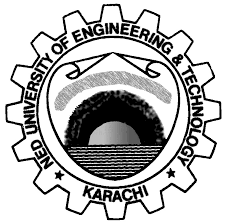
**PSL Management System**

**D.B.M.S Project**

****

**Project By:**

* Bilal Zubairi (CS-21091)
* Muhammad Faizan Khan (CS-21070)
* Salman (CS-21099)

Course Name: Database Management System (D.B.M.S)

Course Code: CS-222

Batch: 2021

Date of Submission:

Department: Computer and Information Systems Engineering

**C E R T I F I C A T E**

Certified that following students have successfully completed the DBMS project titled“PSL Management System” assigned as a mandatory requirement for qualifying the practical examination of the course.

Bilal Zubairi (CS-21091) \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Muhammad Faizan Khan (CS-21070) \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Salman (CS-21099) \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Name & Signature

**Project Examiner**

**Abstract**

The project will require developing a database to store data about teams, players, match timings and names of different grounds where two different teams will play against each other. The categorization of different entities like pitch report, commentary, umpiring will help in finely managing the different operations of cricket ground. The next essential feature is to construct an application for managing different operations including scheduling of matches, standing of teams, stats of different players and seating arrangement of crowd. Umpiring, scoreboard of batting, bowling and drafting of players. Moreover, it will also provide quick search facility to retrieve player information, team information, and match timings.

**Table of Contents**

|  |  |  |
| --- | --- | --- |
| **S. No** | **Topic** | **Page No.** |
| 1 | Introduction | 1 |
| 2 | Requirement Formulation and Analysis | 1 |
| 3 | Entity Relationship Model | 3 |
| 4 | Tables | 4 |
| 5 | Normalization | 4 |
| 6 | D.D.L Statements | 6 |
| 7 | Implementation of front-end and back-end | 9 |
| 8 | Security of Database | 11 |
| 9 | Individual Contributions | 12 |

# Introduction:

This project is basically a website which is developed to show complete statistics of Pakistan Super League (P.S.L). We went through many websites searching for stats of P.S.L season 6 but there were none which could gave the complete stats even the official website of Pakistan Cricket Board (P.C.B) was lacking the information. Therefore, we gathered data from different websites and also referred to the highlights that were uploaded on YouTube and at the end we collectively present the complete statistics within our website.

# REQUIREMENT FORMULATION AND ANALYSIS (DATABASE DESIGNING):

This application is related to the database of different teams,players information of HBL PSL 6th edition.This database will give be applicable in searching player information and inserting their stats and category.In the database each player will be having player id and player name and the data of different teams played in this mega event for example each player will be playing for a particular team and the team will be having a team id and team name and the database will also contain the information about the match schedule because every team will play a match with their opponents so each match have unique attributes like match id and date and the match toss will be decided by umpires so every umpires will have an umpire id and umpire name. This will facilitate in searching each data of every player in read only mode for the user. Only the database administrator will have the access to edit the information about players,match schedules or managing staff of the ground we can further also add more previous HBL PSL edition database about cricketers and match schedules in future for the advancement of the database and applications.

Moreover, each player information will be obtained by just searching player name all their stats will be displayed. Database also maintains the category of each of the team players and retrieving umpires’ information of the matches and team standings. Every match will be played in the stadium so the database will also maintain the information regarding in which stadium the match is played. Every stadium will have a stadium Id and stadium name. In the database each stadium information will also contain the pitch report of the stadium and also stores the location of the stadium and its capacity. scheduling of matches, standing of teams, stats of different players and seating arrangement of crowd. Umpiring, scoreboard of batting and bowling and drafting of players. Moreover, it will also provide quick search facility to retrieve player information, team information, and match timings.

# Entity Relationship Model:

Stadium

Is played in

Umpire

Match

Has a

Team

PLAYER

Decided by

Plays for

# Tables:

1)Team(ID,Name,Location,Owner,coach,captain,performance)

2)Players(ID,Name,DOB,Role,Style,Height,team\_id,Nationality,Performance,category)

3)stadium(st\_ID,st\_Name,st\_pitch\_report,st\_location,st\_capacity)

4)Match(ID,Matchtimings,toss,commentator,Match\_date,umpire\_id,score,stats,st\_id,match\_between,match\_number)

5)Umpire(umpire\_id,umpire\_name,nationality,experience)

# Normalization:

**1NF**

Player­\_ID,Team\_ID,Match\_ID,Stadium­\_ID,Umpire\_ID🡪player\_Name,height,Role,Style,Nationality\_P,category,D.O.B\_P,Age\_P,Performance\_P,team\_Name,owner,coach,team\_loc,captain,performance-t,Match\_date,Match\_Time,teamId,Team-2,commentator,toss-winner,choose\_to target,stats,stadium\_name,stadium\_location,Pitch\_Report,stadium\_capacity,Umpire\_Name,D.O.B-u,Age-u,years of experience,Nationality-u

Player\_ID🡪player\_name, height, role, style, Nationality-P, category, D.O.B, performance

Team\_ID🡪team\_name,owner,coach,team-loc,captain,performance

Match\_ID🡪Match\_Date,Match\_Time,team 1,team 2,commentator,toss-winner

UmpireID🡪Umpire\_Name,D.O.B-U,Age-U,years of experience

**2NF**

FD1:Player­\_ID,Team\_ID,Match\_ID,Stadium­\_ID,Umpire\_ID🡪player\_Name,height,Role,Style,Nationality\_P,category,D.O.B\_P,Age\_P,Performance\_P,team\_Name,owner,coach,team\_loc,captain,performance-t,Match\_date,Match\_Time,teamId,Team-2,commentator,toss-winner,choose\_to target,stats,stadium\_name,stadium\_location,Pitch\_Report,stadium\_capacity,Umpire\_Name,D.O.B-u,Age-u,years of experience,Nationality-u

FD2:Player\_ID🡪player\_name,heightrole,style,Nationality-P,category,D.O.B-p,Age-p,performance-p

FD3:Team\_ID🡪team\_name,owner,coach,team-loc,captain,performance-t

FD4:Match\_ID🡪Match\_Date,Match\_Time,team 1,team 2,commentator,toss-winner

FD5:Umpire\_ID🡪Umpire\_Name,D.O.B-U,Age-U,years of experience

**3NF**

Player\_ID🡪Performance\_P,Age\_P,D.O.B\_P,category,nationality\_P,style,role,height,player\_Name

Team\_ID🡪performance\_t,captain,team\_loc,coach,owner,team\_Name

Match\_ID🡪toss\_winner,commentator,Team\_2,team\_1,Match\_Time,Match\_Date

Umpire\_ID🡪nationality\_u,yearsofexperience,Age\_u,D.O.B\_u,Umpire\_Name

# D.D.L Statements:

CREATE TABLE psl.stadium(

Stadium\_id INT NOT NULL UNSIGNED AUTO\_INCREMENT,

Stadium\_Name VARCHAR(100) NOT NULL,

Location VARCHAR(100) NOT NULL,

Pitch\_condition VARCHAR(100) NOT NULL,

Capacity INT UNSIGNED NOT NULL,

PRIMARY KEY(stadium\_id));

CREATE TABLE psl.team (

Team\_id INT NOT NULL UNSIGNED AUTO\_INCREMENT,

Team\_Name VARCHAR(150) NOT NULL,

Location VARCHAR(100) NOT NULL,

Owner VARCHAR(100) NOT NULL,

Coach VARCHAR(100) NOT NULL,

Captain VARCHAR(100) NOT NULL,

PRIMARY KEY(team\_id));

CREATE TABLE psl.umpire(

Umpire\_ID INT NOT NULL UNSIGNED AUTO\_INCREMENT,

D.O.B DATE NOT NULL,

Umpire\_name VARCHAR(150) NOT NULL,

Nationality VARCHAR(150) NOT NULL,

Years\_of\_experience INT NOT NULL,

PRIMARY KEY(Umpire\_id));

CREATE TABLES psl.players(

Player\_id INT NOT NULL UNSIGNED AUTO\_INCREMENT,

Player\_name VARCHAR(250) NOT NULL,

D.O.B DATE NOT NULL,

Height INT NOT NULL UNSIGNED,

Role VARCHAR(50) NOT NULL,

Style VARCHAR(50) NOT NULL,

Nationality VARCHAR(80) NOT NULL,

Performance VARCHAR(80) NOT NULL,

Team\_id INT NOT NULL,

PRIMARY KEY(player\_id));

CREATE TABLE psl.match(

Match\_id INT NOT NULL UNSIGNED AUTO\_INCREMENT,

Match\_date DATE NOT NULL,

Match\_time TIME NOT NULL,

Team\_1\_id INT NOT NULL,

Team\_2\_id INT NOT NULL,

Toss\_winner INT NOT NULL,

Choose\_to VARCHAR(5) NOT NULL,

Score INT NOT NULL,

Stats VARCHAR(255) NOT NULL,

Match\_Winner INT NOT NULL,

Commentator VARCHAR(100) NOT NULL,

Umpire\_No INT NOT NULL,

Umpire\_id INT NOT NULL,

Stadium\_id INT NOT NULL,

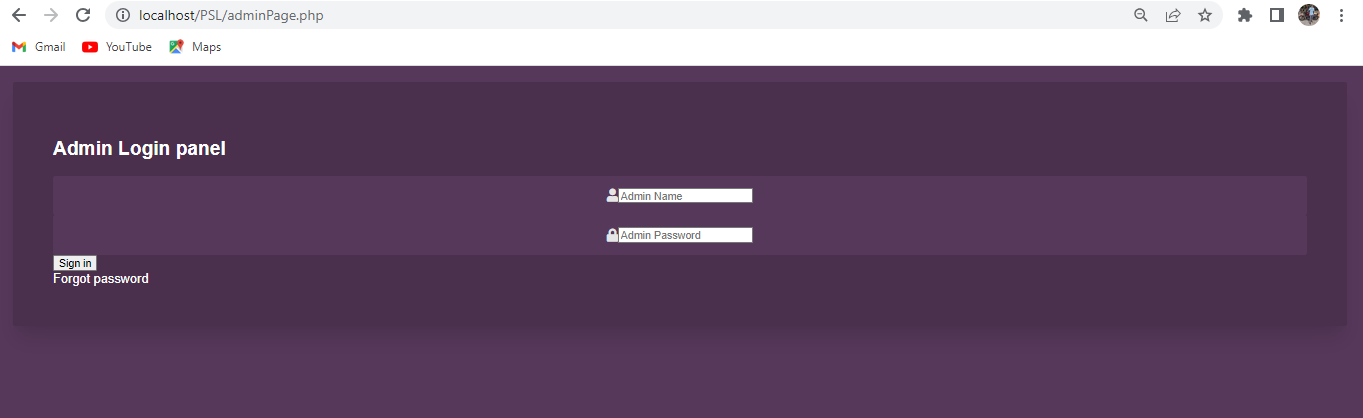
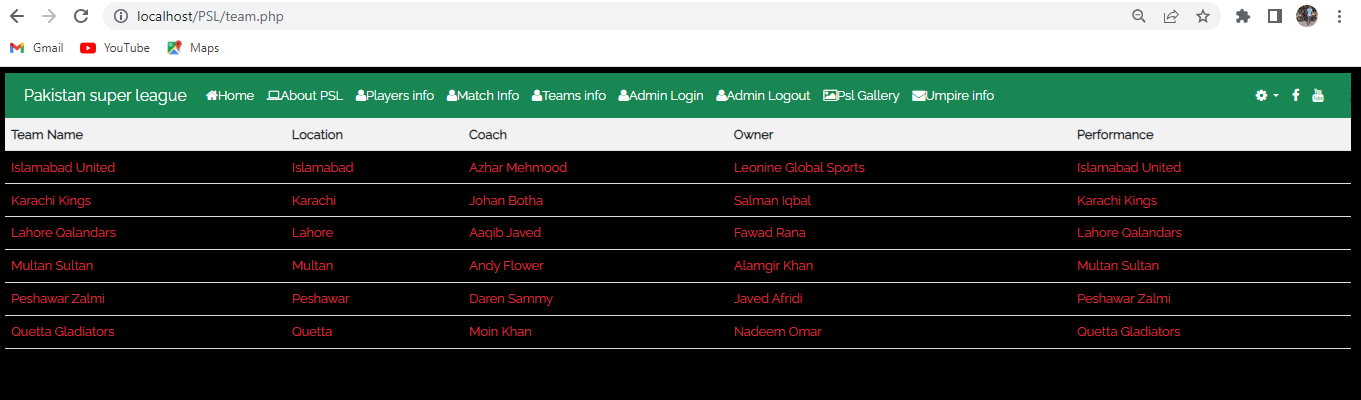
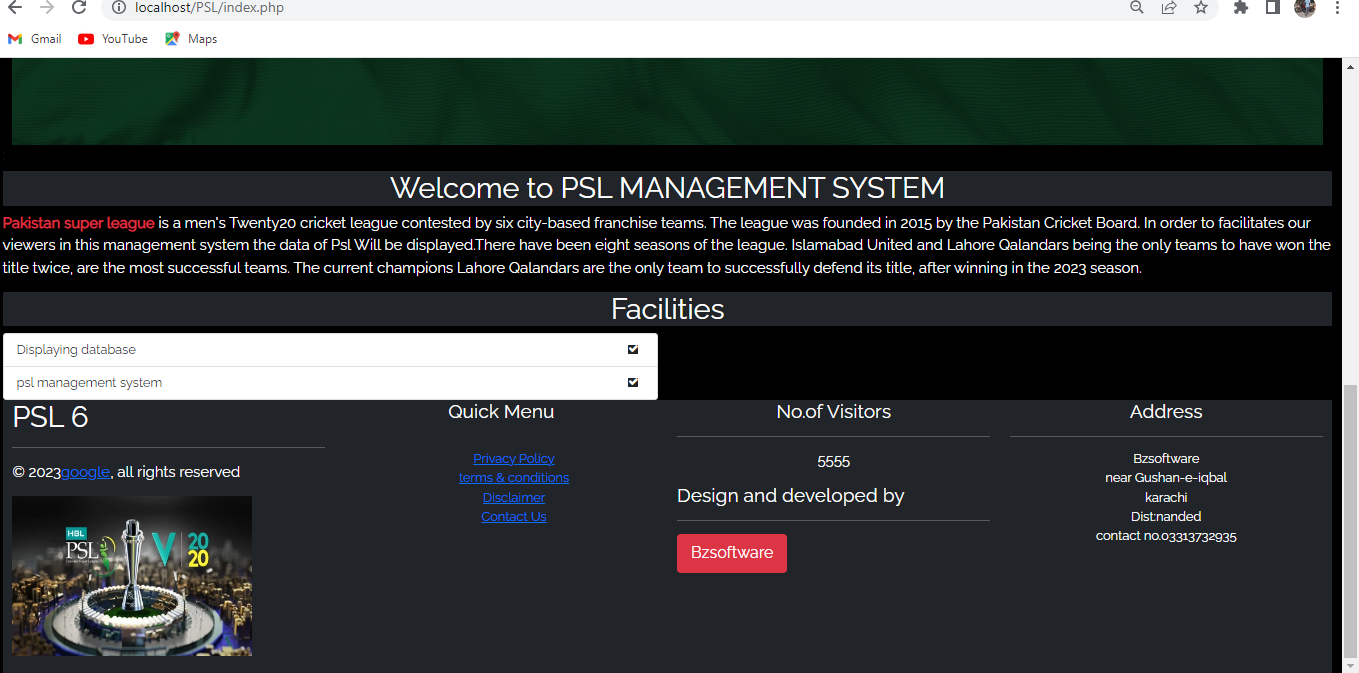
PRIMARY KEY(match\_Id));

# Implementation of Front-End and Back-End:

For the front-end we used HTML, CSS and additional CSS frameworks like Bootstrap and animate CSS.we also implemented Font Awesome Library of css and Also used carousel Slider in order to display a eye-catching frontend.We Also Implemented Javascript query to save the table forms

For back-end we have use PHP which has MySqli extension that fetches and retrieves the data

We also implemented All the files in our project using Php



# Security of Database:

We have used the mysqli extension of PHP which allows integrated security and prevents the database from SQL injection attacks. we have avoided the use of default port for integrating our database.

# Github Repository:

https://github.com/FaizanK116/DBMS-Project

# Individual Contributions:

* Salman made the Entity Relationship Model and normalized the tables.
* Faizan went through different websites to gather the information and collectively created the data set and schema.
* Bilal developed the front-end and back-end of the website using HTML, CSS, and PHP additional frameworks like bootstrap and animate CSS were also used.