COMP4801 Final Year Project [2023/24] Detailed Project Plan

KickInsights: Football Application with Machine Learning

Supervisor   Dr. Chim T W
Student   Fung Hon Yin   3035784497

Table of Contents

Page

1. Project Background

In the ever-evolving world of football, where there is enormous passion from millions of fans across the globe, there is a growing demand for convenient yet comprehensive access to football data. Hence, a variety of mobile applications have been developed in the past decade. This project aims to create a mobile app – KickInsights, to further elevate fans' football experience by tackling the common inefficiencies of current alternatives, and to implement a game-changing match prediction system by leveraging the power of machine learning.

1.1 Motivation

Despite the availability of football applications on mobile platforms, most of the existing options share the following downsides.

Firstly, a lack of functions for insights extraction from static data. These apps only provide pages loaded with fixed information with no tools for evaluation. For example, most apps provide a club or player information page listing their statistics (see Figure 1,2). However, users cannot take advantage of the data provided to perform further evaluation, including seasonal comparison between clubs, or cross-club player comparison. Without functions for further data exploration, under-utilization of the data will likely have resulted.
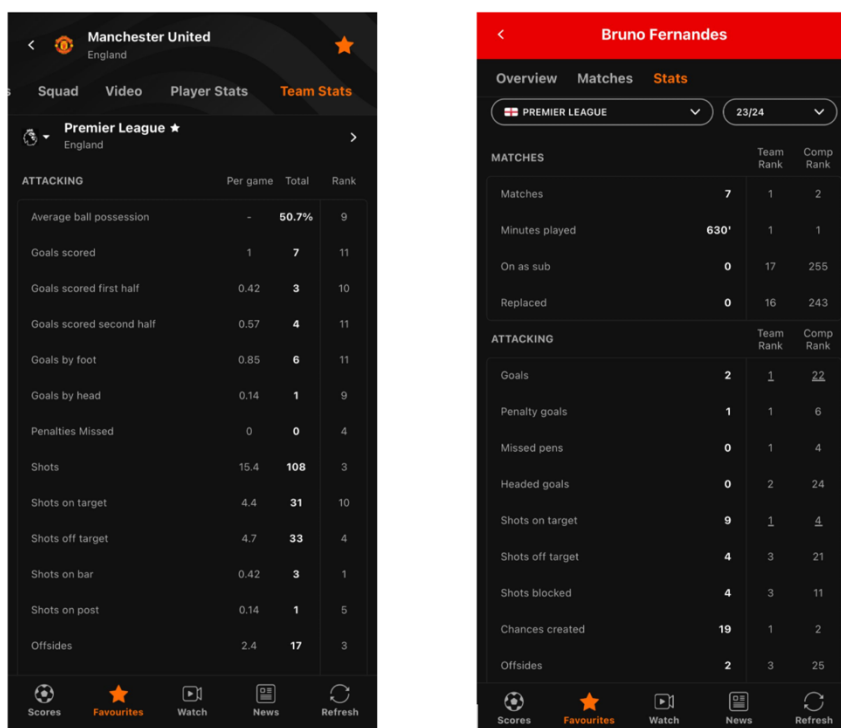

Figure 1&2. Club and Player information page from LiveScore

Secondly, inconsistency and low comprehensiveness of the data provided. For instance, some apps only scratched the surface and offer general club information like points and goal types (see Figure 3). Others offer only a few columns of player data without clear notation (see Figure 4). This leads to a poor overall user experience as they will be unsure whether the data they are looking for is available in the app.

For users who desire detailed in-game statistics, like a player's long pass completion rate in the current season, or a goalkeeper's launch rate during goal kicks, it leaves them no choice but to visit large-scale online databases, which is time-consuming and difficult to navigate on a mobile browser (see Figure 5).
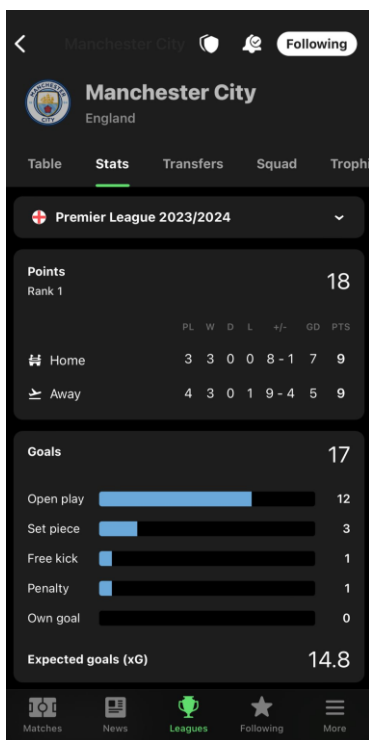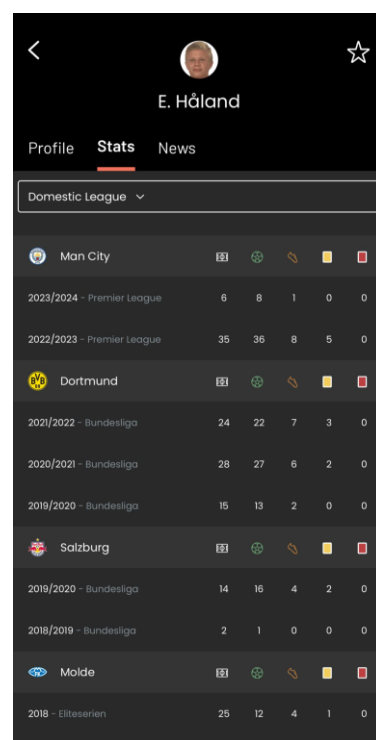

Figure 3. Club data from FotMob
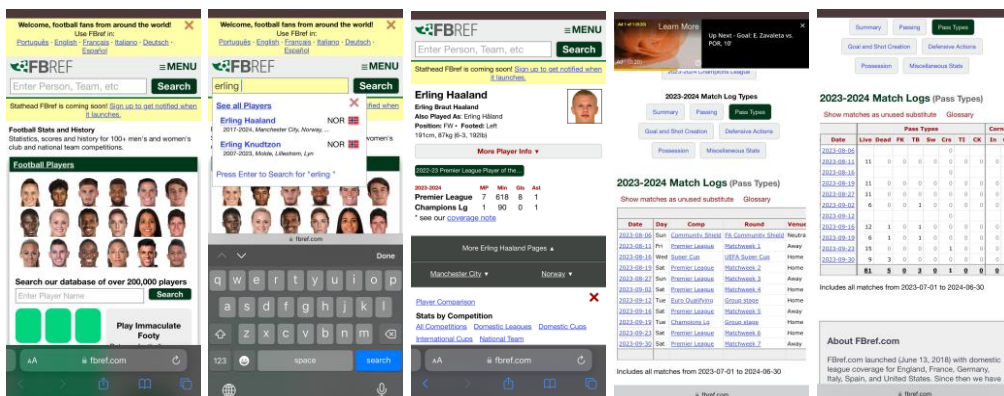

Figure 4. Player data from GOAL

Figure 5. The sequence of actions to locate players' statistics on a web browser.

Thirdly, the absence of an objective match-winner prediction system. Some apps with prediction features offer expert tips and suggestions (see Figure 6,7), which are subjective and prone to errors. Some only provide selective insights to users (see Figure 8), which is unorganized and has a low reference value. Some apps create polls as an indicator of sentiment (see Figure 9,10). However, they are highly biased due to the different amount of supporters for each club.
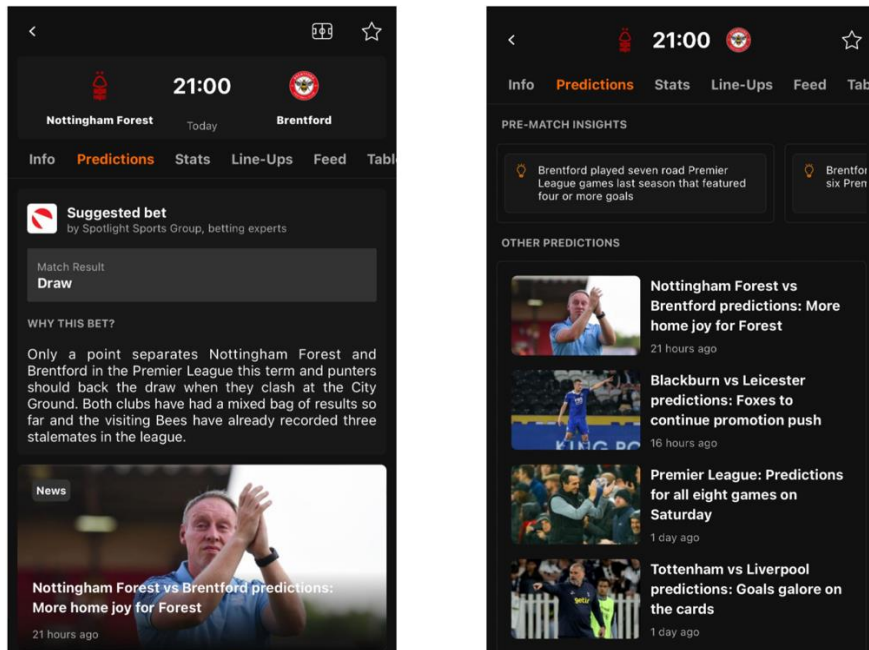

Figure 6&7. Predictions and match result suggestions from LiveScore
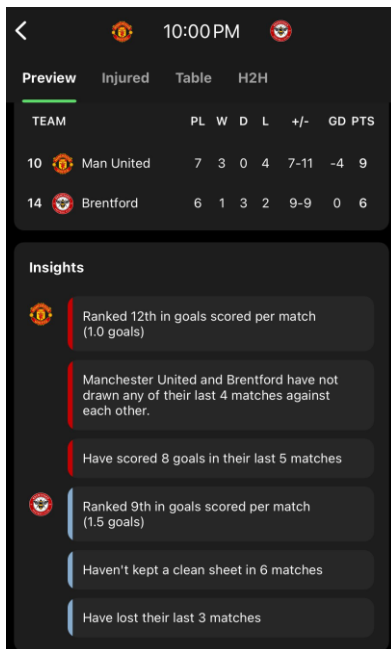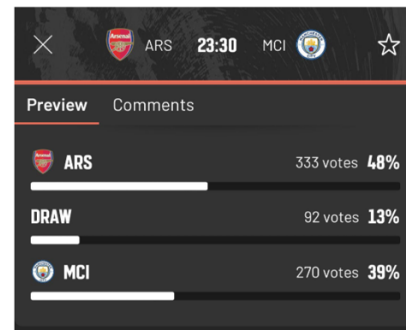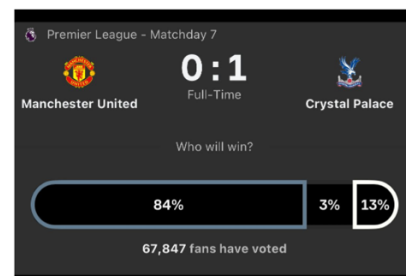

Figure 8. Match insights from FotMob

Figure 9&10. Prediction poll from OneFootball and GOAL

1.2 Existing Football Applications

Features / Functions	AiScore	FotMob	GOAL	LiveScore	One-Football	KickInsights
In-depth team data [1]	✗	✗	✗	✔	✔	✔
In-depth player data [2][3]	✔	✔	✗	✔	✔	✔
Team/Player Comparison	✗	✗	✗	✗	✗	✔
Match prediction (opinion based)	✔	✗	✔	✔	✔	✔
Match prediction (statistical based)	✗	✔	✗	✔	✗	✔

✔ - Provided , ✔ - Partially Provided , ✗ - Not Provided
[1] See Appendix A
[2][3] See Appendix B, C

Table 1. Comparison between popular football apps and KickInsights
(as of 2023-09-13)

As shown in the table, even the best alternatives for obtaining club and player data, LiveScore and OneFootball, provide them only partially. Besides, not one of these apps provides an interface for direct club or player comparison, implying that in order to compare these statistics, users would have to navigate between pages to make comparisons in a one-by-one manner. Lastly, match prediction by these apps (if offered) is not statistical-based in general.

1.3 Proposed Solution

The goal of this project is to develop KickInsights, a brand-new mobile app that brings comprehensive football data onto mobile devices. Football enthusiasts will no longer need to access in-depth statistics via personal computers, or by going through time-consuming and repetitive navigation on web browsers.

Besides offering the data, KickInsights also allows users to extract insights from them via interfaces for statistical comparison, and even further utilize them with a data-driven match prediction system powered by machine learning algorithms.

2. Project Objective and Scope

2.1 Objective

This project aims to develop KickInsights to solve the inefficiencies of current mobile application alternatives, and to integrate machine learning components with the app to enhance utilization of the football data provided. To achieve the outcome, the core functionalities of the app are elaborated upon in section 2.3. The list is not exhaustive as extra features may be incorporated into the app depending on the future progress of the project.

2.2 Scope of data

This project will focus on the Premier League for three reasons. Firstly, it is the most popular football league in the world followed by an estimated 3.2 billion global fanbase [1]. Secondly, it has extensive and reliable data across multiple online databases readily available for scraping. Thirdly, it has been consistently generating high-quality match content for the past decades, assuring that the data possess significant reference value representing football as a whole.

Data used for the app prototype illustrated below is as of September 16, 2023.

2.3 Core Functionalities

2.3.1 League Table

The live Premier League table of the current season will be provided in the "Table" tab so that users can easily access the table in just one click (see Figure 11). The table includes the matches played (PL), wins(W), draws (D), losses(L), goal difference (GD), and points (Pts) of all 20 clubs. To evaluate the home and away performance of each club, users can tap on the navigation bar on top of the screen (see Figure 12,13). In addition, when a club is clicked, users will be redirected to the information page of that club (section 2.3.2).

Figure 11. Table Tab　　Figure 12. Home table　　Figure 13. Away table

2.3.2　Club Information

In the "Clubs" tab, there will be a list containing all clubs in the current Premier League season (see Figure 14). Users can click into a club for more information, including an overview, the last and the next 3 games of the club. Besides, the "Stats" page provides in-depth club statistics, which includes a wide range of seasonal data segmented into General, Shooting, Passing, Pass Types, Goal and Shot Creation, Defensive Actions, Possession, Goalkeeping, and Miscellaneous data. Lastly, the "Players" page stores all players of the club according to their playing positions. Users can view general information about a player by tapping on the player's box (see Figure 15-17).

Figure 14. Clubs Tab　　Figure 15-17. Club Info, Stats, and Players page
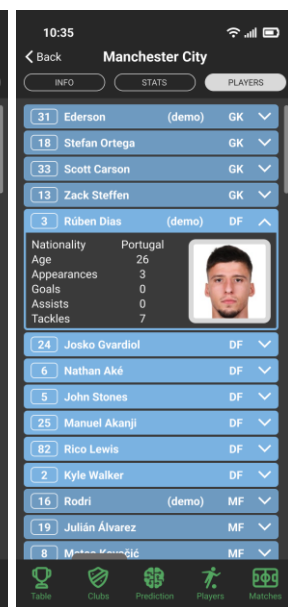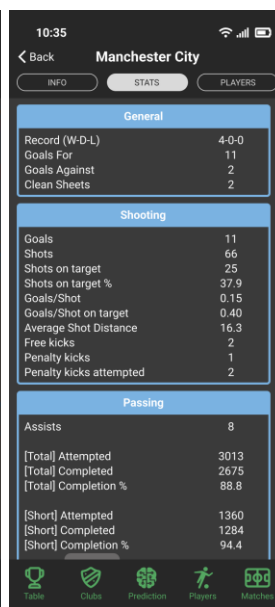
2.3.3 Club Comparison

Next to the club list is the page for club statistics comparison. Users can pick two teams and directly compare their seasonal data in a head-to-head manner (see Figure 18-20), saving the trouble of switching between tabs in other apps or mobile browsers.
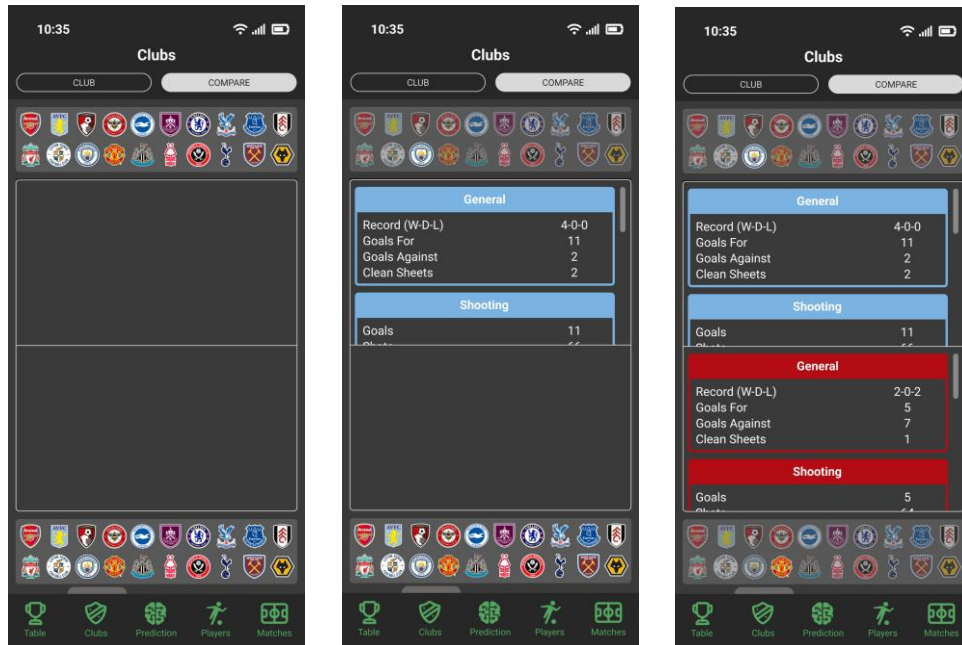
Figure 18-20. Picking two teams for comparison in the Clubs Tab.

2.3.4 Player Information

In the "Players" tab, all Premier League players are listed according to their playing positions, including forwards, midfielders, defenders, and goalkeepers (see Figure 21). Players of the same position are assigned to the same page for easy searching. Users can access the in-depth statistics of each player by simply tapping on the player's box. For goalkeepers, instead of standard statistics for on-field players, there will be specific goalkeeping statistics (see Figure 22,23).
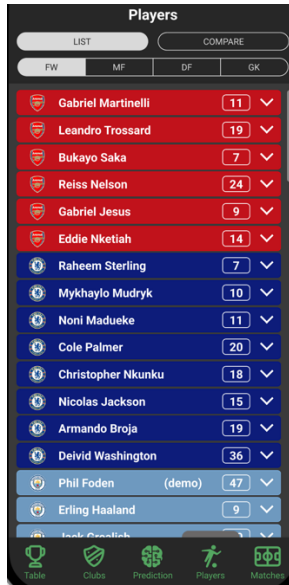
Figure 21. Players Tab   Figure 22&23. Statistics for players and goalkeepers
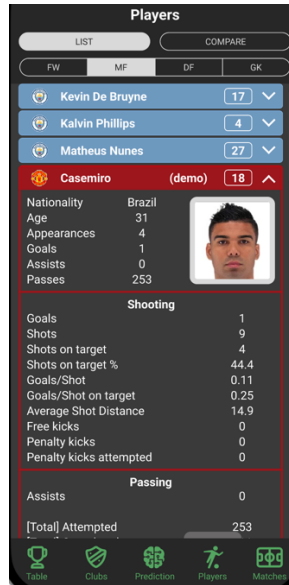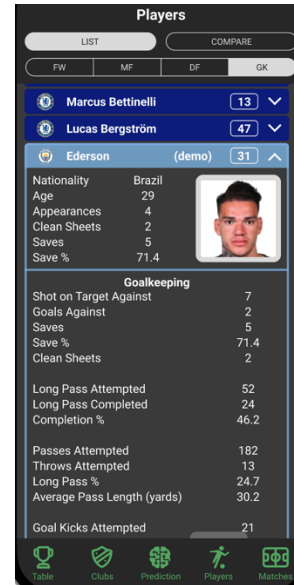
2.3.5 Player Comparison

Similar to the club comparison function, an interface will also be provided in the "Players" tab for direct comparison of player statistics. As a result, Cross-club and cross-positional comparisons can be made easily on a single page (see Figure 24-26). Comparison of players in the same club will also be supported.
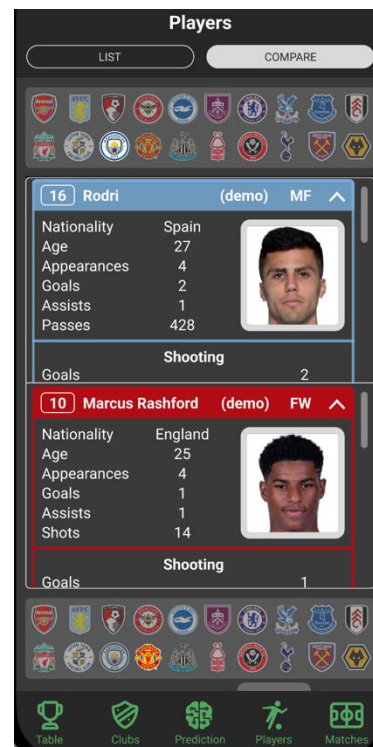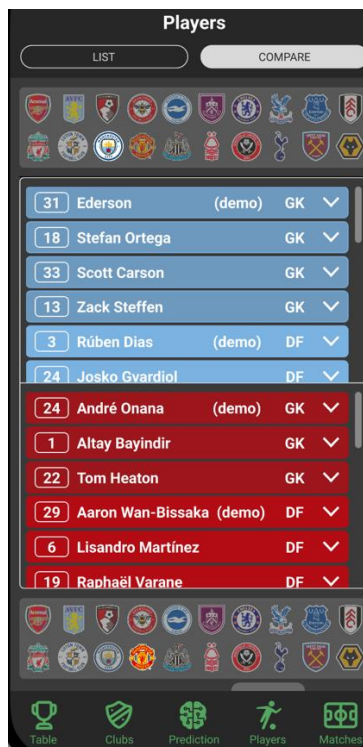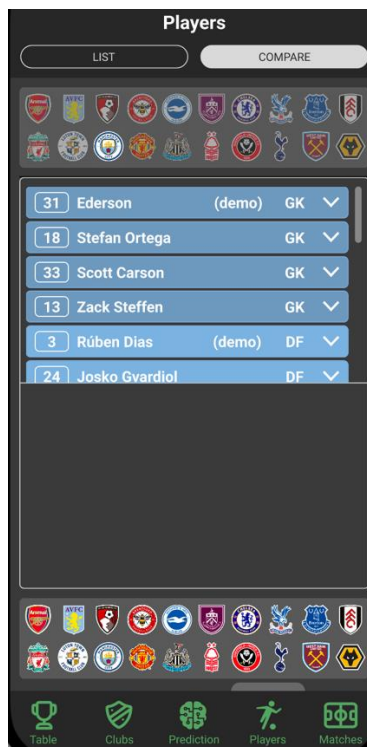
Figure 24-26. Picking two players for comparison in the Players Tab.

2.3.6 Matches Tab

All matches in the current matchweek will be displayed in real time. For example, assume that the match between Newcastle and Brentford is ongoing in the 69th minute, the live scoreline (1-0) is shown (see Figure 27). Other matches in the past are labeled as full-time (FT), while upcoming matches are shown with their starting time. With the navigation bar on top of the screen, users can also view previous results and future schedules for other matchweeks.

2.3.7 Match Centre

For each match, users can access the match centre by clicking the match in the "Matches" tab. The "Squad" page contains in-game statistics for that match, including the goalscorers and head-to-head indicators like possession, passing accuracy, etc. (see Figure 28), while the "Squad" page contains the formation, first team, and substitute players used by both sides in the match (see Figure 29).
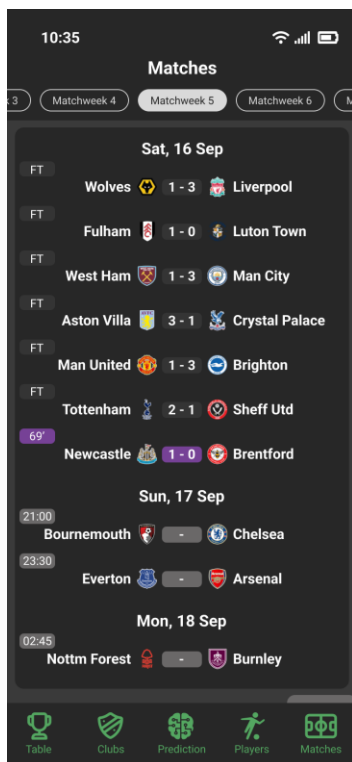
Figure 27. Matches Tab   Figure 28&29. Stats and Squad page in Match Centre
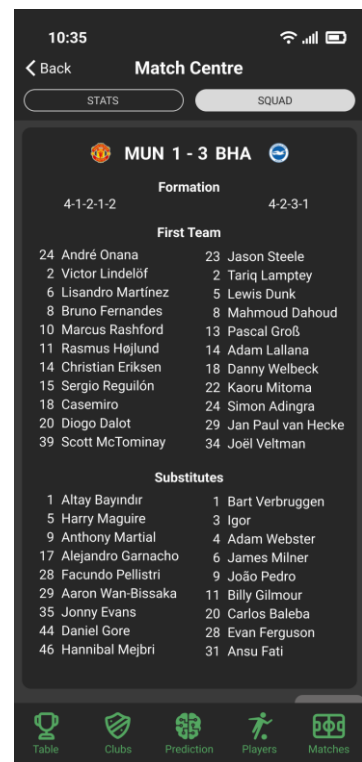
2.3.8 Machine Learning Prediction (Community Models)

The "Prediction" tab, which will be the default page of the app, contains the unique interactive machine learning elements. The "Community" page showcases machine learning predictive models created by app users (section 2.3.9). For each model, the model's name, algorithms selected, and attributes used are displayed (see Figure 30). The accuracy or F1 score attained by the model will also be shown. Users can leave their feedback for each model by clicking the like or dislike button.

To enhance the searching process, the models will be sorted by upload date by default, while users can also perform sorting by model accuracy, F1 score, or the "Likes" received (see Figure 31). In addition, at most two algorithm filters can be applied to include only models that contain the algorithm. For example, in Figure 32, only models that are trained on random forests are displayed.
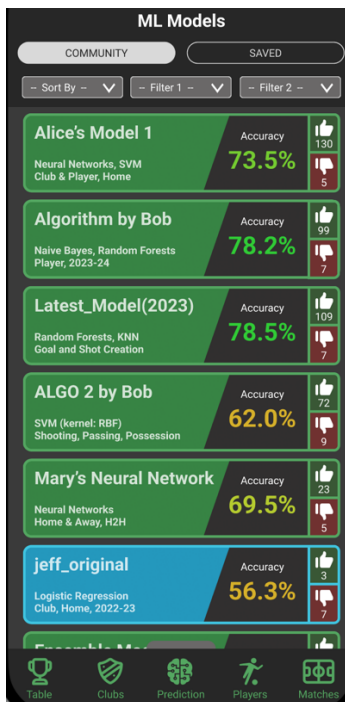
Figure 30. Community
ML models
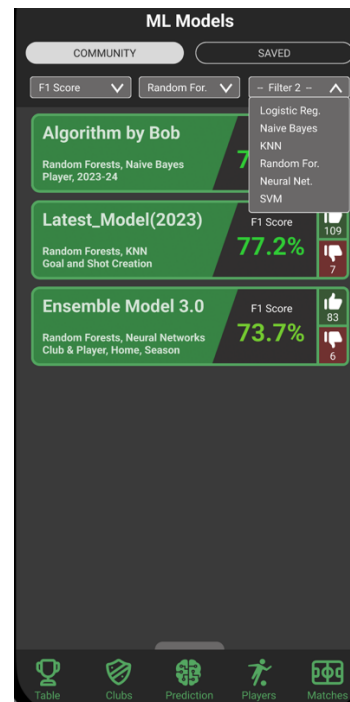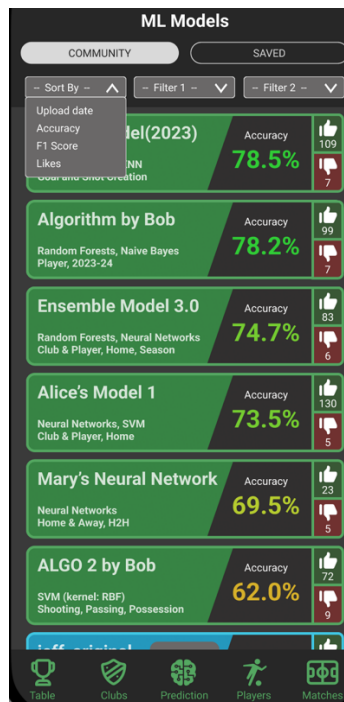
Figure 31&32. Model sorting and filtering

When a model is selected, information about it will be displayed in the "Data", "Training" and "Evaluation" tabs. The "Data" tab (see Figure 33) will list all of the data selected for the model, including the seasons used, data type, and in-game statistics separated into different areas. In each area, users can select all data available, include only specific ones, or not choose any for training the model.

The "Training" tab (see Figure 34) will show the algorithms used for training the model and their respective specifications. For example, the number of trees or neurons used, and the kernel for the support vector machine can be selected. Besides, weights can be assigned for different attributes for model optimization.

The "Evaluation" tab (see Figure 35) summarizes the model performance in different evaluation metrics, including accuracy, confusion matrix with F1 score, AUC-ROC plot, and the categorical cross-entropy loss. It allows users to conveniently access the performance of different models in the community.

To apply the model and obtain match prediction results, users can click the "Apply" button in the bottom left corner. Then, for every match, the predicted winning team will be shown in green, the losing team in red, and yellow for both teams if a draw is predicted (see Figure 36).
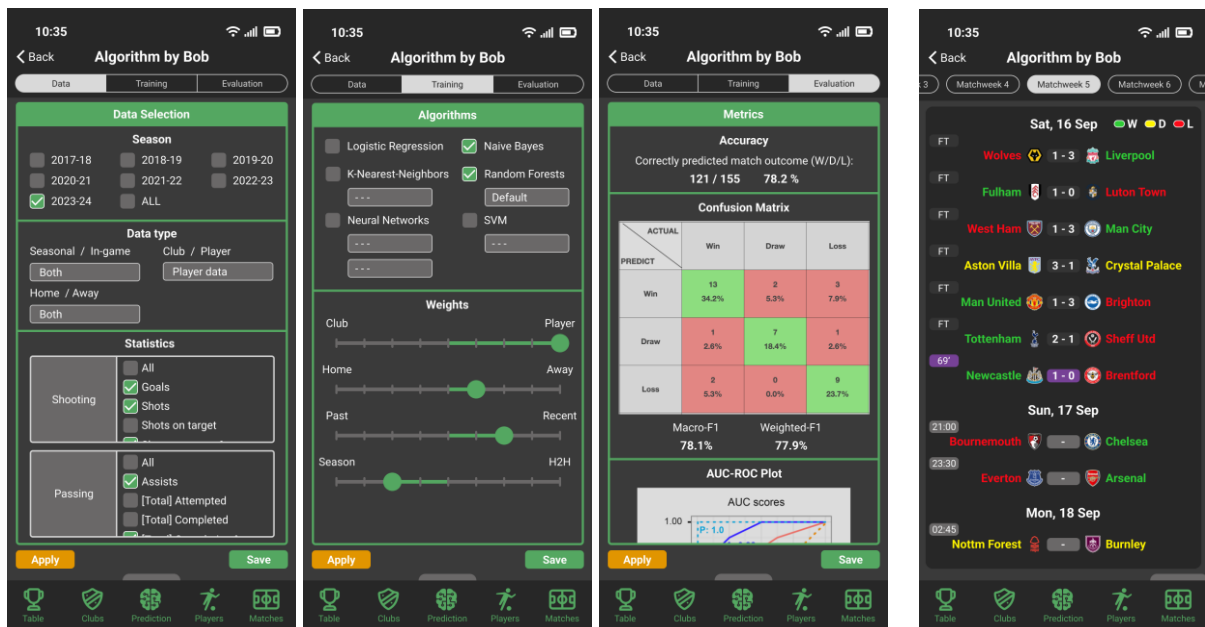
Figure 33.	Figure 34.	Figure 35.	Figure 36.
Data tab	Training tab	Evaluation tab	Prediction results

2.3.9 Creating Machine Learning Models

To facilitate user experience, a consistent interface with the community models' is implemented. To create a model, users should first select the data timeframe, appropriate data types, and statistics used for training the model (see Figure 37).

Then, users should select one or more algorithms for the model, and select the specifications (if necessary) for each algorithm used. Weights can be assigned to different attributes to further calibrate the model (see Figure 38).

Finally, evaluation results will be presented after the model is trained. The same metrics with the community models will be used (see Figure 39).
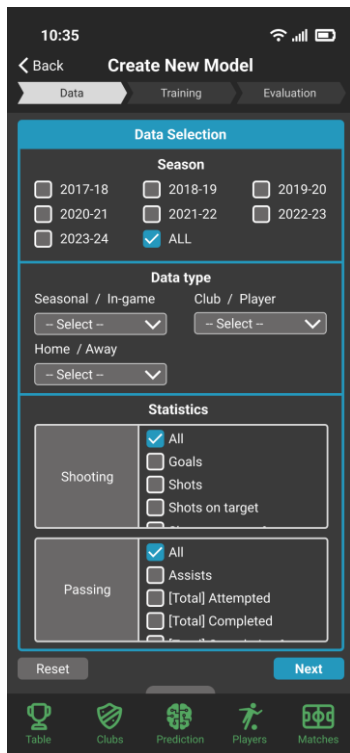
Figure 37. Data Tab
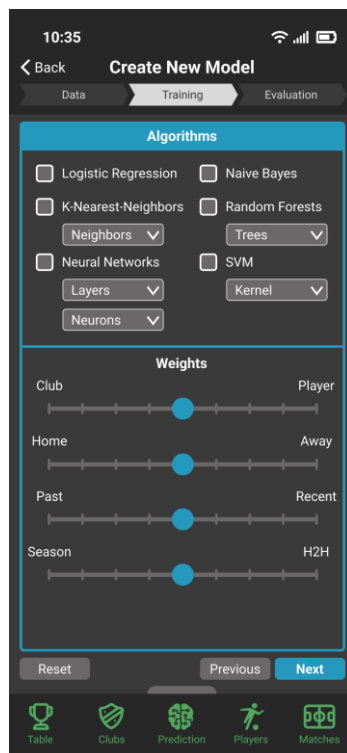
Figure 38. Training tab
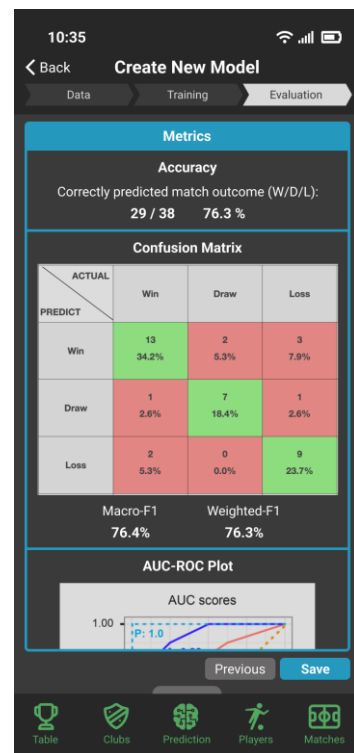
Figure 39. Evaluation tab

2.3.10 Machine Learning Prediction (Saved Models)

In the "Saved" page, there will be two types of models, including the ones saved from the "Community" page and the ones created by the user, which will be shown in green and blue respectively (see Figure 40). Sorting and filtering functions for these saved models are also supported.

When the user selects a model created by himself or herself, the model information page will be provided (see Figure 41,42). Besides applying the model to view match prediction results, users can update it by re-selecting or adding extra parameters. The model can also be renamed and published to the community. If needed, users can unpublish or delete the model anytime.
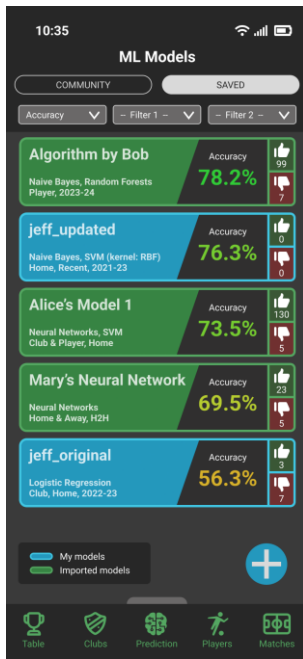
Figure 40. "Saved" page   Figure 41&42. Functions for users' own model
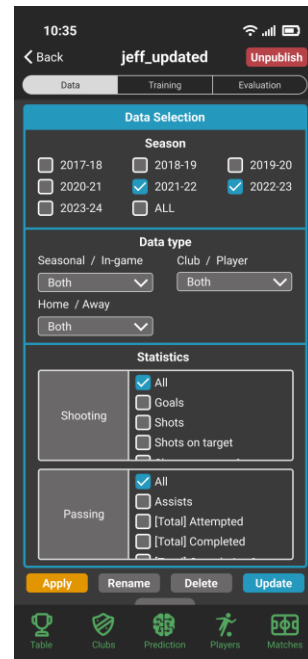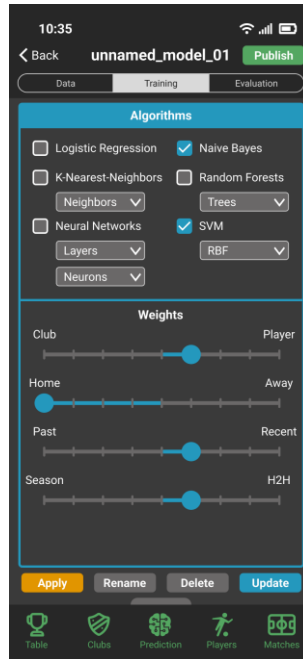
3. Methodology

3.1 Frontend development

3.1.1 Prototype

A complete prototype of the app has been created in Figma (see Figure 43). The GUI designs will be used as a reference for initial frontend development and may be modified and optimized in the latter part of the project.

The narrated prototype demonstration can be accessed via the following link. https://youtu.be/ymzIQAWuqTk

Figure 43. Figma prototype design for KickInsights

3.1.2 UI and UX

Navigation within the app will be streamlined by implementing a convenient and consistent user interface. As shown in the prototype, users can easily switch between the 5 main tabs with the navigation menu at the bottom of the screen, and access each page with the bar on top of the screen. Distinct coloring for different components will be used to allow easy identification of functions. For example, green is used for community machine-learning models, blue is used for user-generated machine-learning models, and a variety of club colors are used for their respective club and players' statistics.

To enhance user experience, all statistics are systematically grouped by teams, players, or matches. Users can locate in-depth data in under 3 clicks, which is lower than current app alternatives in general. Besides, loading screens and transition animations will be deployed to sustain users' immersiveness in the app.

3.1.3 Application development

Launching the KickInsights app on IOS will be the primary focus of this project. However, to allow further expansion into Android devices for possible cross-platform availability in the future, React Native will be used with Visual Studio Code as the IDE to first develop code in JavaScript, and then convert the code into native languages.

Expo will also be used to facilitate the development process. It contains a well-established set of tools and services oriented around React Native for app developers to write, build, submit, and monitor their mobile apps [2]. It also supports testing on emulators and mobile devices.

3.2 Backend development

3.2.1 Data Collection

The data source for seasonal and in-game statistics will be FBREF, one of the best football data providers constantly collecting team and player statistics, covering the sports comprehensively in a data-driven approach [3]. In addition, Transfermarkt will be used mainly as the source of club and player information.

3.2.2 Data Wrangling

Web scraping will be carried out on the following endpoints. In-depth team data, on-field player data, and goalkeeper data will be collected. (For respective definitions, visit Appendix A, B, C)

1) FBREF:
   https://fbref.com/en/comps/9/Premier-League-Stats
2) Transfermarkt:
   https://www.transfermarkt.com/premier-league/startseite/wettbewerb/GB1

To scrape data from FBREF, the following projects will be referenced.

1) soccerdata:
   https://github.com/probberechts/soccerdata
2) worldfootballR:
   https://github.com/JaseZiv/worldfootballR

After obtaining data from the aforementioned data sources, cleaning will then be performed to remove redundant attributes and maintain consistency. For example, players' full names and playing positions will be unified. Besides, additional columns will be created in the database for the shortened club and player names.

3.2.3 Application development

The application backend and database will be constructed by Firebase, Google's BaaS which provides robust infrastructure for data storage. The distributed architecture not only provides load balancing for app usage but also syncs data in real time across data centres, preventing bottlenecks at any specific server.

The Firebase Realtime Database will store the updated clubs and players' statistics, while Firestore will store app resources available for offline access, including club and player images, community models, and statistics since the last update.

Google's Cloud Run API will be used to serve app requests to build, train, and evaluate the machine learning models. Related Python libraries like scikit-learn and TensorFlow will be used for developing the machine learning components.

4. Project Schedule and Milestone

4.1 Budgets

Service	Provider	Price
Firebase BaaS	Google	$400*
Firestore	Google	$50*
Cloud Run	Google	$300*
Estimated total expenditure		$750

* The listed Google services provide a free tier but subsequent usage will be charged.
   The price is approximated for the estimated usage.

4.2 Timeline

The following table outlines the tentative schedule of this project.

Time Period	Tasks / Deliverables	Status
September 2023	- Detailed Project Plan - Project web page - UI / UX Design - App prototype	Completed
October 2023	- Frontend development - Backend and database design - Data collection and wrangling - Research on machine learning models	In-progress
November 2023 - December 2023	- Backend and database development - Integration of machine learning models	TBC
January 2024	- First presentation - Preliminary Implementation - Detailed Interim Report	TBC
February 2024 - March 2024	- Application testing - Codebase review - Improvement in UI / UX - Optimization of machine learning models	TBC
April 2024	- Final presentation - Final tested implementation - Final report	TBC
May 2024	Project exhibition	TBC

5. References

[1] Ádomás. (2023a, August 20). *Top 10 most watched football leagues in the world*. SportingWiki. https://www.sportingwiki.com/top-10/top-10-most-watched-football-leagues/

[2] Expo. (n.d.). *The world's most loved ecosystem of tools that help you develop review & deploy*. Expo. https://expo.dev/

[3] Toby . (2023, May 12). *Best sites for Free Football Statistics: Top Soccer Stats websites*. Punter2Pro. https://punter2pro.com/free-football-statistics-soccer-stats/#:~:text=Fbref%20is%20a%20website%20that,match%20breakdowns%20to%20player%20transfers.

6. Appendices

Appendix A

Definition of in-depth team data:
Contains at least one datum from each of the following areas.

General:
Record (W-D-L), Goals For, Goals Against, Clean Sheets

Shooting:
Goals, Shots, Shots on target, Shots on target %, Goals/Shot, Goals/Shot on target, Average Shot Distance, Free kicks, Penalty kicks, Penalty kicks attempted

Passing:
Assists, [Total] Attempted, [Total] Completed, [Total] Completion %, [Short] Attempted, [Short] Completed, [Short] Completion %,
[Medium] Attempted, [Medium] Completed, [Medium] Completion %, [Long] Attempted, [Long] Completed, [Long] Completion %

Pass Types:
Live-ball passes, Dead-ball passes, Free Kicks, Through Balls, Switches, Crosses, Throw-Ins, Corner Kicks, Offsides, Passes blocked

Goal and Shot Creation:
Shot Creating Actions: [SCA] Live-Ball passes, [SCA] Dead-Ball passes, [SCA] Take-Ons, [SCA] Shots, [SCA] Fouls, [SCA] Defensive actions
Goal Creating Actions: [GCA] Live-Ball passes, [GCA] Dead-Ball passes, [GCA] Take-Ons, [GCA] Shots, [GCA] Fouls, [GCA] Defensive actions

Defensive Actions:
Tackles, Tackles Won, Tackles in Def 3rd, Tackles in Mid 3rd, Tackles in Att 3rd, Challenges, Challenges Won, Blocks, Shots blocked, Passes blocked, Interceptions, Clearances, Errors

Possesion:
Possession, Touches, Touches in Def penalty area, Touches in Def 3rd, Touches in Mid 3rd, Touches in Att 3rd, Touches in Att penalty area, Take-Ons, Take-Ons Success, Take-Ons Tackled, Carries, Total Distance (yards), Progressive Distance (yards), Progressive Carries, Carries into Att 3rd, Carries into Att penalty area, Miscontrols, Dispossessed, Passes Received, Progressive Passes Received

Goalkeeping:
Shot on Target Against, Goals Against, Saves, Save %, Clean Sheets, Long Pass Attempted, Long Pass Completed, Completion %, Passes Attempted, Throws Attempted, Long Pass %, Average Pass Length (yards), Goal Kicks Attempted, Long Goal Kick %, Average GK Length (yards), Crosses Faces, Crosses Stopped, Actions outside penalty area, Average action distance (yards)

Miscellaneous:
Yellow Cards, Red Cards, Second Yellow Cards, Fouls Committed, Fouls Drawn, Penalty Kicks Won, Penalty Kicks Conceded, Own Goals, Ball Recoveries, Aerial Duels Won, Aerial Duels Lost, Aerial Duels Won %

Appendix B

Definition of in-depth on-field player data (Defender, Midfielder, Forward):
Contains at least one datum from each of the following areas.

General:
Nationality, Age, Appearances, Goals, Assists, Tackles

Shooting:
Goals, Shots, Shots on target, Shots on target %, Goals/Shot, Goals/Shot on target,
Average Shot Distance, Free kicks, Penalty kicks, Penalty kicks attempted

Passing:
Assists, [Total] Attempted, [Total] Completed, [Total] Completion %, [Short]
Attempted, [Short] Completed, [Short] Completion %,
[Medium] Attempted, [Medium] Completed, [Medium] Completion %, [Long]
Attempted, [Long] Completed, [Long] Completion %

Pass Types:
Live-ball passes, Dead-ball passes, Free Kicks, Through Balls, Switches, Crosses,
Throw-Ins, Corner Kicks, Offsides, Passes blocked

Goal and Shot Creation:
Shot Creating Actions: [SCA] Live-Ball passes, [SCA] Dead-Ball passes, [SCA]
Take-Ons, [SCA] Shots, [SCA] Fouls, [SCA] Defensive actions
Goal Creating Actions: [GCA] Live-Ball passes, [GCA] Dead-Ball passes, [GCA]
Take-Ons, [GCA] Shots, [GCA] Fouls, [GCA] Defensive actions

Defensive Actions:
Tackles, Tackles Won, Tackles in Def 3rd, Tackles in Mid 3rd, Tackles in Att 3rd,
Challenges, Challenges Won, Blocks, Shots blocked, Passes blocked,
Interceptions, Clearances, Errors

Possesion:
Possession, Touches, Touches in Def penalty area, Touches in Def 3rd, Touches in
Mid 3rd, Touches in Att 3rd, Touches in Att penalty area, Take-Ons, Take-Ons
Success, Take-Ons Tackled, Carries, Total Distance (yards), Progressive Distance
(yards), Progressive Carries, Carries into Att 3rd, Carries into Att penalty area,
Miscontrols, Dispossessed, Passes Received, Progressive Passes Received

Miscellaneous:
Yellow Cards, Red Cards, Second Yellow Cards, Fouls Committed, Fouls Drawn,
Penalty Kicks Won, Penalty Kicks Conceded, Own Goals, Ball Recoveries, Aerial
Duels Won, Aerial Duels Lost, Aerial Duels Won %

Appendix C

Definition of in-depth player data (Goalkeeper):
Contains at least one datum from each of the following areas.

General:
Nationality, Age, Appearances, Goals, Assists, Tackles

Goalkeeping:
Shot on Target Against, Goals Against, Saves, Save %, Clean Sheets, Long Pass Attempted, Long Pass Completed, Completion %, Passes Attempted, Throws Attempted, Long Pass %, Average Pass Length (yards), Goal Kicks Attempted, Long Goal Kick %, Average GK Length (yards), Crosses Faces, Crosses Stopped, Actions outside penalty area, Average action distance (yards)

Miscellaneous:
Yellow Cards, Red Cards, Second Yellow Cards, Fouls Committed, Fouls Drawn, Penalty Kicks Won, Penalty Kicks Conceded, Own Goals, Ball Recoveries, Aerial Duels Won, Aerial Duels Lost, Aerial Duels Won %