

# REPORT

## SUMMER INTERNSHIP - 2022



**MSC-IT SUMMER INTERNSHIP PC649**

**CRICKET MANAGEMENT SYSTEM (DATABASE MANAGEMENT SYSTEM)**

**UNDER SUPERVISION OF PROF. P M JAT**

Yash Solanki | 202112028  
Jay Savani | 202112078  
Jenish Dhanani | 202112106

## Scope of Project

CMS (Cricket Management System) Database based on various information about the various teams participating in the World Cup, in which all the major Teams participates. It also provides information about the various players, participations of tournament. Other Information like Umpire's records, Coach records, Match's Records are Included in CMS Database.

## Project Description

The project includes a complete database of World Cup game involving cricket, with coaches and umpires managing the game with a brief demonstration on data management program including all the entities mentioned below.

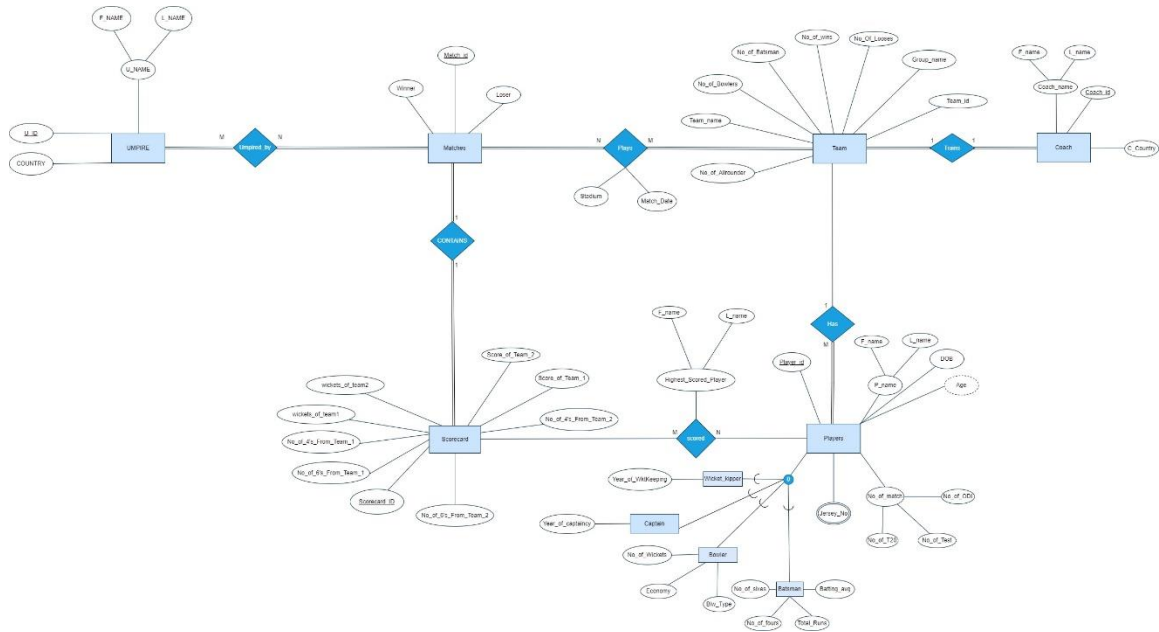
The Schema will provide following entities with complete data information of the according cricket statistics. The information is available team wise and player wise.

- Team
- Players
- Scorecard
- Played
- Umpire
- Umpired by
- Scored
- Coach
- Matches

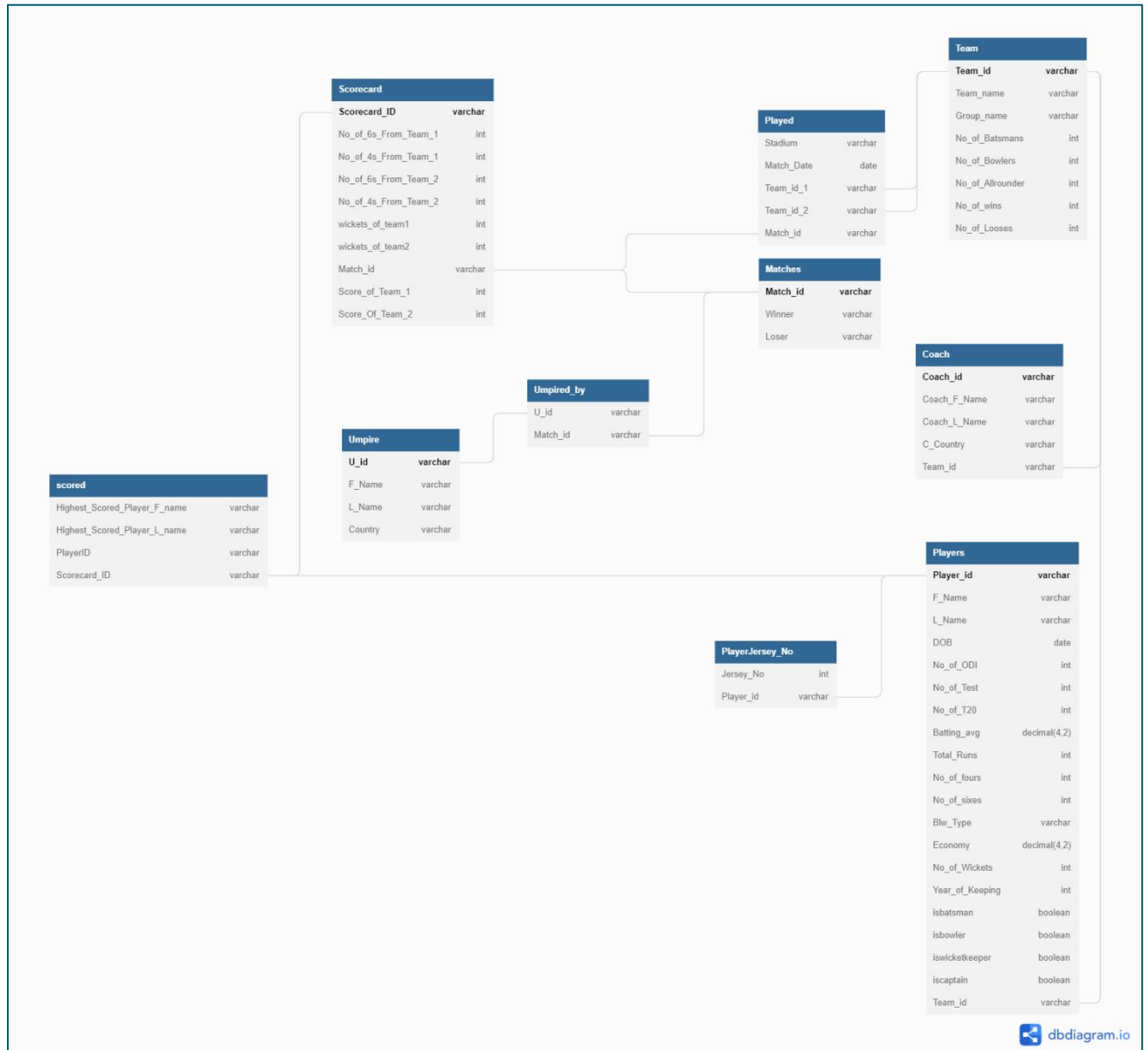
By entering the data of each unit, we can get other details like team, player, coach, umpire which are related to the organizations, in addition, we get more information about the total number of teams in the match, the total wickets, and the runs scored, the order of the teams after the match and the CMS. And the winning team is also announced.

The advantage of this database project is very useful for cricket match broadcasters to get information quickly. In addition, for cricket lovers who are very interested in cricket statistics and also for cricketers and children to understand easily. In other words, we manage the data of the Cricket World Cup, which is played at a world level.

# Entity Relationship Diagram



# Relational Schema



## Data Definition Language

```
create schema cms;
```

```
set search_path to cms;
```

```
-- Team Table
```

```
create table Team (  
    Team_id varchar(30) PRIMARY KEY,  
    Team_name varchar(30) NOT NULL,  
    Group_name varchar(1) NOT NULL,  
    No_of_Batsmans int,  
    No_of_Bowlers int,  
    No_of_Allrounder int,  
    No_of_wins int,  
    No_of_Looses int  
);
```

```
-- Coach Table
```

```
create table Coach(  
    Coach_id varchar(30) PRIMARY KEY,  
    Coach_F_Name varchar(20),  
    Coach_L_Name varchar(20),  
    C_Country varchar(30),  
    Team_id varchar(30) NOT NULL REFERENCES Team(Team_id)  
    ON DELETE SET DEFAULT ON UPDATE CASCADE  
);
```

-- Players Table

```
create table Players(  
    Player_id varchar(30) PRIMARY KEY,  
    F_Name varchar(20),  
    L_Name varchar(20),  
    DOB date,  
    No_of_ODI int,  
    No_of_Test int,  
    No_of_T20 int,  
    Batting_avg decimal(4,2),  
    Total_Runs int,  
    No_of_fours int,  
    No_of_sixes int,  
    Blw_Type varchar(30),  
    Economy decimal(4,2),  
    No_of_Wickets int,  
    Year_of_Keeping int,  
    Year_of_captaincy int,  
    isbatsman boolean,  
    isbowler boolean,  
    iswicketkeeper boolean,  
    iscaptain boolean,  
    Team_id varchar(30) NOT NULL REFERENCES Team(Team_id)  
    ON DELETE SET DEFAULT ON UPDATE CASCADE  
);
```

-- PlayersJersey\_No Table

```
create table PlayersJersey_No(  
    Jersey_No int,  
    Player_id varchar(30) NOT NULL REFERENCES Players(Player_id)  
    ON DELETE SET DEFAULT ON UPDATE CASCADE  
);
```

-- Umpire Table

```
create table Umpire (  
    U_ID varchar(30) PRIMARY KEY,  
    F_NAME varchar(20),  
    L_NAME varchar(20),  
    COUNTRY varchar(30)  
);
```

-- Matches Table

```
create table Matches (  
    Match_id varchar(20) PRIMARY KEY,  
    Winner varchar(30),  
    loser varchar(30)  
);
```

-- Scorecard Table

```
create table Scorecard (  
    Scorecard_ID varchar(30) PRIMARY KEY,
```

```

        No_of_6s_From_Team_1 int,
        No_of_4s_From_Team_1 int,
        No_of_6s_From_Team_2 int,
        No_of_4s_From_Team_2 int,
        wickets_of_team1 int,
        wickets_of_team2 int,
        Score_of_Team_1 int,
        Score_of_Team_2 int,
        Match_id varchar(30) NOT NULL REFERENCES Matches(Match_id)
        ON DELETE SET DEFAULT ON UPDATE CASCADE
    );

```

-- Played Table

```

create table Played (
    Stadium varchar(40),
    Match_Date date,
    Team_id_1 varchar(30) NOT NULL REFERENCES team(Team_id)
    ON DELETE SET DEFAULT ON UPDATE CASCADE,
    Team_id_2 varchar(30) NOT NULL REFERENCES team(Team_id)
    ON DELETE SET DEFAULT ON UPDATE CASCADE,
    Match_id varchar(20) NOT NULL REFERENCES Matches(Match_id)
    ON DELETE SET DEFAULT ON UPDATE CASCADE
);

```

-- Umpired\_by Table

```

create table Umpired_by(

```



```

    U_ID varchar(30) NOT NULL REFERENCES Umpire(U_ID)
    ON DELETE SET DEFAULT ON UPDATE CASCADE,
    Match_id varchar(20) NOT NULL REFERENCES Matches(Match_id)
    ON DELETE SET DEFAULT ON UPDATE CASCADE
);

-- Scored Table
create table scored(
    Highest_Scored_Player_F_name varchar(20),
    Highest_Scored_Player_L_name varchar(20),
    PlayerID varchar(30) NOT NULL REFERENCES Players(Player_id)
    ON DELETE SET DEFAULT ON UPDATE CASCADE,
    Scorecard_ID varchar(30) NOT NULL REFERENCES Scorecard(Scorecard_ID)
    ON DELETE SET DEFAULT ON UPDATE CASCADE
);

```

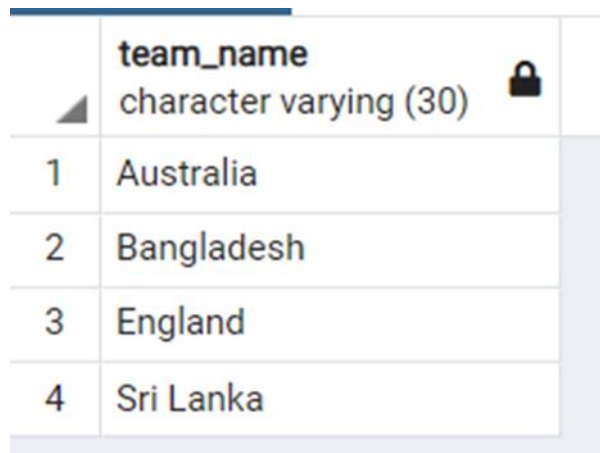
## Queries

### Query: 1 (a)

Display the country whose players have batting average greater than 90.

```
SELECT DISTINCT TEAM_NAME  
FROM TEAM  
NATURAL JOIN PLAYERS  
WHERE BATTING_AVG > 90;
```

Screenshot:



The screenshot shows a database query result in a table format. The table has two columns: an index column and a column named 'team\_name'. The 'team\_name' column is described as 'character varying (30)' and has a lock icon. The table contains four rows of data, indexed 1 through 4.

	team_name character varying (30) 🔒
1	Australia
2	Bangladesh
3	England
4	Sri Lanka

### Query: 1 (b)

Display the country whose players have batting average less than 30.

```
SELECT DISTINCT TEAM_NAME  
FROM TEAM  
NATURAL JOIN PLAYERS  
WHERE BATTING_AVG < 30;
```

Screenshot:

	team_name character varying (30) 🔒
1	Bangladesh
2	Scotland
3	Pakistan
4	Afghanistan
5	India
6	South Africa
7	West Indies

Query: 2

Display name of coach and name of player who has coached a player with total runs greater than 9500.

```
SELECT DISTINCT
    COACH_F_NAME,
    COACH_L_NAME,
    F_NAME AS PLAYER_F_NAME,
    L_NAME AS PLAYER_L_NAME
FROM COACH
INNER JOIN PLAYERS ON COACH.TEAM_ID = PLAYERS.TEAM_ID WHERE
(TOTAL_RUNS > 9500);
```

### Screenshot:

Data Output		Messages		
	coach_f_name character varying (20) 🔒	coach_l_name character varying (20) 🔒	player_f_name character varying (20) 🔒	player_l_name character varying (20) 🔒
1	Russell	Domingo	Tamim	Iqbal
2	Misbah ul	Haq	Asif	Ali
3	Russell	Domingo	DAS	Liton
4	Mickey	Arthur	kusal	perera
5	Lance	Klusener	Najibullah	Zadran
6	Shane	Burger	Kyle	Coetzer
7	Lance	Klusener	Mohammad	Nabi
8	Ravi	Shastri	Virat	Kohli
9	Justin	Langer	Matthew	Wade
10	Chris	Silverwood	jos	Buttler
11	Chris	Silverwood	Ben	Stokes
12	Phil	Simmons	Fafdes	Plessis
13	Lance	Klusener	Rashid	Khan
14	Phil	Simmons	Aides	Markram
15	Mickey	Arthur	angelo	mathews
16	Ravi	Shastri	Rohit	Sharma
17	Russell	Domingo	Shakib Al	Hasan
18	Misbah ul	Haq	Babar	Azam
19	Mark	Boucher	Quinton-de	Kock
20	Mickey	Arthur	dIMUTH	karanarathe

### Query: 3

Number of players used by each team.

```
SELECT COUNT(*) AS TOTAL_TEAM_PLAYER, TEAM_NAME
FROM PLAYERS
    INNER JOIN TEAM
    ON PLAYERS.TEAM_ID=TEAM.TEAM_ID
GROUP BY TEAM_NAME;
```

Screenshot:

	total_player bigint	team_name character varying (30)
1	11	Bangladesh
2	11	Sri Lanka
3	11	England
4	11	Afghanistan
5	11	India
6	11	South Africa
7	11	Scotland
8	11	Pakistan
9	11	Australia
10	11	West Indies

Query: 4

Display All team with number of won match count.

```
SELECT TEAM_NAME,  
       COUNT(MATCHES) AS WINS  
FROM TEAM  
LEFT JOIN MATCHES ON (WINNER = TEAM_NAME)  
GROUP BY TEAM_NAME  
ORDER BY WINS DESC;
```

Screenshot:

team_name	wins
character varying (30)	bigint
Sri Lanka	2
South Africa	1
Bangladesh	1
India	1
West Indies	1
Scotland	1
Australia	1
Pakistan	0
England	0
Afghanistan	0

Query: 5

select team name, player id, player name whose any of the playerscored more than 25000+ runs and 500+ sixes.

```
SELECT TEAM_NAME,  
       PLAYER_ID,  
       F_NAME,  
       L_NAME  
FROM TEAM  
JOIN PLAYERS ON TEAM.TEAM_ID = PLAYERS.TEAM_ID  
WHERE PLAYERS.TOTAL_RUNS > 25000  
       AND PLAYERS.NO_OF_SIXES > 500
```

Screenshot:

Data Output		messages		
	<b>team_name</b> character varying (30) 🔒	<b>player_id</b> character varying (30) 🔒	<b>f_name</b> character varying (20) 🔒	<b>L_name</b> character varying (20) 🔒
1	Afghanistan	AFG001	Rashid	Khan
2	Bangladesh	BAN005	DAS	Liton
3	Sri Lanka	SL005	kusal	perera

## GitHub Link

<https://github.com/Jenish-Dhanani/Database-Project-About-Cricket-Management>