Data and Applications

Project Phase-3

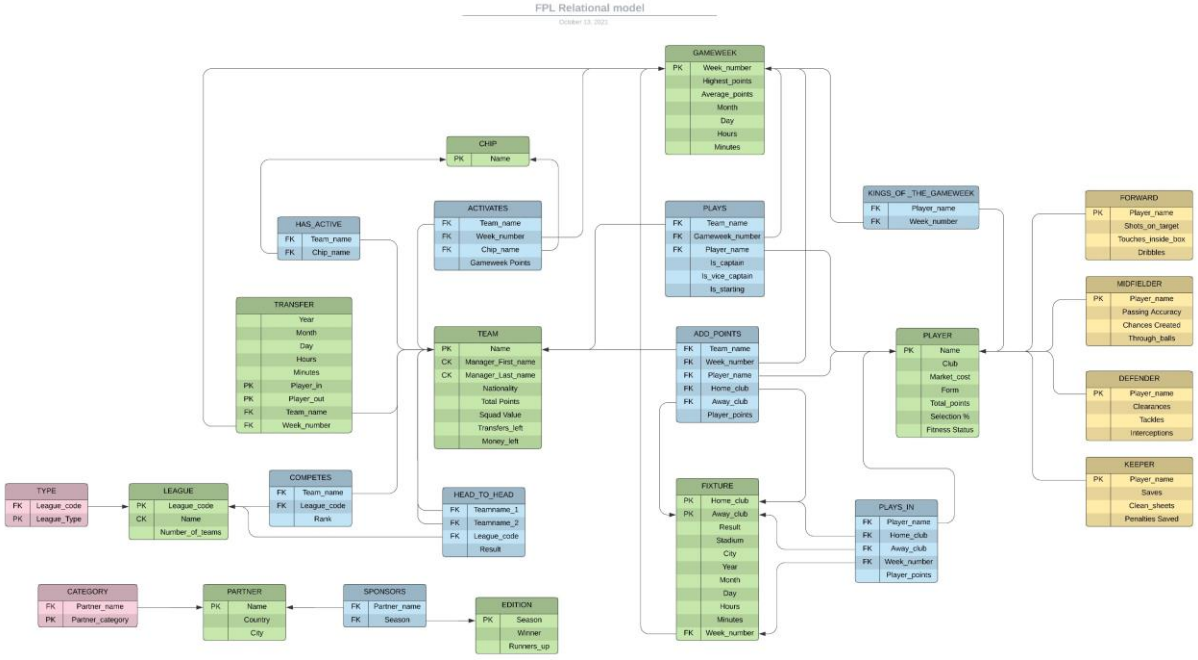
Fantasy Premier League

By Thakkallapally Rohan Rao, Nitin Rajasekar and Ankith Varun J
(2020101037) (2020101117) (2020111019)
~ https://fantasy.premierleague.com

1 RELATIONAL DATA MODEL

The conversion of an Entity-Relationship model into a Relational Data model was done sequentially using the ER-to-Relational Mapping Algorithm. First, an appropriate relation was created for each strong entity with one of the candidate keys forming the primary key of the relation. The relation created for weak entities had, in addition, a foreign key which is the primary key of the parent entity. Mapping of specialization was done by creating multiple relations to represent the superclass and subclasses. Second, appropriate modifications were done to represent the relationships between entities depending on their cardinality ratio, with extra attributes and relations being added as necessary. One-to-one and one-to-many relationships were represented by the addition of key attributes to a relation representing a participating entity, while many-to-many relationships were represented with the addition of a relationship relation by cross-referencing the participating entities. Separate relations were made for multivalued attributes, and composite and derived attributes were modified or removed to ordinary attributes. Finally, ternary and quaternary relationships were accounted for with the addition of a separate relation with its attributes being the keys of the participating relations. The resulting model is the Relational Data model.

A snapshot of the relational model is given below.



A lucidchart link has been attached here for further clarity.

2 FIRST NORMAL FORM

The obtained relational model has only single valued attributes (separate relations have been created for multivalued attributes), and all the attributes are atomic and indivisible. The domains of the attributes are constant and do not change. The attributes of a relation all have unique names, and the order in which the data is stored does not matter. During the conversion to a relational model, all composite attributes were converted to ordinary attributes and derived attributes were ignored. Thus, as all the requirements for 1NF are already satisfied by the above relational model, and there is no need to make any further modifications.

3 SECOND NORMAL FORM

We are required to check if the above relational model is in second normal form (2NF). A relational model is said to be in 2NF if each relation schema R of the model is in 2NF i.e every nonprime attribute A in R is *fully functionally dependent* on the primary key of R. Therefore, it is sufficient to check for the existence of *partial functional dependencies* in any relation schema of the relational model given above. On careful observation, two such functional dependencies arise.

1. In the relation schema *FIXTURE*, we find that the attributes *Stadium* and *City* can be functionally determined from the attribute *Home_club* alone. This results in the functional dependency

   $$\{Home\_club\} \rightarrow \{Stadium, City\}$$

   However, we know that {*Home_club*} is a subset of the primary key {*Home_club,Away_club,Week_number}*. Hence, {*Stadium,City*} is *partially functionally dependent o*n the primary key of *FIXTURE* and violates 2NF.

   FIXTURE is second normalised by decomposing it into two 2NF relations - *FIXTURE1*{*Home_club,Away_club,Week_number,Result,Year,Month,Day,Hours,Minutes*} and *FIXTURE2*{*Home_club,Stadium,City*}.

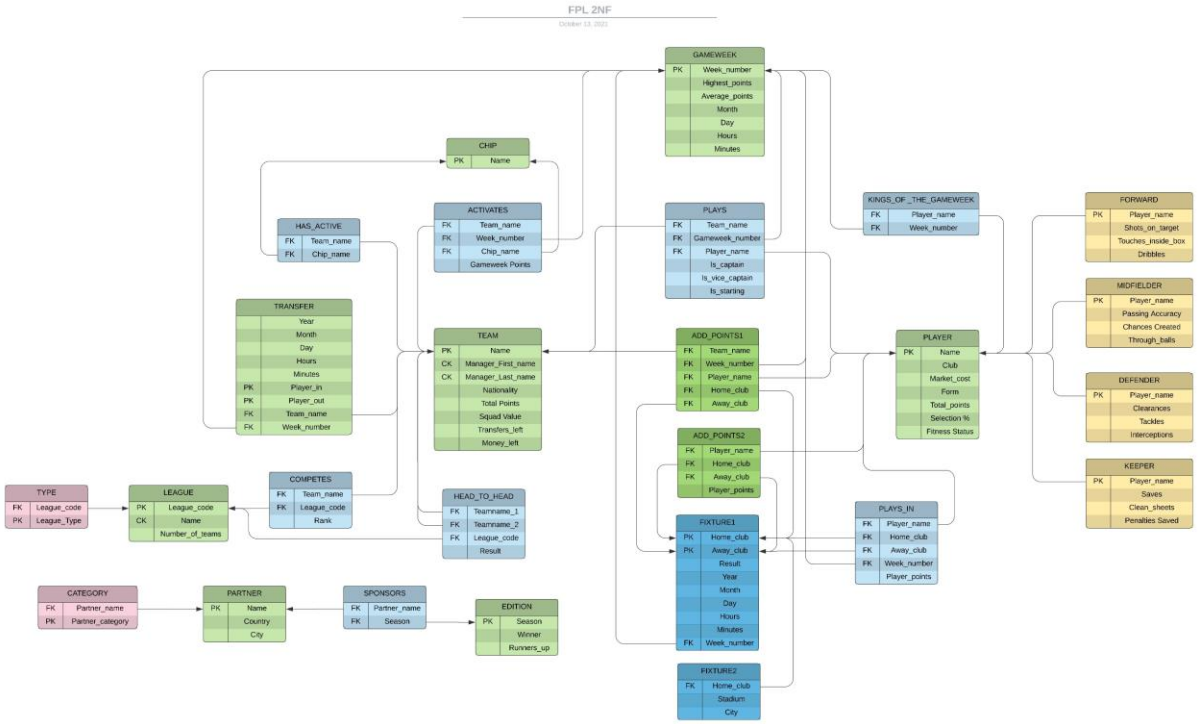
2. In the relation schema *ADD_POINTS*, the attributes *Player_name*, *Home_club* and *Away_club* are sufficient to functionally determine the attribute *Player_points*. This results in the functional dependency

   $$\{Player\_name, Home\_club, Away\_club\} \rightarrow \{Player\_points\}$$

   However, we know that {*Player_name,Home_club,Away_club*} is a subset of the primary key {*Team_name,Player_name,Home_club,Away_club,Week_number}*. Hence, {*Player_points*} is *partially functionally dependent o*n the primary key of *ADD_POINTS* and violates 2NF.

   ADD_POINTS is second normalised by decomposing it into two 2NF relations - *ADD_POINTS1*{*Team_name,Player_name,Home_club,Away_club,Week_number*} and *ADD_POINTS2*{*Player_name,Home_club,Away_club,Player_points*}.

A snapshot of the relational model after 2NF conversion is given below.



A lucidchart link has been attached here for further clarity.

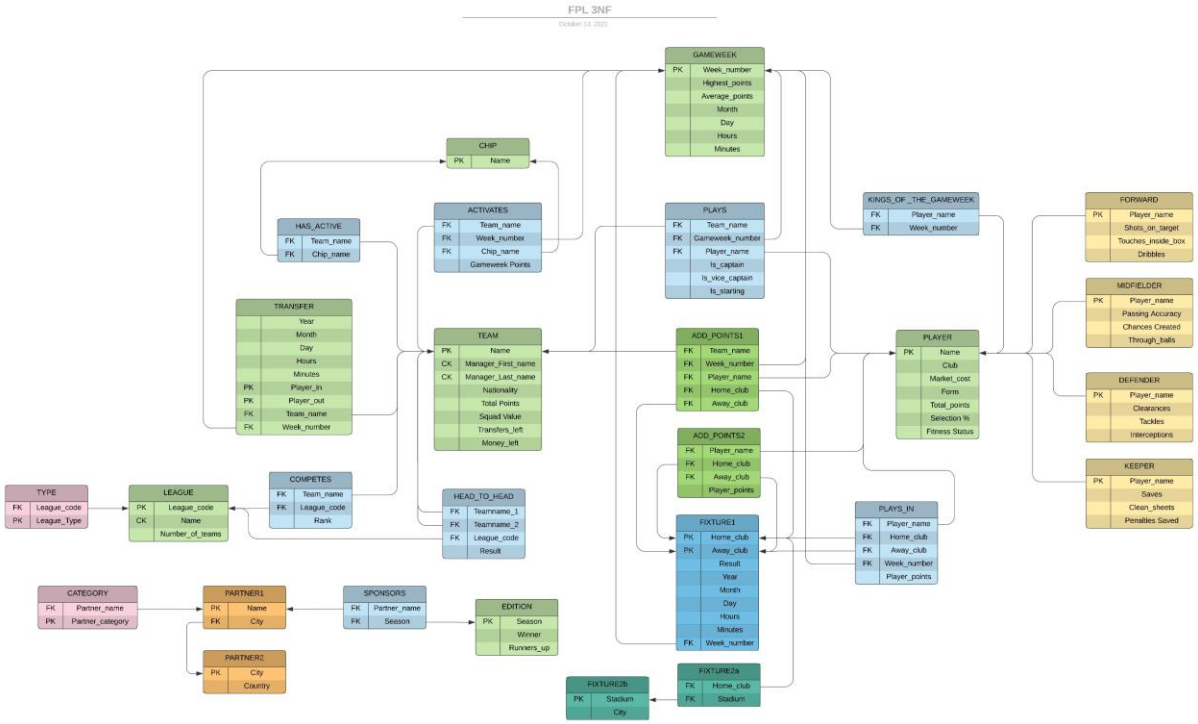
4 THIRD NORMAL FORM

We are required to check if the relational model satisfies third normal form (3NF). A relational model is in 3NF if every relation schema R in the model satisfies 2NF and no nonprime attribute of R is *transitively functionally dependent* on its primary key. The first requirement is already met. All that's left is to check for functional dependencies between nonprime attributes in any relation schema of the model. Again, two such cases can be found.

1. In the relation schema *PARTNER*, it is trivial that the attribute *Country* can be functionally determined from the attribute *City* alone. This results in the functional dependency, {*City*}→{*Country*}. Since {*Name*} is the primary key of *PARTNER*, the functional dependencies {*Name*}→{*City*} and {*Name*}→{*Country*} also exist. Thus, *Country* is *transitively functionally dependent* on *Name* via *City* and violates 3NF.
   PARTNER is third normalised by decomposing it into two 3NF relations - *PARTNER1*{*Name*,*City*} and *PARTNER2*{*City*,*Country*}.

2. Similarly, in the relation schema *FIXTURE2*, the attribute *City* can be functionally determined from the attribute *Stadium* alone. This results in the functional dependency, {*Stadium*}→{*City*}. Since {*Home_club*} is the primary key of *FIXTURE2*, the functional dependencies {*Home_club*}→{*Stadium*} and {*Home_club*}→{*City*} also exist. Thus, *City* is *transitively functionally dependent* on *Home_club* via *Stadium* and violates 3NF.
   FIXTURE2 is third normalised by decomposing it into two 3NF relations - *FIXTURE2a*{*Home_club*,*Stadium*} and *FIXTURE2b*{*Stadium*,*City*}.

A snapshot of the relational model after 3NF conversion is given below.



A lucidchart link has been attached here for further clarity.