**PHASE 1 PILOT (Foundation, not throwaway)**

## Goal

Build a production-grade, reproducible data pipeline + backtest harness + logging framework that future ML models can plug into without rework. Baseline probability engine is acceptable for Phase 1, but the framework must be generic.

## Scope (locked)

- Strategy definition: **market + selection rule + odds provider + odds timestamp rule + staking assumption**.
- Markets: 1 market only (example: Double Chance or Over 1.5).
- Leagues: 1 league for the pilot (optional stretch: run same pipeline on a second league as validation).
- Backtest window: last 2 full seasons if data allows (you propose exact dates).

## Data sources

- Odds: The Odds API (historical snapshot calls).
- Fixtures and match metadata (for match pages and later ML features): API-Football (or the agreed football API).
- The pilot must implement reliable joining logic between the two sources (teams + kickoff window) and store a mapping table.

## Odds timing rule (must be explicit)

- Example rule: “Use the latest snapshot at T-6h before kickoff. If missing, use nearest snapshot before T-6h within X hours, otherwise skip fixture.”
- Define what “closing odds” means if you use kickoff timestamp as proxy.

## Hard rules (integrity)

- Strict walk-forward only, no random splits, no leakage.
- Missing odds handling specified in writing.
- Predictions are logged before kickoff and cannot be edited.

## Deliverables (must be included)

1. **ETL + storage**

    - Pull fixtures/results and pull odds snapshots with timestamps.
    - Database schema (or parquet files) with clear keys and indexing.
    - Documented “as-of” rule for what features are allowed pre-match and what is banned.

2. **Backtest harness**
    - Walk-forward backtest script/notebook that reruns end-to-end.
    - Metrics: ROI, hit rate, number of bets, average odds, max drawdown, longest losing streak, profit curve by month (CLV proxy if available).
3. **Audit logs (bet level)**
    - fixture_id, kickoff_utc, league
    - market, selection, odds_at_pick, odds_timestamp
    - model_prob, implied_prob, edge
    - stake, result, pnl
    - closing_odds and CLV if possible.
4. **Integration output for your site**
    - JSON output schema for picks per match (exact fields agreed).
    - CSV prediction log file.
    - Minimal API endpoint spec or export file format that WordPress/Laravel can ingest.
5. **Reproducibility package**
    - requirements.txt or poetry.lock
    - cached dataset export for the backtest window (csv/parquet)
    - one-command rerun instructions that reproduce identical report output.

**Acceptance criteria (definition of done)**

- I can run the backtest on my machine and reproduce the same metrics from the provided scripts + dataset export (or rebuild script with fixed endpoints/params).
- Walk-forward splits are date-based and documented.

- All picks include odds_timestamp and the agreed odds timing rule is followed.

**Explicitly out of scope for Phase 1**

- Live betting execution, auto-betting, deployment to production, and “guaranteed ROI”. Keep focus on foundation.

And here is **Full project scope (like the “dream model” broken into phases)**

This is why Phase 1 must be built clean.

**Phase 1 (Pilot, foundation)**

- Two-API ingestion + mapping + storage
- Backtest harness + reproducibility + logs
- Baseline probability engine (explainable) as a sanity benchmark

**Phase 2 (First real ML probability engine)**

- Feature engineering from stored history (Elo, rolling stats, home/away splits, rest days)
- Critical: features must be “as known at prediction time” to avoid leakage.
- Model training + walk-forward evaluation inside the same harness
- Compare ML vs baseline apples-to-apples

**Phase 3 (Productization)**

- Daily prediction jobs, monitoring alerts (missing data, odds gaps)
- Prediction registry and versioning (model version, data version)
- Shadow mode logging (publish later, log before kickoff)

**Phase 4 (Website integration and premium UX)**

- Match page pulls: fixtures, H2H, standings, line movement charts, model explanation
- Premium gating based on your WP + Laravel setup
- Public results tracker (ROI, CLV, transparency pages)