**📄 Product Requirements Document (PRD)**

**1. Product Overview**

The product is a **Multi-Agent Investment Advisory System** that leverages AI agents orchestrated through CrewAI. It automates the process of:

1. Gathering company-related news.
2. Retrieving financial data & statements.
3. Analyzing data & context.
4. Generating actionable **investment recommendations (Buy/Hold/Sell)**.

The goal is to provide **retail investors, financial advisors, or fintech apps** with reliable AI-driven investment insights in a structured, explainable, and scalable manner.

**2. Objectives**

* Automate investment research and advisory tasks.
* Deliver **explainable outputs** (analysis + recommendation reasoning).
* Support **multi-stock and multi-market scenarios** (e.g., US, India).
* Enable extensibility for **additional agents** (e.g., risk manager, compliance checker).

**3. Key Features**

* **News Agent**: Collects latest company updates via DuckDuckGo search.
* **Data Agent**: Fetches financial data & income statements from Yahoo Finance.
* **Analyst Agent**: Synthesizes company info, news, and financials into a cohesive analysis.
* **Financial Expert Agent**: Generates investment recommendations (Buy/Hold/Sell) with justification.
* **Custom Tools**:
  + Current Stock Price API
  + Company Info API
  + Income Statements API
* **Output**: Structured outputs saved to Analysis.md and Recommendation.md.

**4. Target Users**

* **Retail Investors** seeking advisory assistance.
* **Financial Advisors/Portfolio Managers** needing rapid pre-analysis.
* **Fintech Developers** integrating AI-driven recommendations into apps.

**5. User Stories**

* *As an investor*, I want to input a stock symbol and receive an **explainable recommendation**.
* *As an advisor*, I want consolidated financial news and analysis to **save research time**.
* *As a fintech developer*, I want an API wrapper to **integrate AI advisory into apps**.

**6. Functional Requirements**

* Input: Stock symbol.
* Processing: Multi-agent sequential task execution.
* Output: Analysis + Recommendation (Buy/Hold/Sell).
* Storage: Results saved as Markdown files.
* Error Handling: Graceful fallback if data sources fail.
* Extensibility: Plug-and-play new tools and agents.

**7. Non-Functional Requirements**

* **Scalability**: Handle multiple requests in parallel.
* **Explainability**: Provide reasoning in human-readable format.
* **Security**: API keys managed via .env.
* **Performance**: Response time < 1 minute per stock.

**8. Constraints**

* Reliance on third-party APIs (Yahoo Finance, DuckDuckGo).
* Requires LLM access (e.g., OpenAI API).
* Regional differences in stock ticker formats (e.g., .NS suffix for Indian stocks).

Note: We will be using LangGraph in place of Crew