CPCS241-Database I- 2025-Project

[Saudi eSports Federation Database]

DB Design

Group No:

Name	ID	Section
Abdulrahman Alfrihidi		CS1
		CS3
		CS1
		CS2
		CS2
		CS1

Contents

PART I: Analysis 4
1 Problem Definition and Data Requirements 4
1.1 Problem Description 4
1.2 Data Requirements 4
1.3 Business Rules 6
1.4 Intended Output of the system 7
PART II: DB DEISGN 9
2 ER Diagram Design 9
2.1 ER diagram 9
2.2 Design of Business Rules 10
3 ER-to-logical schema mapping 11
3.1 Mapping of Regular Entity Types 11
3.2 Mapping of Weak Entity Types 16
3.3 Mapping of binary 1-1 relationship types 16
3.4 Mapping of binary 1-N relationship types 17
3.5 Mapping of binary M-N relationship types 22
3.6 Mapping of multivalued attributes 23
3.7 Mapping of n-ary relationship types 23
3.8 Schema Diagram 24
4 Normalization 25
4.1 First Normal Form 25
4.2 Second Normal Form 25
4.3 Third Normal Form 25
5 Final DB Schema Diagram 26
PART III: IMPLEMENTATION 27
6 Table Creation Script 27
6.1 <*Clan*> TABLE 27
6.2 <*Game*> TABLE 27
6.3 <*Team*> TABLE 27
6.4 <*Mamber*> TABLE 28
6.5 <*Clan_Ach*> TABLE 28
6.6 <*Tournament*> TABLE 29
6.7 <*Participation*> TABLE 29
6.8 < *Match*> TABLE 30
6.9 <*Plays*> TABLE 30
6.10 <*Locations*> TABLE 31
6.11 <*Sponsor* > TABLE 31

6.12 <*Sponsor_Ship*> TABLE 31
7 Constraints Script 32
8 Queries 32
8.1 < *List all teams with their clan names.*> 32
8.2 < *Get all tournaments with their game names.*> 32
8.3 < *TOP 3 sponsorship contracts.*> 33
8.4 < *Show all teams participated on '2025-06-01'. >* 33
8.5 < *Show TOP 5 Average prize pool per tournament.*> 33
8.6 < *List all tournaments with the winning team name and its clan name.*> 34
8.7 < *Show total number of matches played by each team.*> 34

PART I: Analysis

1 Problem Definition and Data Requirements

1.1 Problem Description

We represent the **Saudi eSports Federation Database**, which is designed to manage and organize the structure of eSports in Saudi Arabia. As the industry of eSports is growing rapidly in the world, more than 140 countries have their own eSports federation organization. In Saudi Arabia, there are more than 150 Saudi eSports teams, which require efficient system management. Our comprehensive database will provide an organized way to track and manage players, coaches, staff, clans, tournaments, games, sponsors, schedules.

1.2 Data Requirements

Listing all entities with the required data:

1. **Member**

- email
- name
- nickname
- member_id
- role
- nationality
- date_of_birth

2. **Clan**

- country
- creation_date
- clan_name
- clan_id

3. **Clan Ach**

- date_awarded
- title
- achievement_id

4. **Team**

- team_id
- team_name

5. **Participation**

- participation_id

6. **Tournament**

- end_date
- start_date
- name
- tournament_id
- prize_pool
- champion

7. **Game**

- platform
- genre
- game_name
- game_id
- release_year

8. **Match**

- start_time
- end_time
- location
- match_id
- winner_clan_id
- score

9. **Sponsor**

- name
- owner
- spinosaurid

Tournament-Sponsor Relationship (SPONSORED):

- start_date
- end_date
- contract_value

Participation-Team Participation Relationship (PARTICIPATE):

- participation_date

1.3 Business Rules

Member

- Each member can only belong to one clan.
- Each member must be assigned to a team.

Clan

- Every clan shall have at least 10 members.
- A clan can achieve zero or more clan achievements.
- Each clan can have one or many teams that represent the clan.

Clan Ach

- Only one clan can achieve each specific clan achievement.

Team

- Each team competes in one game.
- Each team should play at least two matches or more.
- Every team must be part of a clan.
- The team must have at least one member assigned to.
- Each team can have zero or many participations.

Participation

- Every participation for a specific team.
- exactly one participation in a specific tournament.

Tournament

- Each tournament shall consist of one game.
- Every tournament can host one or many matches.
- Each tournament shall be sponsored by at least 1 sponsor.
- Each tournament has at least one to start.

Game

- Some games may consist of zero or more tournaments.
- Some games may provide zero or more competitions for teams to compete in.

Match

- Every match shall be hosted by exactly one tournament.
- Each match has exactly two teams playing against each other.

Sponsor

- At least one sponsor for each tournament.

1.4 Intended Output of the system

Queries

- List all teams with their clan names.
- Get all tournaments with their game names.
- TOP 3 sponsorship contracts.
- Show all teams participated on ‘2025-06-01‘.
- Show TOP 5 Average prize pool per tournament.
- List all tournaments with the winning team name and its clan name.
- Show total number of matches played by each team.

Reports

- **Clan Performance Report:** Clan name, number of members, number of achievements, and number of teams.
- **Tournament Overview Report:** Tournament name, game, sponsors, number of participants, matches hosted.
- **Match Summary Report:** Match ID, tournament, teams involved, game
- **Team Composition Report:** For each team: team name, clan, game, list of assigned members.
- **Game Popularity Report:** List of games with counts of tournaments and teams competing in them.

Calculations

1. Clan Metrics

a. Total number of clans.

b. Average number of members per clan.

c. Count of clans with achievements.

2. Team Metrics

a. Average number of members per team.

b. Count of teams per game.

3. Tournament Metrics

a. Total number of tournaments per game.

b. Average number of matches per tournament.

c. Average number of sponsors per tournament.

4. Participation Metrics

a. Number of participants per tournament.

b. Number of matches each team has played.

c. Top clans by number of tournament participants.

5. Game and Match Analytics

a. Game with the most tournaments.

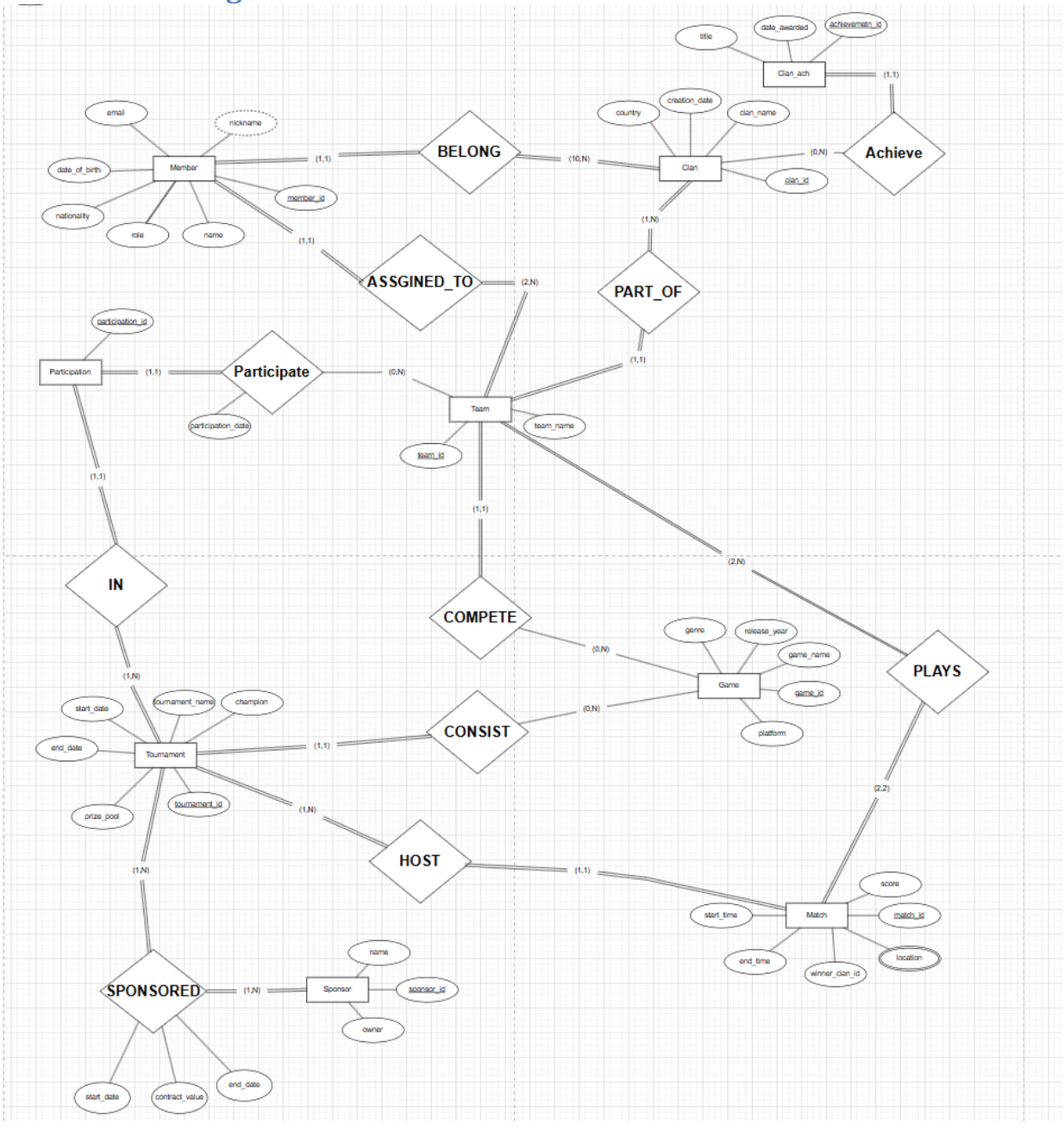
b. Matches played per game.

c. Most active teams cross all matches.

PART II: DB DEISGN

2 ER Diagram Design

2.1 ER diagram



2.2 Design of Business Rules

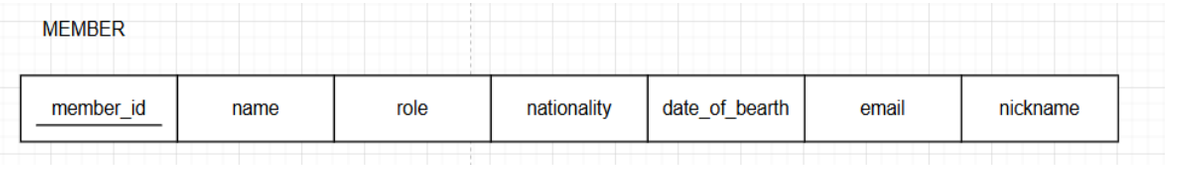
In this subsection, show how the business rules have been translated into design decisions. Some business rules can be deployed during implementation phase only. Provide sufficient justification.

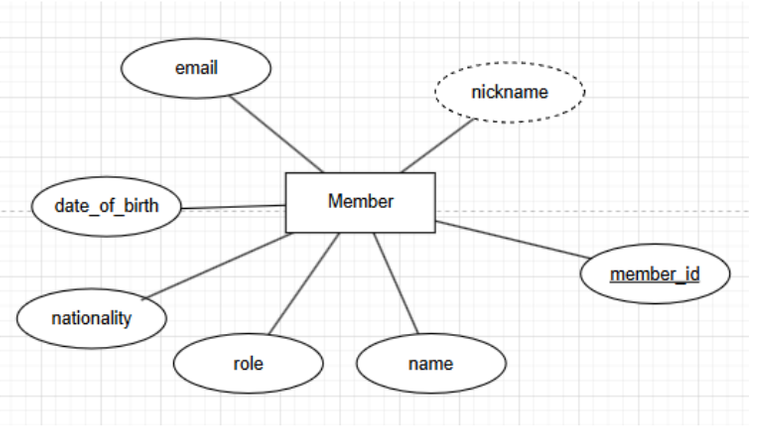
Business Rules	Design Decision	Justification
Each member can only belong to one clan.	Allow members to be linked to one clan only.	Prevents conflicts and maintains clarity in clan membership.
Each member must be assigned to a team.	Require mandatory team assignment for members.	Ensures structured participation and team involvement.
Every clan shall have at least 10 members.	Enforce a minimum member requirement for clans.	Ensures clans are large enough to function competitively.
A clan can achieve zero or more clan achievements.	Allow clans to have optional achievements.	Reflects performance variance across clans.
Each clan can have one or many teams that represent the clan.	Allow clans to have multiple representing teams.	Enables scalability and diversity in clan participation.
Only one clan can achieve each specific clan achievement.	Restrict each achievement to be owned by one clan.	Ensures uniqueness and value of achievements.
Each team competes in one game.	Fix team competition to a single game.	Keeps team focus and specialization.
Each team should play at least two matches or more.	Require teams to participate in a minimum of two matches.	Ensures meaningful competition and team activity.
Every team must be part of a clan.	Require each team to belong to a clan.	Maintains a hierarchical structure within the organization.
The team must have at least two members assigned to.	Enforce a minimum team size of one member.	Ensures no team is empty or non-functional.
Each team can have zero or many participations.	Allow teams to optionally participate in tournaments.	Provides flexibility in team engagement levels.
Every participation for a specific team.	Associate each participation with exactly one team.	Ensures clear accountability and participation tracking.
Exactly one participation in a specific tournament.	Restrict each participation to exactly one tournament.	Maintains clear boundaries and organization for tournament participation.
Each tournament shall consist of one game.	Associate one game with each tournament.	Keeps tournament rules consistent and focused.
Every tournament can host one or many matches.	Allow tournaments to include multiple matches.	Ensures tournaments are scalable and competitive.
Each tournament shall be sponsored by at least 1 sponsor.	Require sponsorship to validate a tournament.	Emphasizes the importance of sponsor involvement.
Each tournament has at least one to start.	Require at least one participating team to start a tournament.	Ensures viability and prevents empty tournaments.
Some games may consist of zero or more tournaments.	Allow games to exist without being used in tournaments.	Supports inactive or upcoming games.
Some games may provide zero or more competitions for teams to compete in.	Allow flexibility in games providing competitions.	Accounts for varying game types and formats.
Every match shall be hosted by exactly one tournament.	Require matches to be part of exactly one tournament.	Maintains organizational hierarchy.
Each match has exactly two teams play against each other.	Enforce exactly two teams per match.	Ensures meaningful competition structure.
At least one sponsor for each tournament.	Mandate a minimum of one sponsor per tournament.	Highlights sponsor relevance and financial support.

3 ER-to-logical schema mapping

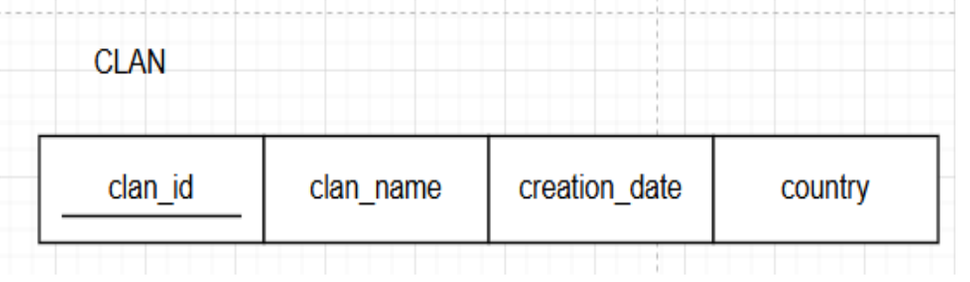
3.1 Mapping of Regular Entity Types

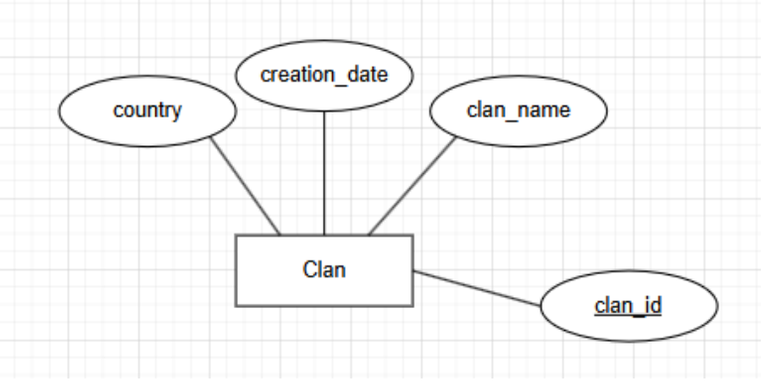
Member:



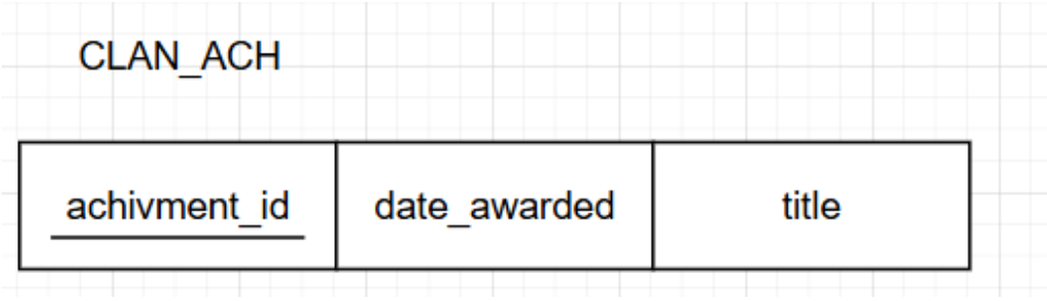


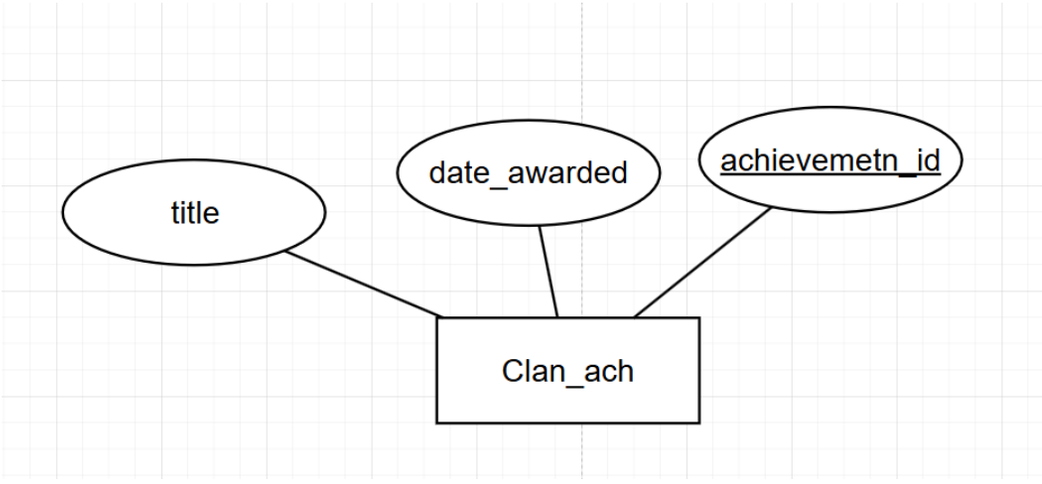
Clan:



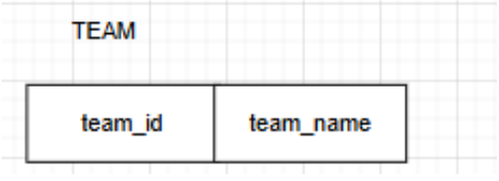


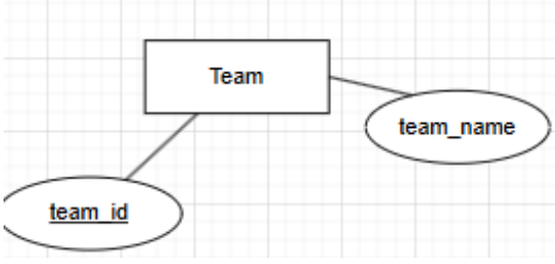
Clan_ach:



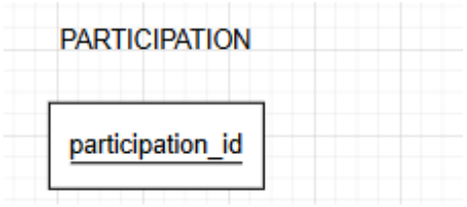


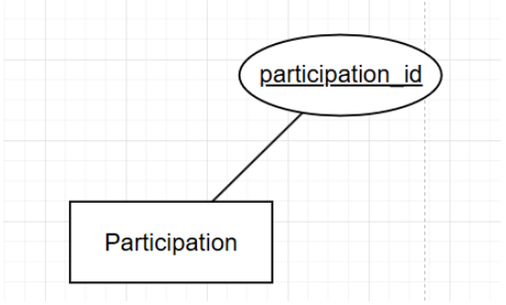
Team:



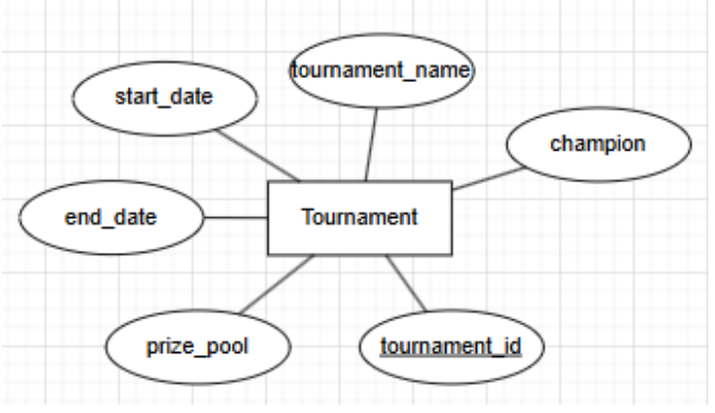


Participation:

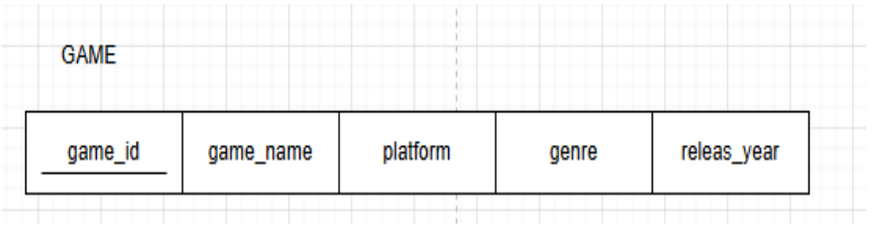


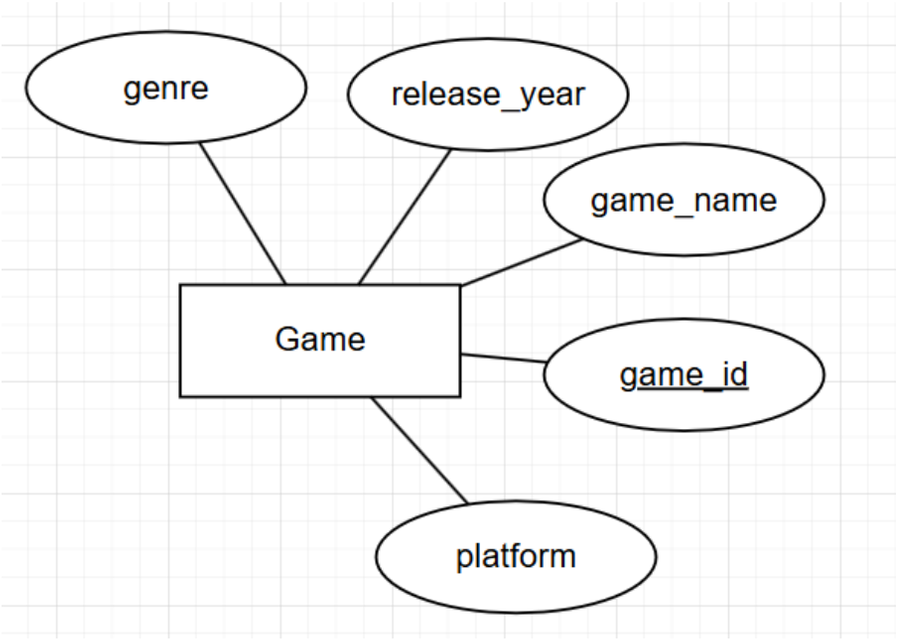


Tournament:

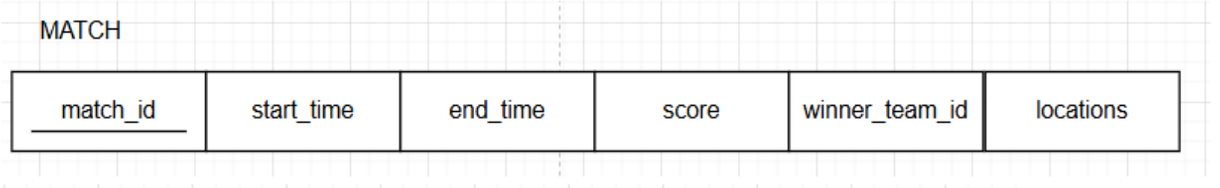


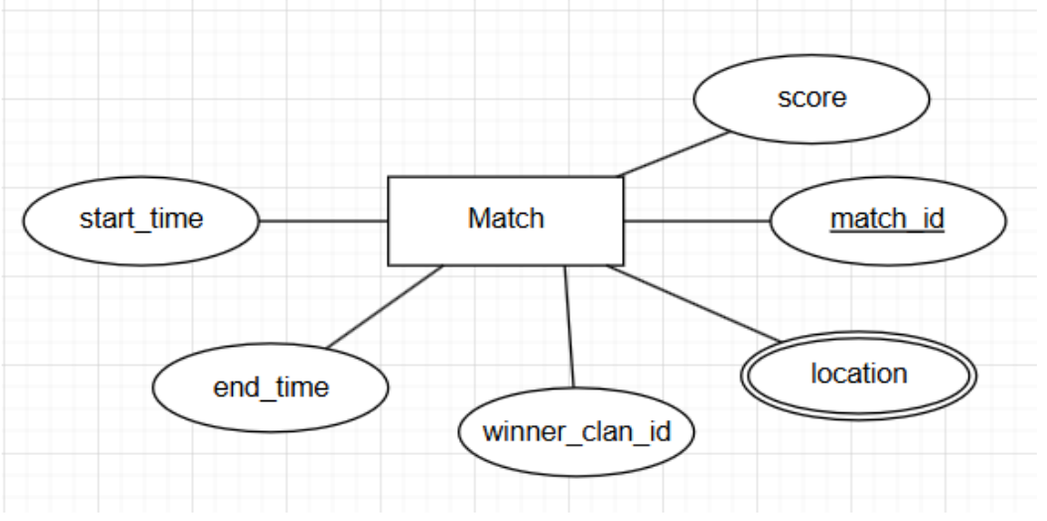
Game:


GAME
game_id
game_name
platform
genre
releas_year

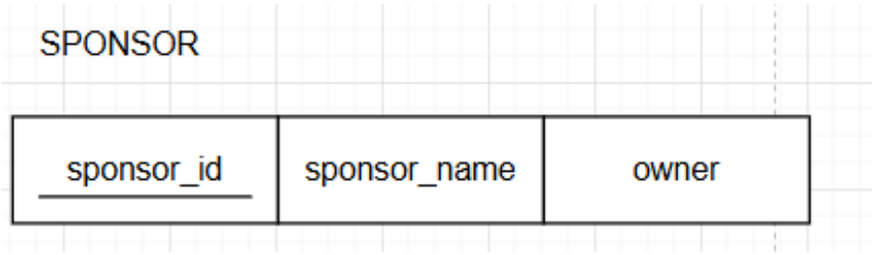

genre
release_year
game_name
Game
game_id
platform

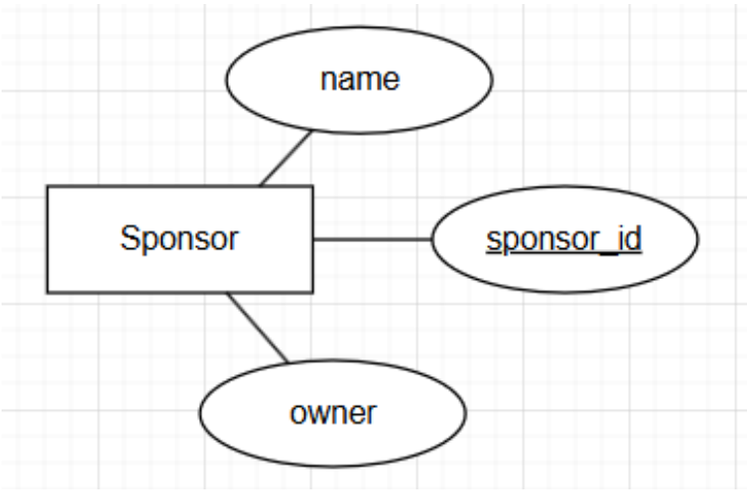
Match:


MATCH
match_id
start_time
end_time
score
winner_team_id
locations


score
start_time
Match
match_id
end_time
location
winner_clan_id

Sponsor:


SPONSOR
sponsor_id
sponsor_name
owner


name
Sponsor
sponsor_id
owner

3.2 Mapping of Weak Entity Types

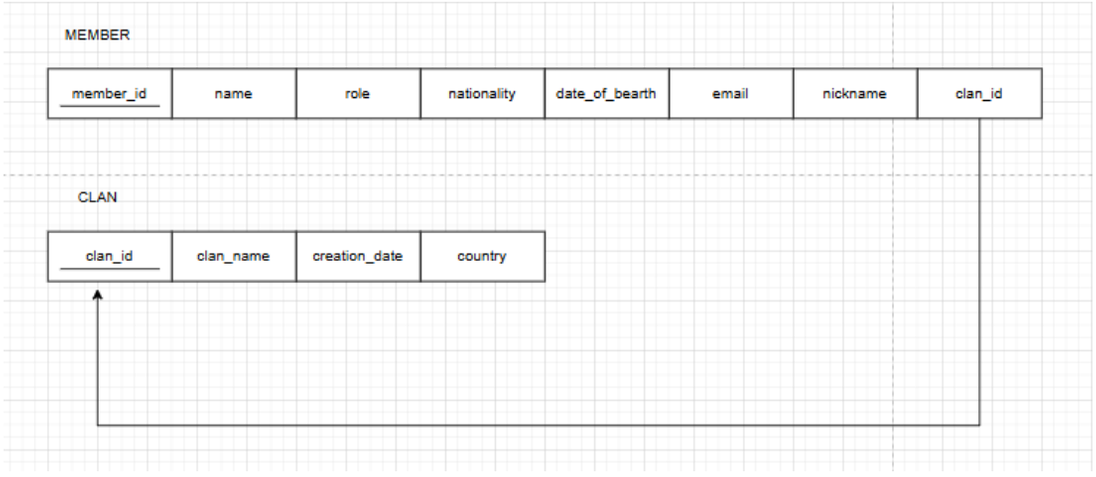
No Weak Entity Types.

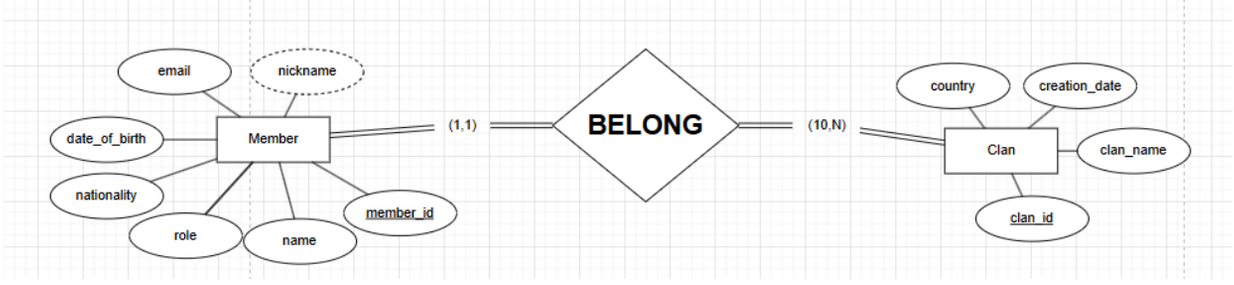
3.3 Mapping of binary 1-1 relationship types

No 1-1 relationship type.

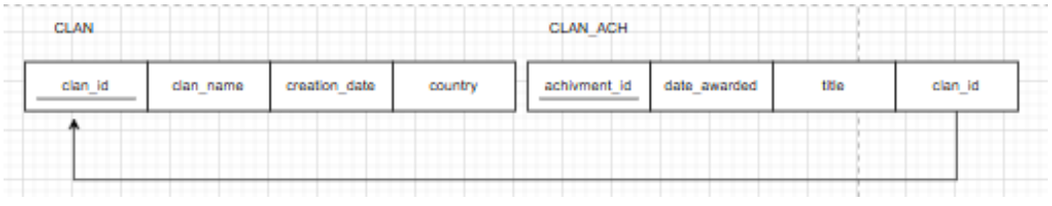
3.4 Mapping of binary 1-N relationship types

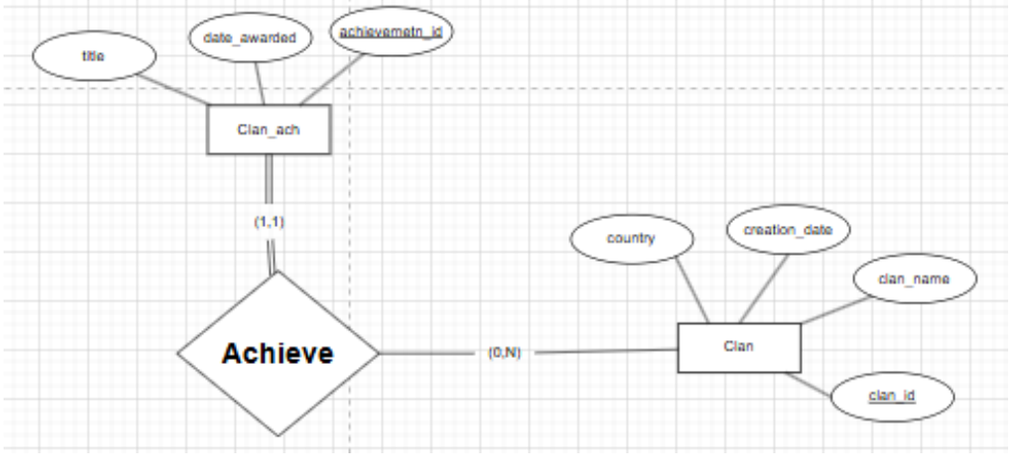
1.



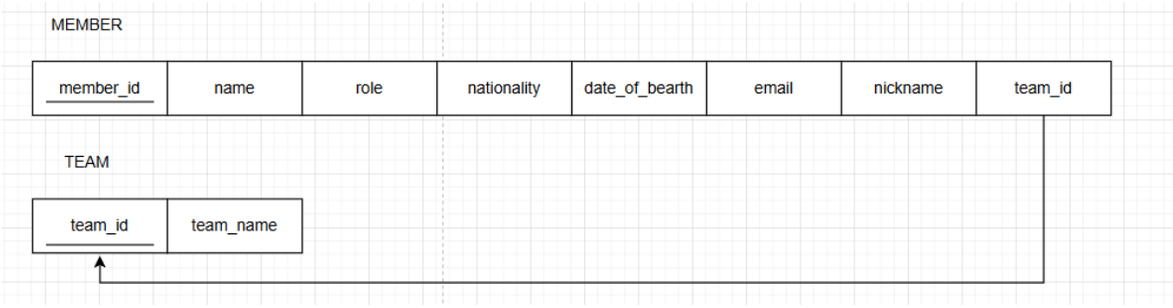


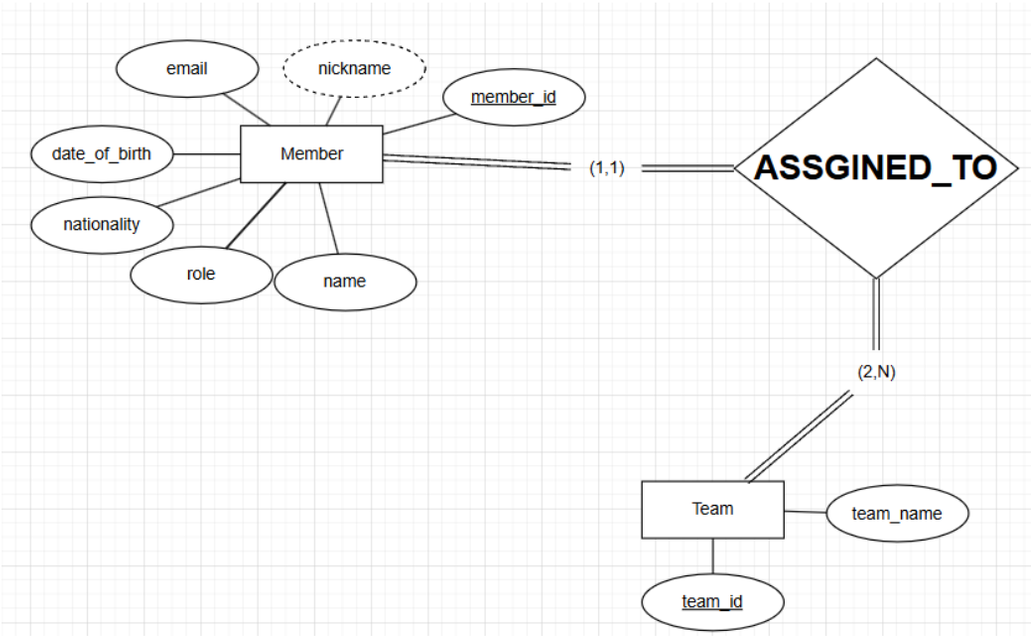
2.



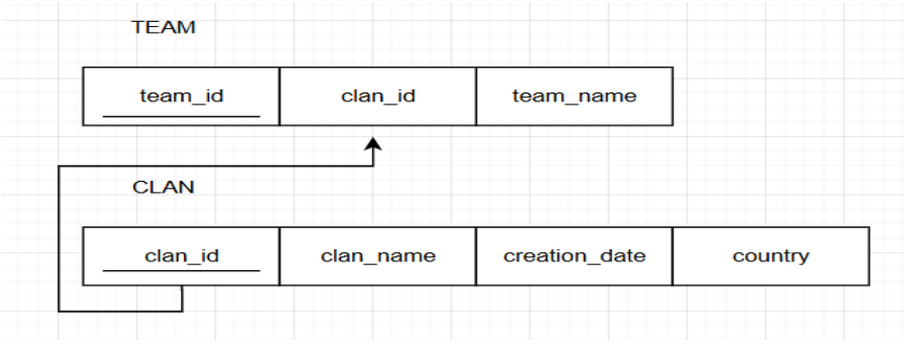


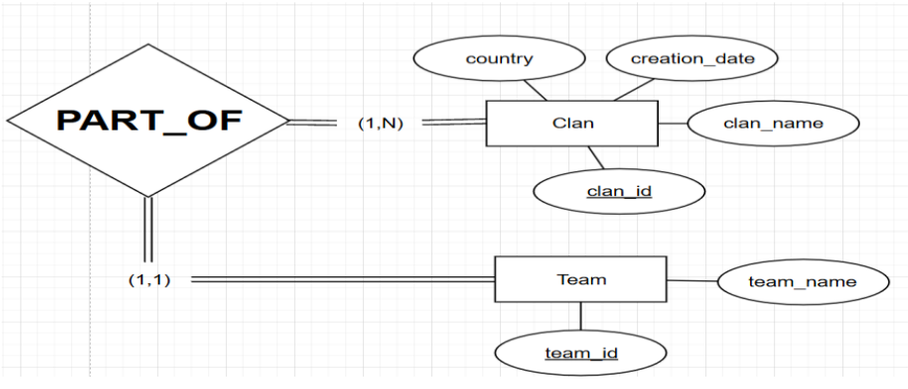
3.


MEMBER
member_id
name
role
nationality
date_of_bearth
email
nickname
team_id
TEAM
team_id
team_name

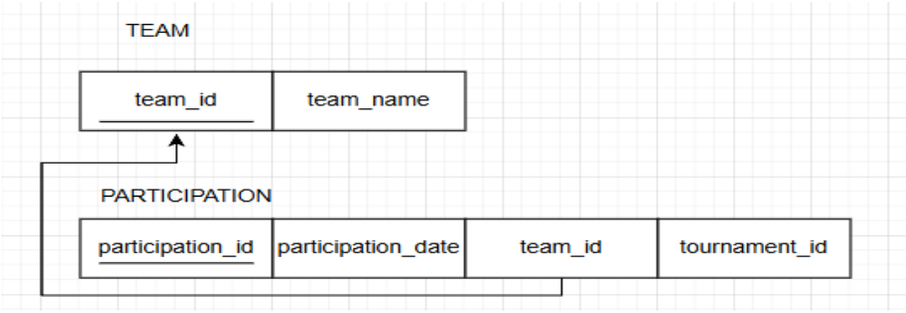

email
nickname
member_id
date_of_birth
Member
(1,1)
ASSGINED_TO
nationality
role
name
(2,N)
Team
team_name
team_id

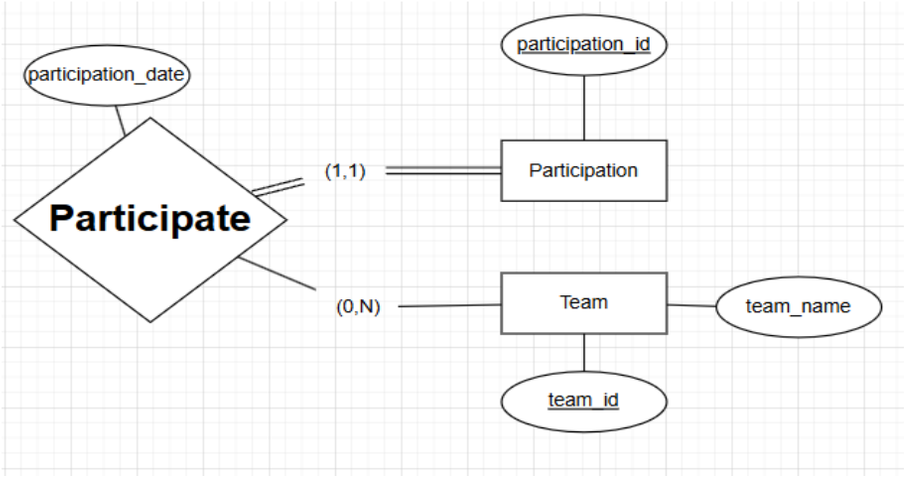
4.


TEAM
team_id
clan_id
team_name
CLAN
clan_id
clan_name
creation_date
country

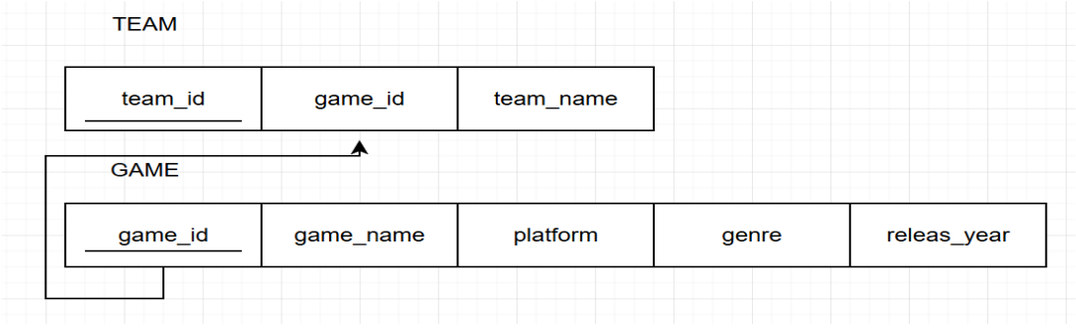

country
creation_date
PART_OF
(1,N)
Clan
clan_name
clan_id
(1,1)
Team
team_name
team_id

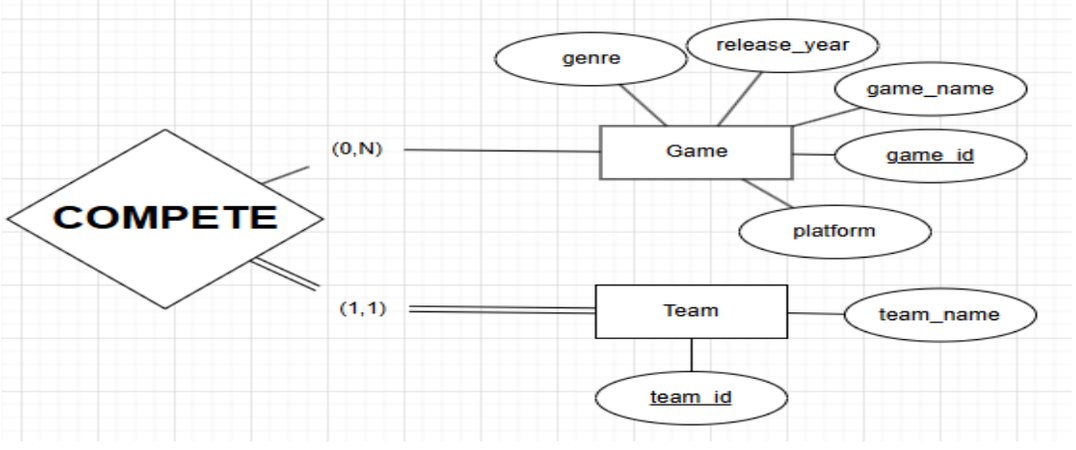
5.


TEAM
team_id
team_name
PARTICIPATION
participation_id
participation_date
team_id
tournament_id

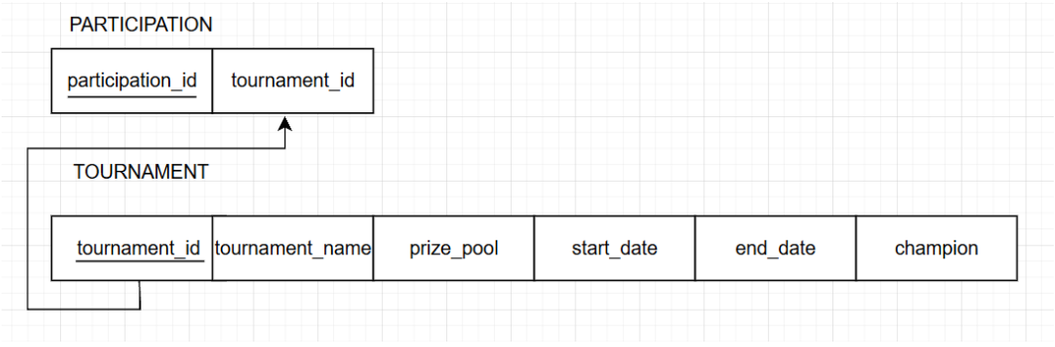

participation_id
participation_date
Participate
(1,1)
Participation
(0,N)
Team
team_name
team_id

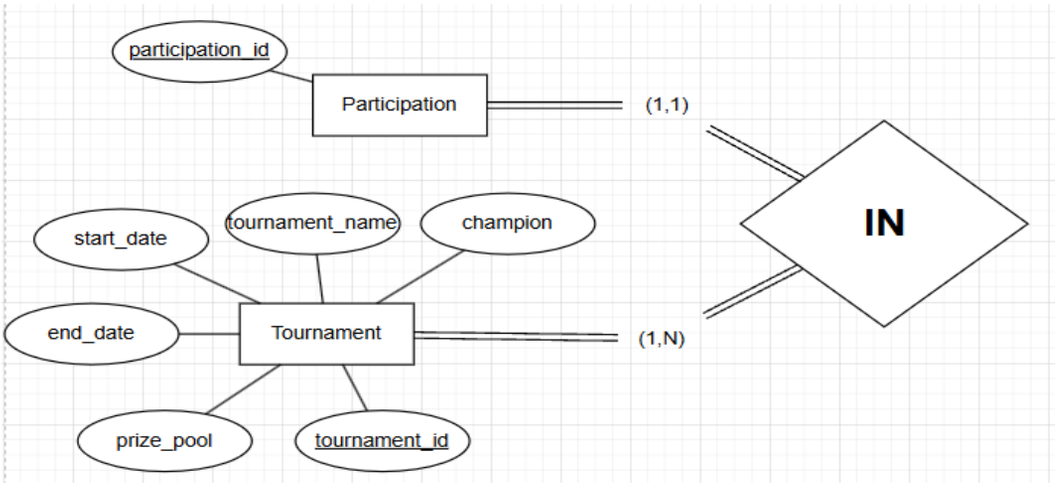
6.


TEAM
team_id
game_id
team_name
GAME
game_id
game_name
platform
genre
releas_year

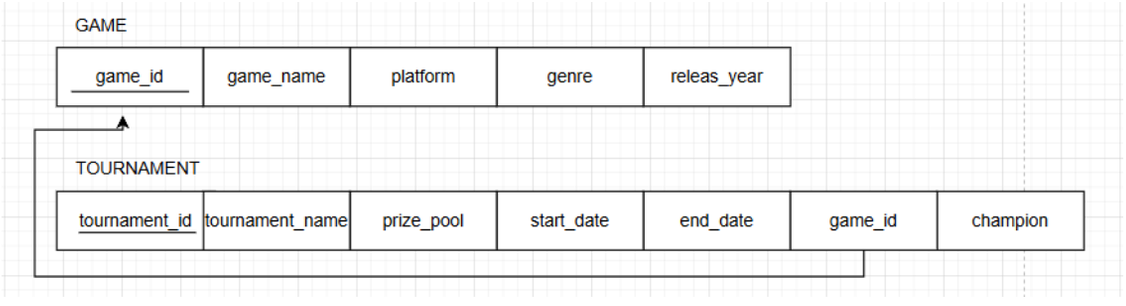

genre
release_year
game_name
COMPETE
(0,N)
Game
game_id
platform
(1,1)
Team
team_name
team_id

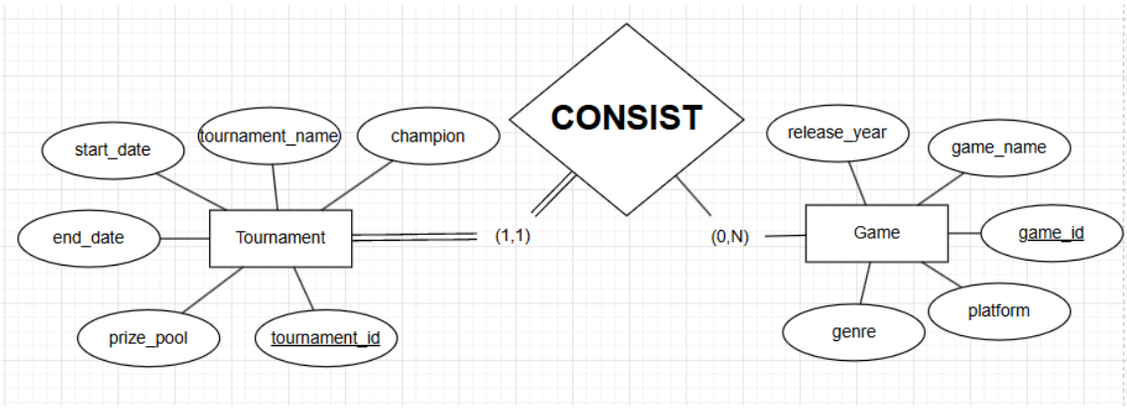
7.


PARTICIPATION
participation_id
tournament_id
TOURNAMENT
tournament_id
tournament_name
prize_pool
start_date
end_date
champion

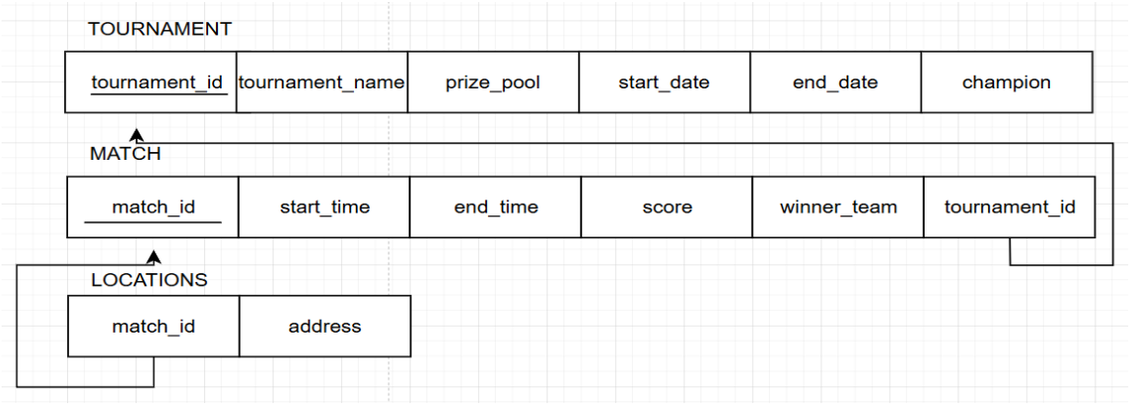

participation_id
Participation
(1,1)
IN
start_date
tournament_name
champion
end_date
Tournament
(1,N)
prize_pool
tournament_id

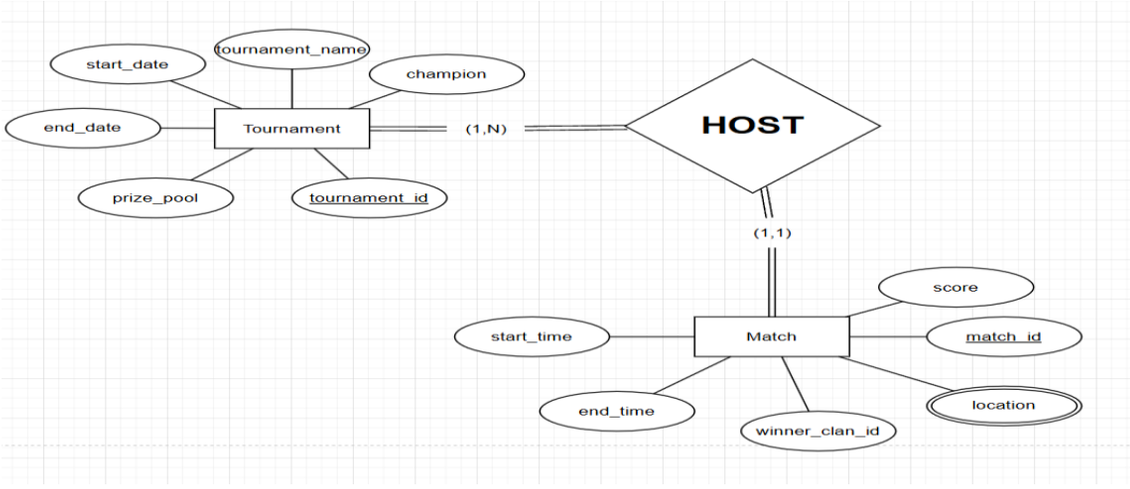
8.


GAME
game_id
game_name
platform
genre
releas_year
TOURNAMENT
tournament_id
tournament_name
prize_pool
start_date
end_date
game_id
champion


CONSIST
start_date
tournament_name
champion
end_date
Tournament
(1,1)
(0,N)
release_year
game_name
Game
game_id
platform
genre
prize_pool
tournament_id

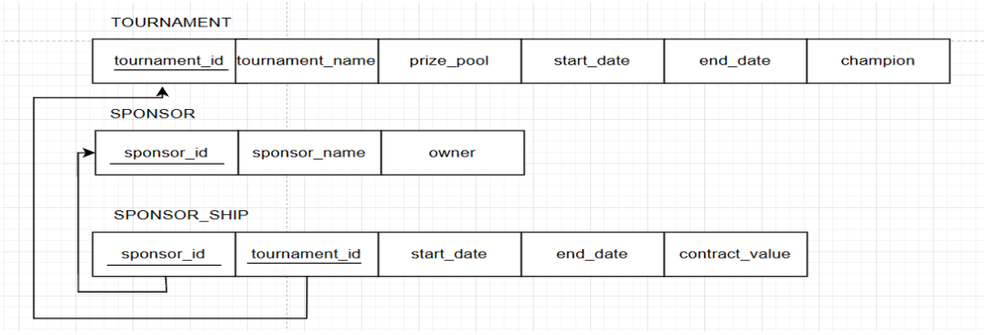
9.

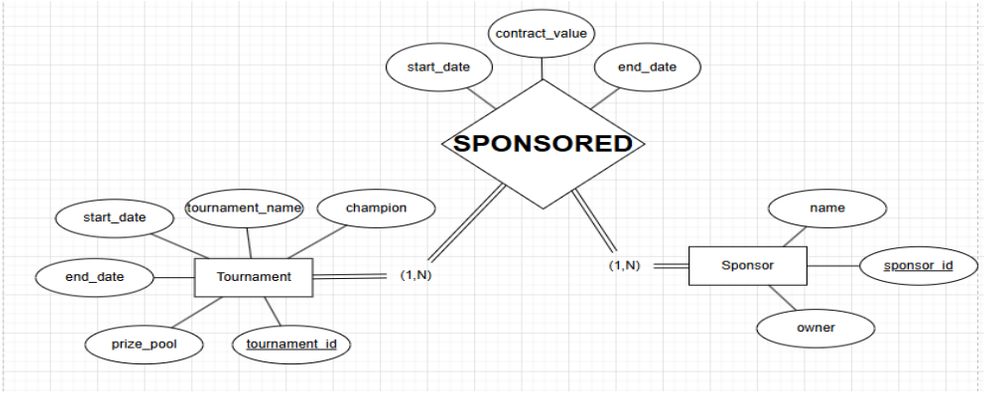

TOURNAMENT
tournament_id
tournament_name
prize_pool
start_date
end_date
champion
MATCH
match_id
start_time
end_time
score
winner_team
tournament_id
LOCATIONS
match_id
address


start_date
tournament_name
champion
end_date
Tournament
(1,N)
HOST
prize_pool
tournament_id
(1,1)
score
start_time
Match
match_id
end_time
location
winner_clan_id

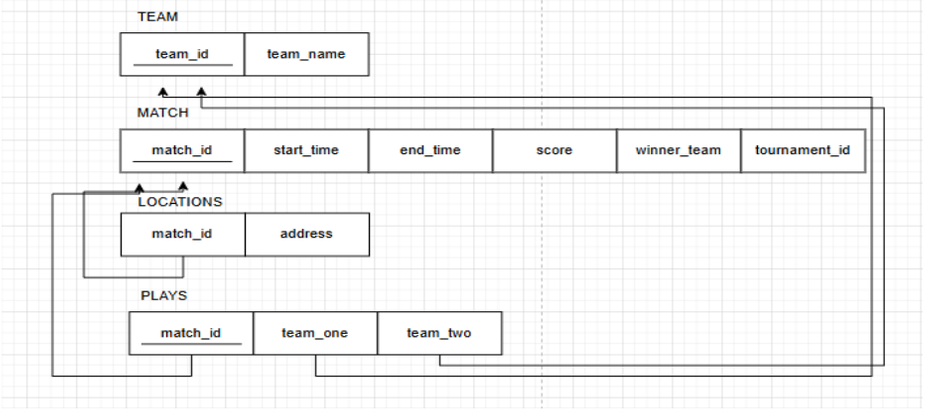
3.5 Mapping of binary M-N relationship types

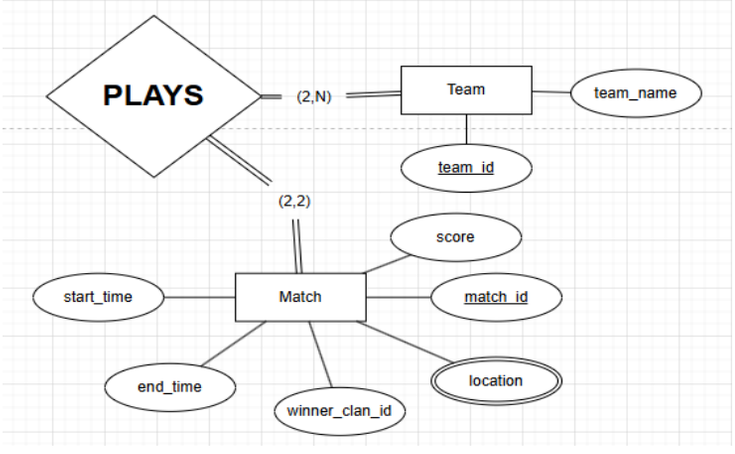
1.



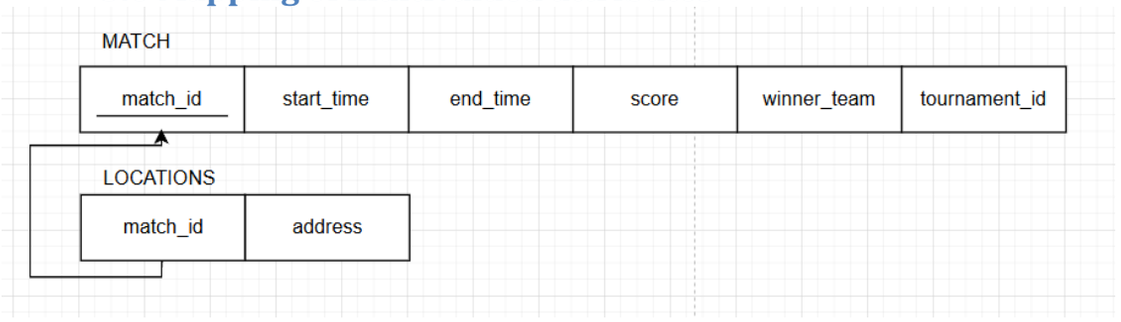


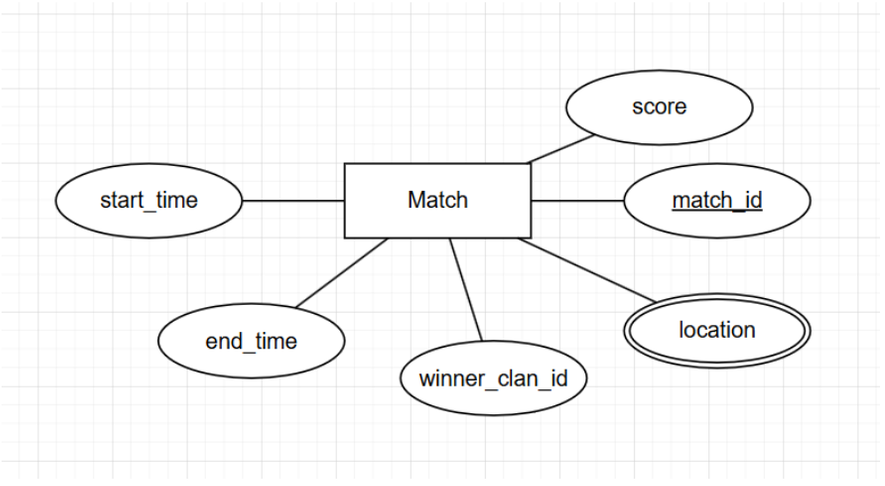
2.





3.6 Mapping of multivalued attributes

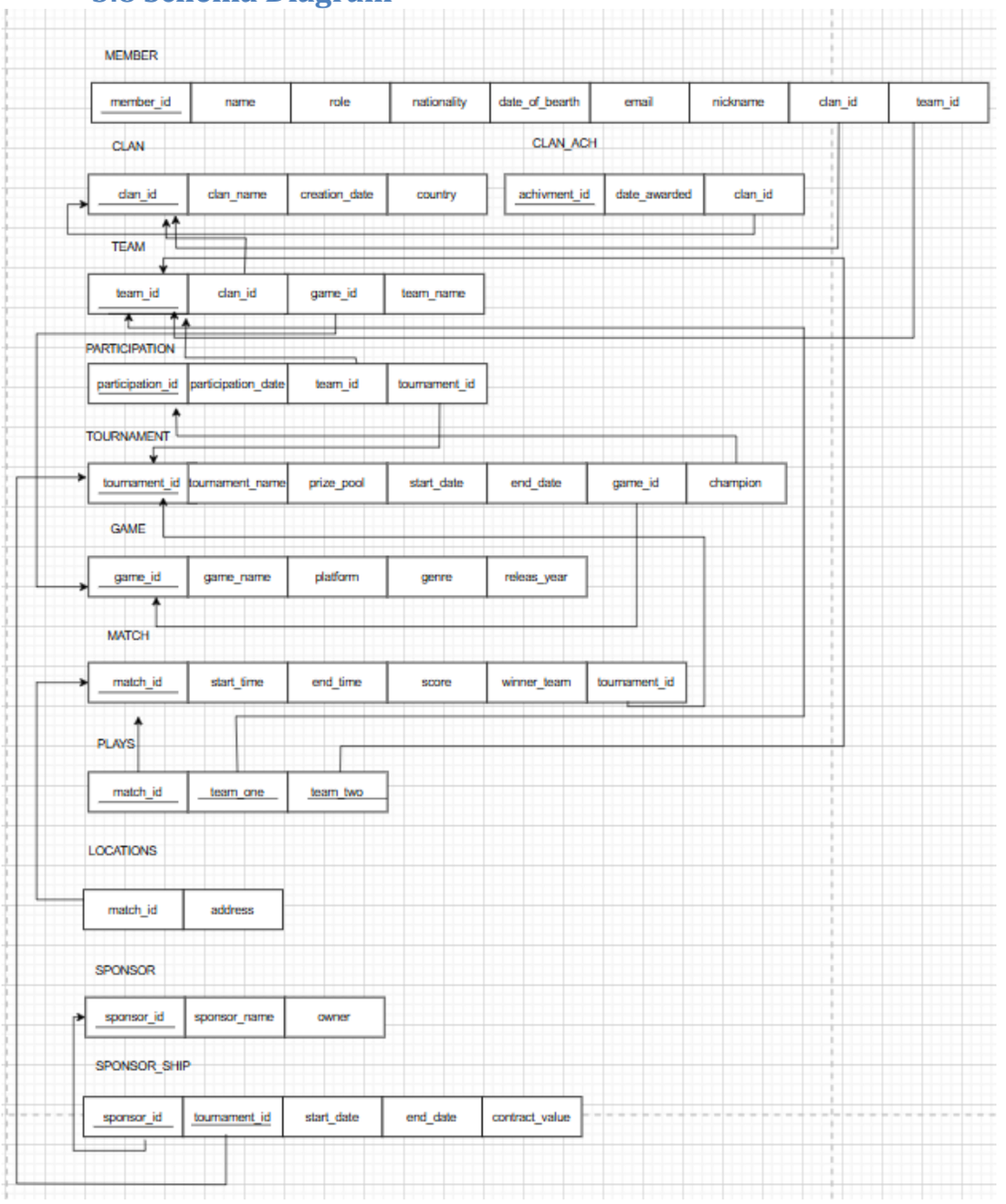




3.7 Mapping of n-ary relationship types

No n-ary relationship type.

3.8 Schema Diagram



4 Normalization

4.1 First Normal Form

- All tables have **atomic attributes**.
- Multivalued attributes like **Locations** were **correctly mapped** into a separate table (LOCATIONS) with a **foreign key** to MATCH.
- **Conclusion**: Your schema **satisfies 1NF**.

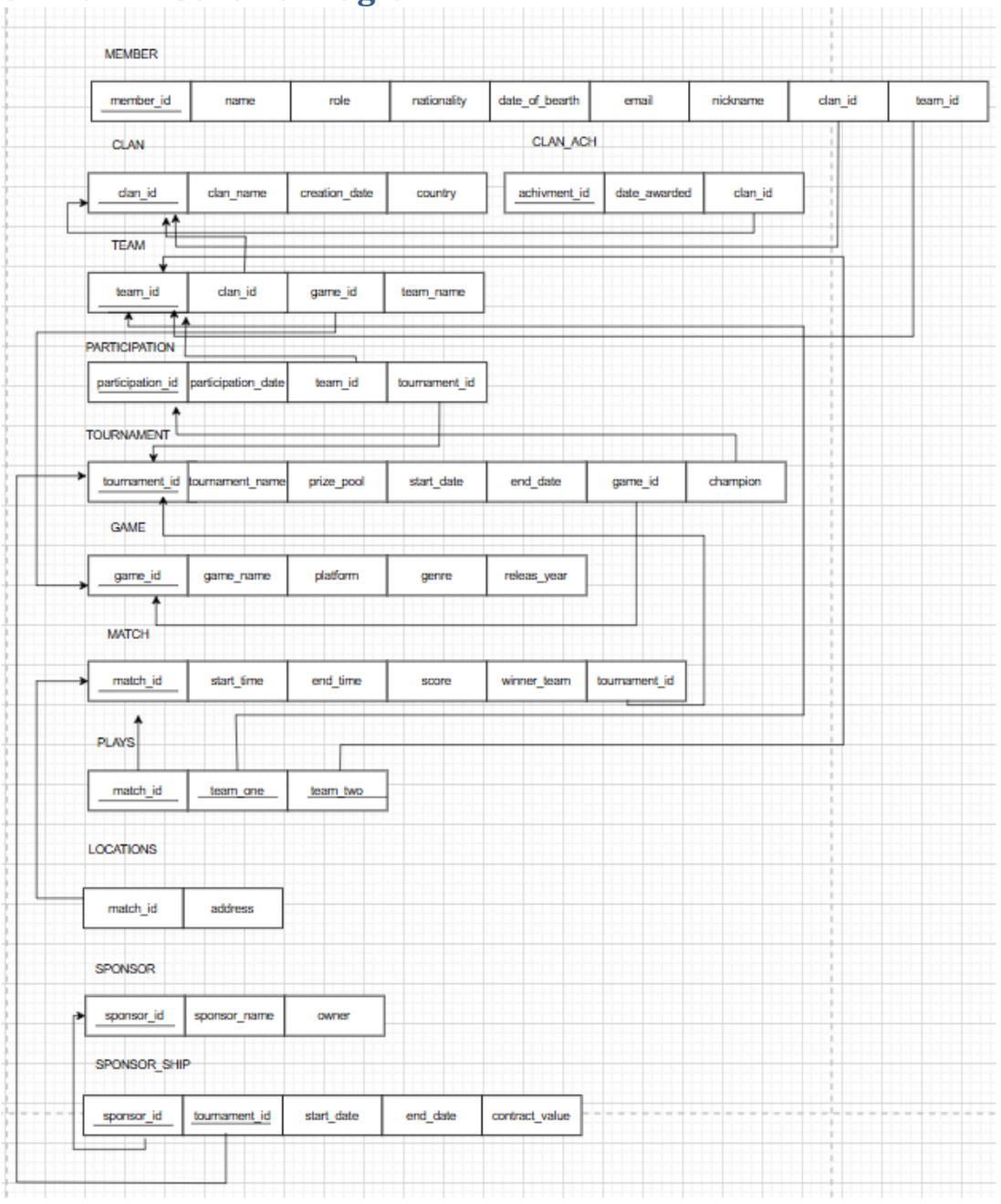
4.2 Second Normal Form

- 2NF checks if **non-prime attributes are fully dependent on the whole primary key**.
- In your tables:
- No table has a **composite primary key** (except Plays, Locations and Sponsor_Ship).
- In LOCATIONS, the primary key is (match_id, address), and address depends fully on both (ok because it's location per match).
- In **Sponsor_Ship**, the (sponsor_id, tournament_id) composite key is necessary for the sponsorship contract.
- No **partial dependencies** detected.

4.3 Third Normal Form

- 3NF requires **no transitive dependency** (non-prime attributes depending on another non-prime attribute).
  For instance:
- In MEMBER, attributes like email, name, nickname all depend **directly on member_id** (Primary Key), not through any other attribute.
- No **transitive dependencies** found.

5 Final DB Schema Diagram



PART III: IMPLEMENTATION

6 Table Creation Script

In this section, list your table creation scripts. The tables should all be in 3NF. If you have 10 tables then you should have 10 subsections. Please highlight (in yellow) all the constraints in the script including referential integrity and constraints on attributes. Please be organized!

6.1 <*Clan*> TABLE

```
CREATE TABLE CLAN (
   clan_id NUMBER PRIMARY KEY,
   clan_name VARCHAR2(50) UNIQUE NOT NULL,
   creation_date DATE,
   country VARCHAR2(25)
);
```

6.2 <*Game*> TABLE

```
CREATE TABLE GAME (
   game_id NUMBER PRIMARY KEY,
   game_name VARCHAR2(100) NOT NULL,
   platform VARCHAR2(50),
   genre VARCHAR2(50),
   release_year NUMBER
);
```

6.3 <*Team*> TABLE

```
CREATE TABLE TEAM (
   team_id NUMBER PRIMARY KEY,
   clan_id NUMBER NOT NULL,
   game_id NUMBER NOT NULL,
   team_name VARCHAR2(50) UNIQUE NOT NULL,
   CONSTRAINT fk_team_clan FOREIGN KEY (clan_id) REFERENCES Clan(clan_id),
   CONSTRAINT fk_team_game FOREIGN KEY (game_id) REFERENCES Game(game_id)
);
```

6.4 <*Mamber*> TABLE

```
CREATE TABLE MEMBER (
   member_id NUMBER PRIMARY KEY,
   name VARCHAR2(50),
   role VARCHAR2(30),
   nationality VARCHAR2(50),
   date_of_birth DATE,
   email VARCHAR2(100) UNIQUE NOT NULL,
   nickname VARCHAR2(20),,
   clan_id NUMBER NOT NULL,
   team_id NUMBER NOT NULL,
   CHECK (role IN ('Player', 'Coach', 'Staff')),
   CONSTRAINT fk_member_clan FOREIGN KEY (clan_id) REFERENCES Clan(clan_id),
   CONSTRAINT fk_member_team FOREIGN KEY (team_id) REFERENCES Team(team_id)
);
```

6.5 <*Clan_Ach*> TABLE

```
CREATE TABLE CLAN_ACH (
   achievement_id NUMBER PRIMARY KEY,
   achievement VARCHAR(100),
   date_awarded DATE,
   clan_id NUMBER,
   CONSTRAINT fk_clanach_clan FOREIGN KEY (clan_id) REFERENCES Clan(clan_id)
);
```

6.6 <*Tournament*> TABLE

```
CREATE TABLE TOURNAMENT (
   tournament_id NUMBER PRIMARY KEY,
   tournament_name VARCHAR2(100),
   prize_pool NUMBER,
   start_date DATE,
   end_date DATE,
   game_id NUMBER NOT NULL,
   champion_team_id NUMBER,
   CHECK (end_date > start_date),
   CONSTRAINT fk_tournament_game FOREIGN KEY (game_id) REFERENCES Game(game_id),
   CONSTRAINT fk_tournament_champion FOREIGN KEY (champion_team_id) REFERENCES
Team(team_id) );
);
```

6.7 <*Participation*> TABLE

```
CREATE TABLE PARTICIPATION (
   participation_id NUMBER PRIMARY KEY,
   participation_date DATE,
   team_id NUMBER NOT NULL,
   tournament_id NUMBER NOT NULL,
CONSTRAINT fk_participation_team FOREIGN KEY (team_id) REFERENCES Team(team_id),
 CONSTRAINT fk_participation_tournament FOREIGN KEY (tournament_id) REFERENCES
Tournament(tournament_id)
);
```

6.8 < *Match*> TABLE

```
CREATE TABLE MATCH (
   match_id NUMBER PRIMARY KEY,
   start_time TIMESTAMP,
   end_time TIMESTAMP,
   score VARCHAR2(20),
   winner_team _id NUMBER,
   tournament_id NUMBER NOT NULL,
   CONSTRAINT fk_match_tournament FOREIGN KEY (tournament_id) REFERENCES
Tournament(tournament_id),
   CONSTRAINT fk_match_clan FOREIGN KEY (winner_clan_id) REFERENCES Clan(clan_id)
);
```

6.9 <*Plays*> TABLE

```
CREATE TABLE PLAYS (
   match_id NUMBER NOT NULL,
   team_one NUMBER NOT NULL,
   team_two NUMBER NOT NULL,
   PRIMARY KEY (match_id, team_one_id, team_two_id),
   CONSTRAINT fk_plays_match FOREIGN KEY (match_id) REFERENCES Match(match_id),
   CONSTRAINT fk_plays_team_one FOREIGN KEY (team_one_id) REFERENCES Team(team_id),
   CONSTRAINT fk_plays_team_two FOREIGN KEY (team_two_id) REFERENCES Team(team_id),
   CHECK (team_one_id != team_two_id)
);
```

6.10 <*Locations*> TABLE

```
CREATE TABLE LOCATIONS (

   match_id NUMBER ,

   address VARCHAR2(100),

   CONSTRAINT pk_locations PRIMARY KEY (match_id, address),

   CONSTRAINT fk_locations_match FOREIGN KEY (match_id) REFERENCES Match(match_id)

);
```

6.11 <*Sponsor* > TABLE

```
CREATE TABLE SPONSOR (

   sponsor_id NUMBER PRIMARY KEY,

   sponsor_name VARCHAR2(100),

   owner VARCHAR2(100)

);
```

6.12 <*Sponsor_Ship*> TABLE

```
CREATE TABLE SPONSOR_SHIP (

   sponsor_id NUMBER NOT NULL,

   tournament_id NUMBER NOT NULL,

   start_date DATE,

   end_date DATE,

   contract_value NUMBER,

   PRIMARY KEY (tournament_id, sponsor_id),

   CONSTRAINT fk_sponsored_tournament FOREIGN KEY (tournament_id) REFERENCES
Tournament(tournament_id),

   CONSTRAINT fk_sponsored_sponsor FOREIGN KEY (sponsor_id) REFERENCES Sponsor(sponsor_id)

);
```

7 Constraints Script

All constraints in Design of Business Rules have been translated fully in detail in the implementation above.

8 Queries

8.1 *< List all teams with their clan names.>*

English Description:
Display the name of each team along with the clan's name.

SQL script:

```
SELECT T.team_name, C.clan_name
FROM Team T, CLAN C
WHERE T.clan_id = C.clan_id;
```

TEAM_NAME	CLAN_NAME
Capital Warriors	Riyadh Champions
Falcon Warriors	Desert Falcons
Desert Storm	Desert Falcons
Desert Titans	Desert Falcons
Nadj Legends	Nadj Knights
Knight Shooters	Nadj Knights
Najd Rivals	Nadj Knights
Nadj Phantoms	Nadj Knights
Eagle Strikers	Hejaz Eagles
Hejaz Titans	Hejaz Eagles
Hejaz Storm	Hejaz Eagles
Hejaz Hunters	Hejaz Eagles
Lions United	Peninsula Lions
Peninsula Kings	Peninsula Lions
Peninsula Titans	Peninsula Lions
Lion Guardians	Peninsula Lions
Riyadh Rangers	Riyadh Champions
Capital Crushers	Riyadh Champions

8.2 *< Get all tournaments with their game names.>*

English Description:
List all tournaments and their game names.

SQL script:

```
SELECT T.name AS tournament_name,
G.game_name
FROM Tournament T, Game G
WHERE T.game_id = G.game_id;
```

TOURNAMENT_NAME	GAME_NAME
Valorant Saudi Cup	Valorant
Peninsula Valorant Battle	Valorant
League of Legends Riyadh Masters	League of Legends
Fortnite Desert Challenge	Fortnite
Apex Legends Saudi Arena	Apex Legends
COD Warzone Battle Royale	Call of Duty: Warzone
Overwatch Saudi Clash	Overwatch
PUBG Gulf Championship	PUBG: Battlegrounds
Rocket League Riyadh Open	Rocket League
Rainbow Six Siege KSA Cup	Rainbow Six Siege

8.3 *< TOP 3 sponsorship contracts.>*

English Description:
Show the top 3 sponsors names with their total contracts values).
SQL script:

```
SELECT S.name AS sponsor_name, SUM(SP.contract_value) AS total_sponsored
FROM Sponsor S, Sponsored SP
WHERE S.sponsor_id = SP.sponsor_id
GROUP BY S.name
ORDER BY SUM(SP.contract_value) DESC
FETCH FIRST 3 ROWS ONLY
```

SPONSOR_NAME	TOTAL_SPONSORED
MBC Play	47000
STC Gaming	41000
Neom Esports	27000

8.4 *< Show all teams participated on '2025-06-01'. >*

English Description:
Display all teams and who participatd in the date '2025-Jun-01' alongside with the tournament name
SQL script:

```
SELECT T.team_name, TM.name AS tournament_name, P.participation_date
FROM Participation P, Team T, Tournament TM
WHERE P.team_id = T.team_id
and P.tournament_id = TM.tournament_id
and P.participation_date = TO_DATE('2025-06-01', 'YYYY-MM-DD');
```

TEAM_NAME	TOURNAMENT_NAME	PARTICIPATION_DATE
Falcon Warriors	Valorant Saudi Cup	01-JUN-25
Peninsula Kings	Valorant Saudi Cup	01-JUN-25
Desert Titans	Valorant Saudi Cup	01-JUN-25

8.5 *< Show TOP 5 Average prize pool per tournament.>*

English Description:
Display the top 5 game names with their average prize pool
SQL script:

```
SELECT G.game_name, AVG(T.prize_pool) AS avg_prize_pool
FROM Game G, Tournament T
WHERE G.game_id = T.game_id
GROUP BY G.game_name
ORDER BY avg_prize_pool DESC
FETCH FIRST 5 ROW ONLY
```

GAME_NAME	AVG_PRIZE_POOL
League of Legends	60000
PUBG: Battlegrounds	55000
Apex Legends	45000
Valorant	41000
Rainbow Six Siege	40000

8.6 *< List all tournaments with the winning team name and its clan name.>*

English Description:

Display a detailed view of each tournament and the winning team alongside with its clan name

SQL script:

```
SELECT TM.name AS tournament_name, T.team_name AS winner_team, C.clan_name
FROM Tournament TM, Team T, Clan C
WHERE TM.champion_team_id = T.team_id
and T.clan_id = C.clan_id;
```

TOURNAMENT_NAME	WINNER_TEAM	CLAN_NAME
Valorant Saudi Cup	Falcon Warriors	Desert Falcons
Fortnite Desert Challenge	Desert Storm	Desert Falcons
League of Legends Riyadh Masters	Nadj Legends	Nadj Knights
COD Warzone Battle Royale	Knight Shooters	Nadj Knights
PUBG Gulf Championship	Najd Rivals	Nadj Knights
Overwatch Saudi Clash	Eagle Strikers	Hejaz Eagles
Apex Legends Saudi Arena	Lions United	Peninsula Lions
Peninsula Valorant Battle	Peninsula Kings	Peninsula Lions
Rainbow Six Siege KSA Cup	Riyadh Rangers	Riyadh Champions
Rocket League Riyadh Open	Capital Crushers	Riyadh Champions

8.7 *< Show total number of matches played by each team.>*

English Description:

Display the total number of matches played by each team.

SQL script:

```
SELECT T.team_name, COUNT(P.match_id) AS matches_played
FROM Team T, Plays P
WHERE T.team_id = P.team_one_id OR T.team_id = P.team_two_id
GROUP BY T.team_name
FETCH FIRST 6 ROW ONLY;
```

TEAM_NAME	MATCHES_PLAYED
Desert Storm	2
Najd Rivals	2
Nadj Phantoms	1
Eagle Strikers	1
Peninsula Titans	1
Peninsula Kings	2