

# Project Task: AI Crypto Hedge Fund

## Introduction

Objective

Assignment

Project Future

## Part 1: Crypto Hedge Fund Concept Presentation

Hedge Fund Model

Risk Management

Portfolio Management

System Architecture

Evaluation Criteria

## Part 2: Hedge Fund Technical Implementation

Baseline Strategy for a Single Cryptocurrency

Adding AI Agents

Portfolio Management on Historical Data

Dynamic Portfolio Rebalancing

Portfolio Expansion

Evaluation Criteria

## Introduction

### Objective

Develop a concept, technical design, and MVP implementation for an AI agent-based automated cryptocurrency trading and risk management system.

### Assignment

This assignment consists of two parts:

- **Part 1 – Conceptual Presentation:**

A more theoretical section focused on your understanding of cryptocurrencies, trading systems, and AI-agent system architecture.

- **Part 2 – Technical Implementation:**

A hands-on coding task where you will demonstrate your technical skills, including programming, system development, agent-based interaction, backtesting, and

automated trading system validation.

The two parts should be closely integrated and reflect a coherent vision.

## Project Future

This project has the potential to be fully automated, executing trades across multiple centralized exchanges via APIs.

Future enhancements could include:

- **Telegram Integration:** A bot interface to display portfolio statistics, close trades manually, or adjust parameters in real time.
- **Data Source Expansion:** Incorporate external data sources like news feeds and sentiment analysis.

## Part 1: Crypto Hedge Fund Concept Presentation

This section requires a presentation that integrates two domains: **cryptocurrencies** and **AI-driven trading bots**.

In this section, you are required to create a presentation that outlines your vision for a cryptocurrency hedge fund. It should include the following four sections:

- Hedge Fund Model
- Risk Management
- Portfolio Management
- System Architecture

Prepare a professional presentation (up to 10 slides) covering the following four sections (recommended 2–3 slides per section).

### Hedge Fund Model

Describe your vision for the structure of an AI-powered cryptocurrency hedge fund.

Sample guiding questions:

- Can classical technical indicators be used in this system?
- Which types of machine learning algorithms are applicable?
- What is the role of AI in trade decision-making? Which AI agents can be used?
- How should multiple AI agents or ML models interact with each other?

## Risk Management

How can risk management be handled using AI?

Sample guiding questions:

- Which risk metrics will you monitor?
- How can AI help mitigate losses during high-volatility periods?
- What methods for estimating volatility and liquidity would you employ?

## Portfolio Management

Discuss existing portfolio theories, their core principles, and how (or whether) AI should be used to implement them.

Sample guiding questions:

- What is an optimal portfolio?
- Which metrics will guide portfolio optimization?
- How will you manage and rebalance the portfolio over time?

## System Architecture

Design and explain a block diagram of your crypto hedge fund's system architecture, including AI agents.

Requirements:

- Must include modules for: data collection, AI agents for signal generation, order execution, monitoring, risk management, and portfolio management.
- Clearly describe interactions between modules.

## Evaluation Criteria

- All four sections must be addressed.
- Depth of theoretical understanding is expected.
- Each design choice must be justified with clear rationale.
- Presentation design is not the primary criteria, but it should be clear, clean, and not text-heavy.

## Part 2: Hedge Fund Technical Implementation

Develop a working codebase for the crypto hedge fund system. The implementation should cover the following five levels:

1. Baseline strategy for a single cryptocurrency
2. Strategy using econometric models, machine learning models, and AI agents for a single cryptocurrency
3. Portfolio management of a small number of cryptocurrencies based on historical data
4. Portfolio management of a small number of cryptocurrencies with dynamic rebalancing
5. Portfolio management of a large number of cryptocurrencies with dynamic rebalancing

You are expected to conduct backtesting on out-of-sample data — i.e., data not used during training. It is recommended to split the data into train/test sets. All evaluations should include both performance and risk metrics. The tests must be reproducible.

### **Technical Requirements:**

- The project must be hosted on a public GitHub or GitLab repository

- The code should be modular
- The code must be reproducible; use of tools like poetry or uv is expected; Docker is a plus
- All necessary data must be included in the project
- The final solution must be presented in a single, self-contained, and reproducible .ipynb notebook
- The solution should be structured according to the sections outlined in Part 2

## Baseline Strategy for a Single Cryptocurrency

Implement a simple strategy (e.g., moving averages) on one crypto pair such as BTC/USDT. This will serve as your baseline.

### Requirements:

- Report metrics like ROI, Sharpe Ratio, and Drawdown. Are there other useful metrics?
- Explain how this simple strategy can evolve into an AI agent.

## Adding AI Agents

Enhance the previously implemented strategy by integrating AI agents, classical econometric time series models, and machine learning models. For now, apply this to a single trading pair (e.g., BTC/USDT).

Be sure to compare the performance of the different approaches.

Sample questions to address:

- What data will be used as features?
- What is the target variable?
- How will the model be trained and tested?
- How frequently should the model be retrained?

- Which metrics will you use to compare results? Why did you choose these metrics?
- How can you verify that the strategy's performance is not due to random chance?

## Portfolio Management on Historical Data

Add 5–7 popular coins and perform static portfolio management based on historical data (last 12 months).

### Requirements:

- Calculate portfolio metrics defined in Part 1.
- Find the optimal portfolio.
- Explain how results could be applied in real trading.

## Dynamic Portfolio Rebalancing

Design an algorithm to adaptively rebalance the portfolio under changing market conditions.

### Requirements:

- Describe rebalancing logic (e.g., time-based, weight deviation, signals).
- How would you select the most effective rebalancing strategy?

## Portfolio Expansion

Expand your system to handle at least 100 cryptocurrency pairs. Consider how your hedge fund would be structured to operate efficiently when trading such a large number of pairs.

- How will you select trading pairs?
- How will you prioritize agent-generated signals?

- How will risk be managed (e.g., another agent? which metrics)?
- How will the portfolio be managed (AI-driven dynamic rebalancing)?
- How will the system be monitored (beyond trading KPIs)?
- How will you track long-term system quality?
- What fail-safes (“stop signals”) will you implement in case AI agents behave unexpectedly?

## Evaluation Criteria

- Project must run and be reproducible on other machines.
- Must include:
  - Data preparation
  - Model validation
  - Strategy backtesting
  - Result visualization & explanation
- Code quality, architecture, and documentation matter.
  - “Spaghetti code” will be penalized.