Criterion A: Planning

Defining the problem

My client, Mr. X, is a passionate player of the online fantasy football game Fantasy Premier League (FPL) and is currently struggling to perform well in the game. The game requires its players to make a team of 11 footballers with a limited budget and various game-specific restrictions (for example, player positions, number of players from one club, etc.) and rewards them for the performances of these players in real-life games. Mr. X is having a hard time making the best possible team, as he explains, "I always end up making the wrong decisions, and waste my money on unnecessary transfers in the game (Appendix 1)." He "currently only uses intuition and his own footballing knowledge without carefully analysing all possible choices to make in-game decisions (Appendix 1)." Thus, Mr. X wants an analysis software that will help him make more successful teams and earn points in his local league (Appendix 1).

Rationale for development

I will develop for him a program that analyses footballer performances based on statistics such as goals scored, assists made, tackles, xG (expected goals), etc. It will then assign each player a score on an index determined by pre-defined criteria. Upon comparing different player performances, cost, budget, and restrictions, an optimum team for the upcoming game week shall be suggested (Appendix 2). It will also allow comparison between different players head-to-head, as the client expresses, "I would also like to compare different players to modify my team according to personal preferences, gut feeling, and instincts (Appendix 2)." Furthermore, Mr. X says, "I would also like the program to consider individual player performances in recent weeks and how many other FPL players have chosen the player in their team. (Appendix 2)." Hence, the program takes into account this request from the client before producing the optimal team.

Additionally, the program will provide a dynamic GUI interface for visualizing the team using a background canvas feature, such that it becomes more interactive for Mr. X (Appendix 2). The program will be written in Python due to various advantages. Firstly, Python enables easier communication with the official FPL API (Application

Programming Interface), as well as allows for convenient data management of the JSON response that is received. It also provides for smooth integration of .csv files and using its Pandas and NumPy libraries, data analysis becomes more efficient. Finally, the CVXPY library in Python is effective with solving linear programming problems, which would be key to solving the problem of the weekly optimum team in the software.

Success criteria

1. The user can filter the list of players based on position and club.
2. The user can directly compare the statistics of two chosen players playing in the same position, for example, goals, assists, xG (expected goals), etc.
3. The program gives a list of most in-form players, in different playing positions, in the recent weeks based on a pre-defined scoring index.
4. The program suggests the most optimum team for the upcoming game week within positional, financial, and club constraints.
5. The user can visualize the starting 11 for the team with different possible playing formations and positional weights through a GUI interface.
6. The user is provided with several tips to create their FPL team.

Word count: 531