# VISVESVARAYA TECHNOLOGICAL UNIVERSITY

JNANASANGAMA, BELAGAVI - 590018

**Mini Project Report**

**on**

## PRO KABADDI LEAGUE DATABASE MANAGEMENT SYSTEM

*Submitted in partial fulfillment for the award of degree of*

**Bachelor of Engineering**
**in**
**COMPUTER SCIENCE AND ENGINEERING**

Submitted by
**AKSHAY ANAND**
1BG17CS010

*Vidyaya Amrutham Ashnuthe*

B.N.M. Institute of Technology

**(Approved by AICTE, Affiliated to VTU, Accredited as grade A Institution by NAAC.**
**All UG branches – CSE, ECE, EEE, ISE & Mech.E accredited by NBA for academic years 2018-19 to 2020-21 & valid upto 30.06.2021)**
Post box no. 7087, 27th cross, 12th Main, Banashankari 2nd Stage, Bengaluru- 560070, INDIA
Ph: 91-80- 26711780/81/82    Email: principal@bnmit.in, www. bnmit.org

**Department of Computer Science and Engineering**

2019-2020

B.N.M. Institute of Technology

**(Approved by AICTE, Affiliated to VTU, Accredited as grade A Institution by NAAC.**
**All UG branches – CSE, ECE, EEE, ISE & Mech.E accredited by NBA for academic years 2018-19 to 2020-21 & valid upto 30.06.2021)**
Post box no. 7087, 27th cross, 12th Main, Banashankari 2nd Stage, Bengaluru- 560070, INDIA
Ph: 91-80- 26711780/81/82 Email: principal@bnmit.in, www. bnmit.org

## Department of Computer Science and Engineering

*Vidyaya Amrutham Ashnuthe*

# CERTIFICATE

Certified that the Mini Project entitled **Pro kabaddi League database Management System** carried out by **Mr. AKSHAY ANAND** USN **1BG17CS010** a bonafied student of V Semester B.E., **B.N.M Institute of Technology** in partial fulfillment for the Bachelor of Engineering in COMPUTER SCIENCE AND ENGINEERING of the **Visvesvaraya Technological University**, Belagavi during the year 2019-20.It is certified that all corrections/ suggestions indicated for internal Assessment have been incorporated in the report. The project report has been approved as it satisfies the academic requirements in respect of Database Management Systems Laboratory with Mini Project as prescribed for the said degree.

**Dr. Sejal Santosh Nimbhorkar**
**Associate Professor & Lab-Incharge**
**Department of CSE**
**BNMIT, Bengaluru**

**Dr. Sahana D.Gowda**
**Professor &HOD**
**Department of CSE**
**BNMIT, Bengaluru**

**Name & Signature**

**Examiner1:**

**Examiner2:**

# Table of Contents

# ABSTRACT

The project illustrates design and implementation of Pro kabaddi League Database Management System. Kabaddi has always been a sport of  great fun in our country, played across different regions .But, with the inception of Pro kabaddi Legue(PKL) in 2014 ,it is no more a game of fun. It had became an international sport with viewership trolling in millions. This massive increase in the popularity of the sport with the inception of PKL also brings together a lot of challenges .As in, viewers want real time live updates of the matches currently going on. Also, audiences are interested in knowing the stats and rankings of players as well as the teams.

The current System is not efficient to cater to all these user requirements. Henceforth, this project aims to design an efficient System that will able to meet all the viewers requirements in accordance with the ever increasing. The system will be a web application which can provide real time updates to viewers about various ongoing and upcoming matches, player stats, Team rankings etc.

This application also intends to provide users/viewers with most simplified interface/UI where they can navigate easily to various tabs with a few clicks. The proposed system also provides easy and efficient management .

# ACKNOWLEDGEMENT

The success and final outcome of this project required a lot of guidance and assistance from many people and I am extremely privileged to have got this all along the completion of my project.

I would like to thank **Shri. Narayan Rao R Maanay**, Secretary, BNMIT, Bengaluru for providing excellent academic environment in the college.

I would like to sincerely thank **Prof. T J Rama Murthy**, Director, BNMIT, Bengaluru for having extended his support and encouragement during the course of the work.

I would like to express my gratitude to **Shri Eishwar Narayan Rao R Maanay**, Dean, BNMIT, Bengaluru for his relentless support, guidance and assistance.

I would like to thank **Dr. Krishnamurthy G N**, Principal, BNMIT, Bengaluru for his constant encouragement.

I would like to thank, **Dr. Sahana D Gowda**, Professor and Head of the Department of Computer Science and Engineering who has shared her opinions and thoughts which helped me in giving my presentation successfully.

I would like to extend my heartfelt, sincere gratitude to **Dr. Sejal Santosh Nimbhorkar**, Associate Professor, Department of Computer Science and Engineering, BNMIT, Bengaluru, for completion of the project.

Finally, I take this opportunity to extend my earnest gratitude and respect to my parents, teaching & non-teaching staffs of the department and all my friends, for giving me valuable advices and support at all times in all possible ways.

AKSHAY ANAND
1BG17CS010

# Chapter 1

# INTRODUCTION

## Overview of Database Management Systems

A Database Management System(DBMS) is a general purposes of software system that allows creation, definition and manipulation of a database, allowing users to store, process and analyze data easily. A Database Management System(DBMS) provides us with an interface or a tool, to perform various operations like creating database, storing data in it, updating data ,creating tables in the database and a lot more. Modern Database Management Systems (DBMS) also provide protection and added security features to the databases. In addition, it also maintains data consistency in case of multiple users. Some examples of the most commonly used Database Management Systems are MySQL, ORACLE DB, IBM DB2, Amazon Simple DB, etcetera.

### Characteristics of a Database Management System

- Reduced redundancy of data stored throughout the database with the help of concepts like normalization which divides the data in the database to reduce repeated data.
- Data consistency is maintained throughout the database with data being constantly added, updated and deleted.
- Multiuser support and concurrent access, allows multiple users to work on the database at the same time.
- Simple query language can be uses to easily fetch, insert, update and delete data from the database.
- Security is built in which restricts unauthorized access to the database. Different users have their own associated permissions.

### Advantages of a Database Management System

- Provides data abstraction and segregation of application program from the data.
- Reduced redundancy of data ensures maximum cost efficiency for the

- Reduced development time while building applications that use database features.
- Seamless integration into the application program allowing DBMS to be used for almost any real world application.

### Disadvantages of Database Management Systems

- DBMS interface with many different changing technologies and have a significant impact on the company's resources and culture.
- Most DBMS software are proprietary and are very expensive to use for simple applications.
- They require sophisticated hardware, software along with highly skilled personnel which further increases the cost of infrastructure and maintenance.

## 1.2 Problem Statement

A pro kabaddi League(PKL) is a is a professional-level Kabaddi league in India. It was launched in 2014 and has gained massive popularity with almost 500 million viewers every season. It is held annually between 12 teams with more teams willing to join representing 12 cities across the country, owned by different franchises.

The PKL's rules are similar to that of the indoor team version of Kabaddi, but with additional rules to encourage more scoring .Basically the teams are divided into groups and initially group matches are held. The top teams of each group qualify for the Quarter-finals, Semi-finals and then the Finals.The team includes Players, Coaching Staff, Managers etc.The System currently in place is not efficient enough to manage various data related to players and teams. Also, viewers today need latest updates of every ongoing and upcoming matches. Hence, there is a demand for a system that fulfills the need of people as well as the kabaddi organization for storing and managing data.

## 1.3 Objective

- The main objective of this project is to develop a database management System for the Pro kabaddi League.
- Investigate the requirements for proper storage and management of data.

- Providing an efficient system to the kabaddi organization. So, that they can add, update or delete their data whenever required
- Providing essential tools that keeps tracks of all the details about every ongoing, upcoming as well as past matches.
- Finally, to design and develop an web application to make all this information available to viewers.
- One of the major objective through this online portal would be also to promote kabaddi as a sport.

## 1.4 Data Set Description

- An entity called **Player** is created with the **player_Id** being the Main Key Attribute also known as the Primary key. The Attributes belonging to the Entity is **type, age, name** and **team_name**

- The Entity **Team** is created with the **tid** being the Primary key. The other attributes **hcity, name**, **tot_match** and **win.**

- The Entity **Match** consists of the **mno** as key attribute. The other attributes it consists are **tname1,tname2,date,time** and **venue**. Team name are referenced to entity Team.

- Entity **stats** consists of **id** as key attribute referenced from entity Player .The other attributes. It consists are **mat_played, not_out, pos** and **tot_point.**

- Entity **teamstatus** consists of **mno** as key attribute referenced from entity Match.The other attributes are tid and status where tid is refrenced to **Team**.

# Chapter 2

# SYSTEM REQUIREMENTS

## 2.1 Software and Hardware Requirements

Software requirements deal with defining software resource requirements and prerequisites that need to be installed on a computer to provide optimal functioning of an application.

### 2.1.1 Software Requirements

**FrontEnd**

- HTML5/CSS
- Bootstrap Boilerplate
- JavaScript
- Google Chrome (Web Browser)

**BackEnd**

- XAMPP Server (v3.2.2) for Apache Server(localhost)
- MySQL (v8.0.12) for Database Management System
- PHP (v7.2.10) for Server Side Scripting
- Atom (Source Code Editor)
- Windows 7 and above

### 2.1.2 Hardware Requirements

The most common set of requirements defined by any operating system or software application is the physical computer resources, also known as hardware.

- CPU: Intel or AMD processor
- Cores: Dual-Core (Quad-Core recommended)
- RAM: minimum 4GB (>4GBrecommended)
- Graphics: Intel Integrated Graphics or AMD Equivalent
- Secondary Storage:250GB
- Display Resolution: 1366x768 (1920x1080recommended)

# Chapter 3

# SYSTEM DESIGN

## 3.1 Entity Relationship Diagram

An Entity Relationship Diagram describes interrelated things of interest in a specific domain of knowledge. A basic ER model is composed of entity types and specifies relationships that can exist between instances of those entity types. In software engineering, an ER model is commonly formed or represent things that a business needs to remember in order to perform business processes. Consequently, the ER model becomes an abstract data model, that defines a data or information structure which can be implemented in a database, typically a relational database.

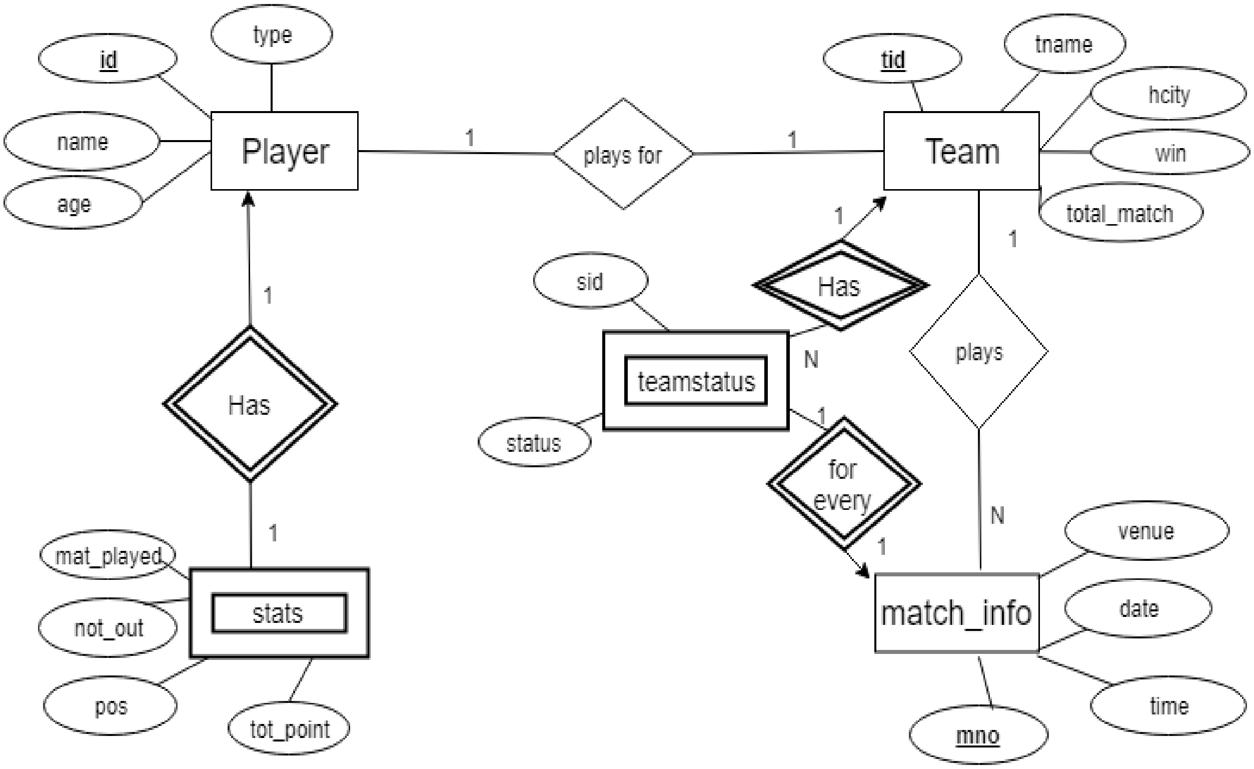


**Figure 3.1 Entity Relationship Diagram**

## 3.2 Schema Database Relationship Diagram

A database schema is the skeleton structure that represents the logical view of the entire database. It formulates all the constraints that are to be applied on the data. A database schema defines its entities and the relationship among them. It contains a descriptive detail of the database, which can be depicted by means of schema diagrams.

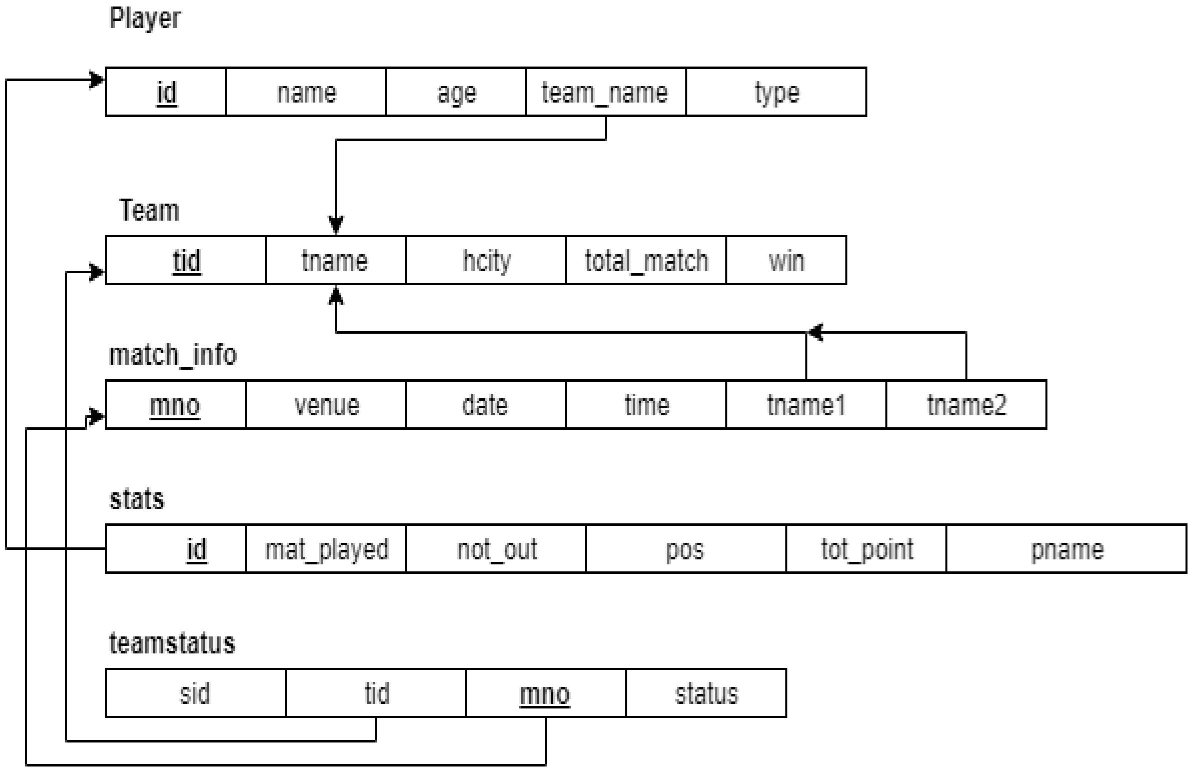


**Figure 3.2 Schema Database Relationship Diagram**

## 3.3 Overview of Graphical User Interface

GUI is program interface that takes advantages of the computer's graphics capabilities to make the program easier to user. Well-designed graphical user interfaces can free the user from learning complex command language. On the other hand, many users find that they work more efficiently with a command-driven interface, especially if they already know the command language.

Hypertext Markup Language (HTML) is the standard markup language for creating web pages and web applications. With Cascading Style Sheets (CSS) and JavaScript it forms a triad of cornerstone technologies for the World Wide Web. Web browsers receive HTML documents from a web server or from local storage and render them into multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document.

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language. Although most often used to set the visual style of web pages and user interfaces written in HTML and XHTML, the language can be applied to any XML document and is applicable to rendering in speech, or on other media. CSS is a cornerstone technology used by most websites to create visually engaging webpages, user interfaces for web applications, and user interfaces for many mobile applications.

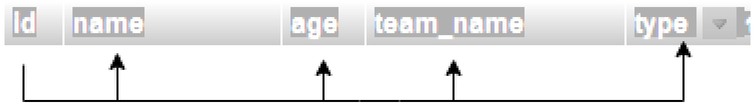
Hypertext Preprocessor ( PHP) is a general-purpose programming language originally designed for web development. PHP code may be executed with a command line interface (CLI), embedded into HTML code, or used in combination with various web template systems, web content management systems, and web frameworks. PHP code is usually processed by a PHP interpreter implemented as a module in a web server or as a Common Gateway Interface (CGI) executable. The web server outputs the results of the interpreted and executed PHP code, which may be any type of data, such as generated HTML code or binary image data. PHP can be used for many programming tasks outside of the web context, such as standalone graphical applications and robotic drone control

## 3.4 Normalization

Normalization rule are divided into following normal form.

1. First Normal Form
2. Second Normal Form
3. Third Normal Form

**PLAYER table**



It is in 1NF as it does not have any composite or multivalued attributes.

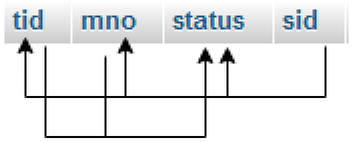
Player type is multivalued, hence it is decomposed to a separate relation.

It is in 2NF as it is fully functionally dependent.

(id) → (name, age, team_name, type)

As the relation does not contain a non-key attribute functionally determining other non-key attributes, it is in 3NF(there is no transitivity in Relation).

**TEAMSTATUS table**



It is in 1NF as it does not have any composite or multivalued attributes.

(sid) → (tid,mno,status)

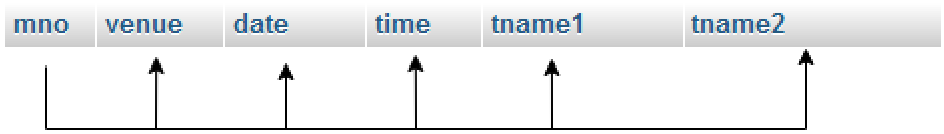
It is in 2NF as it is fully functionally dependent.

(mno) → (tid,status)

Team Status is decomposed in a separate relation so as to decompose it into 2NF.

As the relation does not contain a non-key attribute functionally determining other non-key attributes, it is in 3NF(there is no transitivity in Realation).

### MATCH_INFO table



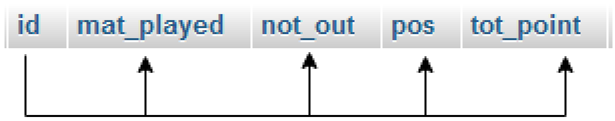
It is in 1NF as it does not have any composite or multivalued attributes.

It is in 2NF as it is fully functionally dependent.

(mno) → (venue, date, time, tname1,tname2)

As the relation does not contain a non-key attribute functionally determining other non-key attributes, it is in 3NF(there is no transitivity in Realation).

### STATS table



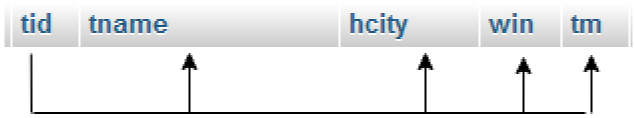
It is in 1NF as it does not have any composite or multivalued attributes.

It is in 2NF as it is fully functionally dependent.

(id) → (mat_played, not_out, pos, tot_point)

As the relation does not contain a non-key attribute functionally determining other non-key attributes, it is in 3NF(there is no transitivity in Realation).

### TEAM table



It is in 1NF as it does not have any composite or multivalued attributes.

It is in 2NF as it is fully functionally dependent.

(tid) → (tname, hcity, win, tm)

As the relation does not contain a non-key attribute functionally determining other non-key attributes, it is in 3NF(there is no transitivity in Realation).

# Chapter 4

# IMPLEMENTATION

## 4.1 Table Creation

**REGISTER**

```
create table register
(
user  varchar(100) primary key,
name varchar(100) not null,
city varchar(100) not null,
dob date not null,
password varchar(10) not null );
```

**PLAYER**

```
create table player
(
id number(3)  primary key,
name varchar(20) not null,
age number(2) not null,
team_name varchar(10),
type varchar(10) not null,
Foreign key(team_name) references team(tname) on delete cascade );
```

**TEAM**

```
create table team
(
tid number(3) primary key,
tname varchar(20) not null,
hcity varchar(10) not null,
win number(2) not null,
tm number(3) not null
);
```

**STATS**

```
create table stats
(
id number(3),
mat_played number(5) not null,
not_out number(5) not null,
pos number(5) not null,
tot_point  number(5) not null,
primary key(id),
Foreign key(id) references player(id) on delete cascade
);
```

**TEAMSTATUS**

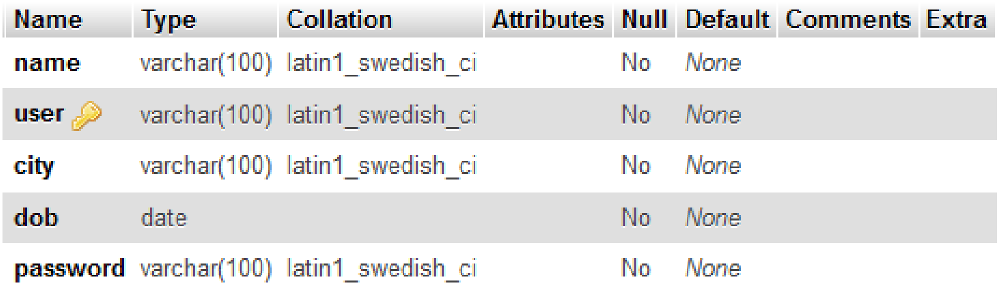
```
create table teamstatus
(
sid int auto_increment,
mno number(3) not null,
tid number(3),
status  varchar(2) not null,
primary key(mno),
Foreign key(mno) references match_info(mno) on delete cascade,
Foreign key(tid) references team(tid) on delete cascade
);
```

**MATCH_INFO**

```
create table match_info
(  mno number(3) primary key,
venue varchar(20) not null,
date date not null,
time time not null,
);
```

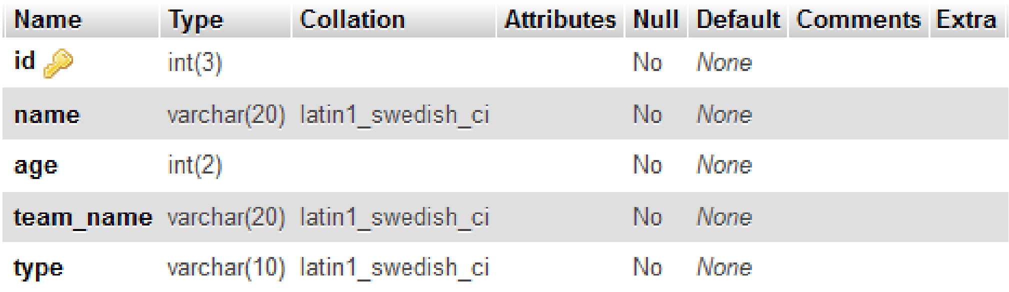
## 4.2 Description of Tables

### Table 4.2.1 – Register Description



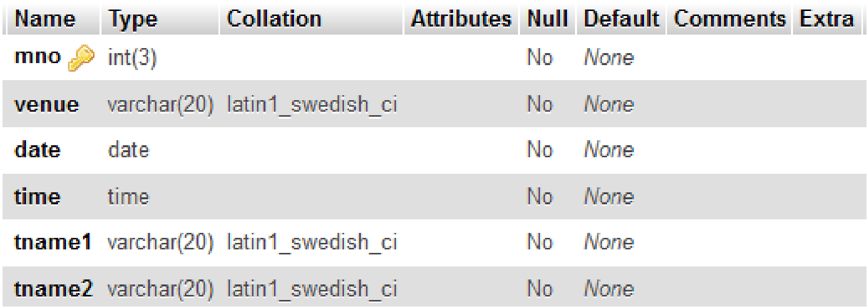
| Name | Type | Collation | Attributes | Null | Default | Comments | Extra |
|---|---|---|---|---|---|---|---|
| **name** | varchar(100) | latin1_swedish_ci | | No | *None* | | |
| **user** 🔑 | varchar(100) | latin1_swedish_ci | | No | *None* | | |
| **city** | varchar(100) | latin1_swedish_ci | | No | *None* | | |
| **dob** | date | | | No | *None* | | |
| **password** | varchar(100) | latin1_swedish_ci | | No | *None* | | |

### Table 4.2.2 – Player Description



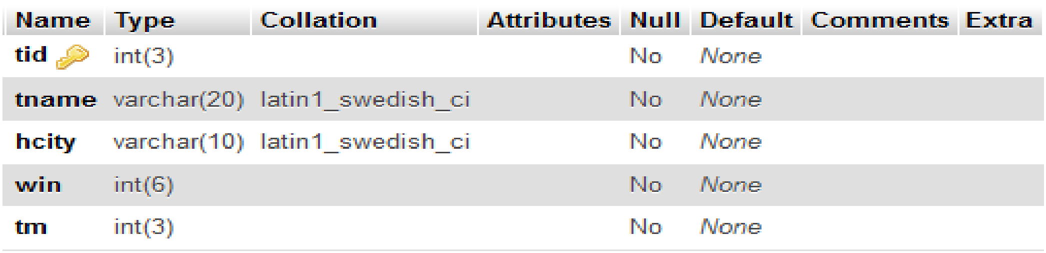
| Name | Type | Collation | Attributes | Null | Default | Comments | Extra |
|---|---|---|---|---|---|---|---|
| **id** 🔑 | int(3) | | | No | *None* | | |
| **name** | varchar(20) | latin1_swedish_ci | | No | *None* | | |
| **age** | int(2) | | | No | *None* | | |
| **team_name** | varchar(20) | latin1_swedish_ci | | No | *None* | | |
| **type** | varchar(10) | latin1_swedish_ci | | No | *None* | | |

### Table 4.2.3 – Match_info Description



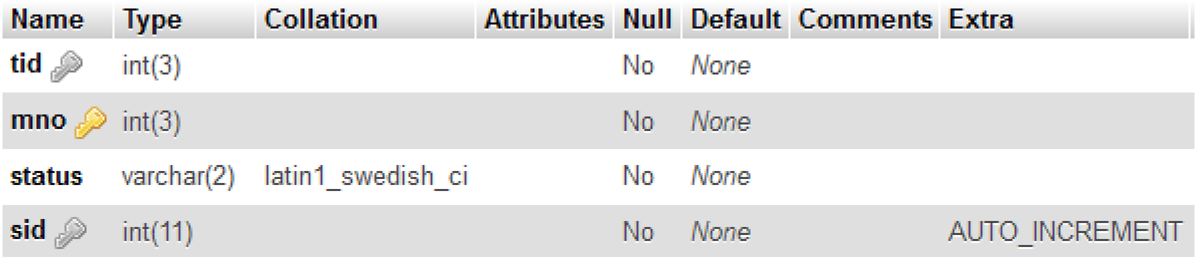
| Name | Type | Collation | Attributes | Null | Default | Comments | Extra |
|---|---|---|---|---|---|---|---|
| **mno** 🔑 | int(3) | | | No | *None* | | |
| **venue** | varchar(20) | latin1_swedish_ci | | No | *None* | | |
| **date** | date | | | No | *None* | | |
| **time** | time | | | No | *None* | | |
| **tname1** | varchar(20) | latin1_swedish_ci | | No | *None* | | |
| **tname2** | varchar(20) | latin1_swedish_ci | | No | *None* | | |

### Table 4.2.4 – Team Description



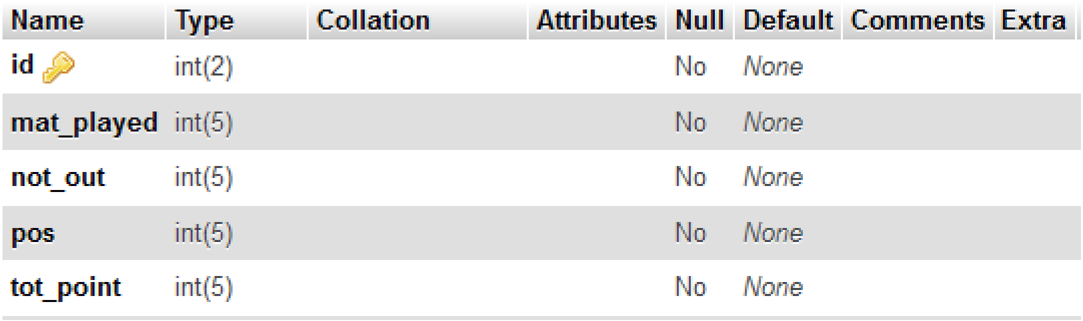
| Name | Type | Collation | Attributes | Null | Default | Comments | Extra |
|---|---|---|---|---|---|---|---|
| **tid** 🔑 | int(3) | | | No | *None* | | |
| **tname** | varchar(20) | latin1_swedish_ci | | No | *None* | | |
| **hcity** | varchar(10) | latin1_swedish_ci | | No | *None* | | |
| **win** | int(6) | | | No | *None* | | |
| **tm** | int(3) | | | No | *None* | | |

### Table 4.2.5 – Teamstatus Description



| Name | Type | Collation | Attributes | Null | Default | Comments | Extra |
|---|---|---|---|---|---|---|---|
| tid | int(3) | | | No | *None* | | |
| mno | int(3) | | | No | *None* | | |
| status | varchar(2) | latin1_swedish_ci | | No | *None* | | |
| sid | int(11) | | | No | *None* | | AUTO_INCREMENT |

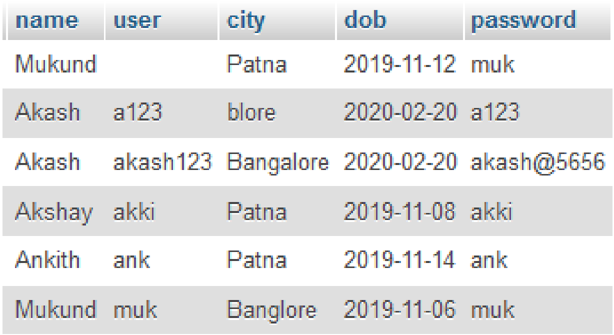
### Table 4.2.6 – stats Description



| Name | Type | Collation | Attributes | Null | Default | Comments | Extra |
|---|---|---|---|---|---|---|---|
| id | int(2) | | | No | *None* | | |
| mat_played | int(5) | | | No | *None* | | |
| not_out | int(5) | | | No | *None* | | |
| pos | int(5) | | | No | *None* | | |
| tot_point | int(5) | | | No | *None* | | |

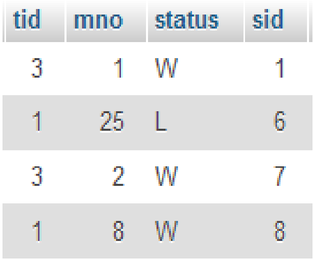
## 4.3 Populated Tables

### Table 4.3.1 – Register Table Values



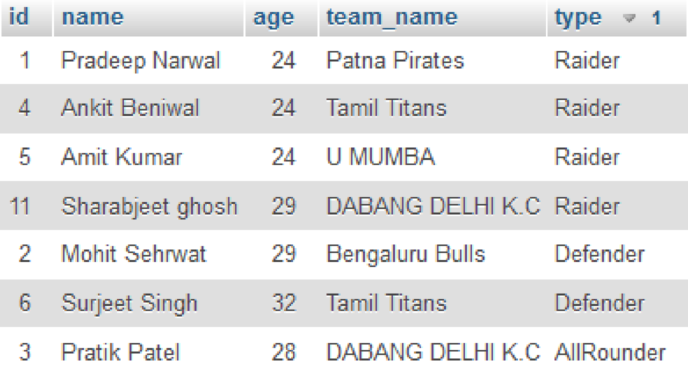
| name | user | city | dob | password |
|---|---|---|---|---|
| Mukund | | Patna | 2019-11-12 | muk |
| Akash | a123 | blore | 2020-02-20 | a123 |
| Akash | akash123 | Bangalore | 2020-02-20 | akash@5656 |
| Akshay | akki | Patna | 2019-11-08 | akki |
| Ankith | ank | Patna | 2019-11-14 | ank |
| Mukund | muk | Banglore | 2019-11-06 | muk |

### Table 4.3.2 – Teamstatus Table Values



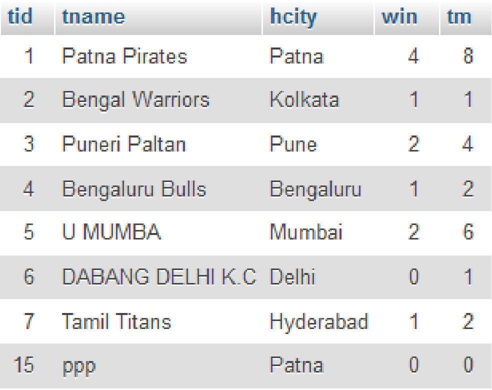
| tid | mno | status | sid |
|---|---|---|---|
| 3 | 1 | W | 1 |
| 1 | 25 | L | 6 |
| 3 | 2 | W | 7 |
| 1 | 8 | W | 8 |

### Table 4.3.3 – Player Table Values



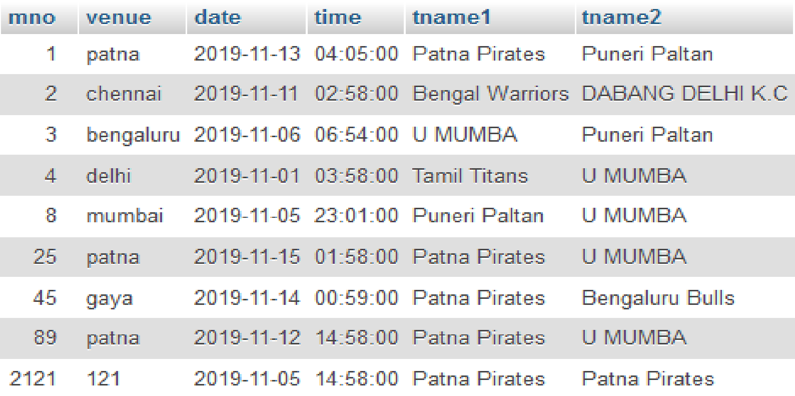
| id | name | age | team_name | type ▼ 1 |
|---|---|---|---|---|
| 1 | Pradeep Narwal | 24 | Patna Pirates | Raider |
| 4 | Ankit Beniwal | 24 | Tamil Titans | Raider |
| 5 | Amit Kumar | 24 | U MUMBA | Raider |
| 11 | Sharabjeet ghosh | 29 | DABANG DELHI K.C | Raider |
| 2 | Mohit Sehrwat | 29 | Bengaluru Bulls | Defender |
| 6 | Surjeet Singh | 32 | Tamil Titans | Defender |
| 3 | Pratik Patel | 28 | DABANG DELHI K.C | AllRounder |

### Table 4.3.4 – Team Table Values



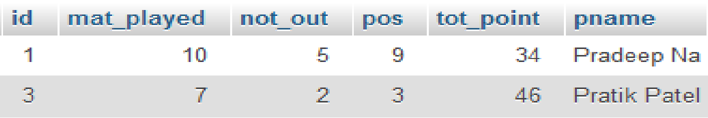
| tid | tname | hcity | win | tm |
|---|---|---|---|---|
| 1 | Patna Pirates | Patna | 4 | 8 |
| 2 | Bengal Warriors | Kolkata | 1 | 1 |
| 3 | Puneri Paltan | Pune | 2 | 4 |
| 4 | Bengaluru Bulls | Bengaluru | 1 | 2 |
| 5 | U MUMBA | Mumbai | 2 | 6 |
| 6 | DABANG DELHI K.C | Delhi | 0 | 1 |
| 7 | Tamil Titans | Hyderabad | 1 | 2 |
| 15 | ppp | Patna | 0 | 0 |

### Table 4.3.5 – Match_Info Table Values



| mno | venue | date | time | tname1 | tname2 |
|---|---|---|---|---|---|
| 1 | patna | 2019-11-13 | 04:05:00 | Patna Pirates | Puneri Paltan |
| 2 | chennai | 2019-11-11 | 02:58:00 | Bengal Warriors | DABANG DELHI K.C |
| 3 | bengaluru | 2019-11-06 | 06:54:00 | U MUMBA | Puneri Paltan |
| 4 | delhi | 2019-11-01 | 03:58:00 | Tamil Titans | U MUMBA |
| 8 | mumbai | 2019-11-05 | 23:01:00 | Puneri Paltan | U MUMBA |
| 25 | patna | 2019-11-15 | 01:58:00 | Patna Pirates | U MUMBA |
| 45 | gaya | 2019-11-14 | 00:59:00 | Patna Pirates | Bengaluru Bulls |
| 89 | patna | 2019-11-12 | 14:58:00 | Patna Pirates | U MUMBA |
| 2121 | 121 | 2019-11-05 | 14:58:00 | Patna Pirates | Patna Pirates |

### Table 4.3.6 – stats Table Values

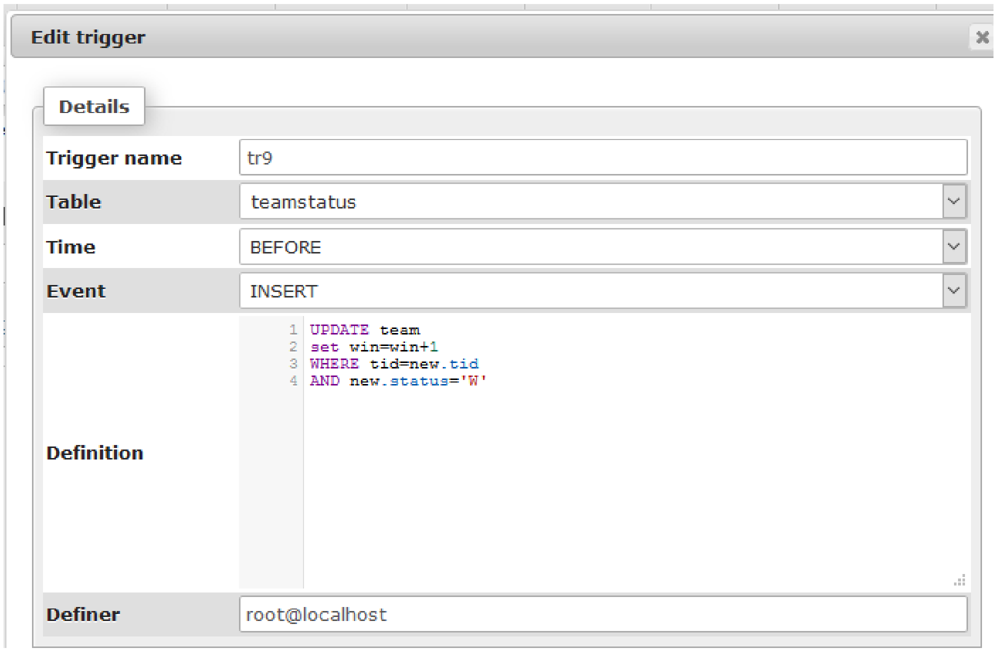


| id | mat_played | not_out | pos | tot_point | pname |
|---|---|---|---|---|---|
| 1 | 10 | 5 | 9 | 34 | Pradeep Na |
| 3 | 7 | 2 | 3 | 46 | Pratik Patel |

# 4.4 SQL Triggers & Stored Procedures

## Triggers

A database trigger is procedural code that is automatically executed in response to certain events on a table or view in a database. The trigger is mostly used for maintaining the integrity of the information on the database.

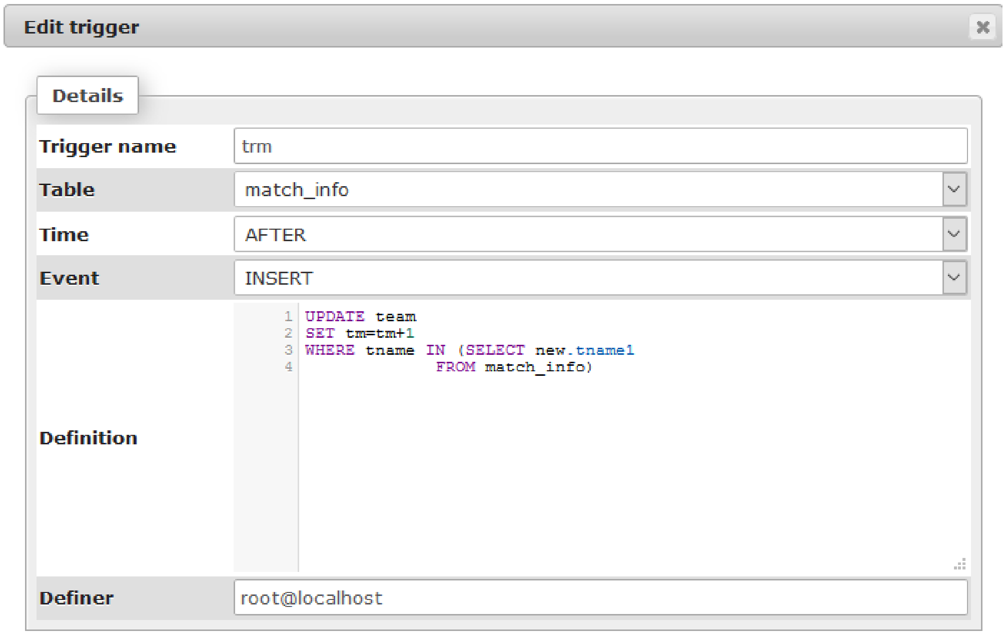


Edit trigger

Details

| | |
|---|---|
| Trigger name | tr9 |
| Table | teamstatus |
| Time | BEFORE |
| Event | INSERT |
| Definition | |
| Definer | root@localhost |

```
UPDATE team
set win=win+1
WHERE tid=new.tid
AND new.status='W'
```



Edit trigger

Details

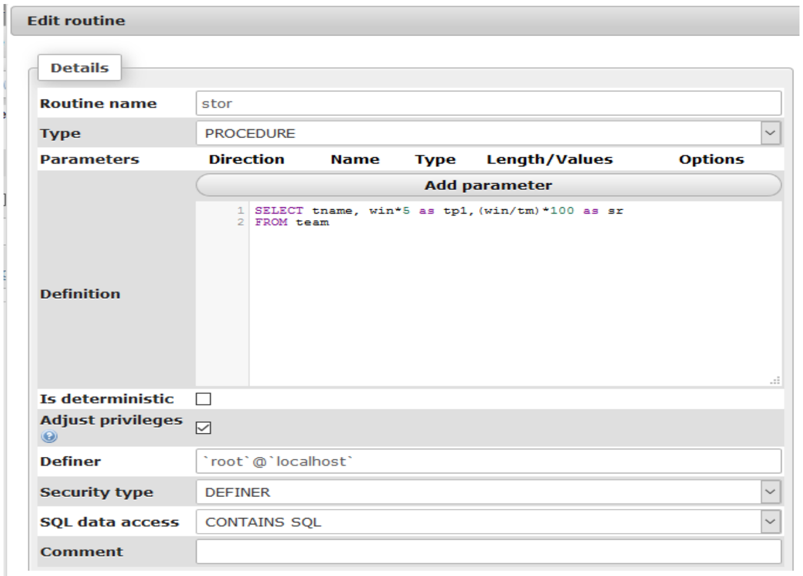
| | |
|---|---|
| Trigger name | trm |
| Table | match_info |
| Time | AFTER |
| Event | INSERT |
| Definition | |
| Definer | root@localhost |

```
UPDATE team
SET tm=tm+1
WHERE tname IN (SELECT new.tname1
             FROM match_info)
```

## Stored Procedure

A stored procedure is a set of Structured Query Language (SQL) statements with an assigned name, which are stored in a relational database management system as a group, so it can be reused and shared by multiple programs.

Code for Stored Procedure is as follows:



Edit routine

Details

Routine name: stor

Type: PROCEDURE

Parameters: Direction | Name | Type | Length/Values | Options

Definition:

```
SELECT tname, win*5 as tp1,(win/tm)*100 as sr
FROM team
```

Is deterministic: □

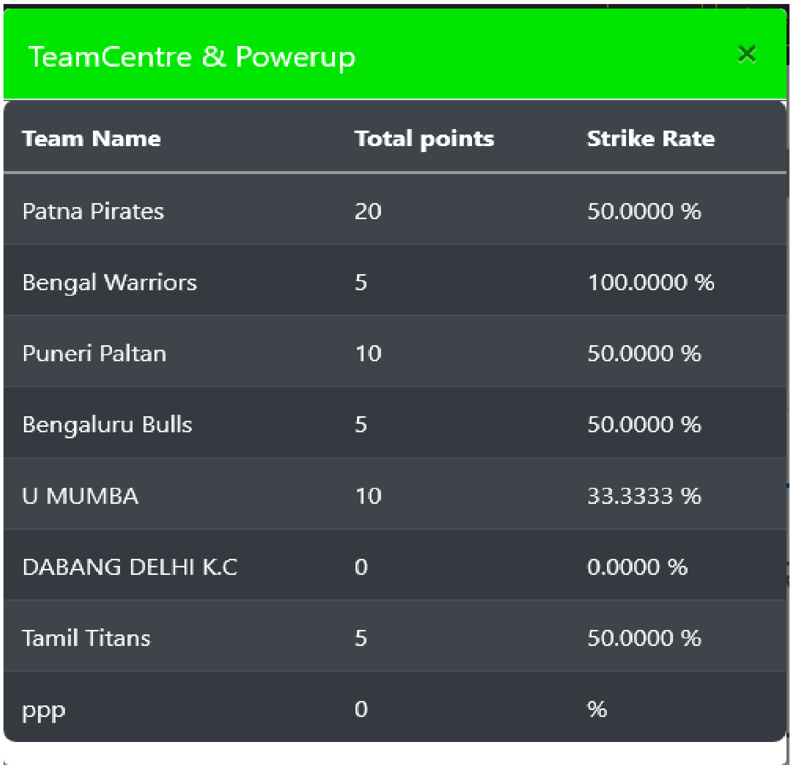
Adjust privileges: ☑

Definer: `root`@`localhost`

Security type: DEFINER

SQL data access: CONTAINS SQL

Comment:



TeamCentre & Powerup

| Team Name | Total points | Strike Rate |
|---|---|---|
| Patna Pirates | 20 | 50.0000 % |
| Bengal Warriors | 5 | 100.0000 % |
| Puneri Paltan | 10 | 50.0000 % |
| Bengaluru Bulls | 5 | 50.0000 % |
| U MUMBA | 10 | 33.3333 % |
| DABANG DELHI K.C | 0 | 0.0000 % |
| Tamil Titans | 5 | 50.0000 % |
| ppp | 0 | % |

## 4.5 Database Connectivity

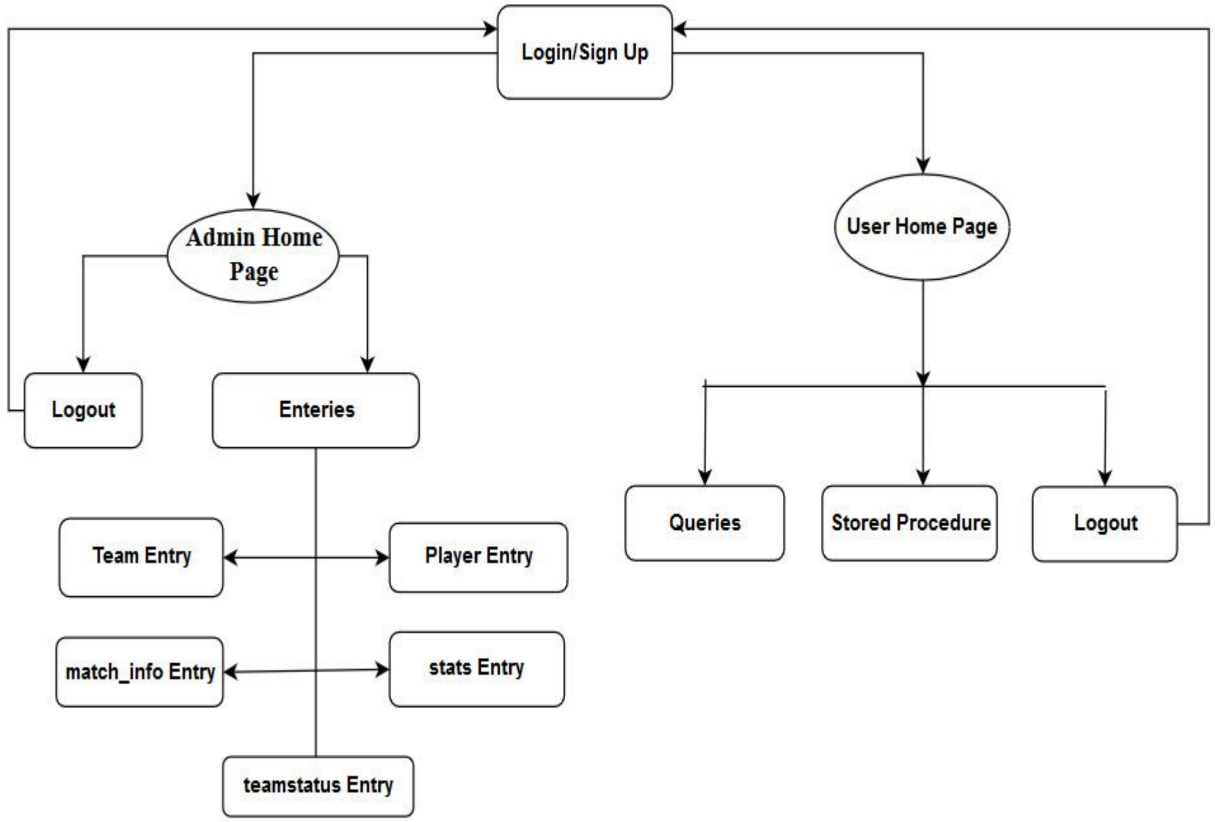
In computer science, a database connection is the means by which a database server and its client software communicate with each other. The term is used whether the client and the server are on different machines. The client uses a database connection to send commands to and receive replies from the server. A database is stored as a file or a set of files on magnetic disk or tape, optical disk, or some other secondary storage device. The information in these files may be broken down into records, each of which consists of one or more fields. The connection to XAMPP database is obtained by the following code snippet:

```
<?php
$servername   = "localhost";
$database = "dbms";
$username = "root";
$password = "";

// Create connection
$conn = new mysqli($servername, $username, $password, $database);
// Check connection
if ($conn->connect_error) {
   die("Connection failed: " . $conn->connect_error);
}
else
{
//Code for proceeding after establishment of connection
}
?
```

## 4.6 Modules

The data flow digram to transit through the various modules in this project are given below:



# Chapter 5

## RESULTS

Home page for the Pro kabaddi League

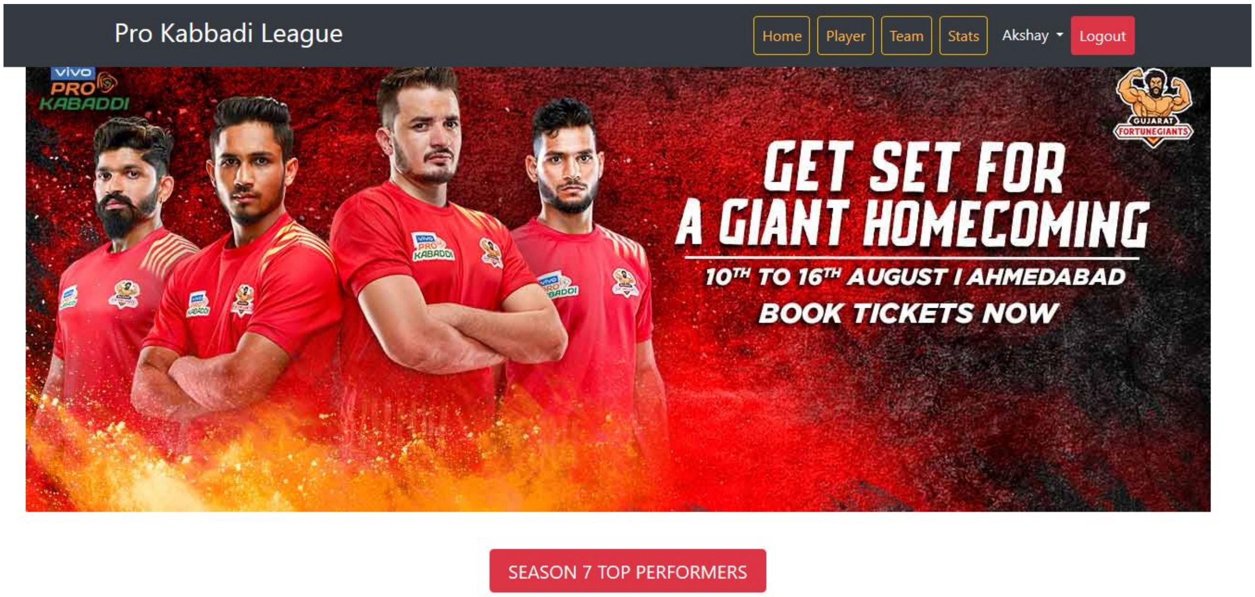


**Figure 5.1: Home page.**

Login page for users

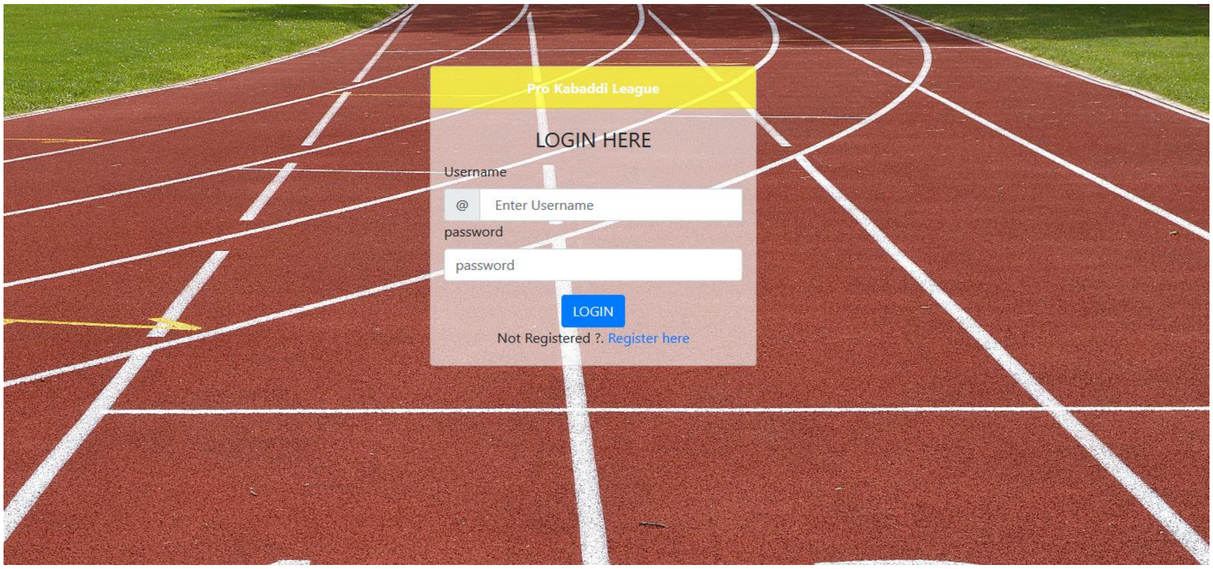


**Figure 5.2: Login Page**

Admin Dashboard

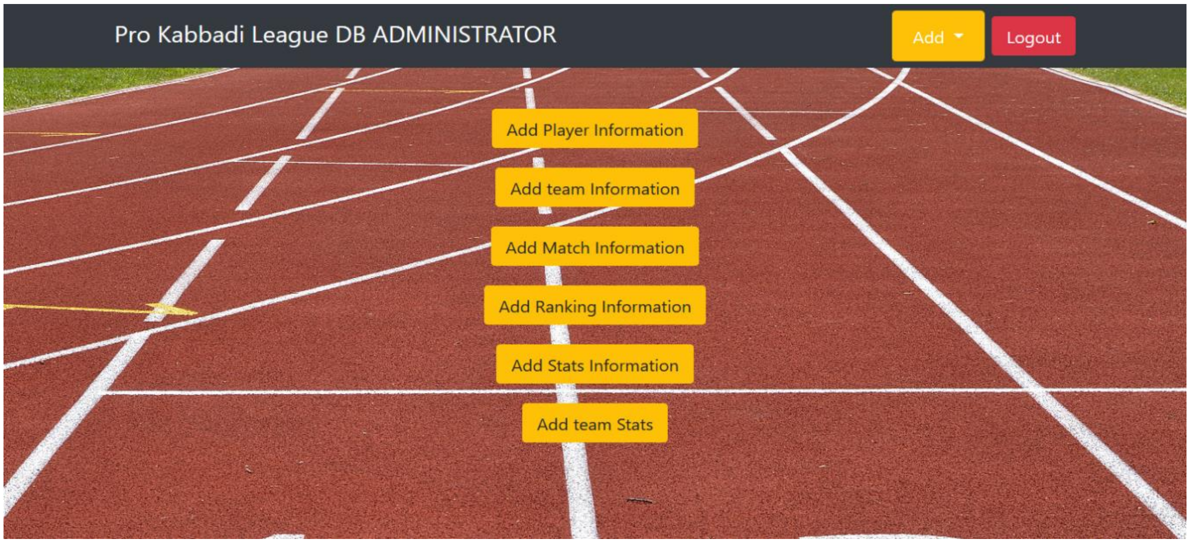
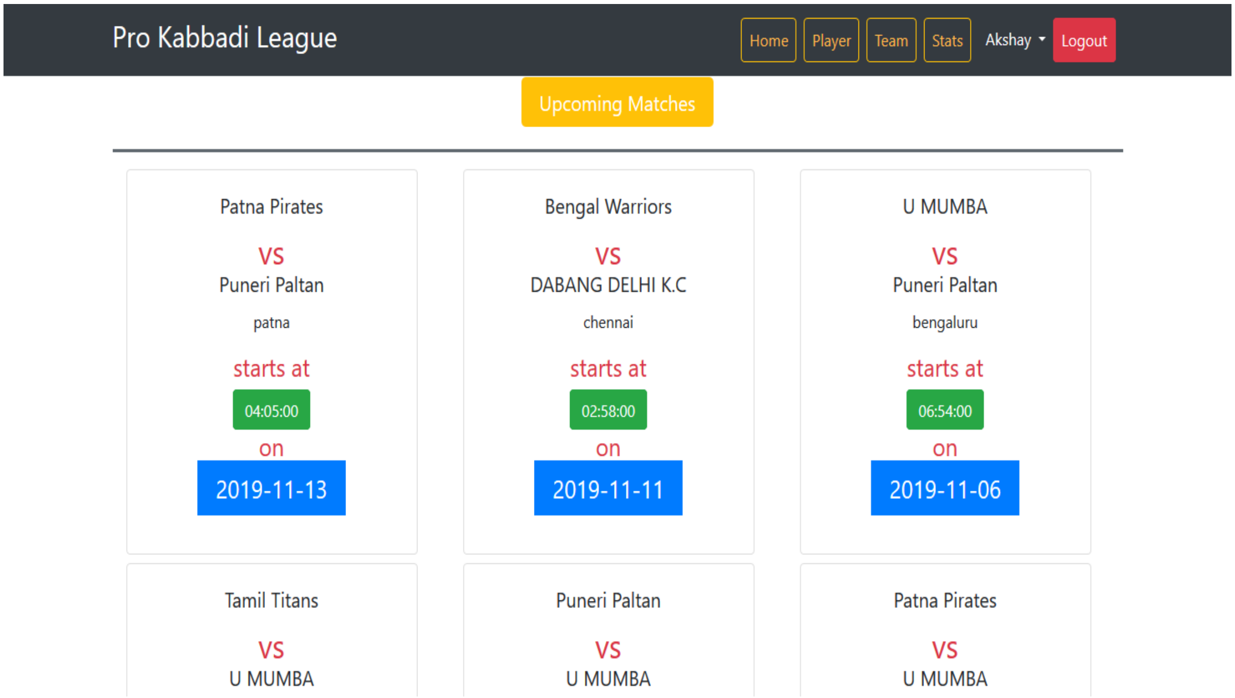


**Figure 5.3: Admin Dashboard**

User dashboard to display upcoming Matches



Patna Pirates
VS
Puneri Paltan
patna
starts at
04:05:00
on
2019-11-13

Bengal Warriors
VS
DABANG DELHI K.C
chennai
starts at
02:58:00
on
2019-11-11

U MUMBA
VS
Puneri Paltan
bengaluru
starts at
06:54:00
on
2019-11-06

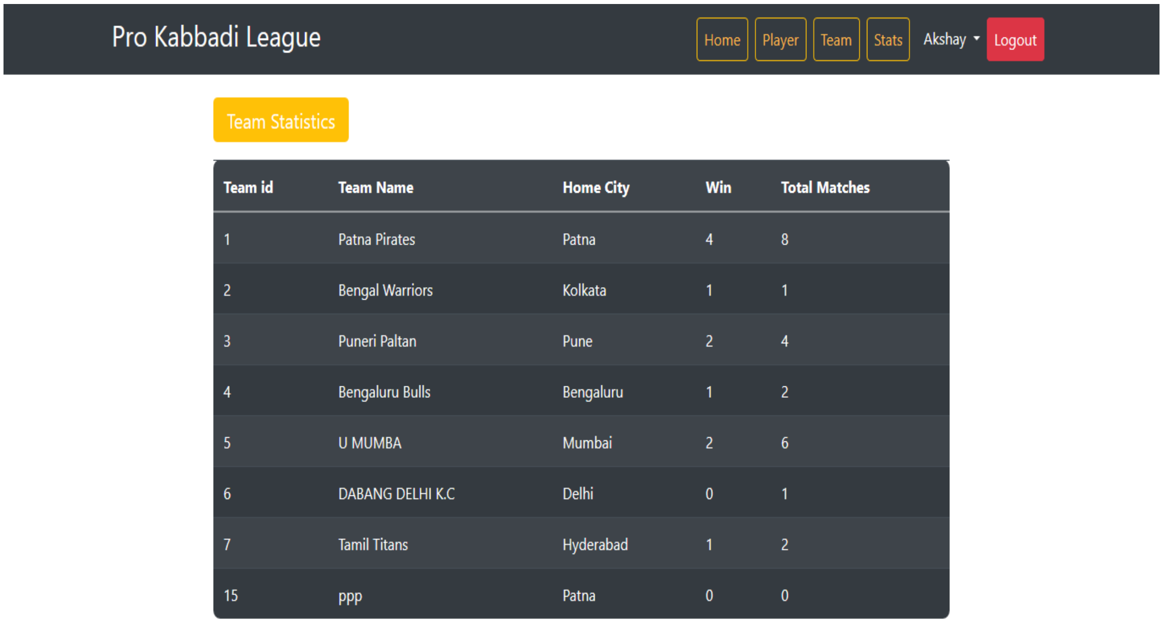
Tamil Titans
VS
U MUMBA

Puneri Paltan
VS
U MUMBA

Patna Pirates
VS
U MUMBA

**Figure 5.4: Upcoming Matches.**

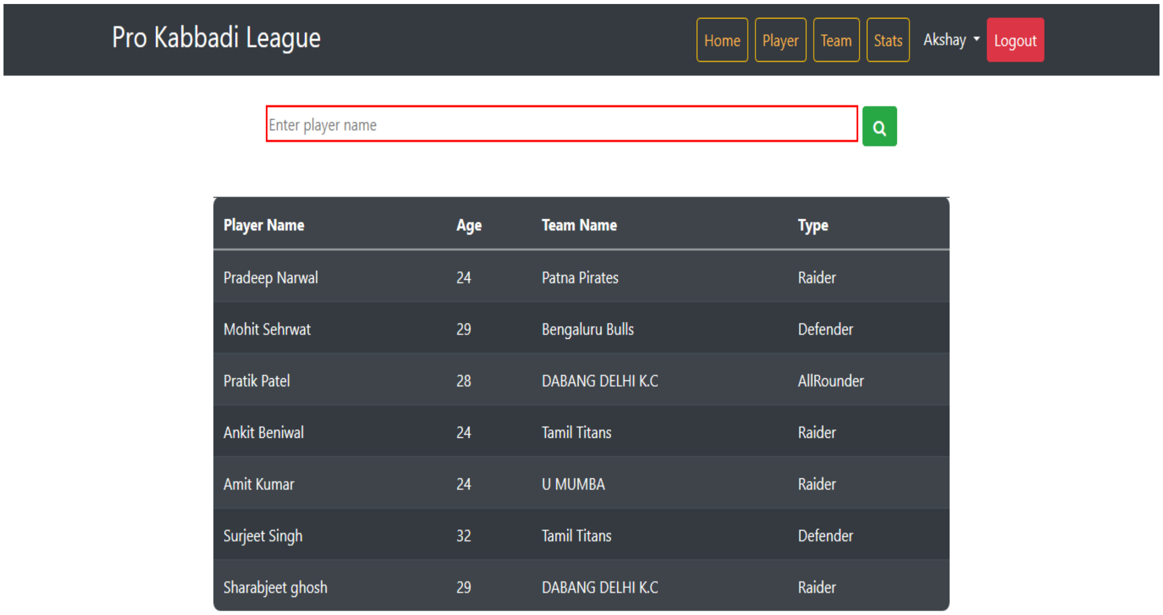
Page to display Team Scoreboard



| Team id | Team Name | Home City | Win | Total Matches |
|---|---|---|---|---|
| 1 | Patna Pirates | Patna | 4 | 8 |
| 2 | Bengal Warriors | Kolkata | 1 | 1 |
| 3 | Puneri Paltan | Pune | 2 | 4 |
| 4 | Bengaluru Bulls | Bengaluru | 1 | 2 |
| 5 | U MUMBA | Mumbai | 2 | 6 |
| 6 | DABANG DELHI K.C | Delhi | 0 | 1 |
| 7 | Tamil Titans | Hyderabad | 1 | 2 |
| 15 | ppp | Patna | 0 | 0 |

**Figure 5.5: Team Scoreboard**

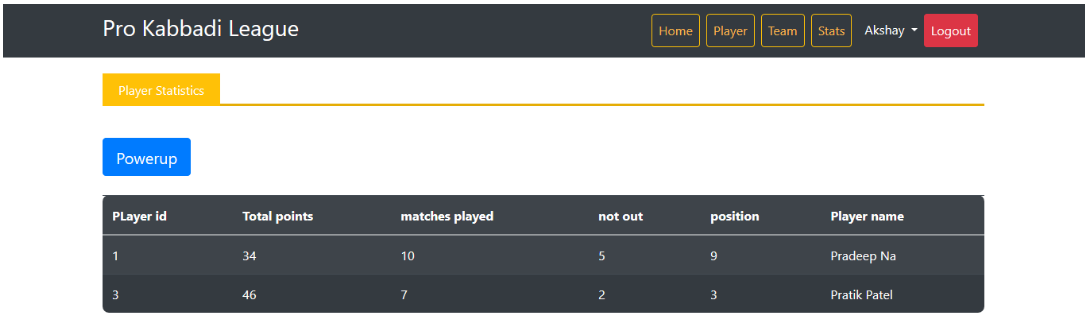
Section to display Players List



| Player Name | Age | Team Name | Type |
|---|---|---|---|
| Pradeep Narwal | 24 | Patna Pirates | Raider |
| Mohit Sehrwat | 29 | Bengaluru Bulls | Defender |
| Pratik Patel | 28 | DABANG DELHI K.C | AllRounder |
| Ankit Beniwal | 24 | Tamil Titans | Raider |
| Amit Kumar | 24 | U MUMBA | Raider |
| Surjeet Singh | 32 | Tamil Titans | Defender |
| Sharabjeet ghosh | 29 | DABANG DELHI K.C | Raider |

**Figure 5.6: Players List**

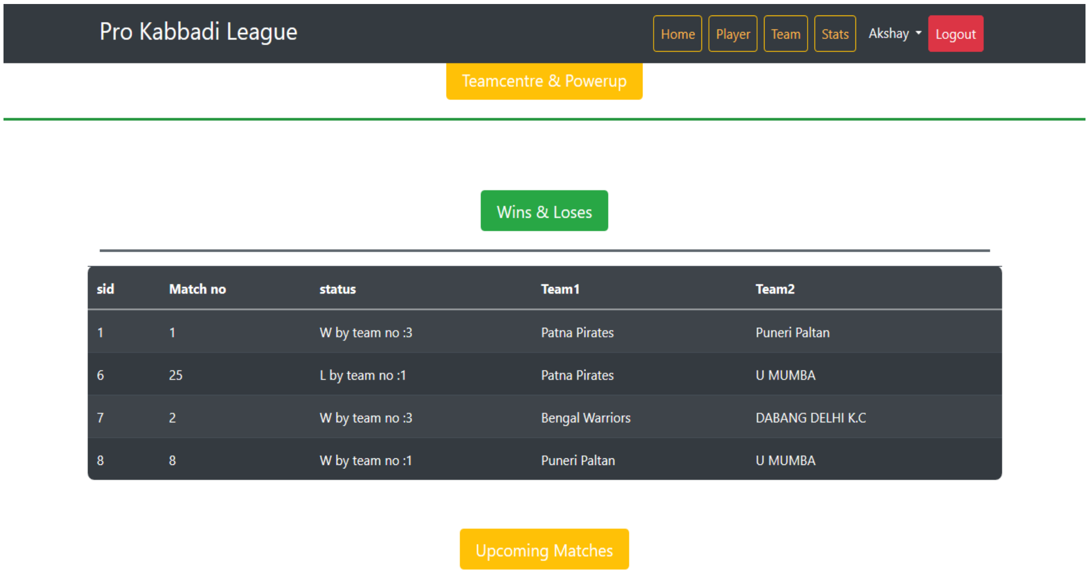
Page to display Player Stats



| PLayer id | Total points | matches played | not out | position | Player name |
|---|---|---|---|---|---|
| 1 | 34 | 10 | 5 | 9 | Pradeep Na |
| 3 | 46 | 7 | 2 | 3 | Pratik Patel |

**Figure 5.7: Players Stats**

Section to display wins & Loses



| sid | Match no | status | Team1 | Team2 |
|---|---|---|---|---|
| 1 | 1 | W by team no :3 | Patna Pirates | Puneri Paltan |
| 6 | 25 | L by team no :1 | Patna Pirates | U MUMBA |
| 7 | 2 | W by team no :3 | Bengal Warriors | DABANG DELHI K.C |
| 8 | 8 | W by team no :1 | Puneri Paltan | U MUMBA |

**Figure 5.8: Wins & Loses**

Search Query

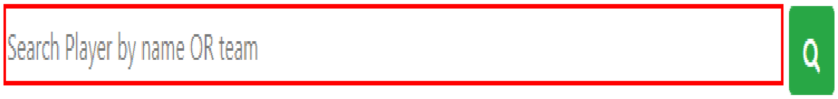
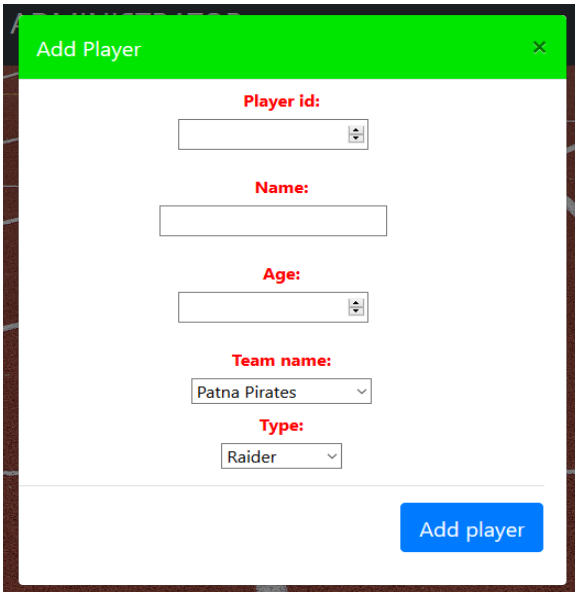


**Figure 5.9: Search Query**

Form to add a new Player



Player id:

Name:

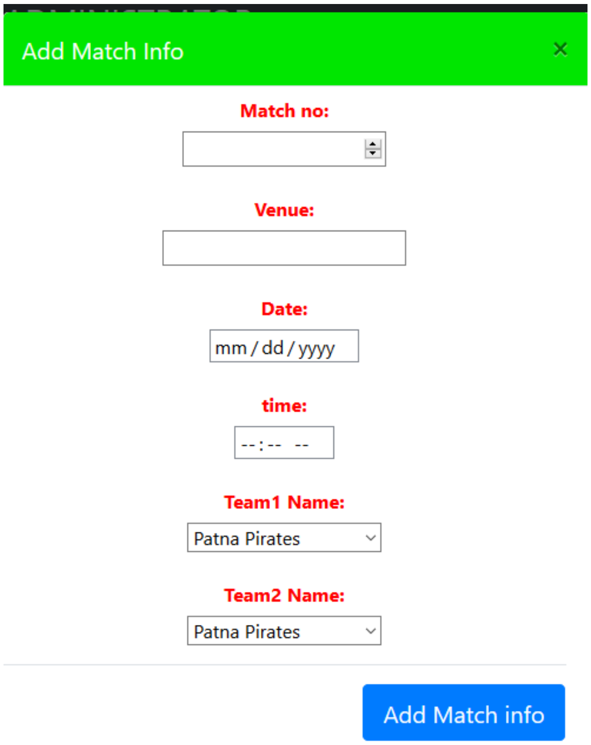
Age:

Team name: Patna Pirates

Type: Raider

**Figure 5.10: Add Player**

Form to add a Match Information



Match no:

Venue:

Date: mm / dd / yyyy

time: --:-- --

Team1 Name: Patna Pirates

Team2 Name: Patna Pirates

**Figure 5.11: Add Match Info**

Form to add a new Team

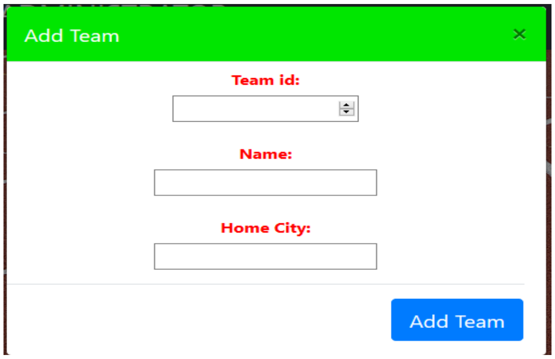
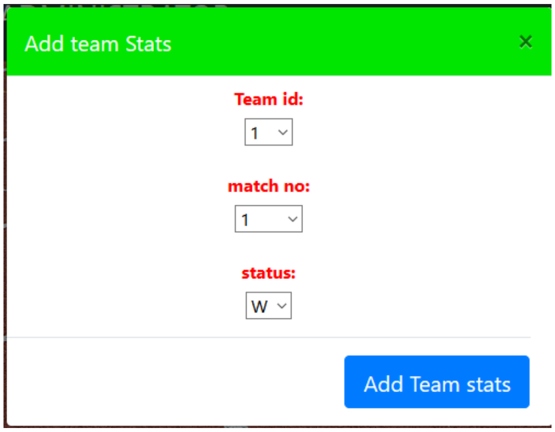


**Figure 5.12: Add Team**

Form to add a MatchStatus



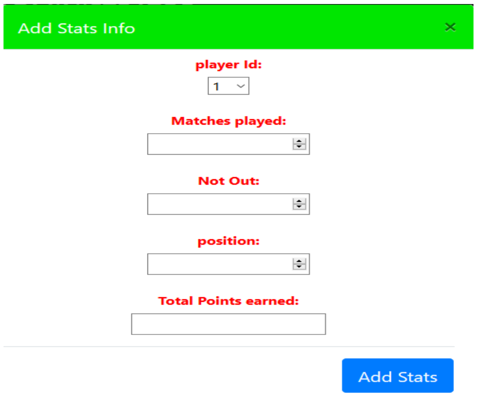
Team id: 1

match no: 1

status: W

**Figure 5.13: Add TeamStatus**

Form to add Player Stats



player Id: 1

**Figure 5.14: Add Player stats**

## Chapter 6

# CONCLUSION AND FUTURE ENHANCEMENT

## Conclusion:

The Pro Kabaddi League DBMS project is designed to meet the requirements of a Pro kabaddi League. The main aim of maintaining the Kabaddi League DBMS data has been achieved in this system. It has been developed in PHP, HTML, CSS keeping in mind the specification of the system.

For designing the system, PHP is being used as the front end in phpMyAdmin Environment. Overall the project teaches us the essential skills like:

- Understanding the database handling and query processing
- Implement, analyze and evaluate the project developed for an application
- Demonstrate the working of different concepts of DBMS

## Scope of Future Enhancement

The Pro Kabaddi League management system designed for the Pro Kabaddi League can sport the following additions in the future through an incremental update:

- Better dashboard design showing more information to the db administator.
- Proving users the option to book tickets on the go whle they are browsing through the list of upcoming matches
- The system design can be tweaked to add Fixtures for every match by a network of Players and experts.
- A data analytics tool can be added to analyze historic data and Match Results using algorithms predict output of a Match to maximize the efficiency of the Pro Kabaddi League DBMS.