# Unified Multi-Bet Research Plan

This document outlines the research and rationale behind the key technical and architectural decisions for the project.

### **1. Core Architectural Principles**

* **Pluggable Model Architecture:** The system is designed with a strict separation between the CoreEngine and the predictive models. This is achieved via the BasePredictiveModel abstract base class.
  + **Rationale:** This allows for independent development, testing, and updating of models without affecting the core business logic. New models (e.g., for a new sport) can be "plugged in" as long as they adhere to the interface, ensuring scalability and maintainability.
* **Two-Layer Feature Store:** The data infrastructure is split into an Online and Offline store.
  + **Online Store (Redis):** Optimized for low-latency reads of feature vectors needed for real-time predictions.
  + **Offline Store (Google BigQuery):** Optimized for large-scale storage and complex queries on historical data, used for model training, backtesting, and analytics.
  + **Rationale:** This separation addresses two different operational needs. A single database would be a compromise, either too slow for live predictions or too expensive and inefficient for large-scale analytics.

### **2. Predictive Model Selection**

The choice of model is tailored to the statistical nature of the betting market being predicted.

* **Racing (Multi-Runner Events): Conditional Logit Model**
  + **Rationale:** Simple binary (win/loss) models are statistically inappropriate for races with multiple potential winners. A Conditional Logit model correctly frames the problem as a choice model, where the probability of one horse winning is dependent on the characteristics of all other horses in the race. This is the industry-standard econometric approach.
* **Team Sports (Head-to-Head): CatBoost Classifier**
  + **Rationale:** Sports data is rich with categorical features (e.g., venue, day of the week, team names). CatBoost is a gradient boosting algorithm specifically designed to handle categorical features natively and effectively, often outperforming other models without extensive manual feature encoding.
* **Player Props (Count-Based): Negative Binomial Regression**
  + **Rationale:** Predicting discrete counts (e.g., number of tries, goals) is not a classification problem. Poisson or Negative Binomial regression models are designed to predict count outcomes. The Negative Binomial model is chosen over the simpler Poisson because it can better handle the overdispersion (high variance) commonly found in sports performance data.
* **Player Props (Time-Series): LSTM Network**
  + **Rationale:** To capture a player's form or momentum, simple rolling averages are insufficient. A Long Short-Term Memory (LSTM) network, a type of recurrent neural network, is specifically designed to learn from sequences of data. It can identify complex temporal patterns in a player's recent performances, providing a more sophisticated prediction of their upcoming performance.

### **3. Advanced Quantitative Logic**

* **Correlation Modeling (Student's t-Copula):**
  + **Rationale:** Simply multiplying the probabilities of two correlated events in a Same-Game Multi is mathematically incorrect and leads to underpriced (or overpriced) bets. A copula function separates the marginal probabilities of the events from their dependence structure. A Student's t-Copula is chosen specifically because it is better at capturing "tail dependence"—the tendency for extreme events to occur together—which is a common feature in financial and sports markets.
* **Explainability (SHAP):**
  + **Rationale:** "Black box" models are a significant risk. SHAP (SHapley Additive exPlanations) is a game theory-based approach to explain the output of any machine learning model. Integrating SHAP provides transparency into why a model made a specific prediction. This is crucial for debugging, building trust, and using the model's own "confidence" as an input for the dynamic staking engine.