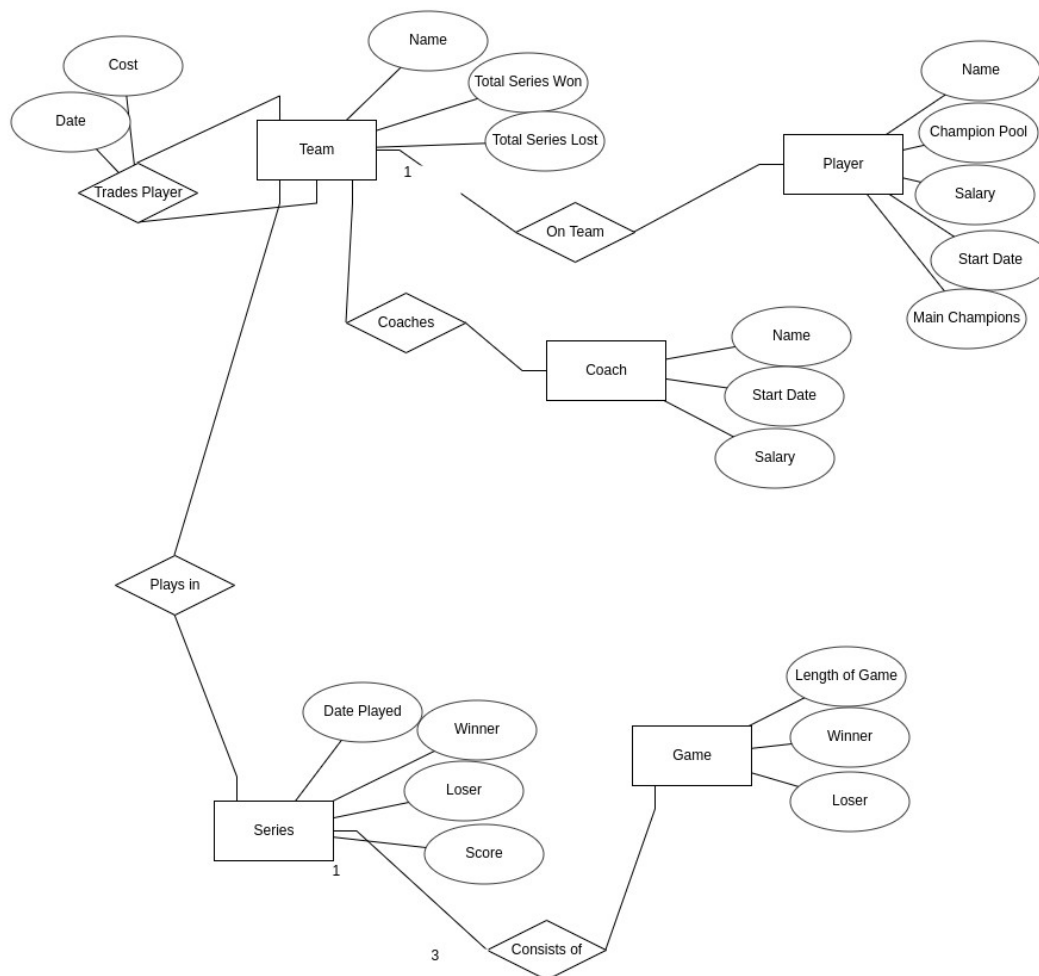


a.

For my PDA, I propose to create a database of teams in the LCS (League of Legends Championship Series, an esports league). This database will contain information about each team, including current roster, wins and losses, current coach, and titles won. For each player on the current roster, information about that player (salary, champions (characters in the game) played, etc) will also be maintained. I also plan on recording matches played in the 2020 season in this database, which would include relationships to two teams and it's own set of information. I imagine the relationship between Team, Series, and Game will be difficult to create cleanly. Teams play in best-of-3 series, but obviously the teams are also playing in those individual games. I considered ordering the relationship “Team → plays in → Game → part of → Series”, but I imagine it will be far more helpful to be able to search a team’s history of series rather than history of individual games.

b.



c. One major function which I would like to add is a *trade player* function, which would take a player and a team (that they are not currently on). This relationship has two attributes, “cost of trade” and “date”. This would remove the player entity’s relationship from their previous team, and establish a new relationship with the other team – as well as setting the start date to the date of the trade.

The second function would be *add game*, which, given a Series, will add details about an individual game. Adding a game would update the score of the series, and adding the final (third) game will establish a winner and loser of the series, which will update the total wins and losses for the teams playing in that series.