Fantasy Premier League
Points Prediction

Strategic Context

Engineering the Solution

Insights and Growth

Premier League

Strategic Context

Engineering the Solution

Insights and Growth

Fantasy Premier League is Popular for Its Strategic Appeal

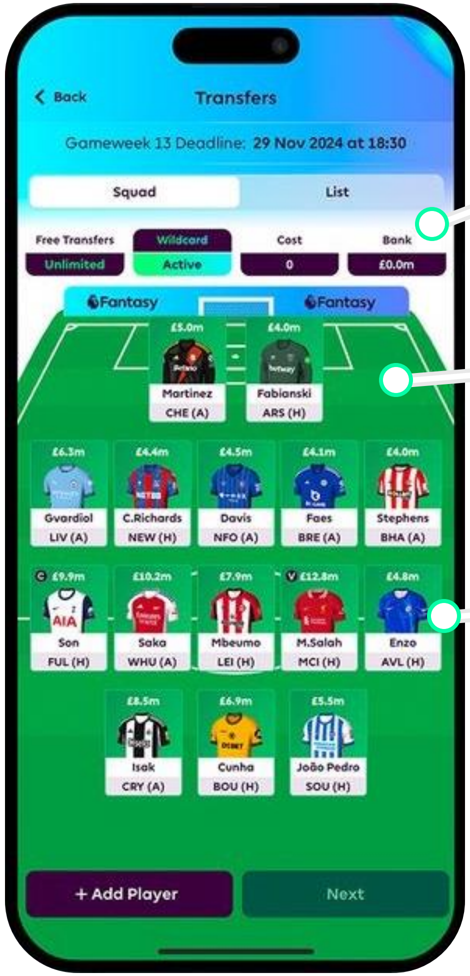
Fantasy Premier League (FPL) Overview

- Global online game where players act as managers
- Over 11 million active players worldwide
- Compete for season-long rankings

Reasons for Popularity

1. Combines football knowledge with planning
2. Encourages community interaction and competition

FPL Rules Shape Strategic Decision-Making



FPL Rules

Budget of **£100m**

15 Players

1 Transfer/week

Starting lineup must be chosen before deadline

Player prices fluctuate due to demand increases/decreases

Budgets add additional pressure to managers due to fluctuating player prices

FPL’s Point System Rewards Key Player Actions

Position Dependent Points

Goalies

- Goal: **10**
- CS*: **4**
- Shots*: **1**
- GA*: **-2**

Defenders

- Goal: **6**
- CS: **4**
- GA*: **-2**

Midfielders

- Goal: **5**
- CS*: **1**

Attackers

- Goal: **3**
- CS*: **0**

CS: Clean Sheet*
Shots: For every 3 shots saved*
GA: For every 2 goals conceded
OG: Own Goal*

Constant Points

Minutes

- < 60 mins: **1**
- > 60 mins: **2**

Player Conduct

- Red: **-3**
- Yellow: **-1**

Additional

- Assist: **3**
- OG*: **-2**

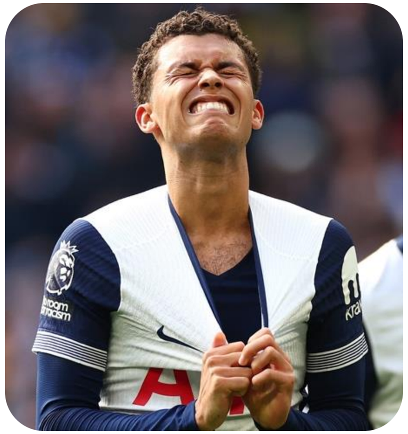
FPL Points System Makes Player Selection Complex

Player Selection Challenges

GW16
CHE (A)
GW17
NFO (H)
GW18
BHA (A)
GW19
ARS (H)
GW20
SOU (A)
GW21
MCI (H)
FIXTURE DIFFICULTY RATINGS
Mbeumo
£7.6m • MID
29.1% Ownership

Trade-offs

- Managers must balance form, fixtures, and budget



Uncertainty

- Risk associated with form, injuries, and rotation

Enzo • MID • £4.9m
4
GW11
ARS (H)
12
GW12
LEI (A)
9
GW13
AVL (H)
5
GW14
SOU (A)
7
GW15
TOT (A)

Inherent Bias

- Managers often balance inherent biases about certain players

The challenges incentivize selecting influential, guaranteed starters

Predicting Player Performances Help Optimize FPL Strategies

Business Objective

Create a predictive system to digest player data efficiently and support decision making

Misclassification Consequences

- False Negatives: Players who will perform well will not be selected, leading to uncaptured points
- False Positives: Players who will not perform well will be selected, leading to wasted resources

Key Performance Indicators

1. Model Test Metric Performance
2. Net Points Gained
3. Team Efficiency Metrics

Scalability and Future Expansion

- Display predictions and key player stats in interactive features in dashboards
- Incorporate external data feeds such as betting odds, live updates, and injury reports

Current Model Excels in Selecting Top-10 Players

Model Business Metrics

Top 10 Accuracy: **80%**

- Measures the proportion of actual top-10 players per position correctly predicted

Top 10 Accuracy (weighted): **52%**

- Gives more weight to correctly predicted players that are highly ranked

Per Position Top 10 Accuracy

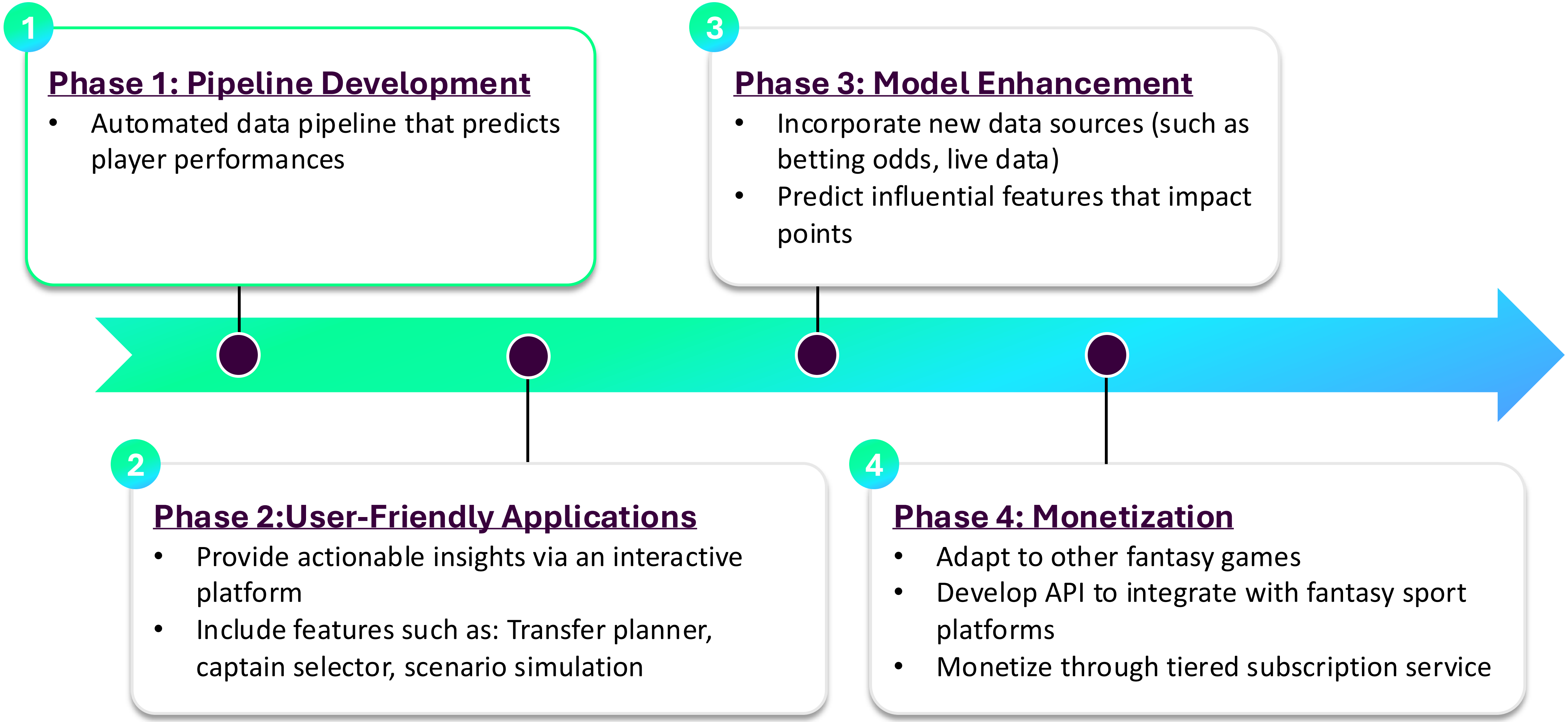
DEF: **79%** MID: **75%** FWD: **92%**

Model 2024-25 Season Selected GW Results

Game Week	Predicted Best	Predicted Players Performance	Best Players Performance
3	54	27	58
4	45	35	58
5	47	40	62
6	51	58	88

While the model can predict top players, it struggles selecting a team

Implementing 4-Phased Approach Unlocks Model’s Full Potential



1

Phase 1: Pipeline Development

- Automated data pipeline that predicts player performances

2

Phase 2:User-Friendly Applications

- Provide actionable insights via an interactive platform
- Include features such as: Transfer planner, captain selector, scenario simulation

3

Phase 3: Model Enhancement

- Incorporate new data sources (such as betting odds, live data)
- Predict influential features that impact points

4

Phase 4: Monetization

- Adapt to other fantasy games
- Develop API to integrate with fantasy sport platforms
- Monetize through tiered subscription service

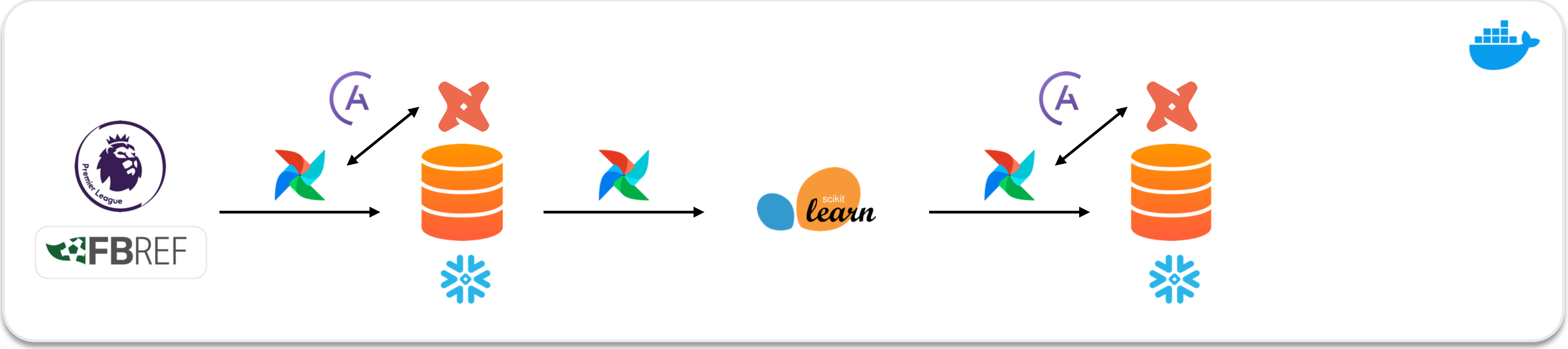
Strategic Context

Engineering the Solution

Insights and Growth

Key Technologies Power a Scalable System

Pipeline Overview



Key Technologies

Apache Airflow

- Orchestrator

Snowflake

- Database

DBT

- Data Transformer

Docker

- Container Platform

A Reliable Data Pipeline Powers Predictions

Pipeline Stages

1 Data Extraction

- Pull raw data from sources
- Key challenge: Managing rate limits

2 Data Transformation

- Clean the raw data into a usable format
- Key challenge: Designing generalizable transformations

3 Data Loading

- Store processed data securely for analysis
- Key challenge: Ensuring low upload costs

Model Requirements Align with FPL's Challenges

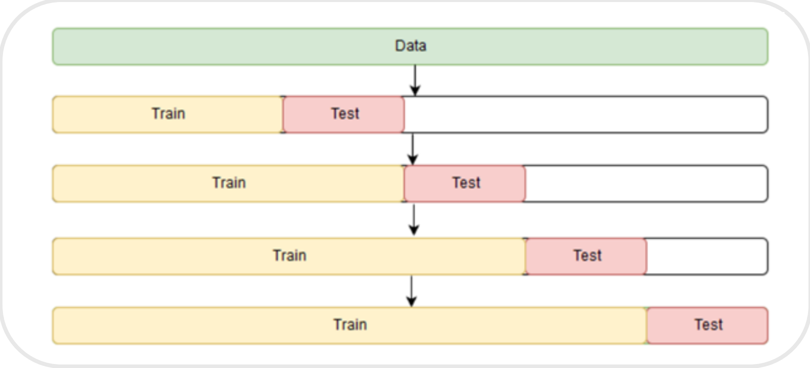
Model Requirements

- Multi-output (goals and assists)
- Interpretability is not a priority
- Performs well with small-sized data

Fixture-Based Components were Necessary

Cross Validation

Train Test Split



Feature Extraction

Raw Data: Goals, Assists, xG, xA

Derived Features: Rolling un/bounded windows

Important to account for players who rarely play

Modeling Results Highlight Strengths and Opportunities

Evaluation Required Creative Solutions

- Track model performance for goal and assist predictions
- Root Mean Squared Error (RMSE) penalized large errors
- Mean Absolute Error (MAE) provided a balanced view

Let: w_g represent the weight for goals ($goal\_weight$),

w_m represent the weight for RMSE vs. MAE ($metric\_weight$)

$$\text{RMSE}_{\text{goals, norm}} = \frac{\text{RMSE}_g}{\text{RMSE}_g + \text{RMSE}_a}, \quad \text{RMSE}_{\text{assists, norm}} = \frac{\text{RMSE}_a}{\text{RMSE}_g + \text{RMSE}_a}$$

$$\text{MAE}_{\text{goals, norm}} = \frac{\text{MAE}_g}{\text{MAE}_g + \text{MAE}_a}, \quad \text{MAE}_{\text{assists, norm}} = \frac{\text{MAE}_a}{\text{MAE}_g + \text{MAE}_a}$$

$$\text{RMSE}_{\text{combined}} = w_g \cdot \text{RMSE}_{\text{goals, norm}} + (1 - w_g) \cdot \text{RMSE}_{\text{assists, norm}}$$

$$\text{MAE}_{\text{combined}} = w_g \cdot \text{MAE}_{\text{goals, norm}} + (1 - w_g) \cdot \text{MAE}_{\text{assists, norm}}$$

$$\text{Combined Metric} = w_m \cdot \text{RMSE}_{\text{combined}} + (1 - w_m) \cdot \text{MAE}_{\text{combined}}$$

Model Results

Goals

RMSE: **0.31**

MAE: **0.20**

Assists

RMSE: **0.28**

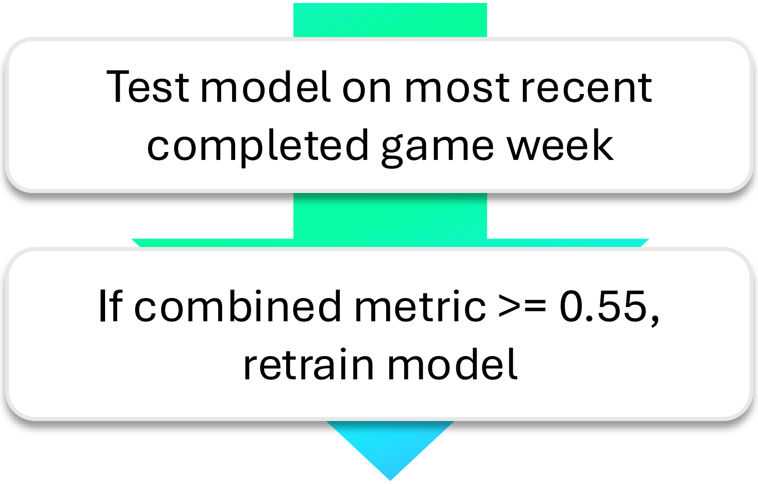
MAE: **0.16**

Combined: 0.50

goal_weight = 0.7
metric_weight = 0.5

Training and Prediction Workflows Ensure Accuracy and Timeliness

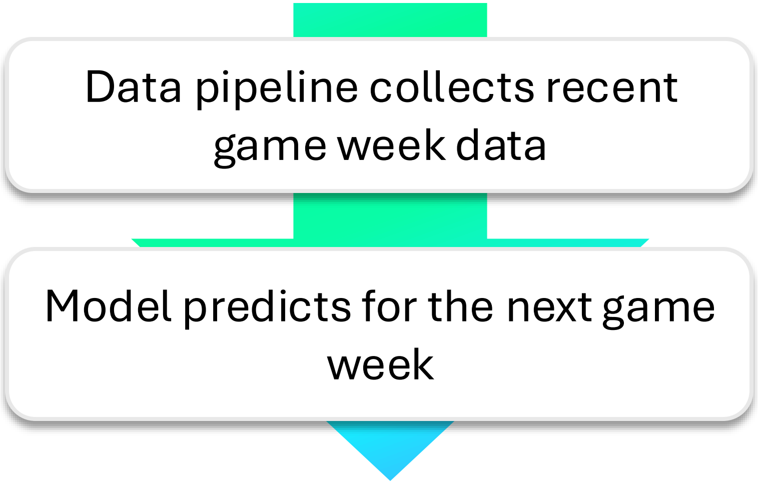
Training Workflow (Monthly)



Test model on most recent completed game week

If combined metric >= 0.55, retrain model

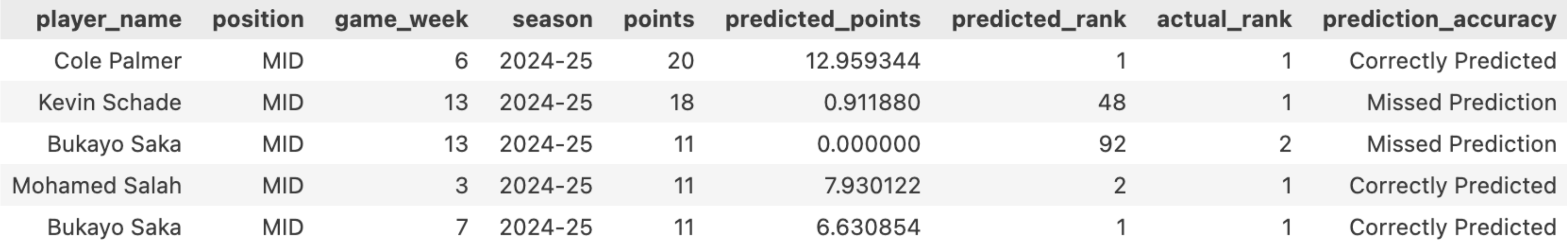
Prediction Workflow (Weekly)



Data pipeline collects recent game week data

Model predicts for the next game week

fct_fpl_prediction_accuracy Table Example



player_name	position	game_week	season	points	predicted_points	predicted_rank	actual_rank	prediction_accuracy
Cole Palmer	MID	6	2024-25	20	12.959344	1	1	Correctly Predicted
Kevin Schade	MID	13	2024-25	18	0.911880	48	1	Missed Prediction
Bukayo Saka	MID	13	2024-25	11	0.000000	92	2	Missed Prediction
Mohamed Salah	MID	3	2024-25	11	7.930122	2	1	Correctly Predicted
Bukayo Saka	MID	7	2024-25	11	6.630854	1	1	Correctly Predicted

Addressing Key Bottlenecks Drastically Enhances the System

Severity of Areas to Improve

1 Optimize Data Collection

- Fbref Data Source unreliable
- 10 requests/minute rate limit creates bottleneck

Explore alternative sources

2 Feature Engineering

- Additional data sources can provide new perspectives on player performance

Incorporate new sources

3 Prediction Evaluation

- Current metrics don’t fully align with actionable insights
- New metrics provide better feedback

Identify patterns in errors

Addressing these key improvements will enhance its reliability, scalability, and overall impact, paving the way for more actionable insights and a seamless user experience

Strategic Context

Engineering the Solution

Insights and Growth

Mastering Modern Tools and Languages

Use Cases for Technologies/Languages

- **Python**: Advanced usage for data preprocessing, feature engineering, and machine learning
- **SQL**: Optimized queries to handle large datasets efficiently
- **Snowflake**: Designed and queried scalable data warehouses for seamless storage and retrieval
- **Airflow**: Built automated workflows to schedule and monitor complex data pipelines
- **dbt**: Created robust transformation models for clean and modular data
- **Docker**: Containerized environments to streamline development and deployment

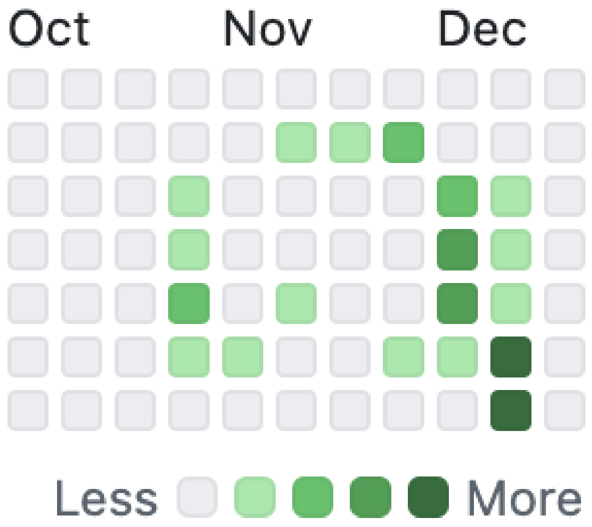
List of Technologies/Languages

Building Scalable and Maintainable Systems

General Software Engineering Principles

- **System Design Principles:** Designed scalable systems from high-level architecture to detailed implementation
- **Reusable Code:** Developed modular functions to streamline workflows and avoid redundancy
- **Version Control with Git:** Adopted professional practices for meaningful and constant commits
- **Key Takeaway:** Developed practices essential for professional software development

Commit History



Leading Through Challenges and Uncertainty

Key Takeaways

- **Leadership Without Deadlines:** Shifted from time-based to task-oriented checklists for better flexibility and focus. Learned that rigid schedules are impractical for exploratory and iterative tasks.

- **Embracing Iteration:** Acknowledged that first attempts often fail. Debugging, redesigning, and reworking were integral to progress.

- **Breaking Down Complexity:** Tackled complex tasks (e.g., Snowflake integration, Airflow pipelines) by dividing them into smaller, manageable chunks.

Thank You!