AI Trading Marketplace & MVP Development Research

# Introduction

This document summarizes the research and development planning for building an AI trading marketplace and scoring system on 0G Blockchain, integrating Uniswap and Hyperliquid for trading execution.

# Idea Solidification

The first step was clarifying the scope of trading. The AI agents would trade on platforms like Uniswap and Hyperliquid, which are cross-chain DEXes. The marketplace app will allow users to discover, fund, and deploy AI trading agents.

# Why Uniswap + Hyperliquid

Both platforms provide liquidity and APIs for trading across chains. Uniswap is widely adopted in DeFi and supports automation. Hyperliquid, an on-chain orderbook DEX, is optimized for speed and efficiency. Together, they provide diverse environments for AI agents.

# 0G Compute Role

0G Compute allows off-chain heavy AI inference while committing verifiable results on-chain. It can be used to run AI trading logic. It does not natively fetch off-chain data, so agents need oracles or APIs for market data. AI results can be stored or scored via 0G Storage.

# MVP Scaffold

The MVP was planned in phases:  
1. ScoreRegistry.sol (on-chain agent scoring contract).  
2. VaultUniswap.sol (ERC-4626 derived vault for execution).  
3. Backend orchestrator (trading execution, storage uploads).  
4. Integration with 0G Compute & Storage for logging and scoring.

# Minimal Demo Flow

A minimal flow was designed:  
1. Agent decision via 0G Compute.  
2. Mock execution of trade.  
3. Upload logs to 0G Storage.  
4. Compute score (PNL, Sharpe ratio).  
  
This proves the pipeline works end-to-end before integrating real Uniswap execution.

# Example Code Skeleton

A Node.js script was drafted to:  
- Call AI agent on 0G Compute.  
- Simulate trade execution.  
- Upload logs to 0G Storage.  
- Compute and output a score.  
  
Later versions will replace simulation with real Uniswap/Hyperliquid trades.

# Next Steps

The project can continue by:  
1. Replacing mock execution with Uniswap/Hyperliquid trades.  
2. Wiring scores back into ScoreRegistry.sol.  
3. Building a frontend dashboard for users to see agent performance.