Player Performance project documentation in French

Introduction to the project

We will analyze a player's performance during the 24/25 season according to various factors (goals, assists, tackles) and prisms (age, price, nationality, etc.).
To do this, we will use two sources: Fbref for statistics and TransferMarkt for the market value of each player.

Data extraction / Data pre-processing

After importing the important libraries for this part, we're going to start by retrieving data from Kaggle users who have web scraped the Fbref sites : https://www.kaggle.com/datasets/hubertsidorowicz/football-players-stats-2024-2025 and TransferMarkt : https://www.kaggle.com/datasets/davidcariboo/player-scores .

For the TransfertMarkt file, we'll filter the data so as to include only players who took part in the 2024/2025 season, and those in one of the 5 major European leagues (Ligue 1, Premier League, Bundesliga, Serie A and Liga). For the Fbref file, we will carry out a transformation of the data requiring to be calculated by 90 minutes. In addition, we will only take into account players who have played more than 600 minutes. Note that players who played for 2 clubs during the season will have their statistics aggregated, depending on the type of variable.

A major step in data pre-processing was to associate the two data providers for each player. To do this, we used several stratagems to include as many players as possible without making mistakes. The 1st was to associate players with the same name + same league + same year of birth + position if it was a goalkeeper (the only position where there was no difference between the providers). This phase covered the majority of players, but given that we noticed certain variations in the names between the two suppliers, we had to go further than this 1st step. We therefore used a fuzzy function, following this logic: if the % resemblance is satisfactory + same league + same year of birth + position if it was a goalkeeper, then it's considered to be the same player. To limit the number of errors, we've split the phases into several parts. Here's the breakdown, with the number of players recovered in each phase:

- Identical names (same year, league, goalkeeper position): 1645

- Matches at 90% (same year, league, goalkeeper position): 39

- Matches at 75% (same year, league, goalkeeper position): 42

- Matches at 90% (same league, goalkeeper position) : 4

- Matches at 65% (same year, league, goalkeeper position): 13

- Matches at 80% (same league, goalkeeper position): 4

- Matches at 60% (same year, league, keeper position): 8

- Total matched: 1751.

It should be noted that, logically, the lower the % of matches on the name, the less likely it is that the correct association will be made between the two suppliers. We have therefore decided not to go below 65% matches, in order to maintain an error-free association.

This list of players, together with the columns of interest for further processing, will be saved in a csv file named database_player.csv.

Pages list

The application will consist of 4 pages:
- Home: Presentation of the project and its resources
- Player analysis: Analysis of the player of your choice through several statistics
- Player Comparison: Comparative analysis of two players in the same position
- Player Ranking: Ranking of players by performance according to a chosen statistic

Project tree for application implementation

Application
- image
- CV
- data
- documentation
- Player_performance.py: Main
- README.md
- requirements.txt

Setting up the application

It's important to point out that the application will contain 2 versions: French and English.

Home

As explained above, the welcome section briefly introduces the project components, and provides access to various resources (documentation, CVs).

Analysis section

At the head of the project

Several functions will be created, such as adapting the format of the value of players on the market, translating countries or positions, categorizing positions into sub-categories (with the list of statistics to be used), or creating a function that calculates the similarity between each player with the same position based on their advanced statistics over the whole season.

Application display

The home page will display the project logo, followed by an explanation of the project and associated resources. Several files will be available for download, such as documentation and CVs.

For the player analysis, firstly, the user will be asked to select the player of his choice. This page will contain the player's profile with basic information (Name, Photo, Position, Club etc.), his statistical radar, and a table of the 5 players who most resemble him statistically. Note that the radar can be customized according to country, league and age group. The statistics displayed on the radar are those that appear to be of interest in relation to the player's position (in blue), and will be compared with the median of the chosen group (in red). A glossary of statistics is also available, and can be unfolded if required.

The player comparison page will follow the same logic, with the option of choosing two players (from the same position) that the user wishes to analyze. Their profile will be displayed side by side, followed by their respective radar (in blue and red).

On the ranking page, users will be asked which statistic they wish to analyze. Based on this choice, a ranking of the top players according to this metric will be displayed, with a podium for the top 3, followed by the table ranking with basic information on these players. The sidebar will contain optional filters (Position, Club, Championship, Age Group, Market Value), as well as a statistics glossary. In addition, an image will be displayed before the user's choice, so as not to leave the page empty (as is the case for every page).