THE UNIVERSITY OF HONG KONG

FITE7001 MSc(FTDA) Project

Detailed Project Plan

Title: Exploring the Use of Prediction Markets as Risk Assessment and Hedging Tools in the Financial Industry

Supervisor: Dr. Juergen Harald Rahmel

Members:

CHAN Yik Chun (3036384153)

CHEUNG King Hung (3036383757)

HO Yau Wai Athena (3035800875)

SY Yat Hei (3036383654)

CHENG Kai Kwong (3035687986)

Due Date: 17 Nov 2025

Objective

With the rising popularity of prediction platforms, such as Polymarket, covering a wide range of event types, from politics to economics to socially related activities, it is observed that the prediction market is an early but growing blockchain-based tool that is under-adopted by large institutions. Due to the potential for growth and increased usage of prediction markets, our team would like to investigate whether market-driven, real-time probability estimates from prediction markets can enhance financial risk management by exploring possible values and hedging opportunities. This project aims to examine how simple derivative-like instruments, that is, binary Yes/No contracts, can be used by rentals and institutional actions to take risk, earn returns or hedge tail exposure.

Scope

To achieve the project objectives, the following are potential areas to explore in this project:

1. Gaps and use cases identification where the prediction market outperforms traditional risk transfer methods.
2. Mathematical/ Algorithmic framework development for exploiting pricing differences and arbitrage across event categories, improving profitability and precision.
3. Institutional application exploration, including parametric insurance with probabilistic triggers and commodity hedging informed by market consensus.
4. Prototype operation model design for binary (yes/no) prediction-market derivatives
5. Oracle and data-extraction protocol integration for ensuring accurate mapping of real-world outcomes to on-chain contracts
6. Governance consideration assessment for supporting a secure public launch, promoting regulatory and cybersecurity alignment to enhance stakeholder confidence

Schedule for Implementation (Oct 2025 - May 2026)

-
-

Estimated Number of Learning Hours

Business research side

Phase	Task	Related Scope	Estimated Hours
Phase 1: Research and Planning	**Understand what Polymarket is: Research Polymarket as a decentralized prediction market platform where users bet on real-world event outcomes using cryptocurrency, with markets resolved by oracles.**	**Gaps and use cases identification where the prediction market outperforms traditional risk transfer methods. Prototype operation model design for binary (yes/no) prediction-market derivatives.**	**50 (business team leads research; 2 people x 25 hours each)**
Phase 1: Research and Planning	**Define your project's scope: Decide if you want a full clone (using pre-made scripts) or build from scratch. Clone scripts can launch in 3-6 months and cost $10,000-$100,000, while from scratch offers more customization but takes longer.**	**Prototype operation model design for binary (yes/no) prediction-market derivatives.**	**30 (business team; 2 people x 15 hours each)**
Phase 1: Research and Planning	**Identify target audience: Think about users interested in politics, sports, crypto events, etc., and niche opportunities like underserved markets.**	**Gaps and use cases identification where the prediction market outperforms traditional risk transfer methods. Institutional application exploration, including parametric insurance with probabilistic triggers and commodity hedging informed by market consensus.**	**40 (business team; 2 people x 20 hours each)**

Phase 1: Research and Planning	**Study competitors: Analyze Polymarket, Kalshi, or Augur for features like yes/no betting, liquidity pools, and resolution mechanisms.**	**Gaps and use cases identification where the prediction market outperforms traditional risk transfer methods. Prototype operation model design for binary (yes/no) prediction-market derivatives.**	**60 (mixed; business 30 hours, coding team assists 30 hours)**
Phase 1: Research and Planning	**Check legal requirements: Research regulations in your jurisdiction (e.g., gambling laws, KYC/AML). Consult a lawyer for compliance to avoid penalties.**	**Governance consideration assessment for supporting a secure public launch, promoting regulatory and cybersecurity alignment to enhance stakeholder confidence.**	**70 (business team leads; 2 people x 35 hours each, including consultations)**
Phase 1: Research and Planning	**Estimate costs and timeline: Budget for development ($10k+), audits, marketing. Plan for 3-12 months depending on approach.**	**Governance consideration assessment for supporting a secure public launch, promoting regulatory and cybersecurity alignment to enhance stakeholder confidence.**	**30 (business team; 2 people x 15 hours each)**
Phase 1: Research and Planning	**Choose a development approach: Opt for a clone script for speed or custom build for uniqueness. If using a clone, select a provider with blockchain expertise.**	**Prototype operation model design for binary (yes/no) prediction-market derivatives.**	**30 (mixed; business 15, coding 15)**
Phase 6: Build the Frontend	**Design UI/UX: Create wireframes for pages like homepage, market list, event details, and user dashboard. Focus on clean, intuitive design with dark mode.**	**Prototype operation model design for binary (yes/no) prediction-market derivatives.**	**50 (mixed; coding 35, business 15 for input)**

Phase 7: Integrate and Test the Full System	**Test on testnet: Deploy to Polygon or Ethereum testnet and invite beta users.**	**Prototype operation model design for binary (yes/no) prediction-market derivatives.**	**50 (mixed; coding 35, business 15 for beta coordination)**
Phase 8: Deployment and Launch	**Set up domain: Register a domain and point it to your app.**	**Governance consideration assessment for supporting a secure public launch, promoting regulatory and cybersecurity alignment to enhance stakeholder confidence.**	**15 (business team; 2 people x ~8 hours each)**
Phase 8: Deployment and Launch	**Ensure compliance: Double-check legal aspects and add KYC if required.**	**Governance consideration assessment for supporting a secure public launch, promoting regulatory and cybersecurity alignment to enhance stakeholder confidence.**	**40 (business team; 2 people x 20 hours each)**
Phase 8: Deployment and Launch	**Market the platform: Use social media, influencers, partnerships, and incentives for early users.**	**Institutional application exploration, including parametric insurance with probabilistic triggers and commodity hedging informed by market consensus.**	**60 (business team; 2 people x 30 hours each)**
Phase 8: Deployment and Launch	**Build community: Engage on Discord, Twitter, or forums to grow users and liquidity.**	**Governance consideration assessment for supporting a secure public launch, promoting regulatory and cybersecurity alignment to enhance stakeholder confidence.**	**50 (business team; 2 people x 25 hours each)**

Technical side

Phase	Task	Related Scope	Estimated Hours
Phase 1: Integrate and Test the Full System	**Perform security audits: Hire third-party auditors to check for vulnerabilities in contracts and code.**	**Governance consideration assessment for supporting a secure public launch, promoting regulatory and cybersecurity alignment to enhance stakeholder confidence.**	**90 (coding team coordinates; 3 people x 30 hours each, but mostly external)**
Phase 1: Integrate and Test the Full System	**Fix issues: Address any bugs, scalability problems (e.g., use Layer 2 for speed), or Sybil attack risks.**	**Governance consideration assessment for supporting a secure public launch, promoting regulatory and cybersecurity alignment to enhance stakeholder confidence.**	**60 (coding team; 3 people x 20 hours each)**
Phase 2: Learn Prerequisites (If You're a Beginner)	**Learn blockchain basics: Study how blockchains like Ethereum work, including smart contracts and decentralized apps (dApps).**	**Prototype operation model design for binary (yes/no) prediction-market derivatives.**	**120 (coding team; 3 people x 40 hours each)**
Phase 2: Learn Prerequisites (If You're a Beginner)	**Learn Solidity: Take online tutorials on Solidity (Ethereum's programming language) for writing smart contracts.**	**Prototype operation model design for binary (yes/no) prediction-market derivatives.**	**150 (coding team; 3 people x 50 hours each)**
Phase 2: Learn Prerequisites (If You're a Beginner)	**Learn web development: Get familiar with HTML, CSS, JavaScript, React.js for frontend, and Node.js/Express.js for backend.**	**Prototype operation model design for binary (yes/no) prediction-market derivatives.**	**180 (coding team; 3 people x 60 hours each)**
Phase 2: Learn Prerequisites (If You're a Beginner)	**Understand oracles: Learn about Chainlink or UMA for fetching real-world data to resolve markets.**	**Oracle and data-extraction protocol integration for ensuring accurate mapping of real-world outcomes to on-chain contracts.**	**90 (coding team; 3 people x 30 hours each)**

Phase 2: Learn Prerequisites (If You're a Beginner)	**Learn Web3 tools: Practice with Web3.js or Ethers.js for blockchain interactions, and tools like Hardhat or Truffle for development.**	**Prototype operation model design for binary (yes/no) prediction-market derivatives.**	**120 (coding team; 3 people x 40 hours each)**
Phase 2: Learn Prerequisites (If You're a Beginner)	**Set up a wallet: Create a MetaMask wallet and get testnet ETH from faucets for experimentation.**	**Prototype operation model design for binary (yes/no) prediction-market derivatives.**	**15 (coding team; 3 people x 5 hours each)**
Phase 3: Set Up Development Environment	**Install tools: Download Node.js, Git, and a code editor like VS Code.**	**Prototype operation model design for binary (yes/no) prediction-market derivatives.**	**15 (coding team; 3 people x 5 hours each)**
Phase 3: Set Up Development Environment	**Choose blockchain: Select Ethereum for security or Polygon for low fees and speed.**	**Prototype operation model design for binary (yes/no) prediction-market derivatives.**	**20 (coding team; 3 people x ~7 hours each)**
Phase 3: Set Up Development Environment	**Set up smart contract dev: Install Hardhat or Foundry. Run npx hardhat to initialize a project.**	**Prototype operation model design for binary (yes/no) prediction-market derivatives.**	**25 (coding team; 3 people x ~8 hours each)**
Phase 3: Set Up Development Environment	**Set up frontend: Create a React app with npx create-react-app my-polymarket.**	**Prototype operation model design for binary (yes/no) prediction-market derivatives.**	**25 (coding team; 3 people x ~8 hours each)**
Phase 3: Set Up Development Environment	**Integrate Web3: Add libraries like @web3-react/core or Ethers.js to connect frontend to blockchain.**	**Prototype operation model design for binary (yes/no) prediction-market derivatives.**	**30 (coding team; 3 people x 10 hours each)**
Phase 3: Set Up Development Environment	**Set up testing: Install Mocha/Chai for unit tests and Ganache for local blockchain simulation.**	**Prototype operation model design for binary (yes/no) prediction-market derivatives.**	**25 (coding team; 3 people x ~8 hours each)**

Phase 4: Develop Smart Contracts	**Use frameworks: Leverage Gnosis Conditional Token Framework (CTF) for core logic, as used in Polymarket.**	**Prototype operation model design for binary (yes/no) prediction-market derivatives.** **Mathematical/ Algorithmic framework development for exploiting pricing differences and arbitrage across event categories, improving profitability and precision.**	**50 (coding team; 3 people x ~17 hours each)**
Phase 4: Develop Smart Contracts	**Write market creation contract: Create a Solidity contract where users define events (e.g., description, outcomes, expiration).**	**Prototype operation model design for binary (yes/no) prediction-market derivatives.**	**60 (coding team; 3 people x 20 hours each)**
Phase 4: Develop Smart Contracts	**Add betting logic: Implement functions for buying/selling shares using ERC-1155 tokens for outcomes, collateralized by USDC.**	**Mathematical/ Algorithmic framework development for exploiting pricing differences and arbitrage across event categories, improving profitability and precision.** **Prototype operation model design for binary (yes/no) prediction-market derivatives.**	**70 (coding team; 3 people x ~23 hours each)**
Phase 4: Develop Smart Contracts	**Integrate AMM: Use Automated Market Makers (e.g., Logarithmic Market Scoring Rule) for dynamic pricing and liquidity.**	**Mathematical/ Algorithmic framework development for exploiting pricing differences and arbitrage across event categories, improving profitability and precision.**	**60 (coding team; 3 people x 20 hours each)**

Phase 4: Develop Smart Contracts	**Add resolution: Integrate oracles (e.g., Chainlink) to fetch outcomes and payout winners automatically.**	**Oracle and data-extraction protocol integration for ensuring accurate mapping of real-world outcomes to on-chain contracts.**	**50 (coding team; 3 people x ~17 hours each)**
Phase 4: Develop Smart Contracts	**Implement security: Add reentrancy guards, input validation, and role-based access using OpenZeppelin libraries.**	**Governance consideration assessment for supporting a secure public launch, promoting regulatory and cybersecurity alignment to enhance stakeholder confidence.**	**40 (coding team; 3 people x ~13 hours each)**
Phase 4: Develop Smart Contracts	**Handle disputes: Add a challenge-and-appeal mechanism for contested resolutions.**	**Oracle and data-extraction protocol integration for ensuring accurate mapping of real-world outcomes to on-chain contracts. Governance consideration assessment for supporting a secure public launch, promoting regulatory and cybersecurity alignment to enhance stakeholder confidence.**	**50 (coding team; 3 people x ~17 hours each)**
Phase 4: Develop Smart Contracts	**Test contracts: Write unit tests for each function, deploy to local Ganache, and simulate bets/resolutions.**	**Prototype operation model design for binary (yes/no) prediction-market derivatives.**	**60 (coding team; 3 people x 20 hours each)**
Phase 5: Build the Backend (If Needed)	**Set up server: Use Node.js/Express.js to handle off-chain data like user sessions or API calls.**	**Prototype operation model design for binary (yes/no) prediction-market derivatives.**	**40 (coding team; 3 people x ~13 hours each)**
Phase 5: Build the Backend (If Needed)	**Integrate storage: Use IPFS for decentralized file storage of event descriptions or images.**	**Prototype operation model design for binary (yes/no) prediction-market derivatives.**	**25 (coding team; 3 people x ~8 hours each)**

Phase 5: Build the Backend (If Needed)	**Add order management: Implement a hybrid Central Limit Order Book (CLOB) for off-chain matching and on-chain execution.**	**Mathematical/ Algorithmic framework development for exploiting pricing differences and arbitrage across event categories, improving profitability and precision.**	**50 (coding team; 3 people x ~17 hours each)**
Phase 5: Build the Backend (If Needed)	**Test backend: Ensure it handles high volumes and integrates securely with smart contracts.**	**Governance consideration assessment for supporting a secure public launch, promoting regulatory and cybersecurity alignment to enhance stakeholder confidence.**	**40 (coding team; 3 people x ~13 hours each)**
Phase 6: Build the Frontend	**Implement navigation: Use React Router for pages.**	**Prototype operation model design for binary (yes/no) prediction-market derivatives.**	**25 (coding team; 3 people x ~8 hours each)**
Phase 6: Build the Frontend	**Connect wallet: Add buttons for MetaMask connection using Web3 libraries.**	**Prototype operation model design for binary (yes/no) prediction-market derivatives.**	**25 (coding team; 3 people x ~8 hours each)**
Phase 6: Build the Frontend	**Display markets: Fetch and show active markets, odds, stakes, and rewards in real-time.**	**Prototype operation model design for binary (yes/no) prediction-market derivatives.**	**35 (coding team; 3 people x ~12 hours each)**
Phase 6: Build the Frontend	**Add betting interface: Create forms for placing bets, showing share prices and potential payouts.**	**Prototype operation model design for binary (yes/no) prediction-market derivatives.**	**50 (coding team; 3 people x ~17 hours each)**

Phase 6: Build the Frontend	**Include features: Add leaderboards, analytics, event creation, and secure wallet integration.**	**Institutional application exploration, including parametric insurance with probabilistic triggers and commodity hedging informed by market consensus. Prototype operation model design for binary (yes/no) prediction-market derivatives.**	**60 (coding team; 3 people x 20 hours each)**
Phase 6: Build the Frontend	**Make responsive: Ensure mobile compatibility with responsive design.**	**Prototype operation model design for binary (yes/no) prediction-market derivatives.**	**25 (coding team; 3 people x ~8 hours each)**
Phase 6: Build the Frontend	**Add fiat on-ramps: Integrate options for buying crypto with fiat.**	**Prototype operation model design for binary (yes/no) prediction-market derivatives.**	**35 (coding team; 3 people x ~12 hours each)**
Phase 6: Build the Frontend	**Test frontend: Check for bugs in betting flows and real-time updates.**	**Prototype operation model design for binary (yes/no) prediction-market derivatives.**	**35 (coding team; 3 people x ~12 hours each)**
Phase 7: Integrate and Test the Full System	**Connect components: Link frontend to smart contracts via Ethers.js and backend APIs.**	**Prototype operation model design for binary (yes/no) prediction-market derivatives.**	**50 (coding team; 3 people x ~17 hours each)**
Phase 7: Integrate and Test the Full System	**Add liquidity incentives: Implement rewards for market makers to encourage activity.**	**Mathematical/ Algorithmic framework development for exploiting pricing differences and arbitrage across event categories, improving profitability and precision.**	**40 (coding team; 3 people x ~13 hours each)**

Phase 7: Integrate and Test the Full System	**Conduct unit tests: Test each module individually.**	**Prototype operation model design for binary (yes/no) prediction-market derivatives.**	**40 (coding team; 3 people x ~13 hours each)**
Phase 7: Integrate and Test the Full System	**Run integration tests: Simulate full user journeys like creating a market, betting, and resolving.**	**Prototype operation model design for binary (yes/no) prediction-market derivatives.**	**50 (coding team; 3 people x ~17 hours each)**
Phase 8: Deployment and Launch	**Deploy smart contracts: Use tools like Hardhat to deploy to mainnet (e.g., Polygon).**	**Governance consideration assessment for supporting a secure public launch, promoting regulatory and cybersecurity alignment to enhance stakeholder confidence.**	**25 (coding team; 3 people x ~8 hours each)**
Phase 8: Deployment and Launch	**Launch frontend: Host on decentralized storage like IPFS or a server like Vercel.**	**Governance consideration assessment for supporting a secure public launch, promoting regulatory and cybersecurity alignment to enhance stakeholder confidence.**	**20 (coding team; 3 people x ~7 hours each)**