

# Sports Stats Plus

By The Select Star Group

URL: <http://flip2.engr.oregonstate.edu:11777/>

*Make sure to vpn to OSU network to use.*

Gary Luong

Tristan Hilbert

5/13/2019

CS340 Online

Step 3 Final

## Fixes Since Step 2

After creating the tables, it seemed to make more sense to have the intermediary table between games and teams to have a single column for a foreign key and a single column for a boolean. The foreign key would hold a team id and the boolean would establish home or visitor status. While outside the bounds of our reality, the reorganization of these tables, made sense with normalizations of the data.

## Fixes for Outline, ERD, and Schema

A couple of changes, made from the step 1 document, include the addition of fields to further solidify the many to many relationships. Identity fields for ID numbers were created for the Player to allow a quick super key. The ID numbers for Team A and Team B were removed with an emphasis on the schema in this document. These will be replaced by two intermediary tables for Home and Away teams. Since many teams can be Home or Visitor, the amount of tables will benefit and still keep intact the relationships for the ERD.

The following are comments we received from Piazza.

1.)

*Outline Comments:*

- *I'm sure you considered breaking the name into first name and last name, but it might be prudent to have it split just because some parts of a website using such a database might only display a player's last name next to their stats.*
- *I'm not sure that defaulting shots taken and points made to 0 is a good idea. If a player gets injured, all of a sudden their average stats will begin to tend towards 0, which isn't true since they haven't been playing. Setting them to NULL might be wiser, but will complicate the semantics at the application level.*
- *Why not use a PlayerID for the MVP?*
- *The "Winning Team" attribute still uses Team A/Team B where I assume it should be Home/Visitor.*
- *To be clear, does the "Player to Team" relationship indicate that a player must belong to one and only one team? Or can a player not belong to a team (ie free agent)? If they're removed from a team are they removed from the db?*

*ERD Comments:*

- *The Team-to-Game relationship is missing a participation indicator.*
- *The outline indicates that a game can have one PotG and a player can be MVP for many games; I'm not sure if how this should be displayed on the ERD, but if it's a valid relationship it should be on the ERD.*

*Schema Comments:*

- *"Team ID" in the Player table should be pointing to the "Team ID" key of the Team table.*
- *The arrows should point from the dependency to the origin key; so all should be reversed except the Position ID reference.*

*Solid work though. The overall design is really well-thought-out.*

2.)

*Good job! Here is my feedback:*

- *Jersey number is unique only within a team, correct? I would suggest updating your description for this attribute as it could be mistaken to mean unique among all players.*
- *This is more of an opinion, but I think it would be interesting to have an attribute in your player table to have the number of shots made (assuming shots taken is 'attempted') as well as an attribute for points attempted (since you have points made).*
- *I think it would make more sense for the 'Most Valuable Player' attribute in the Game entity to be the player id instead of a text field. This could also make your database more useful as you could have another stat showing MVP counts for each player. What you have now is okay for the assignment requirements though.*
- *Is the team color just one color or multiple? How will multiple colors be stored?*
- *The team to game relationship should have either 0 or 1 as the minimum value displayed in the ERD.*
- *Arrows in the schema are pointing in the wrong direction.*

3)

*Howdy Gary and Tristan, I've avoided reading Gregory and Monet's reviews, so this may be repeated information.*

*Outline and Description: Very good, my only real question was about the "Number of Shots Taken" and "Number of Points Made" attributes in the Player entity. Are these totals for the season so far? Don't you want totals for each game? If you want totals for each game then they can't be part of the Player entity. Maybe you really only want the totals for the season so far.*

*Entity Relationship Diagram: The one-to-one relationship between team and mascot makes me think mascot should just be an attribute of team.*

*Schema: The arrows should point from foreign key to primary key. An arrow is needed from Player-TeamID to Team-TeamID.*

*Additionally, I'm wondering about the Player-Game-Position relationship. If players can play more than one position do you care which position(s) a player held in each game? Do player not play some games? As I asked above, do you care about the shots and points for each player in a particular game? I'm imaging a Player-Game-Position(which could be null)-Shots-Points relationship entity that tied all these things together.*

*4.)*

*If a team ID is required for a player, what happens if he is removed from his team for any reason and is a free agent? Does he keep his old team in the database(which would be incorrect) until he gets picked up on a new team? You could allow for a NULL value in player's teamID field which could represent a free agent.*

*5.)*

*So the only fields for statistics you have for players is shots attempted and points scored. Would this be for the whole season, multiple seasons, or game by game? Also, often times shots attempted doesn't reflect too well on the points scored as there are 3s, 2s, and free throws, maybe think about separating these fields or even adding a couple more.*

*6.)*

*Hi Tristan, thanks again for the thorough review. I am happy to see yours is sports related as well!*

*Outline Comments:*

*- Player name: maybe you would like to split it up with first name and last name? For example if someone wants to search for a player via their last name the system would skip over the first name and automatically filter the first names out.*

*- Nickname: would this be something that the user edits and he sees on his side or is it something that the database will have and all users will see the same nickname? Just*

*curious on this one. I can see how this would be helpful either way especially in College basketball with so many players.*

*- Shots taken: in the NBA there is a stat line displayed DNP which means "did not play". Im not sure if this is utilized in NCAA basketball, but if it is it might make this job easier by checking if any minutes were played then automatically populating DNP on all statistics for the specific player. Below Ive posted an example. Looks really good when making a table.*

<http://www.espn.com/nba/boxscore?gameId=401126861>

*ERD Comments:*

*- Looks good, one question. Will other stats be taken into consideration in the future? Such as rebounds, steals and assists?*

*Conclusion*

*These things are minor, but can maybe help clear up where you and your teammate would like the direction of this project. I look forward to your end result!*

## Response to Critique

### Actions Taken

1. Based on feedback from three reviews, it doesn't make much sense to keep track of shots taken per game. Instead, we changed this field to the number of games played by the player. This will increment by one for every game played. The field will be an int value and default to 0.
2. For the Schema, we flipped the arrows for the foreign keys pointing to primary keys
3. The "MVP" attribute in the Schema was changed so it can take the player ID to identify the MVP of the game. This also means that Player and Game will have a "Many to One" relationship. This will be displayed in the Schema and ERD.

4. Team ID will default to NULL if that person is not on a team. It is not required for every player entity. This will also reflect in the participation of the ERD.
5. Player names for the Player entity will have first name and last name attribute fields. This is better practice for finding relationships within data.
6. Primary and Foreign keys will be defined for all attributes within the outline.
7. Mascot will have a Mascot ID field to use as a better primary key, Since Team ID can be NULL.
8. An assumptions list will be added to the end of the document with featured rules of encapsulation for the data within our fantastical space. This is in response to a lot of criticism towards real world scenarios not working with our outline.
9. Clarify Jerseys are not unique to all players. The Jersey Number is different from player ID and it should be depicted as such in the outline.

#### Actions not taken

1. A few reviews mentioned to change the default of “# of points made” in the “Player” table to NULL. Because we are not keeping track of points per game, we will keep it at 0.
2. No additional color fields or entities will be added to the table.
3. No “did not play” fields or entities will be added to the table. We will have a number of games played field, but our database is to host player and game statistics instead of player activity.

## Fixes for Step 3

### Feedback Received

#### ***Comment 1***

Tristan, everything looks swell. I mean it.

- Entities and relationships lists look good
- ERD and Schema are excellent

- DDQ and DMQ look great, DDQ in particular looks complete, I'm jealous.
- HTML: good start

I know this generalized praise is not very useful, but I put a fair amount of time into examining your information and could find no particular criticisms.

If you have any questions or areas of concern I'd be happy to look again with those in mind.

### **Comment 2**

Comments on DDQ:

- Line 74: "location" is spelled wrong
- Your schema has 2 team IDs but Line 91 only shows 1 team ID

Comments on DMQ:

- On Line 12, if I'm reading this correctly, you're updating team's coach by finding the correct row using the team's old coach's name? Would it be easier to find the row by using the team name or team id?
- Missing a comma in Line 100
- Miscellaneous colons missing or semicolons appearing throughout, but I don't know how important syntax is in this since it seems like it's just making a template for the queries we'll be performing in our backend
- Would there ever be a need to update a player's information in the player table?

No real comments on the HTML. ERD and Schema look fine as far as I can tell. Great work.

*No comments were received from Gary's end.*

## Fixes Based on Revisions

We decided to act on everything.

- Corrections on SQL files are changed
- Schema will be changed to have one team in games\_to\_teams
- Update Coach based on team name
- Add Update feature to player for games and points
- Other Quality of Life improvements, improving the robustness. These include:
  - Adding proper functionality previously missing in the html
  - Adding example values for the DDQ file.

# Current Project Outline

Sports Stats Plus is a website and database which stores basketball player and game information. Basketball is a game played by two teams of players against each other. Games at different dates and times, and they involve different sets of players for each team. The database will capture the statistics of the players playing these games and the statistics of their overall team. The overall team is the superset of the possible players that play in a given game. The complexity of using teams of individuals makes these statistics perfect for tracking within a database project. With applications of “March Madness” brackets, individuals may find the site helpful.

## Database Outline and Description

At a bare minimum the following entities and attributes will be in the database. Any additional attributes and entities will be cleared with the grader prior to submission. These entities entail the different objects seen within the above outline.

### Entities

1. Player - This entity will represent the profile of a player.
  - a. **Player ID** - This number is used to identify all possible players as a primary key. This is a mandatory integer field that enumerates for each incoming player entity.
  - b. **Team ID** - This number represents the ID of the associated team for this player. This field must be a valid integer id of a team entity or set to NULL if a player is no longer part of a team. This is a foreign key.
  - c. **Jersey Number** - This number is used to identify the player when playing in a game. This is a unique number for players within teams. Two players on the same team cannot have the same number. This number will be assigned via integer field. This field is not mandatory and can be NULL.
  - d. **Player First Name** - The first name of the player. It will be represented with a 50 character string. This attribute is mandatory and cannot be left blank.
  - e. **Player Last Name** - The last name of the player following the previous field. It will be represented with a 50 character string. This attribute is mandatory and cannot be left blank.



- f. **Nickname** - This is the player's secondary name. This will be represented with a 50 character string. This field is not mandatory and can be left blank. It does not have a default value.
  - g. **Number of Games Played** - This will represent how many games a player has played in for a career. This field is an integer which will have a default value of 0.
  - h. **Number of Points Made** - This represents the cumulative number of points made after each game. After a game occurs the number of points made by shots from this player will be added to this field. This field is an integer which will have a default value of 0 if omitted or if a player has not played any games.
1. **Positions** - This entity will represent the different positions for each player. There will also have to be a positions-player schema/table for the intermediary for the player-position relationship.
    - a. **Position ID** - This will be an integer field that will be assigned a unique identifier. This field is mandatory. It will be used for the primary key within the intermediary player-position table.
    - b. **Position Name** - This will be a 20 character string which is mandatory for this field. It will entail the name of the position played during the game by certain players.
  1. **Team** - This entity will represent an overall team which any player belongs to. A player must belong to a team.
    - a. **Team ID Number** - Unique identifier for intermediary table to use as a primary key. This field will be represented with an enumerated integer field. If a player does not have a team, this will default to NULL.
    - b. **Team Name** - The name of the team. It will have a max 50 characters. This attribute is mandatory and cannot be left blank.
    - c. **Location** - This will represent where the team is located. There is no default. A team must always belong to a location (hometown). This location will be represented with a 50 character string.
    - d. **Color** - This represents the team's primary colors. Each team will have one primary color with a default value of white ("0xFFFFFFFF"). Colors will be represented with a 6 character string for a color code represented in hexadecimal. Ideally this would be done with 3 character strings, but SQL does not provide use of their unsigned chars without use of outside languages.

- e. **Coach** - a Team must always have a coach. The coach can only belong to one team. Coaches will be represented with a 50 character string. There is no default and this field cannot be null.
1. Game - Represents the many games two teams can partake in throughout the season. These are hefty entities with a bulk of information.
    - a. **Game ID** - Unique identifier for intermediary table to use as a primary key. This field cannot be empty and shall be represented with an enumerated number.
    - b. **Date** - The date it occurred held with a SQL date field. This is a mandatory field and must be filled.
    - c. **Location** - This represents the city this game occurred in. This will be represented with a 50 character max String. This is a mandatory field and must be filled.
    - d. **Winning team** - Boolean Field representing if the Home Team won or the Visiting Team. If this field is omitted and is Null then the conclusion will be considered a tie.
    - e. **Most Valuable Player** - This will be denoted by using the Player ID. This field entails the player deemed by statistics to be the most valuable within the game. This field may be Null and is not mandatory.
    - f. **Score Home** - This field will be represented with an integer field of a default value of 0. It represents the point score of the Home team.
    - g. **Score Visiting** - This field will be represented with an integer field of a default value of 0. It represents the point score of the Visiting team.
  1. Mascot - This entity represents the mascot or face of the team. Some teams have them for morale support during games.
    - a. **Mascot ID** - Unique identifier for use as a primary key. This field cannot be empty and shall be represented with an enumerated number.
    - b. **Mascot Name** - This represents a string of up to 50 characters which is mandatory. This should be the mnemonic device used to name the furry mascot for each team.
    - c. **Animal** - This represents a string of up to 50 characters which demonstrates what animal on earth the mascot takes after. This is not mandatory and can be Null.

- d. **Team ID** - This number represents the ID of the associated team for this mascot. This field cannot be empty and it must be a valid integer id of a team entity. This is a foreign key.

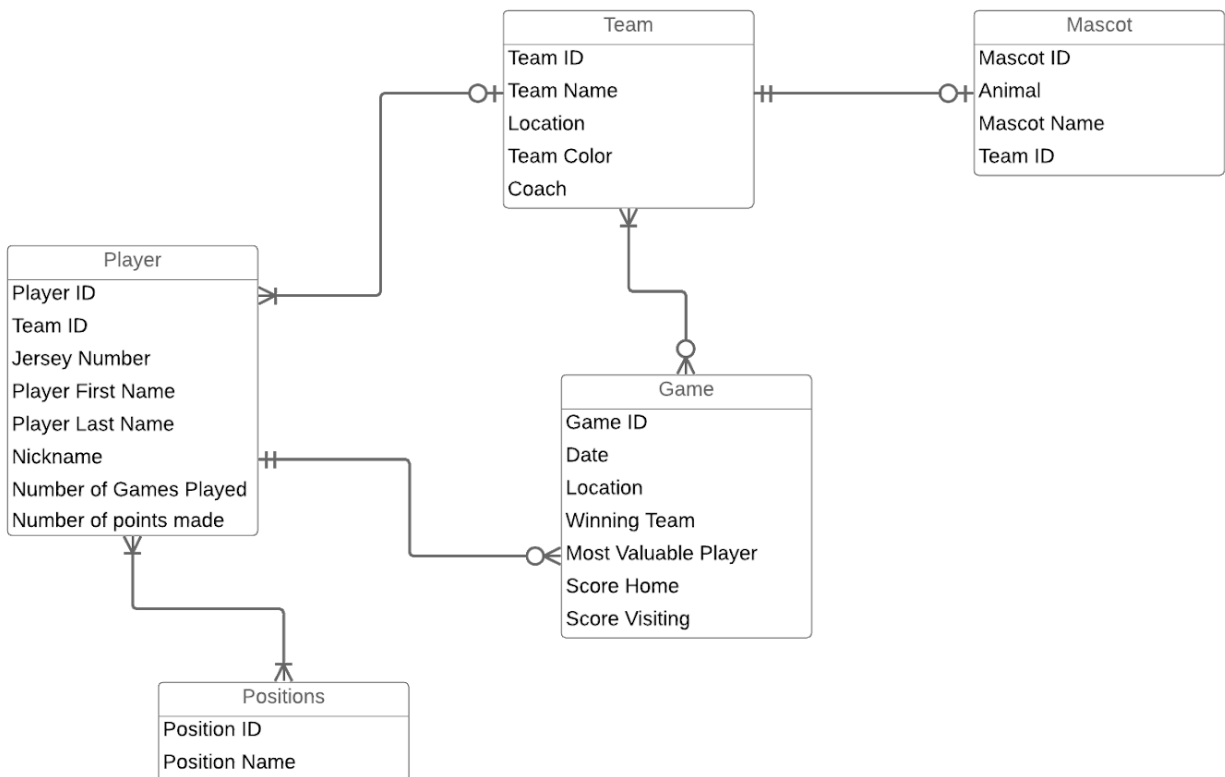
## Relationships

1. Player to Team (one to many) - a player can only belong to one team, but a team can have many players. Every Player must have at least one team.
2. Team to Game (many to many) - there will be many teams that can belong to a game, and many games that can belong to a team. A [boolean](#) on the intermediary table will be used to tell the difference between home and away.
3. Player to Game (one to many) - there can be at most one player of the game (from game info) and each player can be a most valuable player for many games
4. Players to other player positions (many to many) - players can play many positions and each position can be played by multiples players
5. Mascot to Team (one to one) - there will only be at most one mascot for every team. Not every team needs a mascot, but they try to raise morale.

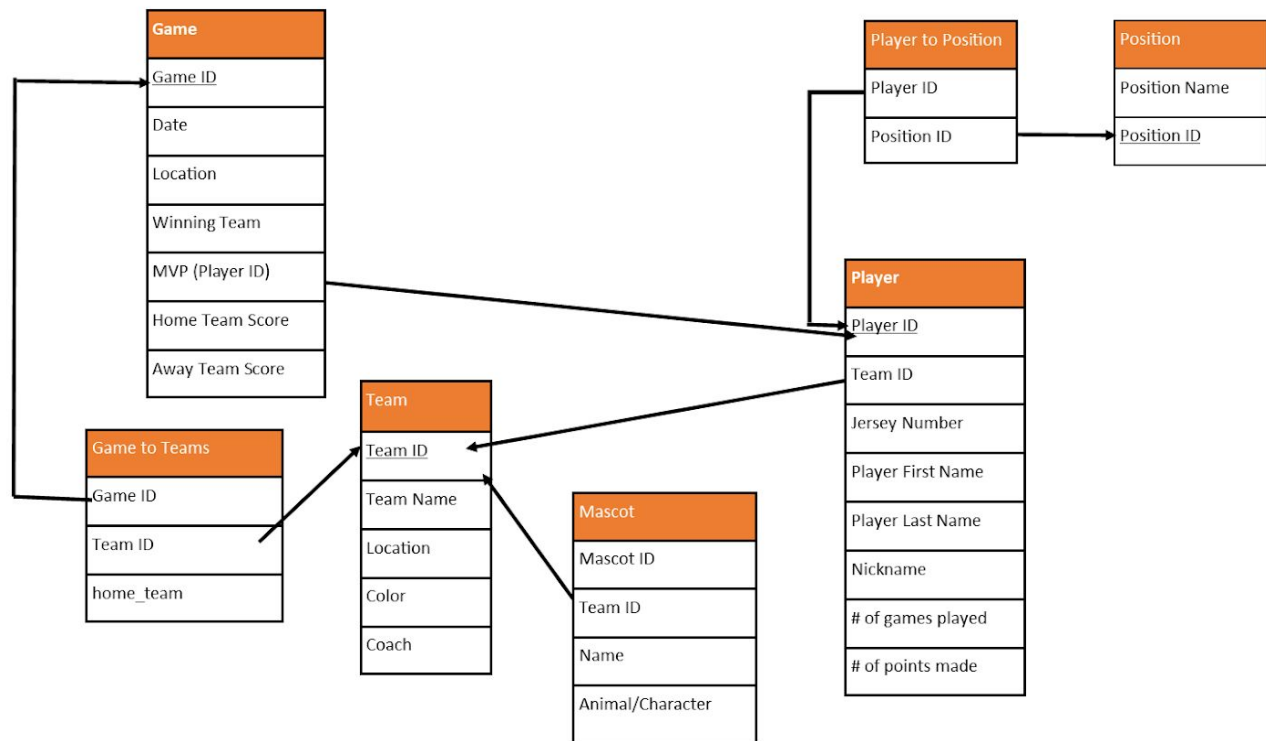
# Entity Relationship Diagram

## Select Star - Sport Stats Plus ERD

Tristan Hilbert | April 27, 2019



## Schema (Table Style)



## Assumptions and Semantics

A few assumptions have been made in the creation of this database. Within sports teams may have multiple colors. The database will only hold the “primary” color. This will be the first color code in a given media page for the team. If multiple colors are used within the jerseys and paraphernalia, only one will be selected. In further development the colors of the team could be an entity.

The coach is a second assumption. In this database the coach is to have no relation with other teams coaches. For this reason the group did not separate the names into first and last name fields. The database also does not treat the coach as its own entity. The attribute field merely represents a small dependent attribute of the Team entity to simplify the database.

The final assumption worth noting, is player participation. We will excuse an injury or issue regarding if a player is currently playing or not. We will assume each player has the opportunity of playing at every game. Therefore the positions they have are kept through their career.

