Milestone 1:

Data Collection and Initial Understanding

The objective of Milestone 1 was to assess whether the problem of player market valuation could be addressed using **reliable, scalable, and publicly available data**, while identifying practical constraints in real-world data acquisition.

After evaluating multiple sources referenced in the *AI Transfer IQ* documentation, the dataset titled **"5.7M+ Records – Most Comprehensive Football Dataset"** from Kaggle was selected and approved by the mentor. This dataset was chosen for its breadth, structural consistency, and pre-cleaned format, significantly reducing risks associated with noisy joins, missing identifiers, or corrupted records.

The dataset provides extensive coverage of player performance metrics, player profiles, injury history, and historical market values. Crucially, it includes **stable linking attributes** such as `player_id` and `player_name` , enabling reliable integration across tables without reliance on approximate matching.

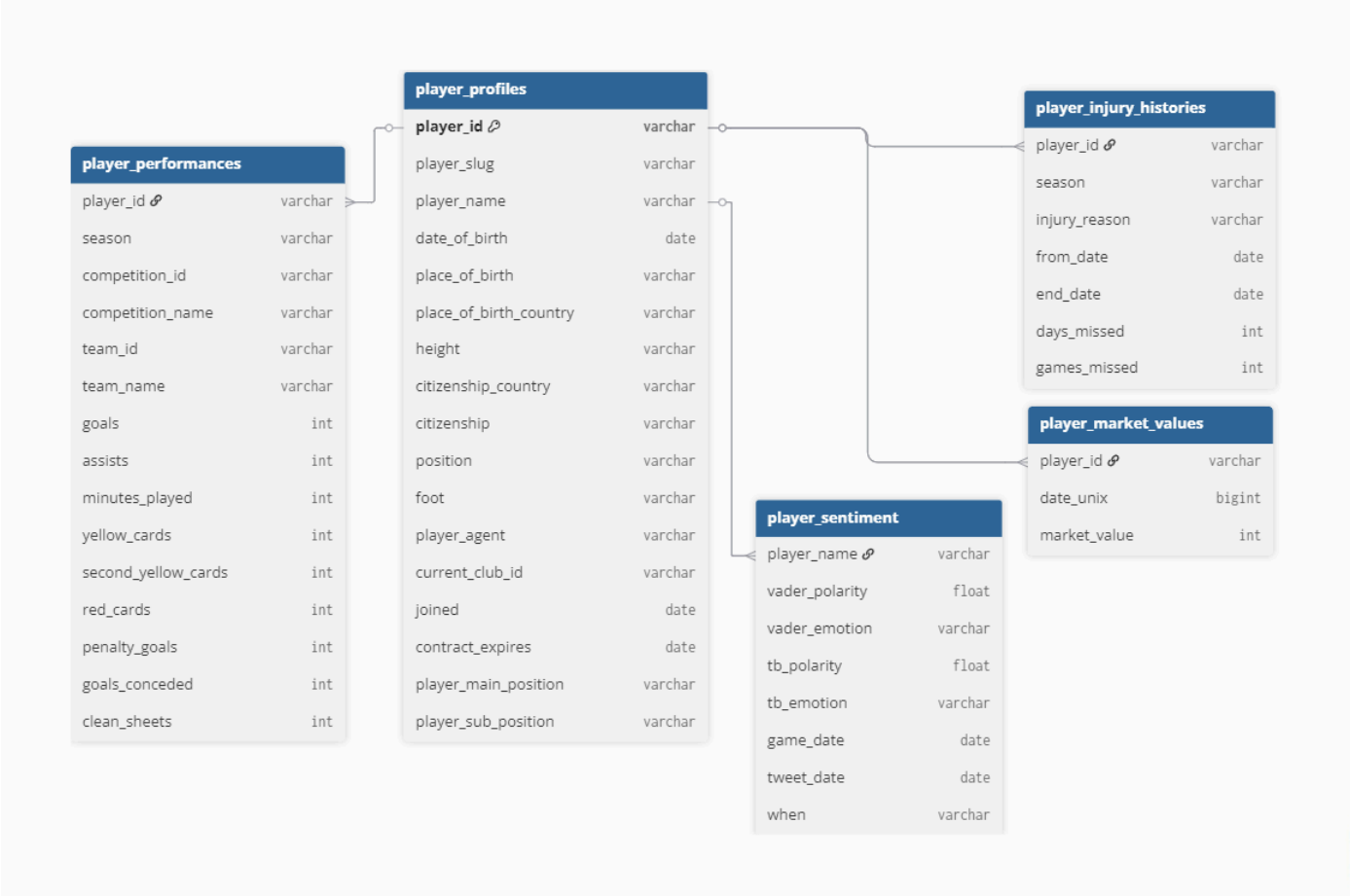
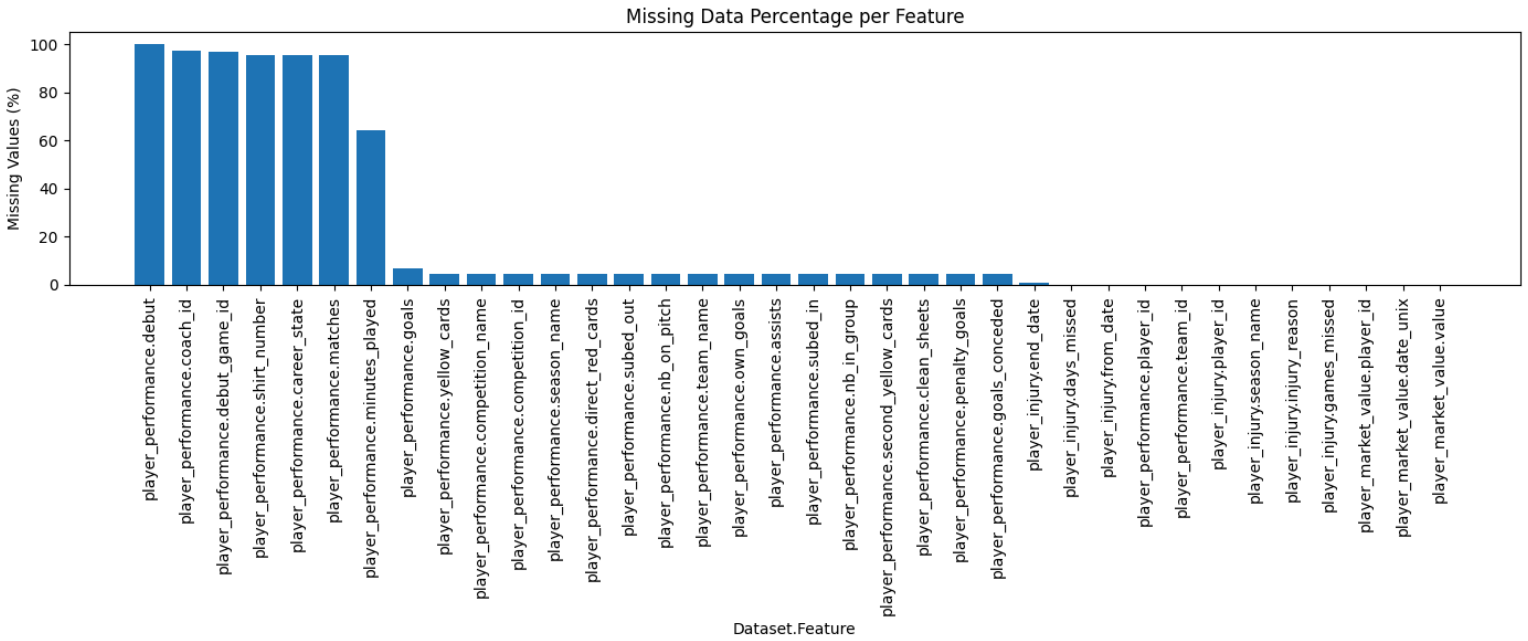
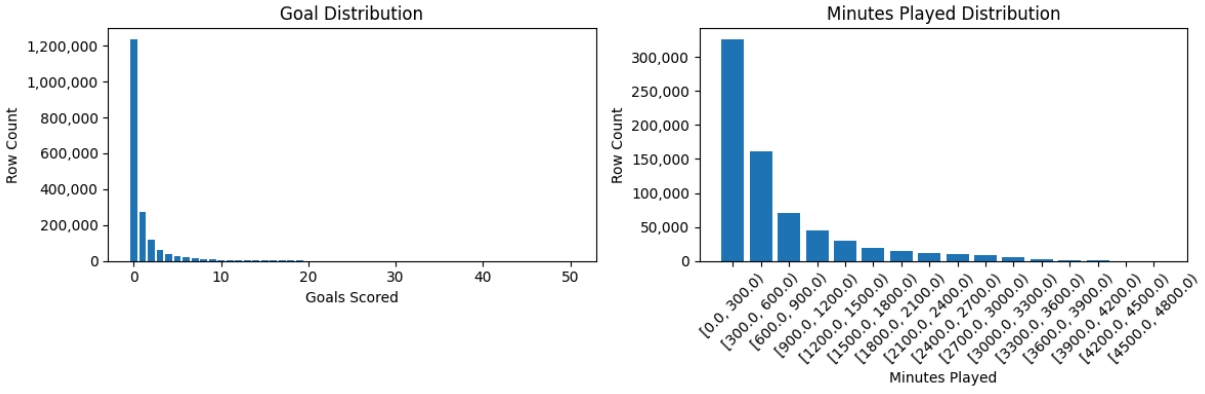


Fig. Entity-Relationship Overview



Initial exploration involved reviewing each file to understand variable definitions, data types, and missing-value distributions (see Missing Data Percentage Histogram per feature). Performance and valuation data were largely complete, while sentiment related data could not be reliably sourced through live social media APIs due to access restrictions, particularly from Twitter.



Alternative platforms such as YouTube and Reddit were explored, but limitations in accessibility, scalability, and consistency made them unsuitable for structured sentiment extraction. These constraints informed the decision to use a mentor- provided sentiment dataset in later stages.

Conclusion

Milestone 1 confirmed the availability of a **clean, comprehensive, and well-linked dataset** capable of supporting meaningful player-level analysis. It also clarified realistic data acquisition boundaries, allowing subsequent milestones to focus on integration, feature reasoning, and modeling rather than fundamental data quality issues.

Milestone 2:

Data Cleaning and Integration

The objective of Milestone 2 was to **integrate multiple player-related datasets into a unified, player-level representation**, while avoiding distortions caused by sparse or session-level records.

The following datasets were used:

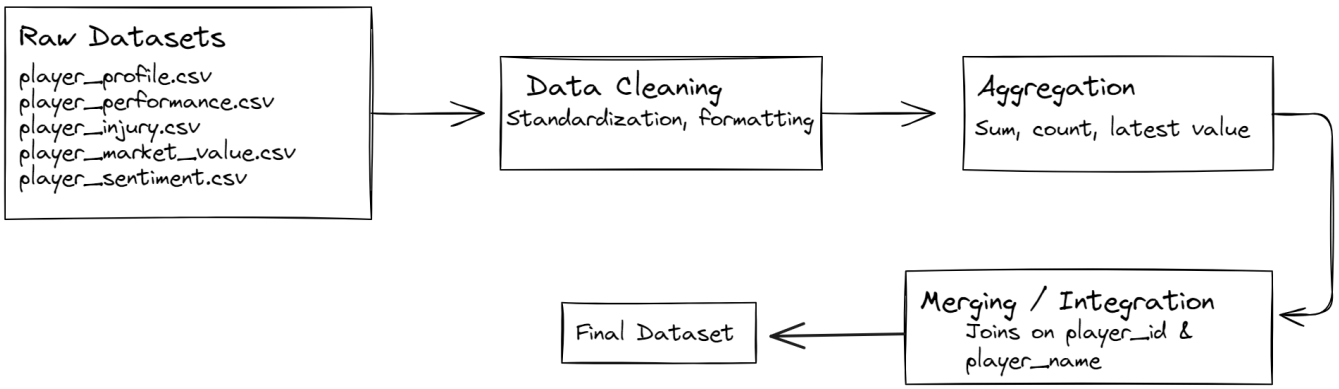
- `player_performance.csv`
- `player_injury.csv`
- `player_market_value.csv`
- `player_profile.csv`

Together, these datasets capture **on-field contribution, availability, financial valuation, and player identity**, which are central to the project objective.

Initial Cleaning Decisions

Because the selected Kaggle dataset was already well-structured, only minimal preprocessing was performed. Text-based fields were standardized to lowercase to ensure consistency. No anomaly correction or imputation was applied at this stage, as the focus was on **structural integration rather than feature correction**.

The diagram below illustrates the workflow for merging raw datasets.



Data Integration & Aggregation Workflow

Aggregation and Integration Strategy

Directly merging session-level records introduced significant sparsity, as players do not participate uniformly across seasons or competitions. To mitigate this, session-level attributes were **aggregated at the player level**:

- Minutes played were summed to form `total_minutes_played`
- Goals, assists, substitutions, and disciplinary actions were summed to represent cumulative contribution
- Injury records were aggregated into total injury counts to reflect long-term availability

This approach ensures that each player is represented by **stable, interpretable signals**, reducing noise and avoiding misleading zero-valued records caused by non-participation.

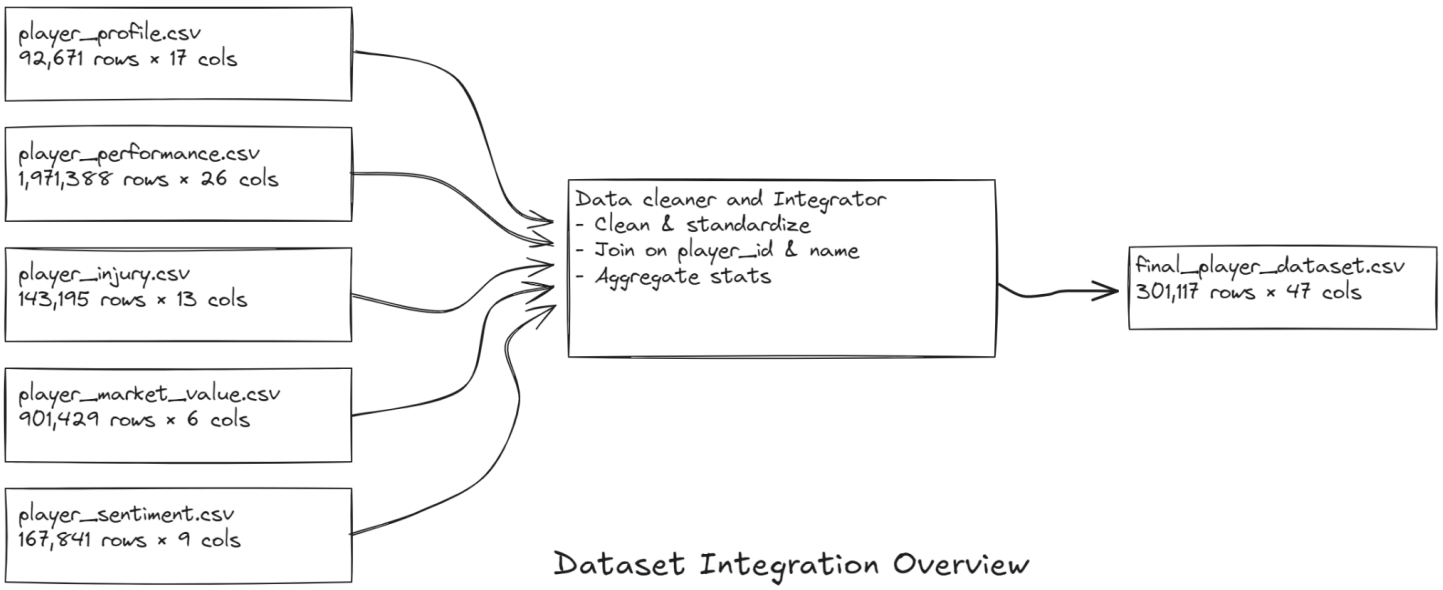
Market Value Representation

Market value was retained in its **most recent available form**, aligning the dataset with the project's objective of predicting current player valuation rather than modeling historical price dynamics.

Sentiment Data Integration

Sentiment data was later merged using `player_name` , the only shared attribute between datasets. While name-based joins introduce ambiguity, this limitation was treated as a **controlled risk** due to the absence of stable identifiers in sentiment Sources.

The figure given below compares dataset sizes before and after integration, highlighting the reduction from multiple session-level datasets to a unified player-level dataset.



Dataset Integration Overview

Conclusion

Milestone 2 produced a **clean, player-centric dataset** by consolidating session-level data into cumulative representations. Although temporal granularity was reduced, the resulting dataset offered improved consistency and interpretability, forming a reliable foundation for feature refinement.

Milestone 3:

> *Milestone 3 is divided in 2 parts, Part A and Part*

Part A:

Feature Refinement and Structured Preprocessing

The objective of Milestone 3 was to convert the integrated player-level dataset into a **modeling-ready representation** by enforcing feature validity, removing non-informative attributes, correcting inconsistencies, and standardizing preprocessing logic. Unlike Milestone 2, which focused on structural integration, this milestone emphasized **feature semantics and reliability**.

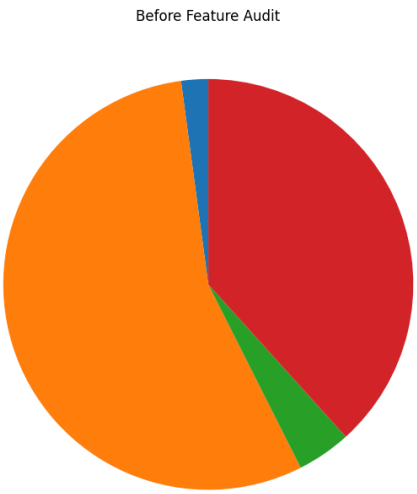
Comprehensive Feature Audit

Following integration, a full audit of all columns was performed to determine whether each feature encoded meaningful information about **player behavior, contribution, availability, or physical capability**.

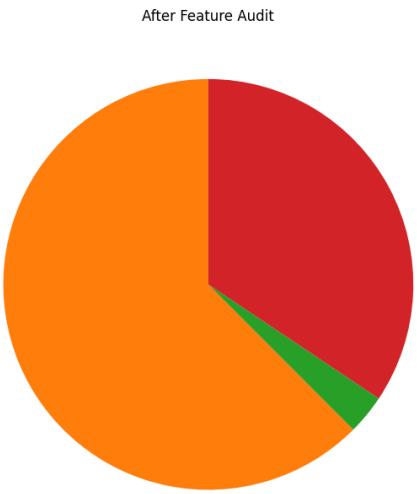
The audit classified features into three categories:

1. **Predictive features** – directly related to performance or availability
2. **Contextual identifiers** – labels without intrinsic predictive meaning
3. **Artifacts** – URLs, slugs, images, or metadata

All features falling into categories (2) and (3) were removed prior to modeling.



Data Type	Before (%)	After (%)
bool	2.1	0.0
float64	55.3	62.5
int64	4.3	3.1
object	38.3	34.4

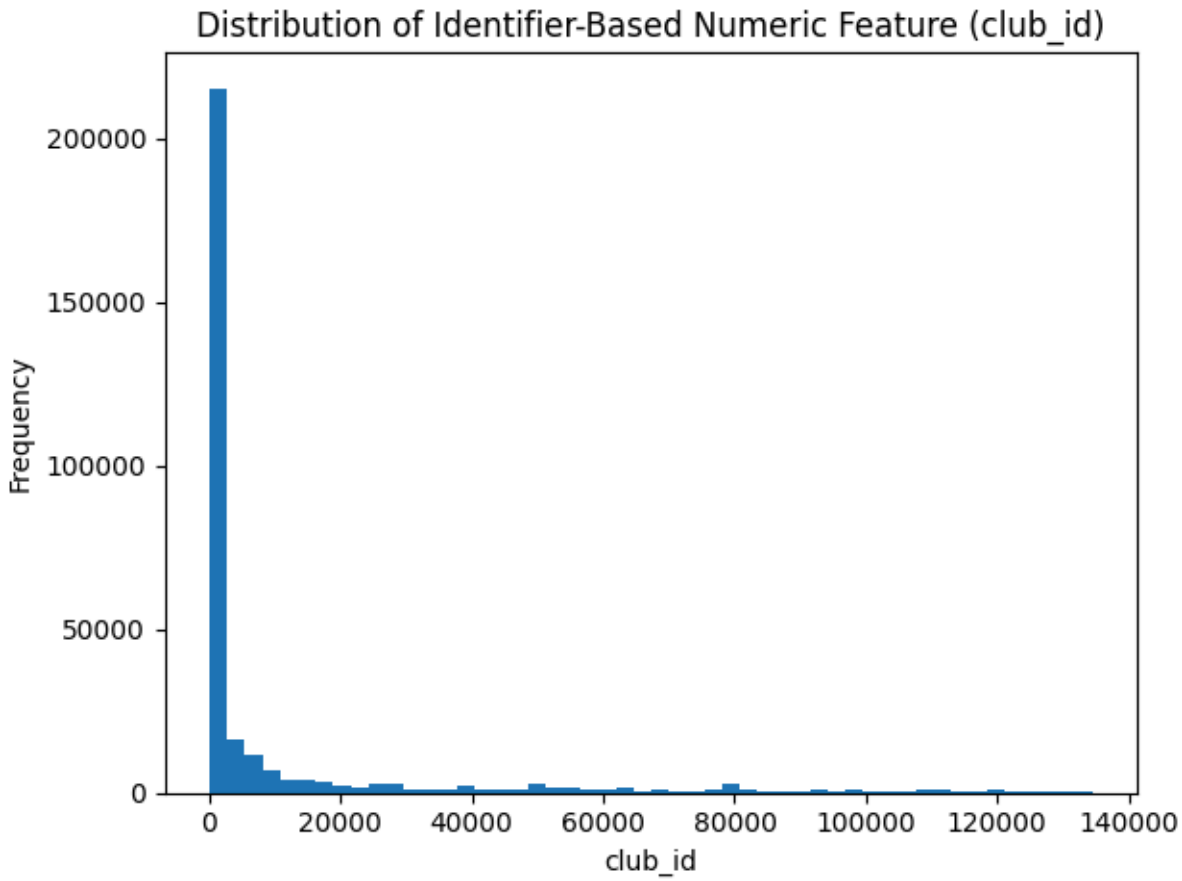


Removal of Identifier-Based Numeric Features

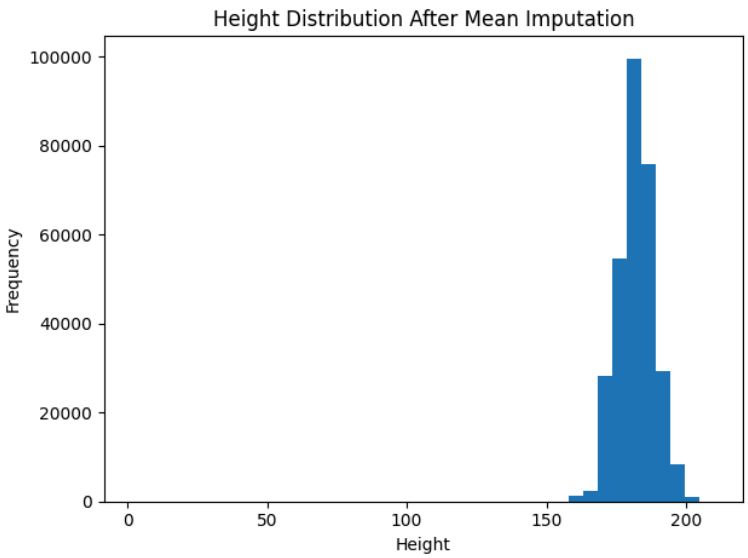
Several columns were numeric in form but categorical in nature (e.g., `club_id`). Despite being integers, these values represent **arbitrary identifiers rather than**

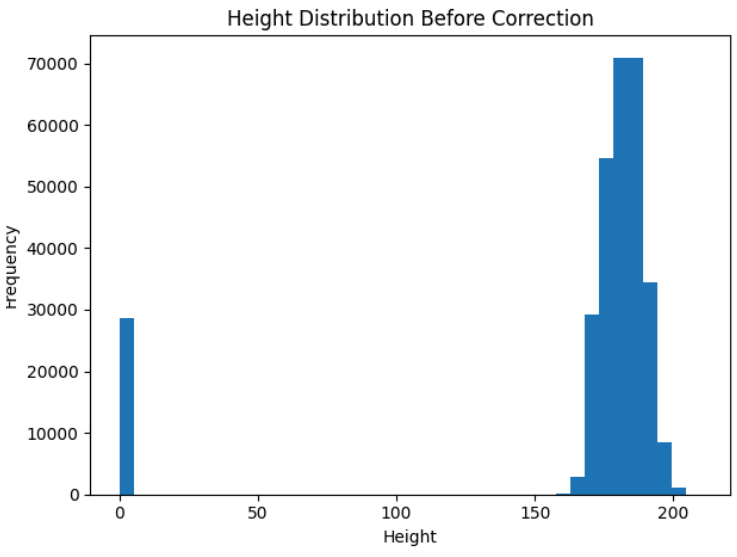
quantitative magnitudes.

Including such features without appropriate encoding would cause models to infer false ordinal or distance-based relationships (e.g., assuming club 10 is “greater” than club 3). Because no complementary data was available to encode club quality or league context, these identifiers were removed to prevent artificial correlations and noise.



Validation and Correction of Physical Attributes





The `height` feature was retained due to its relevance to positional suitability and physical presence. However, inspection revealed invalid values recorded as `0`, which are physically impossible.

Rather than discarding the feature or treating zero as a legitimate category, these values were replaced with the **population mean height**. This approach preserves the feature's distribution while avoiding unrealistic extremes or unjustified assumptions.

Target Variable Semantics and Missing Values

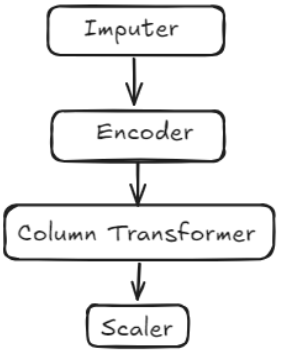
Missing values in the target variable (`current_market_value`) were explicitly assigned a value of `0` . This decision reflects **domain knowledge**: players without recorded market values are typically inactive, retired, or otherwise absent from the active transfer market.

Statistical imputation (e.g., mean or median) would incorrectly inflate the perceived value of such players and distort valuation patterns. Assigning zero therefore preserves semantic correctness and downstream interpretability.

Pipeline-Based Preprocessing Strategy

A unified preprocessing pipeline was implemented to manage all transformations, including:

- Imputation
- Encoding of categorical features
- Column Transformer
- Feature scaling



Using a pipeline ensured that transformations were **applied consistently across training and evaluation splits**, eliminated preprocessing drift, and prevented unintended information leakage. This design choice also supports reproducibility and deployment compatibility.

Outcome of Milestone 3 – Part A

By the end of Milestone 3, the dataset had been transformed into a **clean, semantically valid, and modeling-ready feature space**. All retained features were justified by domain relevance, corrected for anomalies, and processed through a reproducible pipeline, ensuring that downstream model behavior would reflect genuine player characteristics rather than data artifacts.

Part B:

Model-Based Validation of Feature Space

The objective of Milestone 3 part B was to **validate the quality and structure of the engineered feature space** through systematic modeling experiments. At this stage, models were not treated as final predictors, but as **diagnostic instruments** to assess whether the refined features capture meaningful relationships with player market value.

This milestone focused on answering two key questions:

1. Do the engineered features explain player market value under simple assumptions?
2. Do more flexible models reveal non-linear structure and interactions within the data?

Baseline Linear Modeling and Structural Assessment

A Linear Regression model was trained as an initial baseline to test whether player market value could be explained as a linear combination of features. The resulting **low R^2 score (0.29)** indicated that linear assumptions are insufficient.

Rather than being viewed as a failure, this result provided an important structural insight: **player market value depends on conditional relationships, thresholds, and interactions**, rather than independent additive effects. This justified the exploration of non-linear models in subsequent steps.

Exploration of Feature Interactions via Polynomial Expansion

To explicitly test for interaction effects, polynomial feature expansion was applied to the dataset. This transformation enabled the model to represent pairwise and higher-order feature interactions that may jointly influence market value.

However, polynomial expansion dramatically increased feature dimensionality, introducing redundancy and reducing interpretability. To mitigate this, **LassoCV (L1-regularized regression with cross-validation)** was applied to suppress weak or redundant interaction terms.

While this approach confirmed the presence of interaction effects, the **performance gains were marginal** relative to the increase in complexity and computational cost. As a result, explicitly derived polynomial features were not retained. Instead, interaction learning was deferred to models that capture such relationships implicitly.

Evaluation of Non-Linear Models

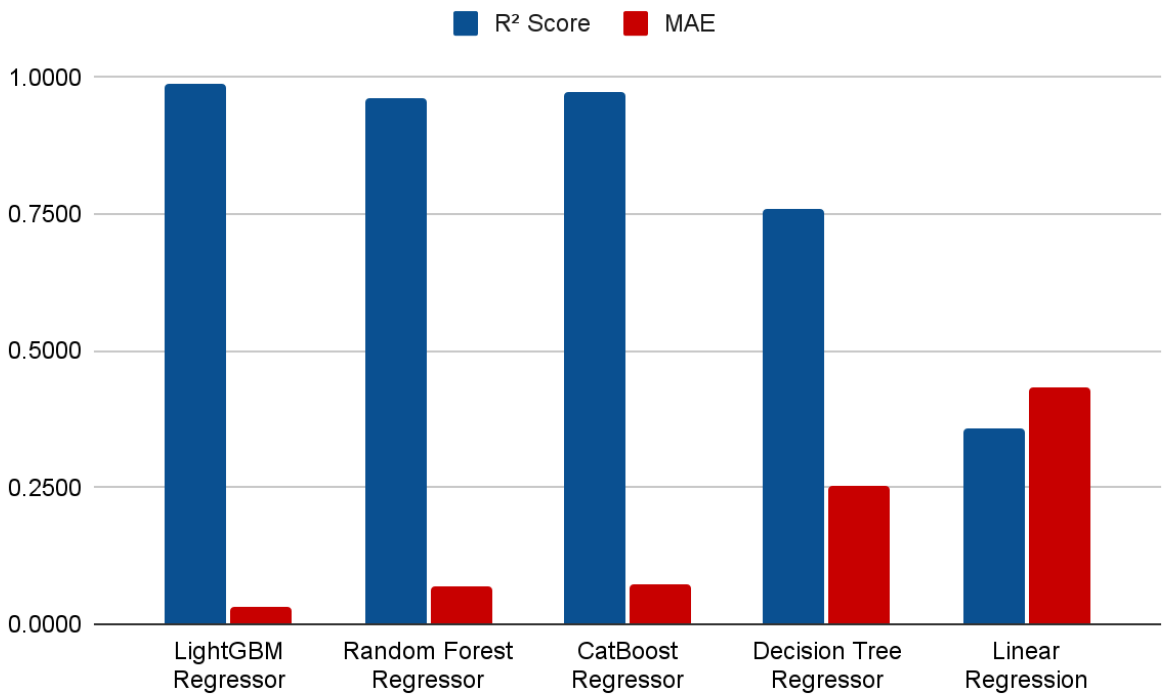
Following the interaction analysis, several non-linear models were trained to further probe the structure of the data:

- Decision Tree Regressor
- Random Forest Regressor
- LightGBM Regressor
- CatBoost Regressor

Model performance was evaluated using **R^2 score** to assess explanatory power.

Model Performance Summary

Model	R^2 Score
Linear Regression	0.36
Decision Tree Regressor	0.753
Random Forest Regressor	0.976
LightGBM Regressor	0.986
CatBoost Regressor	0.992



The substantial performance gains achieved by tree-based and boosting models confirm that the feature space contains **rich non-linear structure**. These models naturally capture hierarchical rules, threshold effects, and conditional dependencies—patterns that align with real-world drivers of player valuation such as minimum participation levels, sustained performance, and availability constraints.

Interpretation of High Performance Scores

Despite achieving very high R^2 values, these results were interpreted **cautiously**. At this stage, models were not evaluated as production-ready predictors, but as validation tools for feature representation.

High performance was therefore treated as evidence that:

- The engineered features encode meaningful information
- Feature preprocessing decisions were structurally sound
- Aggregation and anomaly correction did not distort key signals

Potential risks such as overfitting or target leakage were mitigated through pipeline-based preprocessing and controlled train–test separation established in earlier milestones.

Outcome of Milestone 3 – Part B

Milestone 4 demonstrated that player market value is governed by **non-linear and interaction-driven relationships** that cannot be captured by linear models alone. The results validated the refined feature space developed in Milestone 3 and confirmed that tree-based and boosting models are appropriate tools for downstream analysis.

Importantly, this milestone established the foundation for **feature importance– driven refinement**, which was performed in Milestone 5 through cross-model validation of top-ranked features.

Milestone 4:

Feature Importance–Driven Refinement and Deployment Readiness

The objective of Milestone 5 was to **refine the modeling pipeline by identifying a compact, high-signal feature set**, validate its effectiveness through re-training, and extend the project toward **practical deployment**. Unlike previous milestones, which focused on data preparation and structural validation, this milestone emphasized **feature stability, interpretability, and real-world usability**.

Rationale for Feature Importance–Based Refinement

Following Milestone 4, multiple non-linear models demonstrated strong predictive performance, indicating that the engineered feature space contains rich structural information. However, retaining the full feature set increases redundancy, computational cost, and overfitting risk.

Feature importance analysis was therefore introduced as a **controlled refinement mechanism**, not as an automated selection process. The objective was to identify features that consistently influence predictions **across different model families**, rather than optimizing for any single model's internal metric.

Cross-Model Validation of Top-Ranked Features

Feature importance rankings were independently extracted from the following four models:

- Decision Tree Regressor
- Random Forest Regressor
- LightGBM Regressor
- CatBoost Regressor

For each model, the **top 15 features** were identified based on model-specific importance measures (e.g., split importance, gain, or contribution). These rankings were then **cross-validated across models** to assess feature stability.

Features were selected based on **frequency and consistency of appearance** within the top 15 rankings across the four models. Features that repeatedly appeared as important under different modeling assumptions were treated as **robust predictors**, while features appearing sporadically or in only one model were excluded as model-specific artifacts.

As a result of this cross-model validation process, **16 features** were selected as the final predictive feature set. These features demonstrated stable influence across all

four models and represent meaningful signals related to **player performance, participation, availability, and physical characteristics**.

Two additional columns— `player_id` and `player_name` —were retained exclusively for **traceability and reporting purposes** and were explicitly excluded from model training.

Final Feature Set Used for Modeling

Performance and Participation

- `total_minutes_played`
- `total_matches`
- `total_goals`
- `total_assists`
- `total_penalty_goals`
- `total_own_goals`
- `total_nb_on_pitch`
- `total_nb_in_group`
- `total_subed_in`
- `total_subed_out`

Disciplinary

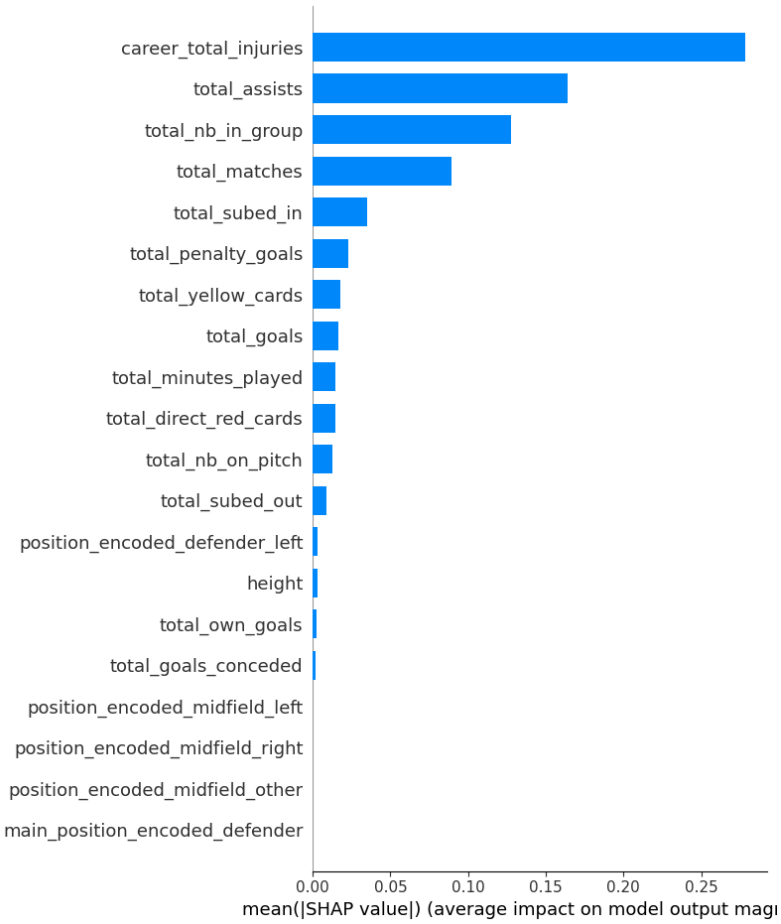
- `total_yellow_cards`
- `total_second_yellow_cards`
- `total_direct_red_cards`

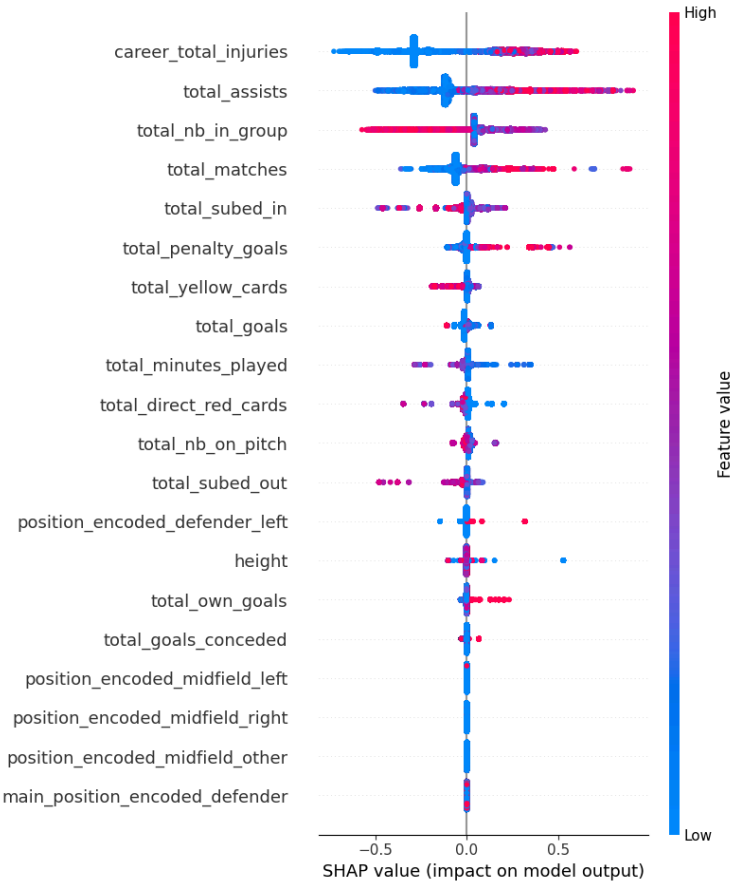
Availability and Physical Attributes

- `career_total_injuries`
- `height`

Positional Encoding

- `main_position_encoded_midfield`





Model Explainability via SHAP Analysis

High R^2 scores confirm predictive power, but **transparency and interpretability** are essential for deployment. SHapley Additive exPlanations (SHAP) analysis was applied to the final LightGBM model to understand *why* predictions are made. SHAP values quantify each feature's contribution, providing accountability.

SHAP Summary Plot (Importance & Direction):

This plot (Insert Plot) shows global feature importance (e.g., total_minutes_played, total_goals). Features are ranked by impact, and the color-coding (low/blue to high/red) shows how feature values influence the prediction, confirming alignment with domain knowledge (e.g., high performance rightarrow higher predicted value).

SHAP Dependency Plot (Feature Relationship):

This plot (Insert Plot) for a key feature (e.g., total_minutes_played) reveals the precise functional relationship. It tests for non-linear relationships and interactions, often showing a threshold effect (e.g., a minimum playtime is needed for positive valuation), ensuring the model is not relying on spurious correlations.

Construction of the Reduced Feature Dataset

Using the selected features, a reduced modeling dataset

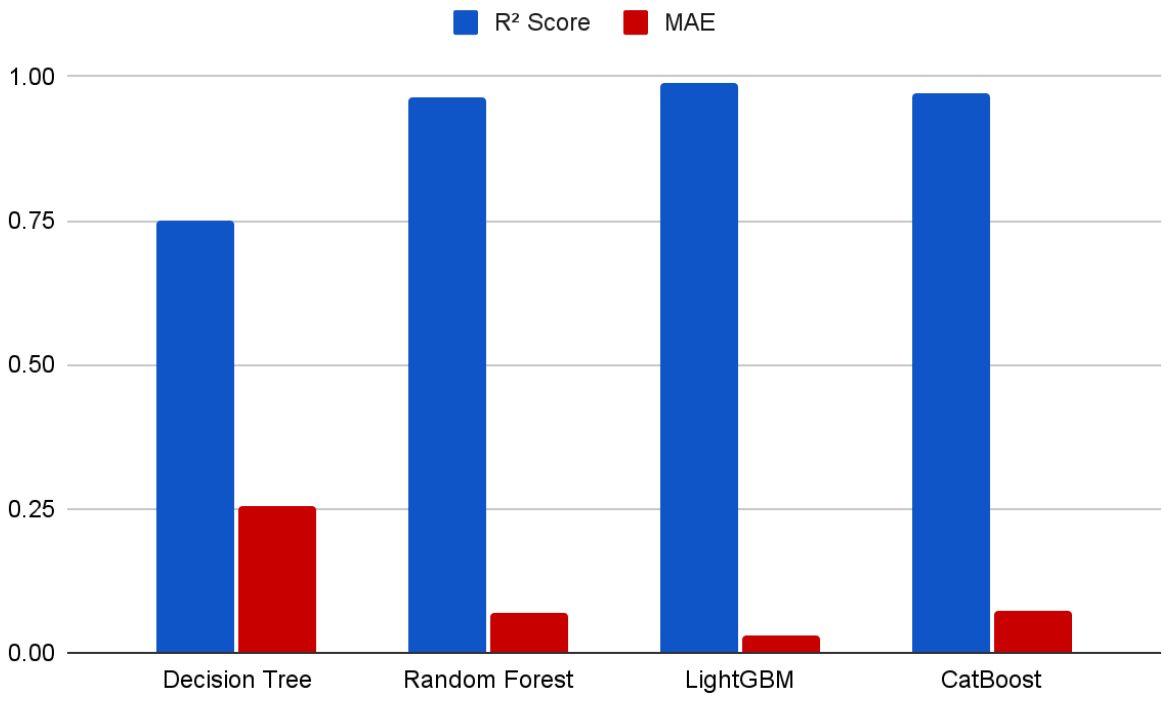
(`dataset_model_fs.csv`) was created. This dataset represents a **high-signal, low-noise feature space**, balancing predictive strength with interpretability and computational efficiency.

Before re-training, the target variable (`current_market_value`) was **log-transformed** to mitigate skewness and reduce the influence of extreme outliers, improving model stability.

Model Re-Training and Evaluation

The same set of models evaluated in Milestone 3 was retrained using the reduced feature set. Model performance was assessed using **R^2 score** and **Mean Absolute Error (MAE)**.

Model	R^2 Score	MAE
Decision Tree	0.749595	0.253420
Random Forest	0.962966	0.068429
LightGBM	0.987230	0.030715
CatBoost	0.972577	0.071171



Although minor performance variations were observed, predictive strength remained high. More importantly, the reduced feature set resulted in **lower training time, improved stability, and clearer interpretability**, confirming that removed features

primarily contributed noise rather than meaningful signal.

Deployment Readiness and Application Layer

To extend the project beyond experimental modeling, the final model was integrated into a **service-oriented architecture**.

Backend: FastAPI

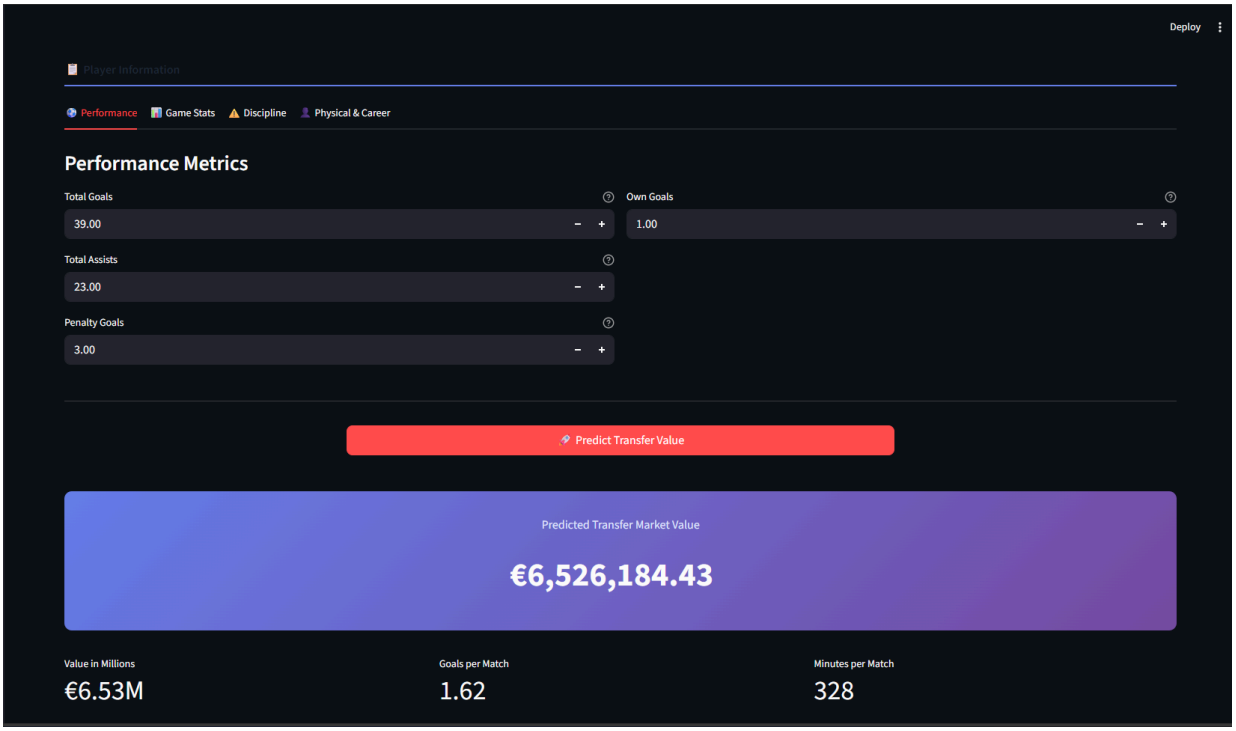
The trained model and preprocessing pipeline were serialized and deployed via a **FastAPI backend**, ensuring:

- Consistent feature ordering and transformations
- Controlled input validation
- Reproducible inference
- Clear separation between modeling and application logic

Frontend: Streamlit

A lightweight **Streamlit frontend** was developed to enable interactive user input and real-time market value prediction. This interface supports accessibility for non-technical users while maintaining transparency of model outputs.

The decoupling of backend and frontend components enhances modularity, maintainability, and scalability.



Interpretation and Trade-Offs

This milestone reinforced that **feature quality outweighs feature quantity**. Feature importance was used as an analytical guide rather than an unquestioned authority. By selecting features that demonstrated cross-model stability, the final pipeline balances predictive accuracy with robustness and interpretability.

While aggressive feature reduction can risk information loss, empirical validation showed that the retained features capture the dominant drivers of player valuation.

Final Conclusion

Milestone 4 distilled the project into a **compact, interpretable, and deployable machine learning solution**. Through cross-model validation of top-ranked features, a stable and high-signal feature set was identified and validated. Subsequent re-training and deployment demonstrated that the model is not only technically sound, but also **practically usable in real-world settings**.