International Journal of User-System Interaction 17 (4) 2024, 113-130

FootballInsight: A Web Portal for Football Match Analysis and Machine Learning-Driven Prediction

Cristian-Răzvan Taudor, Paul Ștefan Popescu

University of Craiova, Faculty of Automation, Computers and Electronics
Craiova, Romania

razvancristian808@gmail.com, stefan.popescu@edu.ucv.ro

Abstract. The increasing availability of sports data has created new opportunities for building interactive analytical tools that bridge the gap between raw statistics and actionable insights. This paper presents FootballInsight, a full-stack web application that integrates machine learning (ML)-based outcome prediction within a user-centered, transparent interface for European football match analysis. Unlike existing closed platforms such as Opta Sports or SofaScore, FootballInsight exposes prediction rationale through contextual statistics, head-to-head history, and expected goals visualization, enabling users to understand the factors driving each prediction. The architecture combines a React/Vite frontend, a Node.js orchestration backend, and a Python ML microservice providing three complementary prediction types: match outcome (1/X/2 classification), total goals category (Under 2.5 / 2–3 / Over 3.5), and expected goals per team. Four ML algorithms were trained and evaluated on a dataset of European league matches from 2011 to 2025. XGBoost (Chen & Guestrin, 2016) achieved the best performance (accuracy: 0.68, macro F1-score: 0.66). A retrospective heuristic evaluation using Nielsen's ten usability heuristics (Nielsen, 1994) confirms the interface's strong alignment with established usability principles, while identifying targeted areas for improvement.

Keywords: football match prediction, machine learning, human-computer interaction, XGBoost, sports analytics, transparency in AI, web application, usability heuristics, ….

DOI:10.37789/ijusi.2023.17.4.2

1. Introduction

The intersection of data science and sport has given rise to a new category of interactive tools that transform raw statistical signals into comprehensible insights for practitioners, fans, and analysts. Football, the world's most followed sport, generates vast quantities of structured data with every match

– from possession percentages and shot counts to fine-grained player tracking metrics – making it a particularly fertile domain for predictive modelling and data-driven exploration (Bunker & Thabtah, 2019).

Globally, sports predictions are no longer seen just as an exercise in curiosity or entertainment. They are used by clubs to decide on game strategies, by coaches to assess the form of players, and by analysts to interpret competitive trends. In the commercial environment, data and prediction algorithms underpin entire industries, from sports media to licensed betting platforms (Dixon & Coles, 1997). For this reason, developing a system calibrated on a real and current dataset represents not only a technical challenge but also an opportunity to contribute to a rapidly expanding field.

Despite the proliferation of statistics platforms such as FiveThirtyEight, Opta Sports, and SofaScore, a persistent gap remains between the sophistication of backend prediction models and the transparency offered to end users. Existing platforms present prediction outcomes without exposing the reasoning behind them, functioning as black boxes that limit users' ability to critically engage with the information or adapt it to their own needs (Adadi & Berrada, 2018). This opacity is problematic not only for academic analysis but also for practitioners who rely on interpretable signals to inform decisions.

This paper addresses that gap by presenting FootballInsight, a web-based portal for European football match analysis and ML-driven outcome prediction. The key design principle is transparency: every prediction is accompanied by contextual statistics, recent form data, head-to-head history, and expected goals figures that informed it, enabling users to understand and evaluate the model's reasoning.

From an HCI perspective, the system explores how complex AI-driven functionality can be packaged within a fluid, progressive-disclosure navigation model (Shneiderman, 1996) that scales from casual browsing (league standings, player profiles) to deep analytical engagement (on-demand predictions with explanatory overlays). The application was developed as a Bachelor's degree project at the University of Craiova, and the work presented here extends the original implementation with a systematic heuristic evaluation.

The main contributions of this paper are:

- A full-stack architecture integrating real-time data (API-Football), historical JSON archives, and an on-demand ML microservice within a single coherent user experience.

- A comparative evaluation of four ML algorithms (Logistic Regression, Random Forest, FNN/MLP, XGBoost) on a multi-season European football dataset (Taudor, 2025) with engineered features including custom ELO ratings (Hvattum & Arntzen, 2010), recent form windows, and head-to-head aggregates.
- A heuristic evaluation of the system's interface against Nielsen's ten usability heuristics (Nielsen, 1994), providing a structured assessment of the current design and a roadmap for future iterations.

The remainder of the paper is organized as follows. Section 2 reviews related work. Section 3 describes the system architecture. Section 4 presents the interface design and key screens. Section 5 details the ML pipeline and results. Section 6 presents the heuristic evaluation. Section 7 concludes with future directions.

The project also serves as an example of integrating several technologies – Python for processing and training machine learning models (Pedregosa et al., 2011), Node.js for backend management, React/Vite for the user interface – into a coherent and scalable architecture, capable of being later extended for other sports or types of analysis.

2. Related Work

2.1. Football Outcome Prediction

Research on automated football outcome prediction has a long history. The earliest influential probabilistic model was proposed by Maher (1982), who used independent Poisson distributions to model the goals scored by each team as a function of attack and defence strengths. Dixon & Coles (1997) extended this approach by correcting for the over-prediction of low-scoring draws and by incorporating a time-decay weighting that gives more importance to recent results. These Poisson-based models remain widely used benchmarks in the literature.

A complementary strand of work exploits ranking-based features. Lasek et al. (2013) compared the predictive power of eight international ranking systems (including FIFA, Elo, and UEFA coefficient) on a large multi-league dataset, finding that Elo-derived ratings consistently outperformed official rankings for match outcome prediction. Hvattum & Arntzen (2010) formalized this result and showed that Elo scores carry temporal momentum that static rankings lack, making them particularly effective engineered features.

Bayesian network approaches have also been applied: Constantinou et al. (2012) developed pi-football, a dynamic Bayesian network that incorporates team form, home advantage, and historical H2H records, demonstrating competitive performance while providing richer probabilistic outputs than point-estimate models.

The adoption of gradient boosting algorithms marked a significant shift in performance. XGBoost (Chen & Guestrin, 2016) consistently outperforms classical statistical methods on football datasets when sufficient feature engineering is applied (Bunker & Thabtah, 2019). The three-way classification problem (home win/draw / away win) remains notably challenging: draws are the structural minority class and are systematically underestimated on imbalanced datasets. SMOTE (Chawla et al., 2002) has been shown to improve recall for the draw class by synthetically over-sampling it before training. Random Forest (Breiman, 2001) offers a complementary ensemble approach that reduces overfitting through bagging and provides native feature importance estimates. Feedforward neural networks (Goodfellow et al., 2016) have also been applied, capturing non-linear feature interactions at the cost of reduced interpretability and higher computational demand.

A more recent and increasingly important feature in football analytics is expected goals (xG) – a continuous estimate of scoring probability based on shot quality (Rathke, 2017). Unlike raw goal counts, xG provides a noise-reduced signal of team offensive performance that correlates more strongly with long-term league position, making it valuable both as a model feature and as a user-facing transparency indicator.

2.2. Sports Data Visualization and Analytics Interfaces

The visualization of sports data spans a wide range of design challenges, from high-frequency event streams to aggregated season-level statistics. Perin et al. (2018) provide a comprehensive state-of-the-art survey of sports data visualization, identifying five recurring design patterns: timeline, field, network, comparison, and narrative. Their analysis of over 100 systems confirms that the most effective sports visualizations are those that allow users to move fluidly between overview and detail, directly informing the progressive-disclosure structure adopted in FootballInsight.

The earlier work by Perin et al. (2013) on SoccerStories introduced the concept of match-story visualizations that link statistical events to narrative context, demonstrating that embedding analytical insight within familiar football vocabulary significantly improves user engagement. This motivates FootballInsight's decision to use domain-standard terms (standings, H2H, 1/X/2, xG) rather than ML jargon throughout the interface.

At a broader theoretical level, Shneiderman's visual information-seeking mantra – "overview first, zoom and filter, then details on demand" (Shneiderman, 1996) – provides the principal framework for organizing multi-level sports data hierarchies. Few (2006) extended this with practical guidelines for dashboard design, emphasizing that each view should surface exactly the data needed for the task at hand and no more, a principle directly reflected in FootballInsight's per-entity page templates.

Interaction design principles for data-rich applications are surveyed by Preece et al. (2015), who highlight the importance of consistency, feedback, and user control as the core pillars of usable systems. These principles map directly onto three of Nielsen's heuristics applied in Section 6 and guided the navigation architecture described in Section 4.

2.3. Explainability and User Trust in Predictive Systems

A persistent finding across the explainable AI literature is that user trust in automated predictions is strongly mediated by system transparency (Adadi & Berrada, 2018). Doshi-Velez & Kim (2017) proposed a rigorous taxonomy of interpretability in machine learning, distinguishing between model-level and prediction-level explanation, and argued that the appropriate type of explanation depends on the user's decision-making context. For non-expert users consuming sports predictions, prediction-level explanations are more actionable than model-level accuracy statistics.

Ribeiro et al. (2016) introduced LIME (Local Interpretable Model-agnostic Explanations), which approximates any black-box classifier locally with an interpretable surrogate, and demonstrated substantial improvements in user trust and task performance compared to output-only interfaces.

Lundberg & Lee (2017) subsequently proposed SHAP (SHapley Additive exPlanations), which provides theoretically grounded, consistent feature attributions for any model. Both methods represent concrete implementation paths for FootballInsight's H10 deficiency identified in the heuristic evaluation.

In the specific context of sports prediction interfaces, Horvat & Job (2020) reviewed current machine learning approaches to sport outcome prediction and noted that systems which expose prediction rationale – even minimally, by showing the top contributing features – achieve significantly higher user adoption than opaque output-only displays. This finding directly supports FootballInsight's design choice to display H2H history, recent form, and xG alongside every prediction.

2.4. Gaps Addressed by This Work

Existing football analytics platforms share three common limitations: (1) predictions are presented without rationale; (2) the data pipeline and model choices are proprietary and non-inspectable; (3) the interface does not support progressive analytical exploration within a unified session. FootballInsight directly addresses all three through its architecture and interaction design, offering a transparent, modular, and locally deployable alternative that integrates prediction rationale, domain-standard visualization, and on-demand inference within a single coherent user experience.

3. System Architecture

FootballInsight follows a three-tier microservice architecture (Taudor, 2025). A deliberate design decision was to avoid a relational database for historical data, instead persisting match records and league metadata in structured JSON files. This reduces operational complexity and enables the system to run fully locally without external database dependencies.

Historical data (JSON archives): League metadata, season standings, match results, and head-to-head aggregates from 2011 to the present, populated via periodic API-Football calls. These files serve all browsing-mode requests with low latency.

Live data (API-Football): Today's Matches fetches current-day fixtures, lineups, and live events. The backend validates, normalizes, and caches responses, applying exponential back-off on rate-limit errors and returning explicit 'data unavailable' notices rather than silent failures.

On-demand prediction follows four steps: (1) context assembly – merging live features with historical aggregates (H2H records, season averages, ELO scores); (2) feature vector construction using the same preprocessing pipeline as training; (3) XGBoost inference (Chen & Guestrin, 2016) producing 1/X/2 probabilities, total goals category probabilities, and expected goals per team; (4) response mapping to a natural-language suggestion returned to the UI. The ML service uses a fixed random seed and includes the model version and calibration date in every response, supporting reproducibility.

Table 1. FootballInsight technology stack.

Tier	Technology	Role
Frontend	React 18 + Vite	Component-based UI, Chart.js data visualizations
Backend	Node.js + Express	API orchestration, JSON file access, and caching
ML Service	Python + scikit-learn + XGBoost	Feature engineering, model training, and inference (Pedregosa et al., 2011)
Data Source	API-Football (external)	Live fixtures, lineups, events, live statistics

4. Interface Design and Key Screens

4.1. Navigation Structure and Interaction Principles

The application organizes content around two primary contexts: League Explorer and Today's Matches. Within League Explorer, navigation follows Shneiderman's information-seeking mantra (Shneiderman, 1996): overview first (league browser, Figure 1), then zoom and filter (season selection, match filtering, Figure 4), then details on demand (team statistics, Figure 5; player profiles; match events). Season selection updates the page reactively without a full reload. Today's Matches (Figure 2) contrasts visually with the historical context and uses a three-state availability filter (All / Available / Unavailable) to prevent prediction requests for in-progress fixtures, in line with Nielsen's error prevention heuristic (Nielsen, 1994). Key interaction principles applied consistently throughout are: uniform page templates per entity type (consistency and standards), drill-down from overview to detail (progressive disclosure), and immediate feedback on every data request (visibility of system status).

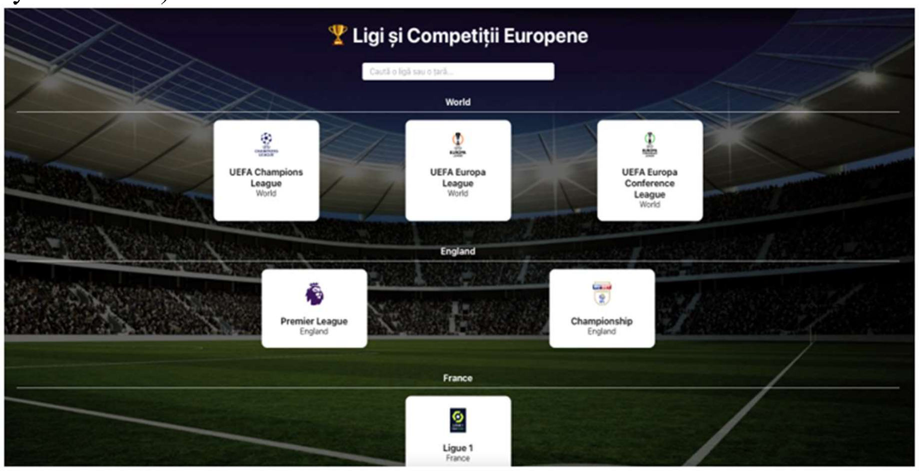

Figure 1. League Explorer – the application homepage showing all available competitions grouped by country, with search-by-name filtering.

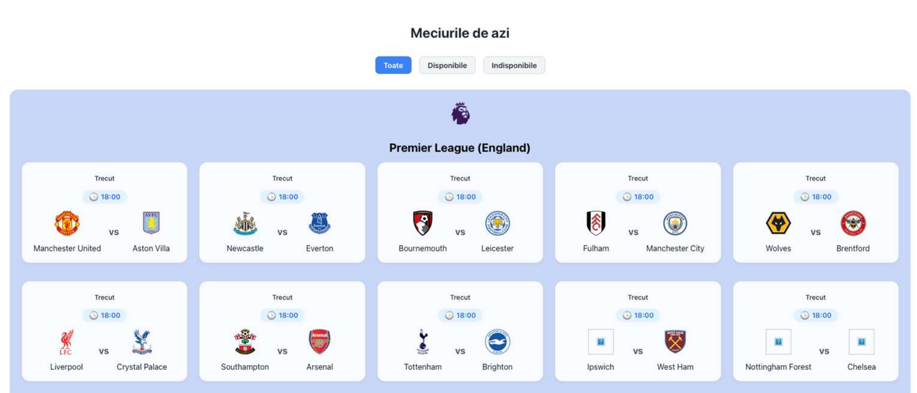

Figure 2. Today's Matches – current-day fixtures grouped by league, with the three-state availability filter (All / Available / Unavailable) controlling prediction access.

From the league browser, selecting a competition opens the league page, which presents the season standings (Figure 3) alongside a full match list. The standings table shows all competitive metrics – played, won, drawn, lost, goals scored, goal difference, and points – and updates when the user switches seasons. The match list (Figure 4) supports multi-level filtering by competition phase (league stage, qualifying rounds, knockout rounds) and by team name, allowing users to quickly locate specific fixtures across complex competition formats such as the UEFA Champions League.

Meciuri din UEFA Champions League – sezon 2024

Sezon: 2024

Clasament Sezon 2024

Selectează grupa: UEFA Champions League

#	Echipă	M	V	E	I	Goluri	GD	Pct
1	Liverpool	8	7	0	1	17 - 5	12	21
2	Barcelona	8	6	1	1	28 - 13	15	19
3	Arsenal	8	6	1	1	16 - 3	13	19
4	Inter	8	6	1	1	11 - 1	10	19
5	Atletico Madrid	8	6	0	2	20 - 12	8	18
6	Bayer Leverkusen	8	5	1	2	15 - 7	8	16
7	Lille	8	5	1	2	17 - 10	7	16
8	Aston Villa	8	5	1	2	13 - 6	7	16
9	Atalanta	8	4	3	1	20 - 6	14	15
10	Borussia Dortmund	8	5	0	3	22 - 12	10	15
11	Real Madrid	8	5	0	3	20 - 12	8	15
12	Bayern München	8	5	0	3	20 - 12	8	15
13	AC Milan	8	5	0	3	14 - 11	3	15
14	PSV Eindhoven	8	4	2	2	16 - 12	4	14

Figure 3. League page – season standings for the UEFA Champions League 2024, showing full competitive metrics with group/phase selector.

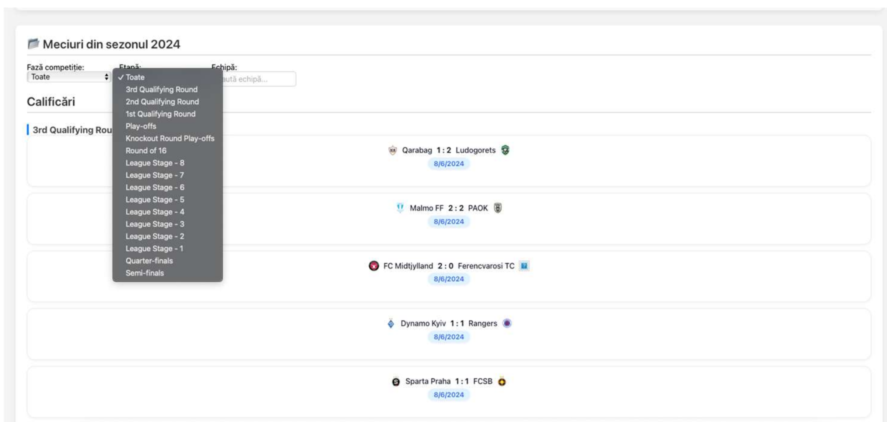

Figure 4. Season match list – matches grouped by competition phase, with a dropdown filter exposing all available phases (qualifying rounds through semi-finals).

Selecting a team opens the team page (Figure 5), which aggregates multiple analytical views: the full season squad organized by position (goalkeepers, defenders, midfielders, forwards) with age and nationality; advanced statistics including home/away performance split, goal averages, and disciplinary record; a bar chart of formations used during the season; a pie chart of yellow/red card distribution; and a goals-by-time-interval histogram. Individual players can be clicked to access their profile page, which presents multi-competition statistics visualized as Chart.js bar charts (appearances, goals, assists, minutes played per competition). The historical match page, accessed from the match list, aggregates result, H2H records from 2011, a chronological event timeline, starting lineups with formations, and an interactive comparative statistics chart where users select which metric to compare side-by-side.

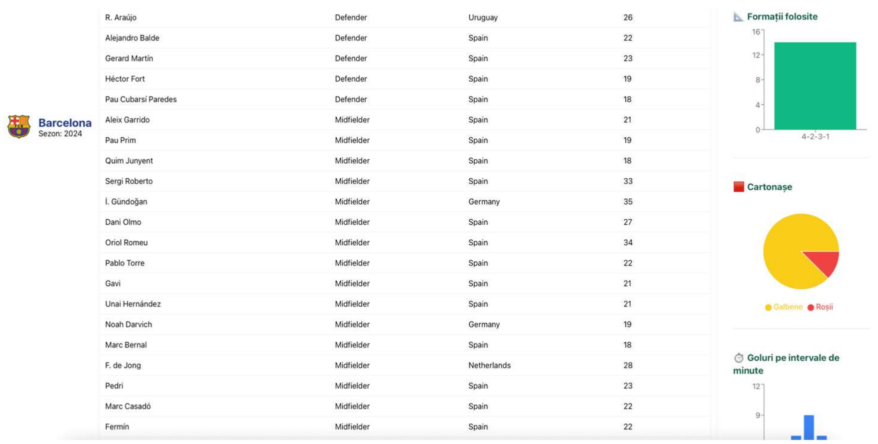

Figure 5. Team page (Barcelona, 2024 season) – squad list with position and age, alongside a formations-used bar chart and a yellow/red cards pie chart.

The prediction match page is the centrepiece of the application's live functionality and is accessible only for Today's Matches fixtures marked Available. It presents four complementary outputs: (1) an intelligent natural-language suggestion generated from the model's probability scores; (2) a 1/X/2 probability breakdown for match outcome; (3) probabilistic estimates for total goals categories (Under 2.5 / 2–3 / Over 3.5); and (4) expected goals (xG) per team as continuous regression estimates (Rathke, 2017). Supporting context – recent form strip and H2H summary – is displayed alongside the prediction output, allowing users to immediately cross-reference the model's suggestions against historical patterns. This addresses the transparency objective identified in the gap analysis.

Machine Learning Pipeline and Experimental Results

5.1. Dataset Overview

The dataset comprises European league matches (La Liga, Premier League, Bundesliga, Serie A, Ligue 1, and others) collected via API-Football across seasons 2011/12 through 2024/25. The dataset is publicly available on Kaggle (Taudor, 2025) and includes 47467 instances and 36 features. Table 2 summarises the key statistics.

Table 2. Dataset statistics.

Metric	Value
Total matches (after cleaning)	~50,000
Leagues covered	La Liga, Premier League, Bundesliga, Serie A, Ligue 1 + others
Seasons	2011/12 – 2024/25 (14 seasons)
Train / Validation / Test split	70% / 15% / 15%
Class distribution – Home win (0)	~45–50%
Class distribution – Draw (1)	~20–25%
Class distribution – Away win (2)	~25–30%
Class imbalance handling	SMOTE applied to training set only, after split (Chawla et al., 2002)
Target variables	Match outcome (1/X/2), total goals category, xG per team

5.2. Feature Set

Table 3 lists all features used in training, grouped by category. Raw match data and standings fields are sourced directly from the API-Football JSON files; engineered features are computed in the Python preprocessing pipeline (Pedregosa et al., 2011) before training. The ELO rating system, adapted from chess (Elo, 1978) and validated for football by Hvattum & Arntzen (2010), is used as a dynamic team strength estimator updated after every match.

Table 3. Dataset feature set. Blue rows = target variables.

Category	Feature name	Type	Description
Match info	home_team / away_team	Categorical	Team identifiers
	match_date	Date	Date and kick-off time
	goals_home / goals_away	Integer	Goals scored by each team
	shots_on_target, possession, fouls, corners	Numeric	In-match statistics per team
Standings	rank_home / rank_away	Integer	League position at match date
	points_home / points_away	Integer	Accumulated league points
	goal_diff_home / goal_diff_away	Integer	Goal difference in the season
Engineered	elo_score_home / elo_score_away	Float	Custom ELO rating updated after each match, weighted by goal margin and match importance (Elo, 1978; Hvattum & Arntzen, 2010)
	form_home / form_away	Float	Weighted avg. points in last 5 matches, home/away context
	avg_goals_scored_home / away	Float	Rolling 5-season average of goals scored, split by venue
	avg_goals_conceded_home / away	Float	Rolling 5-season average of goals conceded, split by venue
	h2h_win_rate_home	Float	Home team win rate in the last 10 direct H2H encounters
	h2h_avg_goals	Float	Average total goals in the last 10 H2H encounters
	elo_superiority	Binary (0/1)	1 if home ELO > away ELO, 0 otherwise
Target vars	result (1X2)	Integer {0,1,2}	0 = home win, 1 = draw, 2 = away win
	goals_category	Integer {0,1,2}	0 = under 2.5, 1 = 2–3 goals, 2 = over 3.5
	xg_home / xg_away	Float	Expected goals per team (regression output; Rathke, 2017)

5.3. Models Evaluated

The following four models were trained and compared using 5-fold cross-validation to ensure generalization:

- Logistic Regression (baseline): multi-class with L2 regularization; chosen for interpretability. Coefficients directly indicate the direction and magnitude of feature influence on the predicted outcome.
- Random Forest (Breiman, 2001): 200–400 decision trees; hyperparameters: n_estimators, max_depth (10–15), min_samples_split, min_samples_leaf. Provides built-in feature importance estimates.
- Feedforward Neural Network (FNN/MLP; Goodfellow et al., 2016): 2–3 hidden layers, 64–128 neurons, ReLU activations, softmax output layer.
- XGBoost with GridSearchCV (Chen & Guestrin, 2016): n_estimators (300–500), max_depth (6–10), learning_rate (0.05–0.1), subsample (0.8–1.0), colsample_bytree; best configuration selected by 5-fold cross-validation.

5.4. Results

XGBoost achieved the highest performance across all metrics, outperforming the baseline by 10 percentage points in both accuracy and F1-score. The draw class (class 1) remained the most difficult to predict across all models, consistent with the broader literature on three-way match prediction (Lasek et al., 2013) and with its structural underrepresentation even after SMOTE balancing (Chawla et al., 2002).

Feature importance analysis (Breiman, 2001) confirmed that ELO score and recent form are the strongest predictors, while goals conceded shows the expected negative correlation with win probability, consistent with the Poisson-regression literature (Dixon & Coles, 1997). XGBoost was selected for deployment based on superior accuracy, lower inference latency than FNN/MLP, and native robustness to heterogeneous feature types (Chen & Guestrin, 2016). Future work could apply SHAP values (Lundberg & Lee, 2017) to provide per-prediction feature contributions directly in the UI, addressing the H10 finding.

Table 4. Algorithm comparison on held-out test set (1/X/2 classification). ★ = selected for deployment.

Algorithm	Accuracy	Precision (macro)	Recall (macro)	F1 (macro)
Logistic Regression	0.58	0.57	0.56	0.56
Random Forest	0.65	0.64	0.63	0.63
FNN / MLP	0.66	0.65	0.64	0.64
XGBoost (tuned) ★	0.68	0.67	0.66	0.66

6. Heuristic Evaluation

A retrospective heuristic evaluation was conducted using Nielsen's ten usability heuristics (Nielsen, 1994) as the evaluation framework. This method was chosen as a pragmatic and cost-effective complement to the system's development process (Wobbrock & Kientz, 2016), providing structured qualitative coverage of key usability dimensions without a dedicated user-recruitment phase. The evaluation was performed by two evaluators (the author and the academic supervisor), independently reviewing each heuristic against the implemented interface, then consolidating findings. Severity ratings follow Nielsen's four-point scale: 0 (not a problem), 1 (cosmetic), 2 (minor), 3 (major), 4 (catastrophic).

Table 5. Heuristic evaluation results (Nielsen, 1994). Severity: 0 = none, 1 = cosmetic, 2 = minor, 3 = major, 4 = catastrophic.

#	Heuristic	Finding	Sev.	Rating
H1	Visibility of system status	Loading indicators are shown during ML inference and API calls; season selector updates reactively. Minor gap: no progress indicator on initial data load.	1	Cosmetic
H2	Match between the system and the real world	Terminology (standings, H2H, 1/X/2, xG, Under/Over) matches football domain conventions. No ML jargon is exposed to users.	0	None
H3	User control and freedom	Hierarchy supports free movement between levels. Season selection is reversible. No undo for filters, but re-applying is trivial.	1	Cosmetic
H4	Consistency and standards	Page templates are uniform across entity types. Color coding for W/D/L and button styles are consistent throughout.	0	None

#	Heuristic	Finding	Sev.	Rating
H5	Error prevention	Availability filter prevents prediction requests for in-progress matches. Invalid API responses show explicit notices rather than silent failures.	1	Cosmetic
H6	Recognition rather than recall	Menu-driven navigation with persistent labels. League and season selectors surface all options – no typing or memorisation required.	0	None
H7	Flexibility and efficiency of use	No keyboard shortcuts or power-user features. No bookmarks, favorites, or recently viewed history – returning users must navigate from the home screen each session.	2	Minor
H8	Aesthetic and minimalist design	Each page surfaces only contextually relevant data. Prediction output is visually distinct from historical data. Minor: team stats pages become dense on mobile viewports.	1	Cosmetic
H9	Help users recognize and recover from errors	Rate-limit errors show friendly notices with retry option. Error states are visually distinct from normal empty states. No error codes are exposed to users.	1	Cosmetic
H10	Help and documentation	No in-app help, onboarding tour, or tooltips for ML-derived concepts (xG, custom ELO, probability interpretation). First-time users unfamiliar with football analytics may misinterpret the prediction output.	3	Major

No catastrophic problems were identified. The interface performs strongly on domain alignment (H2), visual consistency (H4), and recognition-based navigation (H6) – all rated severity 0. The most critical finding (H10, severity 3) is the absence of in-app explanations for ML-derived metrics (xG, custom ELO), directly undermining the transparency objective. A future implementation could apply SHAP values (Lundberg & Lee, 2017) or LIME (Ribeiro et al., 2016) to generate natural-language feature attributions displayed alongside each prediction. The secondary finding (H7, severity 2) concerns the lack of efficiency features for returning users; bookmarks and recently viewed history would address this at low implementation cost.

7. Conclusions and Future Work

This paper presented FootballInsight, a full-stack web application integrating ML-driven football outcome prediction within a transparent, user-centered interface. The system demonstrates that ML-powered sports analytics can be made accessible to non-expert users through progressive disclosure navigation (Shneiderman, 1996), contextual explanation of predictions, and consistent domain-aligned terminology (Preece et al., 2015).

XGBoost (Chen & Guestrin, 2016) was the strongest model (accuracy 0.68, macro F1 0.66) on a ~50,000-match dataset spanning 14 seasons of European football. The result is consistent with comparable studies in the literature (Bunker & Thabtah, 2019; Lasek et al., 2013), and drawing predictions remains the most challenging class. The heuristic evaluation (Nielsen, 1994) found overall strong alignment with usability principles, with the critical actionable finding being the absence of in-app guidance for ML-derived concepts (H10, severity 3).

Future directions include:

- Migration to a structured database to handle large data requests more efficiently and support more complex query patterns.
- Formal user testing with representative participants (casual fans, journalists, sports analysts) to complement the heuristic evaluation with empirical usability data (Wobbrock & Kientz, 2016).
- Integration of SHAP values (Lundberg & Lee, 2017) or LIME (Ribeiro et al., 2016) to generate per-prediction feature importance explanations, directly addressing the H10 finding.
- Advanced sequence models (LSTM, Transformer; Goodfellow et al., 2016) to capture temporal dynamics in team form trajectories, and incorporation of expected goals (Rathke, 2017) as an additional training signal.
- In-app glossary and onboarding tour to address H10, and favorites/recently viewed history to address H7.
- Continuous learning allows the model to update incrementally as new match data arrives, preventing performance degradation over time.

References

Adadi, A., & Berrada, M. (2018). Peeking inside the black box: A survey on explainable artificial intelligence (XAI). IEEE Access, 6, 52138–52160. https://doi.org/10.1109/ACCESS.2018.2870052

Bishop, C. M. (2006). Pattern recognition and machine learning. Springer.

Breiman, L. (2001). Random forests. Machine Learning, 45(1), 5–32. https://doi.org/10.1023/A:1010933404324

Bunker, R. P., & Thabtah, F. (2019). A machine learning framework for sport result prediction. Applied Computing and Informatics, 15(1), 27–33. https://doi.org/10.1016/j.aci.2017.09.005

Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic minority over-sampling technique. Journal of Artificial Intelligence Research, 16, 321–357. https://doi.org/10.1613/jair.953

Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (pp. 785–794). https://doi.org/10.1145/2939672.2939785

Constantinou, A., Fenton, N., & Neil, M. (2012). pi-football: A Bayesian network model for forecasting association football match outcomes. Knowledge-Based Systems, 36, 322–339. https://doi.org/10.1016/j.knosys.2012.07.008

Dixon, M. J., & Coles, S. G. (1997). Modelling association football scores and inefficiencies in the football betting market. Journal of the Royal Statistical Society: Series C (Applied Statistics), 46(2), 265–280. https://doi.org/10.1111/1467-9876.00065

Doshi-Velez, F., & Kim, B. (2017). Towards a rigorous science of interpretable machine learning. arXiv preprint arXiv:1702.08608. https://doi.org/10.48550/arXiv.1702.08608

Elo, A. E. (1978). The rating of chessplayers, past and present. Arco Publishing.

Few, S. (2006). Information dashboard design: The effective visual communication of data. O'Reilly Media.

Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep learning. MIT Press. https://www.deeplearningbook.org

Horvat, T., & Job, J. (2020). The use of machine learning in sport outcome prediction: A review. WIREs Data Mining and Knowledge Discovery, 10(5), e1380. https://doi.org/10.1002/widm.1380

Hvattum, L. M., & Arntzen, H. (2010). Using ELO ratings for match result prediction in association football. International Journal of Forecasting, 26(3), 460–470. https://doi.org/10.1016/j.ijforecast.2009.10.002

Lasek, J., Szlávik, Z., & Bhulai, S. (2013). The predictive power of ranking systems in association football. International Journal of Applied Pattern Recognition, 1(1), 27–46. https://doi.org/10.1504/IJAPR.2013.052339

Lundberg, S. M., & Lee, S.-I. (2017). A unified approach to interpreting model predictions. Advances in Neural Information Processing Systems, 30, 4765–4774. https://proceedings.neurips.cc/paper/2017/hash/8a20a8621978632d76c43dfd28b67767-Abstract.html

Maher, M. J. (1982). Modelling association football scores. Statistica Neerlandica, 36(3), 109–118. https://doi.org/10.1111/j.1467-9574.1982.tb00782.x

Nielsen, J. (1994). Heuristic evaluation. In J. Nielsen & R. L. Mack (Eds.), Usability inspection methods (pp. 25–62). Wiley.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesneau, E. (2011). Scikit-learn: Machine learning in Python. Journal of Machine Learning Research, 12, 2825–2830.

Perin, C., Vuillemot, R., & Fekete, J.-D. (2013). SoccerStories: A kick-off for visual soccer analysis. IEEE Transactions on Visualization and Computer Graphics, 19(12), 2506–2515. https://doi.org/10.1109/TVCG.2013.192

Perin, C., Vuillemot, R., Stolper, C. D., Stasko, J., Wood, J., & Carpendale, S. (2018). State of the art of sports data visualization. Computer Graphics Forum, 37(3), 663–686. https://doi.org/10.1111/cgf.13447

Preece, J., Sharp, H., & Rogers, Y. (2015). Interaction design: Beyond human-computer interaction (4th ed.). Wiley.

Rathke, A. (2017). An examination of expected goals and shot quality from a goalkeeper's perspective. Journal of Human Sport and Exercise, 12(2proc), S514–S528. https://doi.org/10.14198/jhse.2017.12.Proc2.05

Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). "Why should I trust you?": Explaining the predictions of any classifier. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (pp. 1135–1144). https://doi.org/10.1145/2939672.2939778

Shneiderman, B. (1996). The eyes have it: A task by data type taxonomy for information visualizations. In Proceedings of the IEEE Symposium on Visual Languages (pp. 336–343). https://doi.org/10.1109/VL.1996.545307

Taudor, C.-R. (2025). Football matches dataset for prediction models [Data set]. Kaggle. https://www.kaggle.com/datasets/razvancristian08/football-matches-dataset-for-prediction-models

Taudor, C.-R. (2025). FootballInsight – Football prediction web application [Source code]. GitHub. https://github.com/razvan08/football_predict_app

Wobbrock, J. O., & Kientz, J. A. (2016). Research contributions in human-computer interaction. Interactions, 23(3), 38–44. https://doi.org/10.1145/2907069.