CSU34041 Project Report

Brian Lynch
17325256
blynch5@tcd.ie

Contents

Application Description

The database that I have designed and implemented describes some of the data that is required by a Football League. I have modelled this database off the Leinster Senior League (LSL). The Leinster Senior League is Irelands largest amateur soccer league and contains over 20 divisions and over 150 teams, including two teams from Trinity College. As this organisation is so large and I have a decent understanding of them, I thought they would be perfect for this assignment. The LSL must keep track of all players, teams, coaches, fixtures, pitches, organisational bodies within the league and the staff within the bodies.

The first thing the LSL will have to keep track of is players. The players will have a unique identifier as their primary key. They will also have two foreign keys, the team they play for and the organisational body they are registered with. A player can exist without a team as there can be free agents within the LSL, however a player cannot exist within the database without being registered to one of the organisational bodies in the LSL. Players will have other information stored in the database such as name, address, nationality, if they have paid their membership fee and date of birth. Players compose a team and register to the organisational body.

The second thing is the coaches. The coaches will have a unique identifier as their primary key. They will also have a foreign key, the team they coach. A coach can exist without a team as they are still a registered coach within the FAI. Coaches will have other information stored in the database such as name, address, coaching license type and date of birth. Coaches coach a team.

The third thing is teams. Each Team must have a unique name which will be used as the primary key. A team must have a registered coach and a pitch in order to exist within the LSL. A team does not need players or fixtures in order to exist, this is to facilitate new start up teams. Teams will have other information such as address, email and team colours stored in the database. Teams are coached by a coach, are composed of players, own a pitch and play in fixtures.

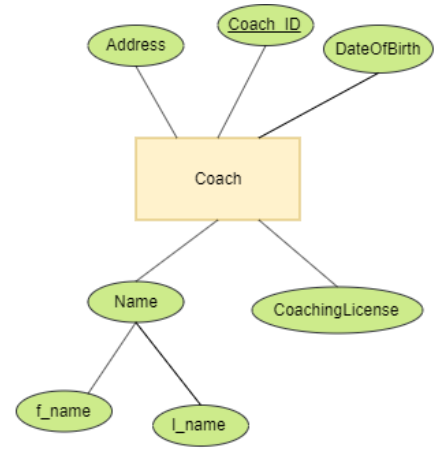
The fourth thing is pitches. Each pitch must have a unique identifier as the primary key. It will have one foreign key which is which team owns the pitch. A pitch can only be owned by one team however the pitch may exist in the database without a team owning it as the league needs neutral pitches to host certain games. Pitches will have other information such as address, capacity and type stored in the database. Pitches are owned by a team and hold fixtures.

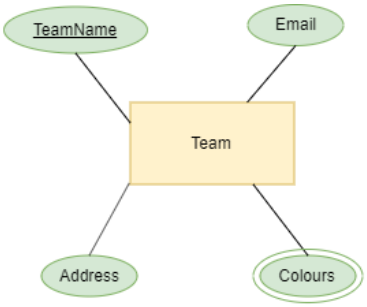
The fifth is the fixtures. Each fixture must have a unique identifier as the primary key. It will have two foreign keys, the ID of the pitch it is on and the ID of the organising body that scheduled it. A fixture cannot exist within the database without an organiser or a pitch. Fixtures will have other information such as type (cup, league etc.), time and date stored in the database. Goals and points of each team (A and B) will also be recorded. Fixtures are organised by the organising body and are held on pitches.

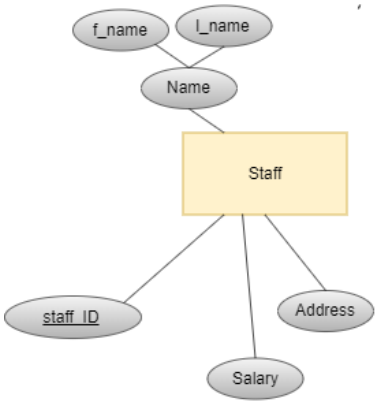
The sixth entity is the organisational body. There is a different body for each sub-section of leagues such as adult, youth, junior and over 40s. Each body will have a unique name as their primary key. No foreign keys. Organisational bodies will have information such as address, email and divisions managed. Organisational bodies schedule fixtures and are register players.

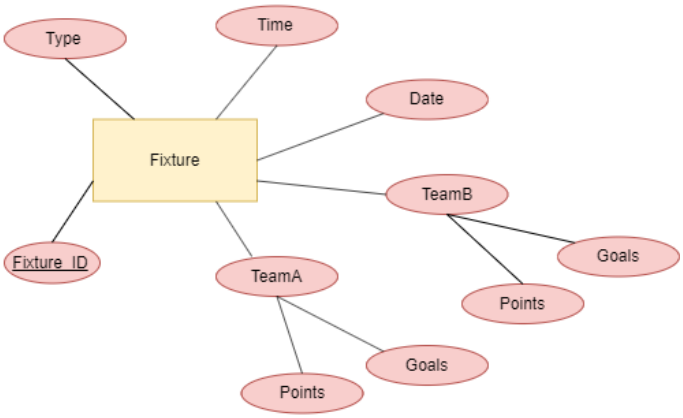
The final entity is the staff. The staff work in the organising body and have a unique identifier as the primary key. As it is not a huge organisation staff can work for more than one body. Staff can not exist without a body hiring them. Staff will have information such as address, salary and name. Staff are employed by the organising body.

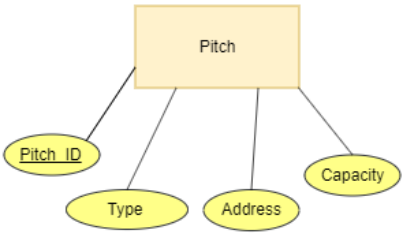
Entity Diagram

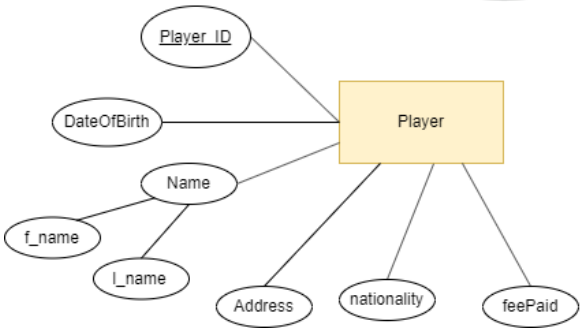


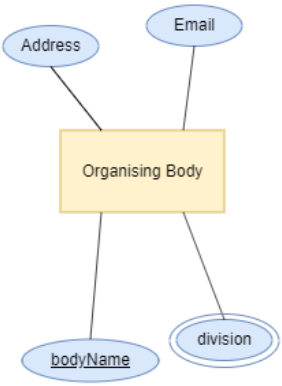




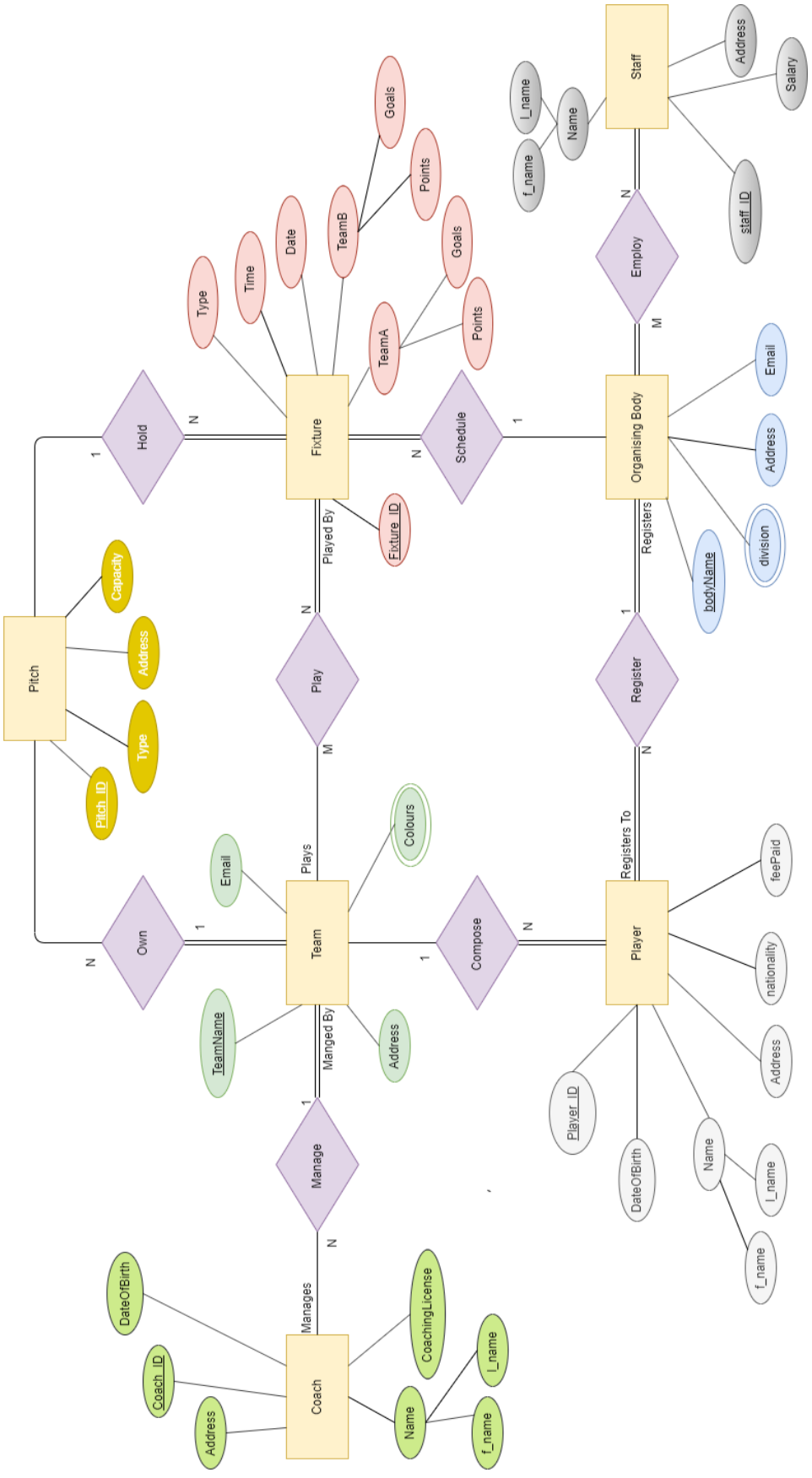




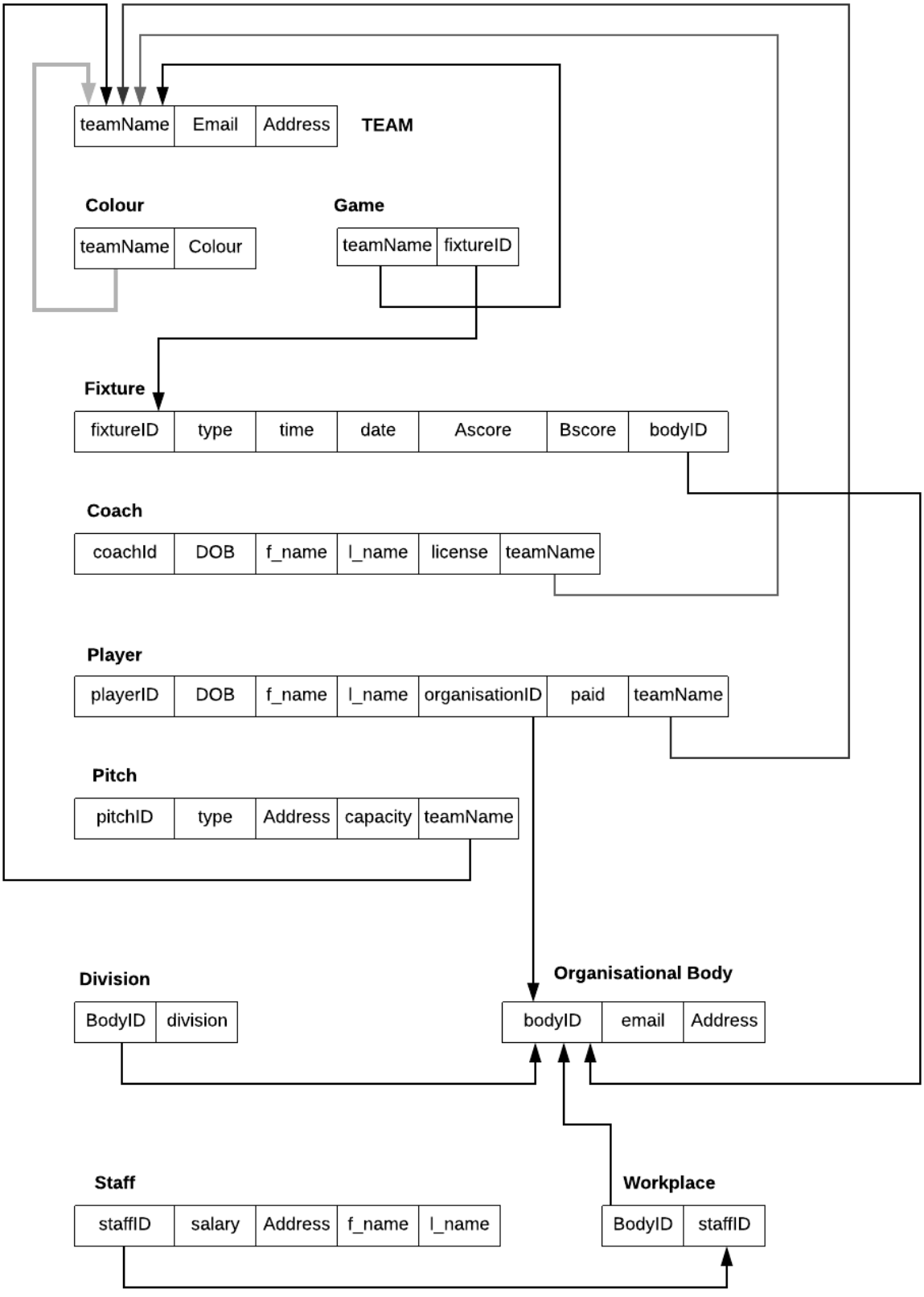




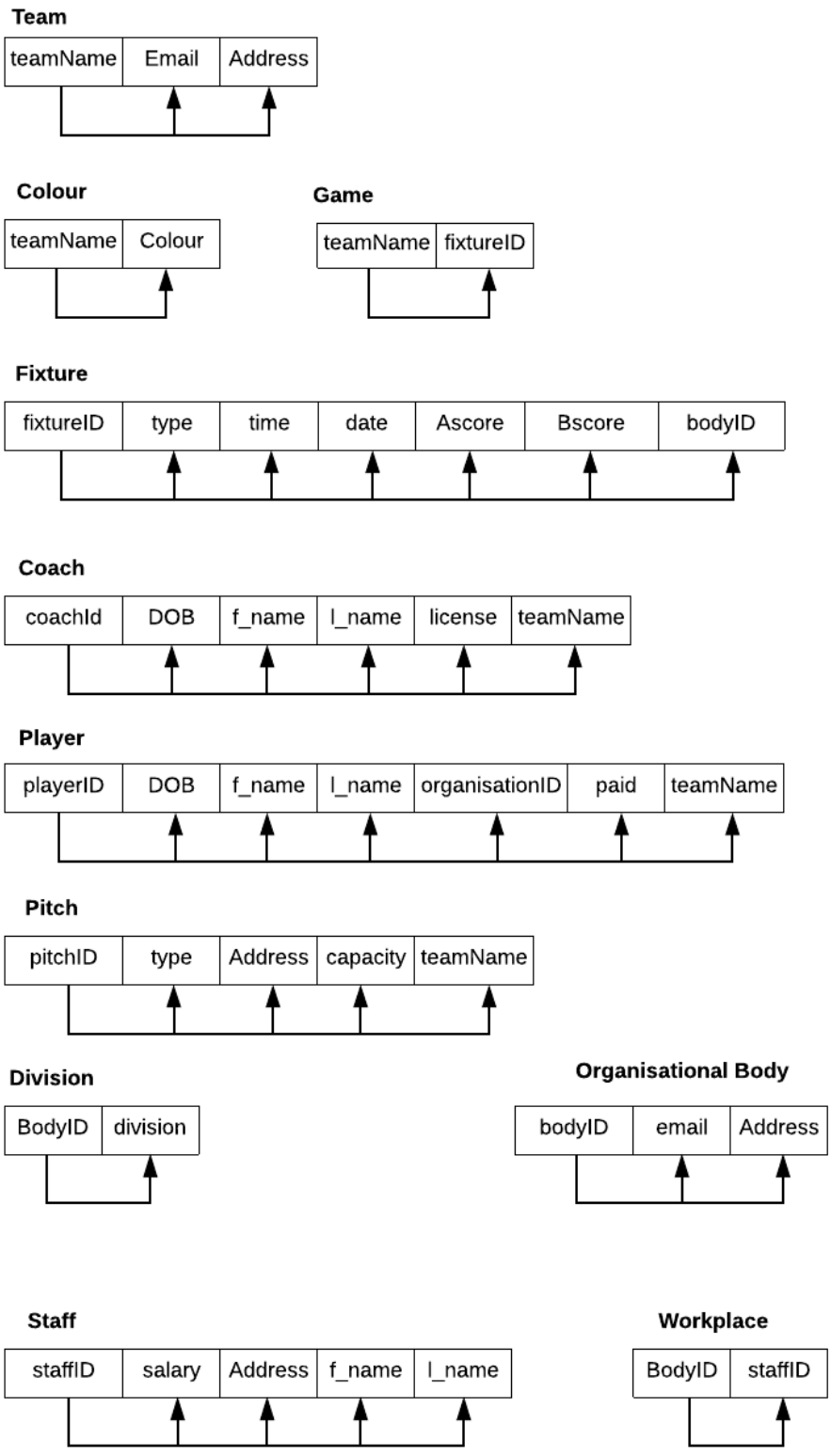
Entity Relationship Diagram



Relational Schema



Functional Dependency



Normalization

My original design was already normalized. I added in two tables with composite primary keys to model the many to many relationships and by having them the many to many attributes were normalized. I also added two tables to represent the multivalued attributes such as colours and division.

When I first approached normalization I thought DOB would decide on the body/league a player would register with however I then realised a youth can register to play adult instead of playing youths and a 46 year old could also apply to play adult instead of over 45s. Therefore, this could not be normalized.

Semantic Constraints

Several semantic constraints were defined for the database to help ensure its integrity and avoid the accidental corruption of information. Semantic constraints can't be expressed in the relational schema.

Entity Integrity Constraints

Specifies that there may not be any NULL values in the Primary Key attribute. Having a NULL in a Primary Key implies that we cannot identify some tuples. In this database that there are to be no NULL values in any of the primary key attributes. I have set all primary keys to NOT NULL.

Referential Integrity

Integrity constraints are specified between two relations. It maintains consistency among tuples in the two relations. A tuple in one relation that refers to another relation, must refer to an existing tuple in that relation.

A Foreign Key formally specifies a Referential Integrity Constraint between two relations. Foreign keys can be null however I have made the decision in this database that they cannot be null. All foreign keys are NOT NULL.

Other Constraints

I use the check function in order to restrict the values that can be entered for an attribute. CHECK is performed for every tuple that is inserted or modified. I implemented the following checks:

- ```
  CONSTRAINT check_coach_license CHECK (license IN ('A', 'B', 'C', 'D'));
  ```

  This ensures each coach has a valid coaching license, they require a minimum of a D license to coach in this league.

- ```
  CONSTRAINT check_paid CHECK (license IN ('Y', 'N'));
  ```

  This ensures the attribute telling the Leinster senior league if a player has paid membership is always yes or no.

- `CONSTRAINT email_check CHECK(email LIKE '%___@___%')`

  I use this constraint to ensure all emails are in the correct format. This checks for anything then an @ and then anything again. This is used twice.

- `CONSTRAINT pitch_type CHECK (type IN ('4G', '3G', 'grass'));`

  This ensures the pitch type is correct, LSL pitches can only be AstroTurf (4G or 3G) or grass type pitches.

Security

The database will be secured using account privileges. There will be four different accounts available, they are:

- Player Account.
- Coach Account.
- General Staff account.
- Database admin account.

Player Account

A public account will be given read access to all fixtures, pitch, team and organisational Body information. None of this contains any private information so it can be shared with the public. This account would be given on request to players as there is not any reason all players should automatically be given this access.

```
Create ROLE Player_Role IDENTIFIED by iPlayBall;
Grant select on Table pitch to Player_Role;
Grant select on Table fixture to Player_Role;
Grant select on Table team to Player_Role;
Grant select on Table OrganisationalBody to Player_Role;

Grant Player_Role TO playerxyz;
```

Coach Account

A coach account will have the same privileges as the player account. A coach however will also be able to access a 'hasPlayerPaid' table/view which contains a list of all player names within the club they coach and whether or not each player has paid their league membership.

Trigger & view – Manager Role

```
CREATE TRIGGER coachRole
BEFORE INSERT ON Coach
FOR EACH ROW

BEGIN
```

```
CREATE VIEW hasPlayerPaid AS
SELECT f_name,l_name,paid,teamName
FROM Players
WHERE Players.paid = new.teamName

Create ROLE Coach_Role IDENTIFIED by iCoach;
Grant select on Table pitch to Coach_Role;
Grant select on Table fixture to Coach_Role;
Grant select on Table team to Coach_Role;
Grant select on Table OrganisationalBody to Coach_Role;

Grant select on Table hasPlayerPaid to Coach_Role;
Grant Coach_Role TO new.coachID;

END coachRole;
```

General Staff Account

Staff will be given read access to all tables within the database. As it is not a huge database it is not necessary for staff to able to modify the database in any way.

Trigger – Staff Role

```
CREATE TRIGGER staffRole
BEFORE INSERT ON Staff
FOR EACH ROW

BEGIN

Create ROLE Staff_Role IDENTIFIED by iStaff;
Grant select to Staff_Role;
Grant Staff_Role TO new.staffID;

END staffRole;
```

Database Admin Account

The database admin will have full privileges to the database and is responsible for user access creation and privilege granting and revoking.

Appendix

Create Tables

```
CREATE TABLE Player (
   id              VARCHAR(12) NOT NULL
  ,f_name          VARCHAR(15) NOT NULL
  ,l_name          VARCHAR(15) NOT NULL
  ,DOB             DATE  NOT NULL
  ,paid            VARCHAR(1) NOT NULL
  ,orgID VARCHAR(7) NOT NULL
  ,teamID          VARCHAR(30)
  PRIMARY KEY (id),
  FOREIGN KEY (orgID) REFERENCES OrganisationalBody(bodyID),
  FOREIGN KEY (teamID) REFERENCES Team(teamName),
```

```
  CONSTRAINT check_paid CHECK (license IN ('Y', 'N'));
);

CREATE TABLE Team(
   teamName VARCHAR(25) NOT NULL
  ,email    VARCHAR(35) NOT NULL
  ,address  VARCHAR(75) NOT NULL,
  PRIMARY KEY (teamName),
  CONSTRAINT email_check CHECK(email LIKE '%___@___%')
);

CREATE TABLE Colour(
    teamName VARCHAR(25) NOT NULL
    ,colour VARCHAR(20) NOT NULL,
    PRIMARY KEY (teamName),
    FOREIGN KEY (teamName) REFERENCES Team(teamName)
);

CREATE TABLE Game(
    teamName VARCHAR(25) NOT NULL,
    ,fixtureID VARCHAR(15) NOT NULL,
    PRIMARY KEY (teamName),
    FOREIGN KEY (teamName) REFERENCES Team(teamName)
);

CREATE TABLE Fixture(
   fixtureID VARCHAR(8) NOT NULL
  ,time      VARCHAR(5) NOT NULL
  ,date      DATE  NOT NULL
  ,Ascore    INTEGER
  ,Bscore    INTEGER
  ,bodyID    VARCHAR(10) NOT NULL
  ,pitchID   VARCHAR(8) NOT NULL,
  PRIMARY KEY (fixtureID),
  FOREIGN KEY (bodyID) REFERENCES OrganisationalBody(bodyID),
  FOREIGN KEY (pitchID) REFERENCES Pitch(pitchID)
);

CREATE TABLE Coach(
   coachID  VARCHAR(10) NOT NULL
  ,DOB      DATE  NOT NULL
  ,f_name   VARCHAR(15) NOT NULL
  ,l_name   VARCHAR(15) NOT NULL
  ,license  VARCHAR(1) NOT NULL
  ,teamName VARCHAR(25),
  PRIMARY KEY (coachID),
  FOREIGN KEY (teamName) REFERENCES Team(teamName),
  CONSTRAINT check_coach_license CHECK (license IN ('A', 'B', 'C', 'D'));
);

CREATE TABLE Pitch(
   pitchID  VARCHAR(8) NOT NULL
  ,pitchType     VARCHAR(5) NOT NULL
  ,address  VARCHAR(37) NOT NULL
  ,capacity INTEGER
  ,teamName VARCHAR(25)
  PRIMARY KEY (pitchID),
  FOREIGN KEY (teamName) REFERENCES Team(teamName),
  CONSTRAINT pitch_type CHECK (type IN ('4G', '3G', 'grass'));
);
```

```
CREATE TABLE Division(
   bodyID  VARCHAR(10) NOT NULL
  ,division     VARCHAR(15) NOT NULL,
  PRIMARY KEY (bodyID),
  FOREIGN KEY (bodyID) REFERENCES OrganisationalBody(bodyID)
  );

   CREATE TABLE OrganisationalBody(
   bodyID   VARCHAR(10) NOT NULL
  ,email    VARCHAR(25) NOT NULL,
  address   VARCHAR(20) NOT NULL,
  PRIMARY KEY (bodyID),
  FOREIGN KEY (bodyID) REFERENCES OrganisationalBody(bodyID)
  );

  CREATE TABLE Staff(
   staffID    VARCHAR(10) NOT NULL
  ,address    VARCHAR(40) NOT NULL
  ,f_name     VARCHAR(15) NOT NULL
  ,Salary     INTEGER  NOT NULL
  ,l_name     VARCHAR(15) NOT NULL
  ,department VARCHAR(20) NOT NULL
  ,PRIMARY KEY (staffID)
);

CREATE TABLE Workplace(
bodyID  VARCHAR(10) NOT NULL,
   staffID VARCHAR(8) NOT NULL,
PRIMARY KEY (bodyID)
);
```

Triggers

```
CREATE TRIGGER coachRole
BEFORE INSERT ON Coach
FOR EACH ROW

BEGIN
CREATE VIEW hasPlayerPaid AS
SELECT f_name,l_name,paid,teamName
FROM Players
WHERE Players.paid = new.teamName;

Create ROLE Coach_Role IDENTIFIED by iCoach;
Grant select on Table pitch to Coach_Role;
Grant select on Table fixture to Coach_Role;
Grant select on Table team to Coach_Role;
Grant select on Table OrganisationalBody to Coach_Role;
Grant select on Table hasPlayerPaid to Coach_Role;
Grant Coach_Role TO new.coachID;
END coachRole;
RUN;



CREATE TRIGGER staffRole
BEFORE INSERT ON Staff
FOR EACH ROW
```

```
BEGIN
Create ROLE Staff_Role IDENTIFIED by iStaff;
Grant select to Staff_Role;
Grant Staff_Role TO new.staffID;
END staffRole;
RUN;
```

Insert Statements

```
INSERT INTO Player (id,f_name,l_name,DOB,paid,orgID,teamID) VALUES
('Player1','Pace','Yakunin','1980/02/02','Y','adult','DCU AFC');
INSERT INTO Player (id,f_name,l_name,DOB,paid,orgID,teamID) VALUES
('Player2','Hort','Goodbody','y1985/04/04','Y','adult','DCU AFC');
INSERT INTO Player (id,f_name,l_name,DOB,paid,orgID,teamID) VALUES
('Player3','Constance','Verrechia','18/06/1995','Y','adult','Dublin
University');
INSERT INTO Player (id,f_name,l_name,DOB,paid,orgID,teamID) VALUES
('Player4','Leela','Cowwell','23/10/1998','Y','adult','UCD AFC');
INSERT INTO Player (id,f_name,l_name,DOB,paid,orgID,teamID) VALUES
('Player5','Bernardina','Perrie','21/05/1998','Y','adult','Dublin
University');
INSERT INTO Player (id,f_name,l_name,DOB,paid,orgID,teamID) VALUES
('Player6','Cyrill','Maplestone','10/03/1980','N','adult','Dublin
University');
INSERT INTO Player (id,f_name,l_name,DOB,paid,orgID,teamID) VALUES
('Player7','Bonnie','Bedborough','13/06/1988','Y','adult','Dublin
University');
INSERT INTO Player (id,f_name,l_name,DOB,paid,orgID,teamID) VALUES
('Player8','Chad','Hounson','19/10/1976','Y','adult','Cherry Orchard');
INSERT INTO Player (id,f_name,l_name,DOB,paid,orgID,teamID) VALUES
('Player9','Dulcea','Roscow','02/01/1971','N','adult','UCD AFC');
INSERT INTO Player (id,f_name,l_name,DOB,paid,orgID,teamID) VALUES
('Player10','Emlynne','Timmons','25/10/1988','N','adult','DCU AFC');
INSERT INTO Player (id,f_name,l_name,DOB,paid,orgID,teamID) VALUES
('Player11','Clari','Beagin','25/04/1991','Y','adult','Cherry Orchard');
INSERT INTO Player (id,f_name,l_name,DOB,paid,orgID,teamID) VALUES
('Player12','Kendricks','Tilley','02/08/1987','Y','adult','Cherry
Orchard');
INSERT INTO Player (id,f_name,l_name,DOB,paid,orgID,teamID) VALUES
('Player13','Griswold','Brecknall','27/07/1993','N','adult','Cherry
Orchard');
INSERT INTO Player (id,f_name,l_name,DOB,paid,orgID,teamID) VALUES
('Player14','Rriocard','Carse','19/07/1971','N','adult','Dublin
University');
INSERT INTO Player (id,f_name,l_name,DOB,paid,orgID,teamID) VALUES
('Player15','Brandie','Fealey','17/01/1980','N','adult','DCU AFC');
INSERT INTO Player (id,f_name,l_name,DOB,paid,orgID,teamID) VALUES
('Player16','Chad','Fick','05/06/1972','Y','adult','UCD AFC');
INSERT INTO Player (id,f_name,l_name,DOB,paid,orgID,teamID) VALUES
('Player17','Winnie','Swale','17/12/1992','Y','adult','Cherry Orchard');
INSERT INTO Player (id,f_name,l_name,DOB,paid,orgID,teamID) VALUES
('Player18','Sunny','Wiggett','22/08/1980','N','adult','DCU AFC');
INSERT INTO Player (id,f_name,l_name,DOB,paid,orgID,teamID) VALUES
('Player19','Artemas','Gilchrest','07/09/1979','Y','adult','UCD AFC');
```

```
INSERT INTO Player (id,f_name,l_name,DOB,paid,orgID,teamID) VALUES
('Player20','Nerte','Tunnoch','09/07/1991','Y','adult','Cherry Orchard');
INSERT INTO Player (id,f_name,l_name,DOB,paid,orgID,teamID) VALUES
('Player21','Theressa','Fassum','20/08/1984','N','adult','Dublin
University');
INSERT INTO Player (id,f_name,l_name,DOB,paid,orgID,teamID) VALUES
('Player22','Obie','Lewsam','25/04/1975','N','adult','Dublin University');
INSERT INTO Player (id,f_name,l_name,DOB,paid,orgID,teamID) VALUES
('Player23','Pren','Lardner','19/01/1986','Y','adult','Cherry Orchard');
INSERT INTO Player (id,f_name,l_name,DOB,paid,orgID,teamID) VALUES
('Player24','Tabby','McGarrell','17/01/1997','Y','adult','UCD AFC');
INSERT INTO Player (id,f_name,l_name,DOB,paid,orgID,teamID) VALUES
('Player25','Lydia','Schneidar','09/03/1999','Y','adult','DCU AFC');
```

```
INSERT INTO Team (teamName,email,address) VALUES ('DCU
AFC','dcuSoccer@gmail.com','sports centre, DCU, DUBLIN');
INSERT INTO Team (teamName,email,address) VALUES ('Dublin
University','DUAFC@gmail.com','sports centre, Trinity College, DUBLIN');
INSERT INTO Team (teamName,email,address) VALUES ('UCD
AFC','UCDAFC@gmail.com','sports centre, UCD, DUBLIN');
INSERT INTO Team (teamName,email,address) VALUES ('Cherry
Orchard','CheeryOrchard@gmail.com','Cherry Orchard, Drumcondra, Dublin');
```

```
INSERT INTO Colour (teamName,colour) VALUES ('DCU AFC', 'Royal Blue');
INSERT INTO Colour (teamName,colour) VALUES ('DCU AFC', 'White');
INSERT INTO Colour (teamName,colour) VALUES ('UCD AFC', 'Baby Blue');
INSERT INTO Colour (teamName,colour) VALUES ('UCD AFC', 'Navy');
INSERT INTO Colour (teamName,colour) VALUES ('Dublin University', 'Red');
INSERT INTO Colour (teamName,colour) VALUES ('Dublin University', 'Green');
INSERT INTO Colour (teamName,colour) VALUES ('Cherry Orchard', 'Red');
INSERT INTO Colour (teamName,colour) VALUES ('Dublin University', 'Baby
Blue');
```

```
INSERT INTO Game (teamName,fixtureID) VALUES ('DCU AFC', 'fixture1');
INSERT INTO Game (teamName,fixtureID) VALUES ('DCU AFC', 'fixture3');
INSERT INTO Game (teamName,fixtureID) VALUES ('UCD AFC', 'fixture2');
INSERT INTO Game (teamName,fixtureID) VALUES ('UCD AFC', 'fixture3');
INSERT INTO Game (teamName,fixtureID) VALUES ('Dublin University',
'fixture1');
INSERT INTO Game (teamName,fixtureID) VALUES ('Dublin University',
'fixture4');
INSERT INTO Game (teamName,fixtureID) VALUES ('Cherry Orchard',
'fixture2');
INSERT INTO Game (teamName,fixtureID) VALUES ('Cherry Orchard',
'fixture4');
```

```
INSERT INTO Fixture(fixtureID,time,date,Ascore,Bscore,bodyID,pitchID)
VALUES ('fixture1','13:00','12/12/2019',3,3,'adult','pitch1');
INSERT INTO Fixture(fixtureID,time,date,Ascore,Bscore,bodyID,pitchID)
VALUES ('fixture2','13:00','12/12/2019',2,4,'adult','pitch2');
INSERT INTO Fixture(fixtureID,time,date,Ascore,Bscore,bodyID,pitchID)
VALUES ('fixture3','13:00','19/12/2019',1,5,'adult','pitch3');
```

```
INSERT INTO Fixture(fixtureID,time,date,Ascore,Bscore,bodyID,pitchID)
VALUES ('fixture4','13:00','19/12/2019',4,4,'adult','pitch4');


INSERT INTO Coach(coachID,DOB,f_name,l_name,license,teamName) VALUES
('coach1','05/05/1980','John','Boyle','B','DCU AFC');
INSERT INTO Coach(coachID,DOB,f_name,l_name,license,teamName) VALUES
('coach2','06/05/1980','Barry','Byrne','B','UCD AFC');
INSERT INTO Coach(coachID,DOB,f_name,l_name,license,teamName) VALUES
('coach3','07/05/1980','Richard','Maguire','D','Dublin University');
INSERT INTO Coach(coachID,DOB,f_name,l_name,license,teamName) VALUES
('coach4','08/05/1980','Steve','Madden','C','Cherry Orchard');
INSERT INTO Coach(coachID,DOB,f_name,l_name,license,teamName) VALUES
('coach5','08/05/1980','David','Madden','C','Cherry Orchard');


INSERT INTO Pitch(pitchID,pitchType,address,capacity,teamName) VALUES
('pitch1','Grass','bowl,UCD,Dublin',4000,'UCD AFC');
INSERT INTO Pitch(pitchID,pitchType,address,capacity,teamName) VALUES
('pitch2','Grass','college park,trinity college,dublin',NULL,'Dublin
University');
INSERT INTO Pitch(pitchID,pitchType,address,capacity,teamName) VALUES
('pitch3','4G','DCU college sports,dcu,dublin',NULL,'DCU AFC');
INSERT INTO Pitch(pitchID,pitchType,address,capacity,teamName) VALUES
('pitch4','Grass','DCU college sports,dcu,dublin',NULL,'DCU AFC');
INSERT INTO Pitch(pitchID,pitchType,address,capacity,teamName) VALUES
('pitch5','Grass','cherry orcharcd fc, drumcondra,dublin',NULL,'Cherry
Orchard');


INSERT INTO Division(bodyID,division) VALUES ('adult','Saturday 1');
  INSERT INTO Division(bodyID,division) VALUES ('adult','Saturday 2');
  INSERT INTO Division(bodyID,division) VALUES ('adult','Sunday 3');
  INSERT INTO Division(bodyID,division) VALUES ('adult','Sunday 4');
  INSERT INTO Division(bodyID,division) VALUES ('youth','youth 1');
  INSERT INTO Division(bodyID,division) VALUES ('over-45','over-45 1');


  INSERT INTO OrganisationalBody(bodyID,email,address) VALUES
('adult','adult@lsl.ie','4 park row, cabra, dublin');
  INSERT INTO OrganisationalBody(bodyID,email,address) VALUES ('over-
45','over-45@lsl.ie','4 park row, cabra, dublin');
  INSERT INTO OrganisationalBody(bodyID,email,address) VALUES
('college','college@lsl.ie','4 leland road, cabra, dublin');
  INSERT INTO OrganisationalBody(bodyID,email,address) VALUES ('over-
60','over-60@lsl.ie','4 park row, cabra, dublin');
  INSERT INTO OrganisationalBody(bodyID,email,address) VALUES
('youth','youth@lsl.ie','8 street way, walkinstown, dublin');


  INSERT INTO Staff(staffID,address,f_name,Salary,l_name,department) VALUES
('staff1','0747 Sullivan Avenue','Benton','Gerholz',20000,'child welfare');
INSERT INTO Staff(staffID,address,f_name,Salary,l_name,department) VALUES
('staff2','18565 Cottonwood Crossing','Sunny','Freshwater',25000,'Club
relations');
```

```
INSERT INTO Staff(staffID,address,f_name,Salary,l_name,department) VALUES
('staff3','6400 Daystar Pass','Frannie','Capaldo',25000,'Secretary');
INSERT INTO Staff(staffID,address,f_name,Salary,l_name,department) VALUES
('staff4','55991 Heffernan Place','Tami','Davern',25000,'database admin');
INSERT INTO Staff(staffID,address,f_name,Salary,l_name,department) VALUES
('staff5','26 Blackbird Way','Rhodia','Aaronson',40000,'Head');
INSERT INTO Staff(staffID,address,f_name,Salary,l_name,department) VALUES
('staff6','8 Anzinger Center','Codi','Rubinov',25000,'Services');
```

```
INSERT INTO Workplace(staffID,bodyID) VALUES ('staff1','youth');
INSERT INTO Workplace(staffID,bodyID) VALUES ('staff2','over-45');
INSERT INTO Workplace(staffID,bodyID) VALUES ('staff2','adult1');
INSERT INTO Workplace(staffID,bodyID) VALUES ('staff2','youth');
INSERT INTO Workplace(staffID,bodyID) VALUES ('staff3','adult1');
INSERT INTO Workplace(staffID,bodyID) VALUES ('staff3','youth');
INSERT INTO Workplace(staffID,bodyID) VALUES ('staff3','over-45');
INSERT INTO Workplace(staffID,bodyID) VALUES ('staff4','adult1');
INSERT INTO Workplace(staffID,bodyID) VALUES ('staff4','over-45');
INSERT INTO Workplace(staffID,bodyID) VALUES ('staff5','adult1');
INSERT INTO Workplace(staffID,bodyID) VALUES ('staff5','over-45');
INSERT INTO Workplace(staffID,bodyID) VALUES ('staff6','adult1');
INSERT INTO Workplace(staffID,bodyID) VALUES ('staff6','over-45');
INSERT INTO Workplace(staffID,bodyID) VALUES ('staff6','over-60');
INSERT INTO Workplace(staffID,bodyID) VALUES ('staff6','college');
```