CMPE321

Project 3

Updated Part 2

Ahmet Fırat Gamsız - 2020400180

Yunus Emre Özdemir - 2020400153

CREATE TABLE User (

username *CHAR*(50),

password *CHAR*(50) NOT NULL,

name *CHAR*(50),

surname *CHAR*(50),

*PRIMARY KEY* (username)

);

CREATE TABLE Player (

username *CHAR*(50),

date\_of\_birth *DATE*,

height *INT*,

weight *INT*,

*PRIMARY KEY* (username),

*FOREIGN KEY* (username) *REFERENCES* User (username)

);

CREATE TABLE Coach (

username *CHAR*(50),

nationality *CHAR*(50) NOT NULL,

*PRIMARY KEY* (username),

*FOREIGN KEY* (username) *REFERENCES* User (username)

);

CREATE TABLE Jury (

username *CHAR*(50),

nationality *CHAR*(50) NOT NULL,

*PRIMARY KEY* (username),

*FOREIGN KEY* (username) *REFERENCES* User (username)

);

CREATE TABLE Position (

position\_ID *INT*,

position\_name *CHAR*(50) NOT NULL,

*PRIMARY KEY* (position\_ID)

);

CREATE TABLE Channel (

channel\_ID *INT*,

channel\_name *CHAR*(50) NOT NULL,

*PRIMARY KEY* (channel\_ID)

);

CREATE TABLE Team (

team\_ID *INT*,

team\_name *CHAR*(50) NOT NULL,

channel\_ID *INT* NOT NULL,

*PRIMARY KEY* (team\_ID),

*FOREIGN KEY* (channel\_ID) *REFERENCES* Channel (channel\_ID)

);

CREATE TABLE Stadium (

stadium\_id *INT*,

stadium\_name *CHAR*(50) NOT NULL,

stadium\_country *CHAR*(50) NOT NULL,

*PRIMARY KEY* (stadium\_id)

);

CREATE TABLE Date\_Time (

date\_of *DATE*,

timeslot *INT*,

*PRIMARY KEY* (date\_of, timeslot),

*CHECK* (timeslot >= 1 AND timeslot <= 3)

);

CREATE TABLE Match\_Sessions (

session\_ID *INT*,

team\_ID *INT* NOT NULL,

stadium\_id *INT* NOT NULL,

date\_of *DATE* NOT NULL,

timeslot *INT* NOT NULL,

*PRIMARY KEY* (session\_ID),

UNIQUE (team\_ID, date\_of, timeslot),

UNIQUE (stadium\_id, date\_of, timeslot),

*FOREIGN KEY* (team\_ID) *REFERENCES* Team (team\_ID),

*FOREIGN KEY* (stadium\_id) *REFERENCES* Stadium (stadium\_id),

*FOREIGN KEY* (date\_of, timeslot) *REFERENCES* Date\_Time (date\_of, timeslot)

);

CREATE TABLE Can\_Play (

username *CHAR*(50),

position\_id *INT*,

*PRIMARY KEY* (username, position\_id),

*FOREIGN KEY* (username) *REFERENCES* Player (username),

*FOREIGN KEY* (position\_id) *REFERENCES* Position (position\_ID)

);

CREATE TABLE Registered (

username *CHAR*(50),

team\_ID *INT*,

*PRIMARY KEY* (username, team\_ID),

*FOREIGN KEY* (username) *REFERENCES* Player (username),

*FOREIGN KEY* (team\_ID) *REFERENCES* Team (team\_ID)

);

CREATE TABLE In\_Contract (

username *CHAR*(50),

team\_ID *INT* NOT NULL UNIQUE,

contract\_start *DATE* NOT NULL,

contract\_finish *DATE* NOT NULL,

*PRIMARY KEY* (username),

*FOREIGN KEY* (username) *REFERENCES* Coach (username),

*FOREIGN KEY* (team\_ID) *REFERENCES* Team (team\_ID)

);

CREATE TABLE Rates (

session\_ID *INT*,

username *CHAR*(50),

rating *DECIMAL*(1, 1) NOT NULL,

*PRIMARY KEY* (session\_ID),

*FOREIGN KEY* (session\_ID) *REFERENCES* Match\_Sessions (session\_ID),

*FOREIGN KEY* (username) *REFERENCES* Jury (username)

);

CREATE TABLE Player\_Plays\_In (

username *CHAR*(50),

session\_ID *INT*,

position\_ID *INT* NOT NULL,

*PRIMARY KEY* (username, session\_ID),

*FOREIGN KEY* (username) *REFERENCES* Player (username),

*FOREIGN KEY* (session\_ID) *REFERENCES* Match\_Sessions (session\_ID),

*FOREIGN KEY* (position\_ID) *REFERENCES* Position (position\_ID)

);

Discussion:

What we did:

- We again started with User table. This user has attributes from the ER diagram and same primary key (let’s call this technique standard conversion). To inherit this table, we create Player table with username field as both primary and foreign key that references to User table. Coach and Jury follows the same. We made design choices on which attributes should be NOT NULL. For example, password can't be NULL, but height can be so that it can be filled after registration to the system.

- Position and Channel are simple tables with standard conversion. Team has a foreign key channel\_ID that references to Channel table. Since each team has exactly one channel we dropped the relationship from ER and used foreing key instead.

- Stadium is also follows standard conversion. Date\_Time table (we modified the name a bit to stay safe from reserved words) has primary key as date\_of (again modified due to reserved words) and timeslot tuple. Also we have check condition here as timeslot must be more than or equal to 1 and less than or equal to 3. This condition is implemented because matches take two timeslots and a match cannot start at 4th slot (it won’t be able to continue in a non-existent 5th slot). A match-session has many attributes and conditions. Since it has relationships denoted with bold arrow with team, stadium and datetime; it has primary keys of those tables as foreign key. Note that we are dropping the relationships of “exactly one” conditions (key and participation constraints) and use foreign keys to directly form this relation. These fields that are in foreign keys must be NOT NULL to make sure the relations exist.

- Can\_Play table provides the relationship with the same name. It has two foreign keys username of Player and position\_id of Position. These foreign keys constitute the primary key so they are implicitly NOT NULL. Registered table also follows the same for username and team\_ID fields. The use of primary keys here enable us to Player play in more than one position and team.

- In\_Contract table keeps start and finish dates of the contract. If a contract is made than it must have finish and start dates therefore NOT NULL conditions. Each team has exact one Coach so using username as primary key is enough but team\_ID must be UNIQUE and NOT NULL.

- Rates table has rating with DECIMAL(1,1) since ratings are given like “4,5”. Also only one jury rates so there is a foreign key of username.

- Lastly, Player\_Plays\_In table is used as the player list that plays in the match. Primary keys are username and session\_ID to ensure there is at least one player in the match. We also have NOT NULL position\_ID here to notify that which position player plays at.

What we couldn’t do:

- Although we created User, Player, Coach, Jury tables, we couldn't prevent a User that is not in other tables or in multiple other tables.

- We couldn't enforce every Player having a position\_list or a team\_list / having at least one position and team to play in.

- Although we could make sure that every team that is directed by a coach is directed by a unique coach, we couldn't enforce every team being directed by a coach.

- We couldn't enforce every match having a jury.

- We couldn't implement the constraints related to the match session overlaps, but at least we were able to make sure match session are not in the exactly same time slot.

Common with the Part 1 Discussion:

- Our logical design doesn't prevent players playing in matches where there are time conflicts.

- We were able to make sure that players play in only one position in a match session, but we couldn't prevent players from playing in a position they can't play (is registered).

- We couldn't implement the constraint "Juries can't edit/change their ratings."

- We can't check if a player that plays in a match is registered in the team that plays in that match.