University of British Columbia, Vancouver
Department of Computer Science

CPSC 304 Project Cover Page

Milestone #: 2_____

Date: 2022/2/26_____

Group Number:73 _____

Name	Student Number	CS Alias (Userid)	Preferred E-mail Address
Yuechang Liu	38612263	y1g7t	lyc021202@gmail.com
Cong Cheng	31070519	v5a3b	ccheng0715@gmail.com
Toby Li	45239803	Liyt03	litoby1203@gmail.com

By typing our names and student numbers in the above table, we certify that the work in the attached assignment was performed solely by those whose names and student IDs are included above. (In the case of Project Milestone 0, the main purpose of this page is for you to let us know your e-mail address, and then let us assign you to a TA for your project supervisor.)

In addition, we indicate that we are fully aware of the rules and consequences of plagiarism, as set forth by the Department of Computer Science and the University of British Columbia

University of British Columbia, Vancouver
Department of Computer Science

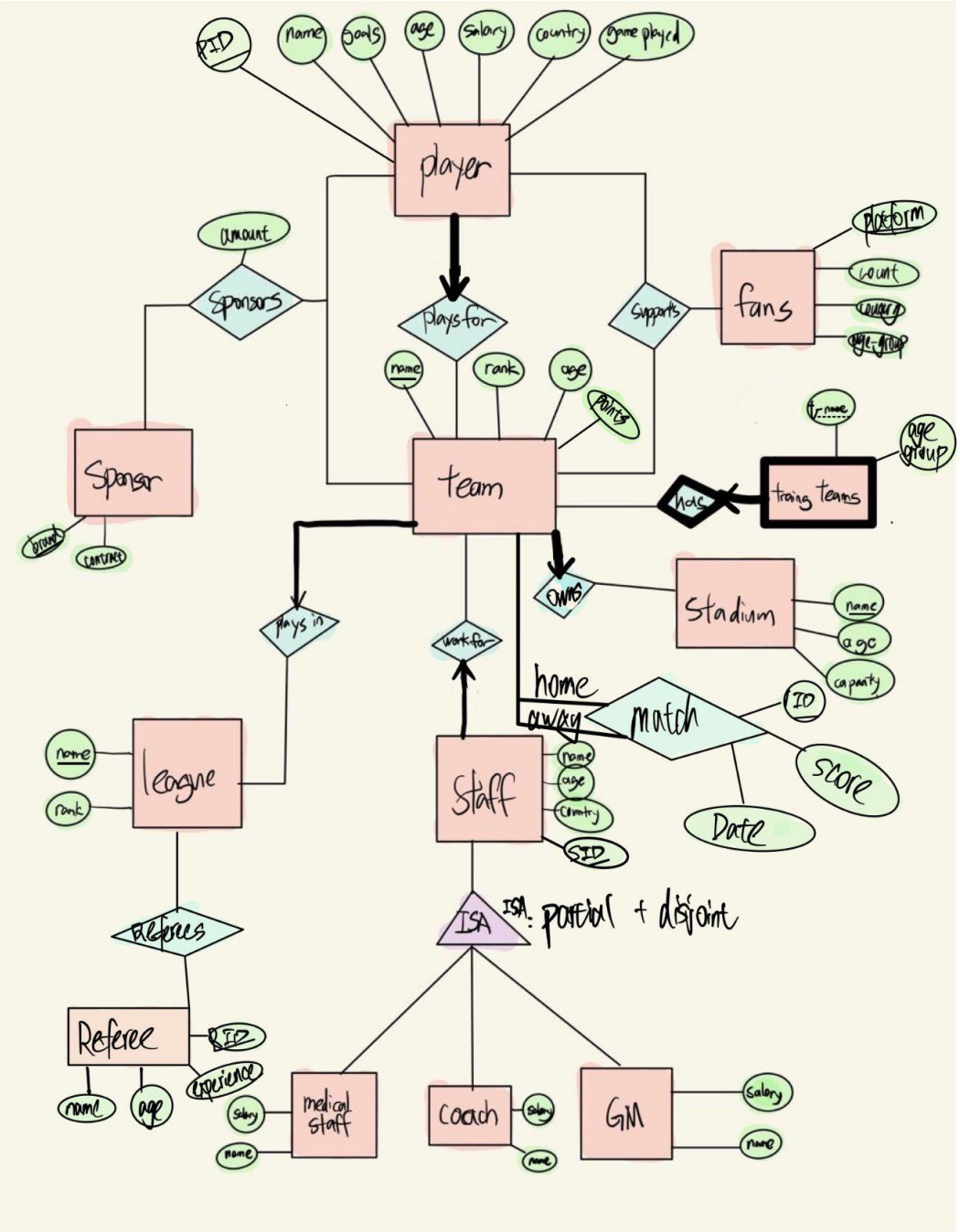
1. A completed cover page (template on Canvas)

2. A brief (~2-3 sentences) summary of your project. Many of your TAs are managing multiple projects so this will help them remember details about your project.

Our project is a database management system that stores information centered around a football team. It stores important information about a team such as the people and properties a club owns and also other entities as well. We hope that our application can help a football team manage their club more conveniently and efficiently.

3. The ER diagram you are basing your item #3 (below) on. This ER diagram may be the same as your milestone 1 submission or it might be different. If you have made changes from the version submitted in milestone 1, attach a note indicating what changes have been made and why.
If you have decided not to implement the suggestions given by your project mentor, please be sure to leave a note stating why. This is not to say that you must do everything that your project mentor says. In many instances, there are trade-offs between design choices and your decision may be influenced by different factors. That being said, your TAs will often leave suggestions that are meant to help massage your project into a form that will fit with the requirements in future project milestones. If you choose not to take their advice, it would be helpful for them to know why in order to better assist the group moving forward.

University of British Columbia, Vancouver
Department of Computer Science



changes have been made:

- correct the cardinality constraint arrow
- add keys for all entities
- add a unary relationship Match
- add ISA constraint
- add weak entity arrow

4. The schema is derived from your ER diagram (above). For the translation of the ER diagram

to the relational model, follow the same instructions as in your lectures. The process should be reasonably straightforward. For each table:
a. List the table definition (e.g., Table1(attr1: domain1, attr2: domain2, ...)). Make sure to include the domains for each attribute.
b. Specify the primary key (PK), candidate key, (CK) foreign keys (FK), and other constraints (e.g., not null, unique, etc.) that the table must maintain.

players(PID: int,name:char[20],goals:int,age:int,salary:int,country:char[10],game played:int,**team-name:**char[15])(team-name not null)
team(name:char[20],rank:int,age:int,**League-name:**char[20],**stadium-name:**char[20],points:int)(League-name,stadium-name not null)
fans(platform:char[20],count:int,country:char[10],age-group:char[20])
sponsor(brand:char[10],contract:int,amount:int)
league(name:char[20],rank:int)
referee(ID:int,name:char[20],age:int,experience:int)
staff(ID:int,name:char[20],age:int,country:char[10],**team-name:**char[15])(team-name not null)
medical_staff(**ID:int**,salary:int,name:char[20],speciality:char[20])
coach(**ID:int**,salary:int,name:char[20])
GM(**ID:int**,salary:int,name:char[20])
stadium(name:char[20],age:int,capacity:int)
training_team(t-name:char[20]**,team-name:**char[20],age-group:char[20])(team-name not null)(team-name unique)
sponsors(**brand:**char[10],**PID:int**,**team-name:**char[20],amount:int)
match(ID:int,score:char[5],date:date,**home-team-name:**char[20],**away-team-name:**char[20])(home-team-name,away-team-name not null)

5. Functional Dependencies (FDs)
a. Identify the functional dependencies in your relations, including the ones involving all candidate keys (including the primary key).
PKs and CKs are considered functional dependencies and should be included in the list of FDs. You do not need to include trivial FDs such as A → A.
Note: In your list of FDs, there must be some kind of valid FD other than those identified by a PK or CK. If you observe that no relations have FDs other than the PK and CK(s), then you will have to intentionally add some (meaningful) attributes to show valid FDs. We want you to get a good normalization exercise. Your design must go through a normalization process.

player:
PID ->name, goals, age, salary, country, game played,team-name

University of British Columbia, Vancouver
Department of Computer Science

game_played, goals, age -> salary

team:
name ->rank,age,**League-name**,**stadium-name**,points
stadium name, league name -> name

fans:
platform->count, country,age-group

league:
name->rank

referee:
ID ->name, age, experience

staff:
SID->name, age, country
team-name -> country

medical_staff:
ID-> name
specialty -> salary

coach:
ID->salary, name

GM:
ID->salary, name

stadium:
name->age, capacity

training_team:
t-name,team-name->age-group

sponsors:
brand, PID,team-name->amount

match:
ID -> score, date,home-team-name,away-team-name

home-team-name, date -> away-team-name, score
away-team-name, date -> home-team-name, score

6. Normalization
a. Normalize each of your tables to be in 3NF or BCNF. Give the list of tables, their primary keys, their candidate keys, and their foreign keys after normalization. You should show the steps taken for the decomposition. Should there be errors, and no work is shown, no partial credit can be awarded without steps shown. The format should be the same as Step 3, with tables listed similar to Table1(attr1:domain1, attr2:domain2, …). ALL Tables must be listed, not only the ones post-normalization.

Player:
Decompose on game_played, goals, age -> salary
Player_R1(PID, name, goals, age, country, game played,team-name)
Player_R2(game_played, goals, age, salary)

Team:
Decompose **stadium name, league name** -> name to BCNF
R1(rank,age,**League-name**,**stadium-name**,points,name)

Fans:
R1(count, country,age-group,platform)

League:
R1(name, rank)

Referee:
R1(ID,name,age,experience)

Staff:
Decompose on team-name -> country
R1(ID,name,age,**team-name**)
R2(**team-name**, country)

Medical-staff:

Decompose ID-> name
R1(**ID**, name), R2(**ID,**salary,speciality)

Decompose specialty -> salary
R3(specialty ,salary), R4(**ID**,speciality)

Final Answer: R1(**ID**, name),R3(specialty ,salary), R4(**ID**,speciality)

Coach:
R1(**ID**,salary, name)

GM:
R1(**ID**,salary,name)

Sponsor:
R1(brand,contract,amount)

Sponsors:
R1(brand, PID,team-name,amount)

Stadium:
R1(name,age, capacity)

Training_team:
R1(t-name,**team-name**,age-group)

Match:
Decompose to BCNF:
R1(**ID**,score, date,**home-team-name,away-team-name)**

7. The SQL DDL statements are required to create all the tables from item #6. The statements should use the appropriate foreign keys, primary keys, UNIQUE constraints, etc.

CREATE TABLE PLAYER_R1(

```
  PID int,
  name char[20],
  goals int,
  age int,
  country char[20],
  game_played int,
  team_name char[20] NOT NULL,
  PRIMARY KEY (PID),
  FOREIGN KEY (team_name) REFERENCES TEAM
)

CREATE TABLE PLAYER_R2(
  game_played int,
  goals int,
  age int,
  salary int,
  PRIMARY KEY (game_played, goals, age)
)

CREATE TABLE TEAM(
  rank int,
  age int,
  League-name char[20] NOT NULL,
  stadium-name  char[20] NOT NULL,
  points int,
  name char[20],
  PRIMARY KEY (name),
  FOREIGN KEY ( League-name) REFERENCES LEAGUE
  FOREIGN KEY (stadium-name) REFERENCES STADIUM
)

CREATE TABLE Fans(
  count int,
  country char[20],
  age-group char[20],
  platform char[20],
  PRIMARY KEY (platform)
)


CREATE TABLE League(
  name char[20],
  rank int,
```

```
  PRIMARY KEY (name)
)

CREATE TABLE Referee(
  ID int,
  name char[20],
  age int,
  experience int,
  PRIMARY KEY (ID)
)

CREATE TABLE Staff_R1(
  ID int,
  name char[20],
  age int,
  team-name char[20 NOT NULL],
  PRIMARY KEY (ID, team-name),
  FOREIGN KEY (team-name) REFERENCES TEAM
)

CREATE TABLE Staff_R2(
  team-name char[20],
  country char[10],
  PRIMARY KEY (team-name),
  FOREIGN KEY (team-name) REFERENCES TEAM

)

CREATE TABLE medical_staff_R1(
  ID int,
  name char[20],
  PRIMARY KEY (ID),
  FOREIGN KEY (ID) REFERENCES STAFF_R1
)

CREATE TABLE medical_staff_R3(
  speciality char[20],
  salary int,
  PRIMARY KEY (speciality)
)

CREATE TABLE medical_staff_R4(
  ID int,
```

```
  speciality char[20],
  PRIMARY KEY (ID),
  FOREIGN KEY (ID) REFERENCES STAFF_R1
)

CREATE TABLE Coach(
  ID int,
  salary int,
  name char[20],
  PRIMARY KEY (ID),
  FOREIGN KEY (ID) REFERENCES STAFF_R1
)

CREATE TABLE GM(
  ID int,
  salary int,
  name char[20],
  PRIMARY KEY (ID),
  FOREIGN KEY (ID) REFERENCES STAFF_R1
)

CREATE TABLE Sponsor(
  brand char[10],
  contract int,
  amount int,
 PRIMARY KEY (brand)
)

CREATE TABLE Sponsors(
  brand char[10],
  PID int,
  team-name char[20],
  amount int,
  PRIMARY KEY (brand, PID, team-name)
  FOREIGN KEY (brand) REFERENCES SPONSOR
  FOREIGN KEY (PID) REFERENCES PLAYER_R1
  FOREIGN KEY (team-name) REFERENCES TEAM
)

CREATE TABLE Stadium(
  name char[20],
  age int,
  capacity int,
```

```
 PRIMARY KEY (name)
)

CREATE TABLE Training_team(
  t-name char[20],
  team-name  char[20] NOT NULL,
  age-group  char[20],
  PRIMARY KEY (t-name, team-name)
  FOREIGN KEY (team-name) REFERENCES TEAM
)



CREATE TABLE Match(
  ID int,
  score char[5],
  date date,
  home-team-name char[20] NOT NULL,
  away-team-name char[20] NOT NULL,
  PRIMARY KEY (ID)
  FOREIGN KEY (home-team-name) REFERENCES TEAM
  FOREIGN KEY (away-team-name) REFERENCES TEAM
)
```

8. INSERT statements to populate each table with at least 5 tuples. You will likely want to have more than 5 tuples so that you can have meaningful queries later on.

```
INSERT INTO Player_R1  VALUES(12345, “Leo Messi”, 1000, 35, “Argentina”, 100, “PSG”);
INSERT INTO Player_R1 VALUES(23456, "Cristiano Ronaldo", 800, 36, "Portugal", 90,
"Manchester United");
INSERT INTO Player_R1 VALUES(34567, "Neymar Jr.", 700, 29, "Brazil", 80, "PSG");
INSERT INTO Player_R1 VALUES(45678, "Kylian Mbappe", 600, 23, "France", 95, "PSG");
INSERT INTO Player_R1 VALUES(56789, "Mohamed Salah", 500, 29, "Egypt", 90, "Liverpool");

INSERT INTO Player_R2  VALUES(100, 1000, 35, 1,000,000);
INSERT INTO Player_R2  VALUES(1 1230, 15, 200,000);
INSERT INTO Player_R2  VALUES(10, 900, 32, 12,000,000);
INSERT INTO Player_R2  VALUES(20, 140, 39, 2,000,000);
INSERT INTO Player_R2  VALUES(30, 471, 31, 2,000,000);


INSERT INTO Team VALUES(1,100, “a”,” home”,23,”barca”);
```

University of British Columbia, Vancouver
Department of Computer Science

INSERT INTO Team VALUES(2,10, “b”,”hoe”,13,”baca”);
INSERT INTO Team VALUES(11,99, “c”,”ome”,22,”arca”);
INSERT INTO Team VALUES(3,20, “d”,”he”,45,”bara”);
INSERT INTO TeamVALUES(9,11, “e”,”hom”,33,”bar”);

INSERT INTO Fans VALUES(101, “USA”,”30-40”, “Twitter”);
INSERT INTO Fans VALUES(102, “India”,”20-30”, “Facebook”);
INSERT INTO Fans VALUES(103, “Brazil”,”40-50”, “Instagram”);
INSERT INTO Fans VALUES(104, “Russia”,”30-40”, “Instagram”);
INSERT INTO Fans VALUES(105, “Nigeria”,”20-30”, “Twitter”);

INSERT INTO League VALUES(“Premier League”, 1);
INSERT INTO League VALUES("La Liga", 2);
INSERT INTO League VALUES("Serie A", 3);
INSERT INTO League VALUES("Bundesliga", 4);
INSERT INTO League VALUES("Ligue 1", 5);

INSERT INTO Referee VALUES(1,”a”,52,10);
INSERT INTO Referee VALUES(2, "b", 46, 12);
INSERT INTO Referee VALUES(3, "c", 55, 8);
INSERT INTO Referee VALUES(4, "d", 49, 14);
INSERT INTO Referee VALUES(5, "e", 40, 6);

INSERT INTO Staff_R1 VALUES(1, “a”,23,”Barcelona FC”);
INSERT INTO Staff_R1 VALUES(2, "b", 28, "Real Madrid");
INSERT INTO Staff_R1 VALUES(3, "c", 32, "Manchester United");
INSERT INTO Staff_R1 VALUES(4, "d", 25, "Juventus FC");
INSERT INTO Staff_R1 VALUES(5, "e", 30, "Bayern Munich");

INSERT INTO Staff_R2 VALUES(”Barcelona FC”, “Spain”);
INSERT INTO Staff_R2 VALUES("Real Madrid", "Spain");
INSERT INTO Staff_R2 VALUES("Manchester United", "England");
INSERT INTO Staff_R2 VALUES("Juventus FC", "Italy");
INSERT INTO Staff_R2 VALUES("Bayern Munich", "Germany");

INSERT INTO Medical_staff_R1 VALUES(1, “a”);
INSERT INTO Medical_staff_R1 VALUES(2, "b");
INSERT INTO Medical_staff_R1 VALUES(3, "c");
INSERT INTO Medical_staff_R1 VALUES(4, "d");
INSERT INTO Medical_staff_R1 VALUES(5, "e");

University of British Columbia, Vancouver
Department of Computer Science

```
INSERT INTO Medical-staff_R2 VALUES(1,100000, “Eye surgent”);
INSERT INTO Medical_staff_R2 VALUES(2, 90000, "Orthopedic surgeon");
INSERT INTO Medical_staff_R2 VALUES(3, 80000, "Dentist");
INSERT INTO Medical_staff_R2 VALUES(4, 100000, "Neurosurgeon");
INSERT INTO Medical_staff_R2 VALUES(5, 95000, "Cardiologist");

INSERT INTO Medical_staff_R3 VALUES(“Eye surgent”, 100000);
INSERT INTO Medical_staff_R3 VALUES("Orthopedic surgeon", 90000);
INSERT INTO Medical_staff_R3 VALUES("Dentist", 80000);
INSERT INTO Medical_staff_R3 VALUES("Neurosurgeon", 100000);
INSERT INTO Medical_staff_R3 VALUES("Cardiologist", 95000);

INSERT INTO Medical_staff_R4 VALUES(1, “Eye surgent”);
INSERT INTO Medical_staff_R4 VALUES(2, "Orthopedic surgeon");
INSERT INTO Medical_staff_R4 VALUES(3, "Dentist");
INSERT INTO Medical_staff_R4 VALUES(4, "Neurosurgeon");
INSERT INTO Medical_staff_R4 VALUES(5, "Cardiologist");

INSERT INTO Coach VALUES(1, 1000000, “Pep Guadiola”);
INSERT INTO Coach VALUES(2, 800000, "Jurgen Klopp");
INSERT INTO Coach VALUES(3, 700000, "Jose Mourinho");
INSERT INTO Coach VALUES(4, 600000, "Carlo Ancelotti");
INSERT INTO Coach VALUES(5, 500000, "Diego Simeone");

INSERT INTO GM VALUES(1, 999999, “Xavi”);
INSERT INTO GM VALUES(2, 800000, "Andoni Zubizarreta");
INSERT INTO GM VALUES(3, 700000, "Leonardo Araujo");
INSERT INTO GM VALUES(4, 600000, "Michael Zorc");
INSERT INTO GM VALUES(5, 500000, "Monchi");

INSERT INTO Sponsor VALUES(“NIKE”, 2,1000000);
INSERT INTO Sponsor VALUES("Adidas", 1, 800000);
INSERT INTO Sponsor VALUES("Puma", 3, 700000);
INSERT INTO Sponsor VALUES("Under Armour", 4, 600000);
INSERT INTO Sponsor VALUES("New Balance", 5, 500000);
```

INSERT INTO Sponsors VALUES(“NIKE”, 1, “Barca”,1000000);
INSERT INTO Sponsors VALUES("Adidas", 2, "Real Madrid", 800000);
INSERT INTO Sponsors VALUES("Puma", 3, "Manchester City", 700000);
INSERT INTO Sponsors VALUES("Under Armour", 4, "Chelsea", 600000);
INSERT INTO Sponsors VALUES("New Balance", 5, "Liverpool", 500000);

INSERT INTO Stadium VALUES(“Old Trafford”,100, 20000);
INSERT INTO Stadium VALUES("Camp Nou", 101, 99000);
INSERT INTO Stadium VALUES("Anfield", 102, 54074);
INSERT INTO Stadium VALUES("Allianz Arena", 103, 75000);
INSERT INTO Stadium VALUES("Santiago Bernabeu", 104, 81044);

INSERT INTO Training_team VALUES(“Barca U17”,”Barcelona”, “17”);
INSERT INTO Training_team VALUES("Juventus U14", "Juventus", "14");
INSERT INTO Training_team VALUES("Arsenal U16", "Arsenal", "16");
INSERT INTO Training_team VALUES("Bayern Munich U18", "Bayern Munich", "18");
INSERT INTO Training_team VALUES("Real Madrid U19", "Real Madrid", "19");

INSERT INTO Match VALUES(1, “5:2”, 2002-12-23, “Barcelona”, “Real Madrid”);
INSERT INTO Match VALUES(2, "3:1", "2022-01-15", "Manchester United", "Chelsea");
INSERT INTO Match VALUES(3, "2:2", "2021-11-21", "Juventus", "AC Milan");
INSERT INTO Match VALUES(4, "4:0", "2023-05-07", "Bayern Munich", "Borussia Dortmund");
INSERT INTO Match VALUES(5, "1:0", "2022-03-19", "Paris Saint-Germain", "Lille");