A picture containing drawing

Description automatically generatedChris Park

Database Management Systems



**Table of Contents**

Cover Page…………..………………………………………………………………….1

Table of Contents……………………………………………..……………………..2

Introduction/Abstract………………………………………………………………3

UML-Complaint ER Model……………………………………………………….4

Business Rules…………………………………………………………………………5

Entity/Attribute Descriptions…………………………………………………..6

Entity/Attribute Descriptions (cont.)………………………………………..7

Entity/Attribute Descriptions (cont.)…………………….………………….8

Closing Section…………………………………………………………………………9

**\*\*\***SQL Script, 10 SQL Queries, and List of Questions are all in *PittClubSports.sql*

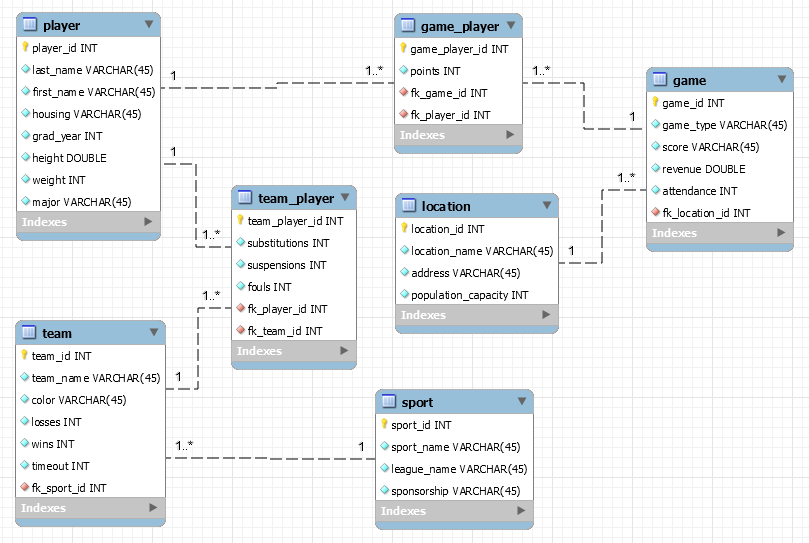
**Introduction/Abstract**

My club sports database is targeted towards the students and faculty at the University of Pittsburgh who are interested in or taking part in a club sport. Students who play club sports are usually those who genuinely are fans of a sport whether it be basketball, soccer, or baseball. In the real world, everyone can track the schedule and statistics of their favorite professional teams and players. I wanted to create something that would allow students experience what it is like to be an athlete in their point of view. My potential club sports database will be able to allow students and even faculty members to access specific information or statistics about certain club sports that are available at the University of Pittsburgh.

At the University of Pittsburgh, there are various club and intramural sports that students can sign up for and play. As a freshman, I played intramural basketball during the Fall of 2019. As a player, I was able to track our game schedule and the intramural team’s schedule through an app called “IMLeagues.” Also, the Pitt club teams’ information was on the app too. It possessed information about each player: email, gender, grade, graduation year, and birthday. However, Pitt club teams did not use this app to its fullest like Pitt intramurals. For example, the app’s ability to track statistics and the leaders among each sport was not utilized at all. My database will utilize these unused features and will possess even more attributes that will give the user an in-depth description and analysis of various sports, teams, players, and games.



**UML-Compliant ER Model**



**Business Rules**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Entity 1 | Entity 2 | Cardinality on Entity 1 side | Cardinality on Entity 2 side | Business Rules |
| Player | Team | 1..\*  1:M | 1..\*  1:M | **M:N**  One Player can be in Many Teams   * A player can play for multiple teams (a person can play volleyball and basketball at the same time).   One Team can have Many Players   * For obvious reasons.   A junction table called team\_player is created in result.   * Consists of fk\_player\_id and fk\_team\_id. |
| Player | Game | 1..\*  1:M | 1..\*  1:M | **M:N**  One Player can have Many Games   * A player could have a soccer game and a basketball game in the same week.   One Game can have Many Players   * A game will always be between two teams, each of which have many players.   A junction table called game\_player is created in result.   * Consists of fk\_player\_id and fk\_team\_id. |
| Location | Game | 1..\*  1:M | 1..1  1:1 | **1:M**  One Location can have Many Games   * Many different sport events could occur at different locations such as a large sports complex.   One Game has One Location   * A game cannot happen simultaneously in multiple locations. |
| Sport | Team | 1..\*  1:M | 1..1  1:1 | **1:M**  One Sport can have Many Teams   * There are various teams in one sport. If there was not, there would be no competition.   One Team is a part of One Sport   * A team is focused on one sport |





**Entity/Attribute Descriptions**

|  |  |  |
| --- | --- | --- |
| player | | |
| player\_id (pk) | INT | Primary Key  Auto incremented |
| last\_name | VARCHAR(45) | A player’s family name |
| first\_name | VARCHAR(45) | A player’s personal name |
| housing | VARCHAR(45) | Name of location where the player lives |
| grad\_year | INT | Year when a player is going to graduate college |
| height | DOUBLE | How tall a player is in feet |
| weight | INT | How heavy a player is in pounds |
| major | VARCHAR(45) | Name of study a player is under |

|  |  |  |
| --- | --- | --- |
| team | | |
| team\_id (pk) | INT | Primary Key  Auto incremented |
| team\_name | VARCHAR(45) | Team identification created by the players themselves |
| color | VARCHAR(45) | Color to distinguish and identify between other teams |
| losses | INT | Number of games a team lost |
| wins | INT | Number of games a team won |
| timeout | INT | Number of timeouts a team has available |
| fk\_sport\_id (fk) | INT | Foreign Key that references the primary key of the sport table |



|  |  |  |
| --- | --- | --- |
| game | | |
| game\_id (pk) | INT | Primary Key  Auto incremented |
| game\_type | VARCHAR(45) | Whether a game is home or away |
| score | VARCHAR(45) | The outcome of a game (very arbitrary) |
| revenue | DOUBLE | Total amount of money made in dollars in a game |
| attendance | INT | Total number that was present during the game |
| fk\_location\_id (fk) | INT | Foreign Key that references the primary key of the location table |



|  |  |  |
| --- | --- | --- |
| location | | |
| location\_id (pk) | INT | PRIMARY KEY  Auto incremented |
| location\_name | VARCHAR(45) | Name of facility that hosts sporting events |
| address | VARCHAR(45) | House number and street name of certain facilities |
| population\_capacity | INT | Total number of people the location can hold |

|  |  |  |
| --- | --- | --- |
| sport | | |
| sport\_id (pk) | INT | Primary Key  Auto incremented |
| sport\_name | VARCHAR(45) | Different types of sports |
| league\_name | VARCHAR(45) | League name that different teams are a part of |
| sponsorship | VARCHAR(45) | Company that financially supports a team |





|  |  |  |
| --- | --- | --- |
| team\_player | | |
| team\_player\_id (pk) | INT | Primary Key  Auto incremented |
| substitutions | INT | Number of times players on a team is subbed in/out |
| suspensions | INT | Number of times players on a team is banned for a certain amount of time |
| fouls | INT | Number of times players on a team, commits a penalty |
| fk\_player\_id (fk) | INT | Foreign Key that references the primary key of the player table. |
| fk\_team\_id (fk) | INT | Foreign Key that references the primary key of the team table. |

|  |  |  |
| --- | --- | --- |
| game\_player | | |
| game\_player\_id (pk) | INT | Primary Key  Auto incremented |
| points | INT | Amount of times a player scored in a game |
| fk\_game\_id (fk) | INT | Foreign Key that references the primary key of the game table |
| fk\_player\_id (fk) | INT | Foreign Key that references the primary key of the player table |



**Closing Statement**

 Overall, I was happy with the database that I created. It was interesting to choose a topic that I am passionate about: sports. I liked the amount of freedom we were given with our project topic. The easiest part about this project was creating the tables in SQL and INSERTING our data. However, before this step, thinking of all the entities and relationships was the difficult part. Also, creating 10 SQL queries that show value was a bit difficult. This was mainly because I really had to think about applying my database in real life. Ultimately, I gained a lot of insight on how databases work. My main takeaway was to always have a good, strong foundation before moving on to creating the database itself.