing tasks like text extraction and entity recognition.
  - Pandas : For data manipulation, cleaning, and handling structured data.
  - Requests / BeautifulSoup : For web scraping and data collection from sources like Crunchbase and LinkedIn.
- Regarding data handling, we need to ensure data accuracy, consistency, and normalization. Missing data should be handled automatically using predefined rules or imputation techniques. Data conflicts should be resolved by determining the most reliable source or aggregating results.

### 4. Alternative Approaches and Trade-offs

While a multi-agent system with LLM integration is a powerful and flexible approach, there are alternative strategies that could be considered:

1. Rule-based System : Instead of using an LLM, a more traditional, rule-based approach could be used to extract and analyze competitor information. This would involve creating predefined templates and patterns for extracting data. However, this approach lacks the flexibility and adaptability of LLMs.
2. Standalone NLP Models : NLP models could be used to analyze and generate competitor reports without the need for an LLM. However, this approach would likely result in less comprehensive reports and may not capture the nuanced insights that an LLM can provide.
3. Manual Reporting : Instead of automating the process, a manual reporting system could be implemented, where human analysts generate the competitor analysis. While this would provide high-quality insights, it would not scale and would involve a lot of manual effort.

Each of these alternative approaches comes with trade-offs regarding flexibility, accuracy, scalability, and complexity. The multi-agent system with LLM integration strikes a balance between these factors, offering a scalable, automated, and high-quality solution.