**Streamlit URL:** https://fantasypremierpredictions.streamlit.app/

**Model Confidence:**

**Goalkeeper Model -**

**MSE:** 0.4091279629629629, **R2:** 0.9332728636436856

**Defender Model -**

**MSE:** 0.9503198634066367, **R2:** 0.8756530747069646

**Midfielder Model -**

**MSE:** 0.8707418365598735, **R2:** 0.9161520777850422

**Forward Model -**

**MSE:** 0.5189986725925925, **R2:** 0.9654291791993306

**Fantasy Premier League Prediction App**

A web application designed to provide player performance predictions and team-building recommendations for Fantasy Premier League enthusiasts.

**Overview**

This app leverages historical player performance data and advanced prediction algorithms to help users optimize their Fantasy Premier League teams. Users can explore predicted points for players, get recommendations for value-based team building, and interact with a chatbot for specific queries.

**Features**

Player Performance Predictions

Search Players: Get predicted points for individual players based on historical data.

Team Recommendations: Suggests players based on position, value constraints, and playing status.

Interactive Chatbot

Player Insights: Ask about a player's predicted points or value.

Recommendations: Get position-specific recommendations based on budget constraints.

Team Management Tools

Formation Selector: Dynamically choose formations (e.g., 4-4-2, 3-5-2).

Bench and Starting XI Management: Visualize and organize your Fantasy team.

**Technologies Used**

Streamlit: For building the interactive web app.

Pandas: For efficient data manipulation and analysis.

FPL API: To fetch live and historical Fantasy Premier League data.

Setup and Usage

**Usage**

Enter player names in the chatbox to get their predicted points.

Use the recommend keyword to get position-based player suggestions within a budget (e.g., "Recommend a midfielder under 6.5M").

Customize your formation and organize your Fantasy team.

**Code Structure**

Main.py: calls model cleaning functions and prepares API data for predictions.

Function.py : Runs Azure time triggered function to update csv files with player data

FPL\_Predictions.py: Main application script with Streamlit integration.

tabs.py: Contains reusable functions for team and player management.

utils.py: Helper functions for API requests and data manipulation.

assets/: Includes visual assets like player jerseys and team logos.

**Future Enhancements**

Enhanced Prediction Accuracy:

Incorporate machine learning models for more precise forecasts.

Graphical Analysis:

Add graphs to visualize player and team performance trends.

Mobile Responsiveness:

Optimize for better mobile user experience.

Team Comparison Feature:

Allow users to compare predicted points across different team setups.