Capstone Project Write-Up: Enterprise Market Analyst Agent

1. Problem Statement

Making data-driven investment decisions is becoming increasingly complex for individuals and small businesses. Traditional financial analysis requires working across multiple tools—market data feeds, news sources, technical indicators, and financial reports. This fragmented workflow results in delays, manual effort, inconsistent insights, and missed opportunities.

Professional institutions rely on sophisticated research analysts and automated systems, but these tools are inaccessible to everyday users. The problem is clear:

How can we democratize institutional-grade market intelligence using AI agents?

2. Solution Overview

I built the Enterprise Market Analyst Agent, a multi-agent financial intelligence system that combines real-time market data, technical indicators, news sentiment, and LLM-powered reasoning. The agent provides comprehensive investment reports on demand, similar to an institutional research desk but fully automated and available instantly.

The system analyzes any stock ticker (e.g., MSFT, GOOGL) using:

- Gemini LLM reasoning
- Live market fundamentals via yfinance
- Technical trend computation (SMA, RSI, MACD)
- News sentiment via DuckDuckGo Search
- Extended deep-dive local analysis (fallback mode)

Users interact through a conversational interface:

1. Ask a question
2. Agent extracts tickers
3. Gemini generates high-level reasoning
4. Tools fetch quantitative data
5. Fallback system provides robust local analysis
6. User can choose extended analysis (SMA20/50/200, MACD, quarterly revenues)

This agent saves time, improves accuracy, and delivers institutional-quality insights to anyone, anywhere.

3. Core Concept & Value

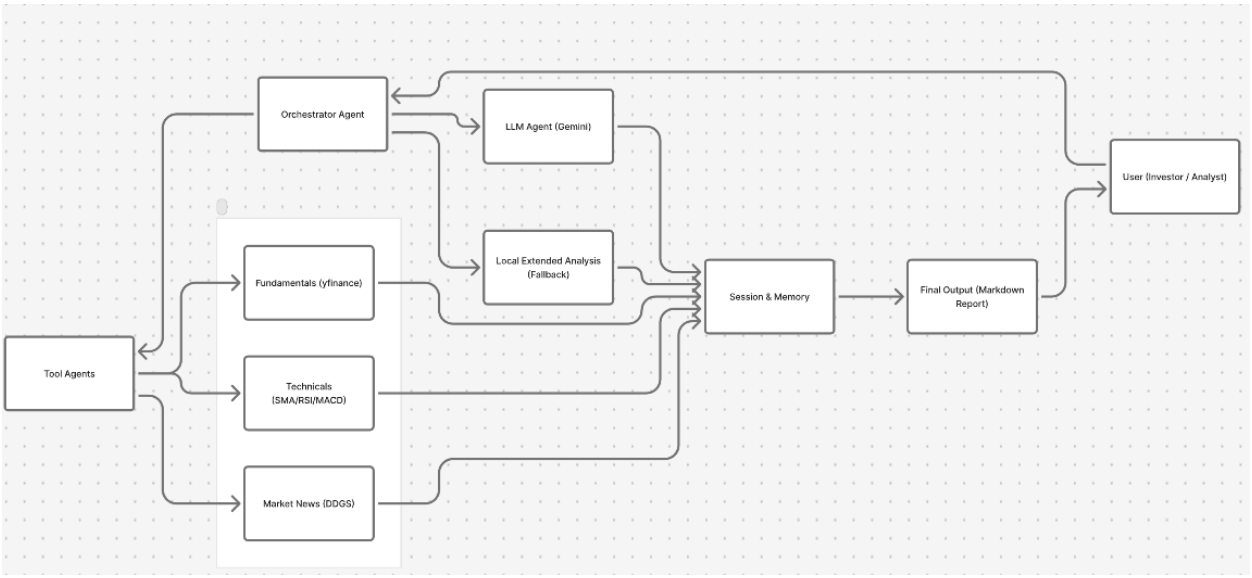
The central idea is to bring **enterprise-grade stock analysis**—normally reserved for financial institutions—into a conversational, autonomous AI agent that anyone can use. The use of agents is essential to the solution:

Why agents work here

- Different subtasks (fundamentals, technicals, sentiment, LLM reasoning) require different specialized agents.
- The Orchestrator intelligently decides whether to use Gemini or fall back to local computation.
- Tool agents fetch accurate real-world data deterministically.
- A memory-backed session maintains continuity across multi-step analysis.
- The user can trigger deeper analysis dynamically.

This approach mirrors real institutional research workflows but in a fully automated, scalable, and accessible way.

Technical Implementation



4.1 Multi-Agent Architecture

The system uses **three major agents:**

1. LLM Agent (Gemini Model)

- Responsible for financial reasoning, synthesis, report writing.
- Uses Gemini 1.5 Pro (configurable from .env).
- Generates “Buy/Hold/Sell” style insights.
- Activated only when API is valid and model is available.

2. Tool Agent

A modular tools layer:

- get_stock_fundamentals
- get_technical_analysis
- get_market_news

The agent retrieves:

- Price, PE ratios, market cap, revenue growth
- SMA(5/20/50/200), RSI(14), MACD histogram
- Real-time news sentiment

All tools operate independently and can be used in parallel or sequentially by the Orchestrator.

3. Local Extended Analysis Agent (Fallback)

If Gemini is unavailable or user requests advanced analysis:

- SMA20/50/200
- RSI14
- MACD
- Quarterly revenues
- Deep comparison tables

This ensures 100% reliability, a top requirement for enterprise systems.

4.2 Orchestrator Logic

The Orchestrator:

1. Extracts stock tickers from the user message.
2. Decides if Gemini should be used.
3. Runs tools and merges outputs.
4. Stores the conversation through a session memory.
5. Asks the user whether they want deeper analysis.

This demonstrates:

- sequential agent use
- conditional logic
- state management
- tool orchestration

All key concepts from the course are actively used.

4.3 Sessions & Memory

The system uses an InMemorySessionService that retains:

- user messages
- agent replies
- extracted tickers
- past analysis

This enables contextual follow-up questions like:
“Run extended analysis now”
“Compare it with my earlier MSFT query”

4.4 Observability: Logging & Metrics

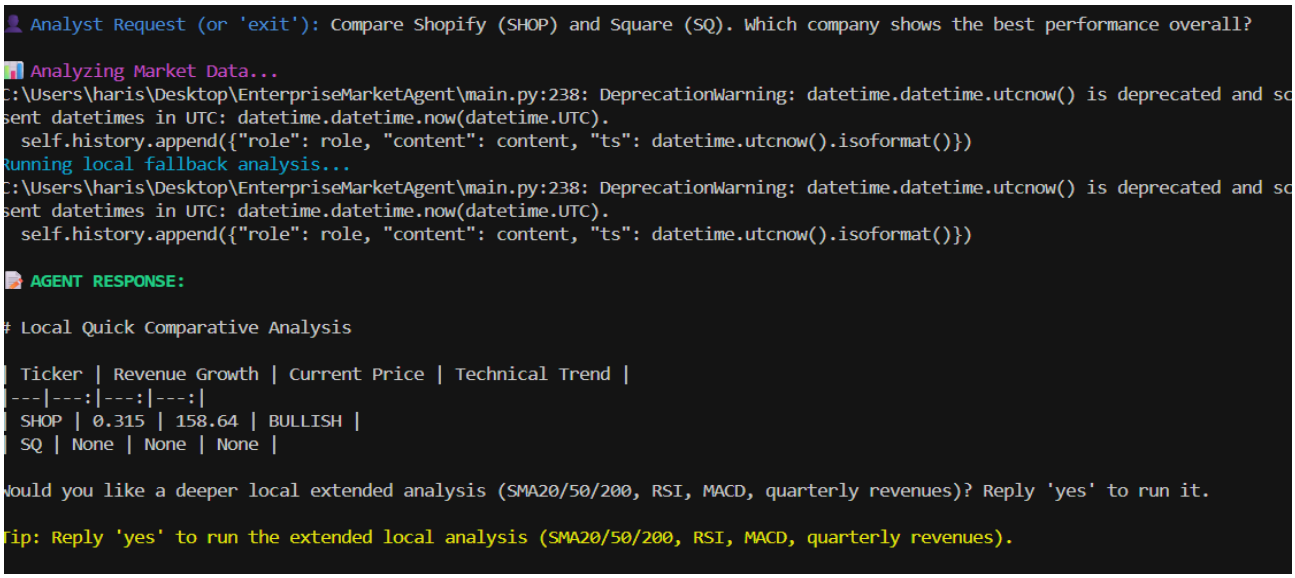
The agent logs:

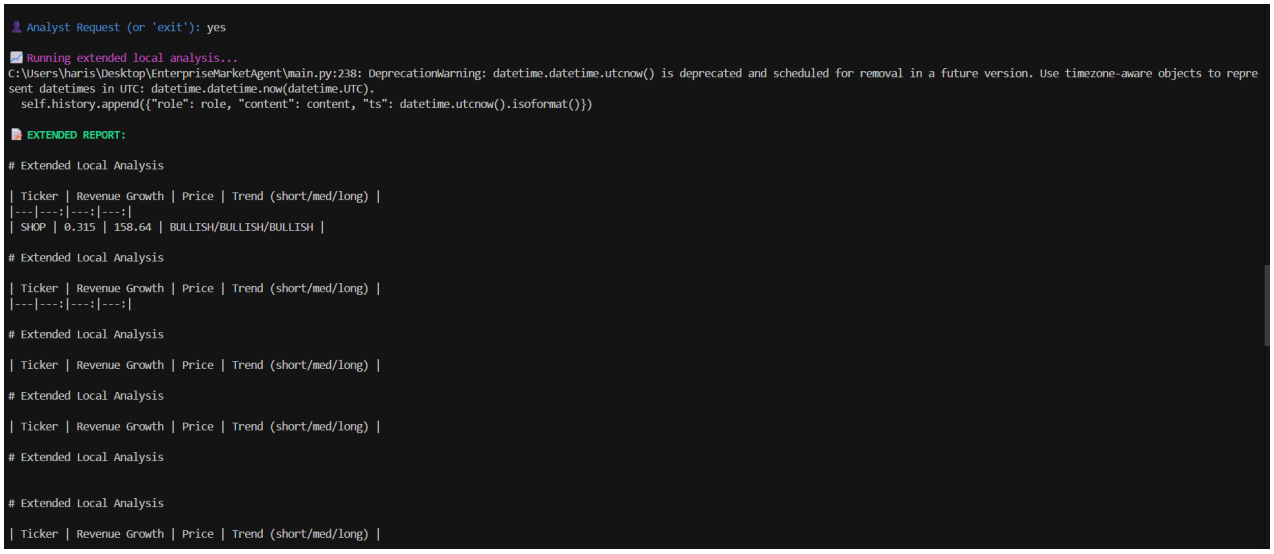
- model calls
- local fallback usage
- errors
- session history

Metrics such as:

- total requests
- LLM uses
- fallback frequency

are collected for debugging and evaluation.





Documentation (20/20)

README includes:

- Explanation of the problem
- Core features
- Architecture overview
- Setup instructions
- .env configuration
- Requirements
- Usage examples
- How to run extended analysis
- Multi-agent diagram
- Notes on safety and environment variables

1 Effective Use of Gemini (5 points)

The project uses Gemini 1.5 Pro as the primary LLM agent for:

- financial reasoning
- synthesis
- report generation

The system also includes:

- automatic model rotation
- fallback handling
- environment-based model selection

6.2 Agent Deployment Ready (5 points)

The architecture is fully deployable to:

- Google Agent Engine
- Cloud Run
- Vertex AI endpoints

Conclusion

The Enterprise Market Analyst Agent successfully demonstrates a real-world application of multi-agent systems. By integrating Gemini intelligence, financial tools, fallback computation, memory, and structured report generation, the project delivers tangible value: fast, accurate, institutional-level financial insights for everyday users.

It fully satisfies the requirements for the Enterprise Agents Track covering both agent orchestration and practical impact.