

VISVESVARAYA TECHNOLOGICAL UNIVERSITY

BELAGAVI-590018



A Database Management System Mini Project Report

on

"Cricket World Cup Database "

*Submitted in partial fulfillment of the requirements for the V semester
and award of the degree of Bachelor of Engineering in Computer Science
and Engineering of Visvesvaraya Technological University, Belagavi*

Submitted by:

SHREYAS B R 1RN20CS145
SIDDHARTH SHETTAR 1RN20CS150

*Under the Guidance of:
Mrs. SOUMYA N G
Assistant Professor
Dept. of CSE*



Department of Computer Science and Engineering
(NBA Accredited upto June 2025)
RNS Institute of Technology
Channasandra, Dr.Vishnuvardhan Road, Bengaluru-560 098
2022-2023

RNS Institute of Technology

Channasandra, Dr. Vishnuvardhan Road, Bengaluru-560098

DEPARTMENT OF COMPUTER SCIENCE ENGINEERING

(NBA Accredited for academic years 2022-25)



CERTIFICATE

Certified that the mini project work entitled "**CRICKET WORLD CUP DATABASE**" has been successfully carried out by **SHREYAS B R** bearing USN **1RN20CS145** and **SIDDHARTH SHETTAR** bearing USN **1RN20CS150**, bonafide students of **RNS Institute of Technology** in partial fulfillment of the requirements for the 5th semester of "**Bachelor of Engineering in Computer Science and Engineering of Visvesvaraya Technological University**", Belagavi, during academic year 2022-2023. It is certified that all corrections/suggestions indicated for Internal Assessment have been incorporated in the report deposited in the departmental library. The project report has been approved as it satisfies the DBMS laboratory requirements of 5th semester BE, CSE.

Signature of the Guide
Ms. Soumya N G
Assistant Professor
Dept. of CSE

Signature of the HoD
Dr. Kiran P
Professor & HOD
Dept. of CSE

Signature of the Principal
Dr. M K Venkatesha
Principal
RNSIT

External Viva:

Name of the Examiners
1.
2.

Signature with Date

Acknowledgement

Any achievement, be it scholastic or otherwise does not depend solely on the individual efforts but on the guidance, encouragement and cooperation of intellectuals, elders and friends. A number of personalities, in their own capacities have helped us in carrying out this project work. We would like to take this opportunity to thank them all.

We are grateful to Management and **Dr. M K Venkatesha**, Principal, RNSIT, Bangalore, for his support towards completing this mini project.

We would like to thank **Dr. Kiran P**, Professor & Head, Department of Computer Science & Engineering, RNSIT, Bangalore, for his valuable suggestions and expert advice.

We deeply express our sincere gratitude to our guide **Mrs. Soumya N G**, Assistant Professor, Department of CSE, RNSIT, Bangalore, for her able guidance, regular source of encouragement and assistance throughout this project.

We would also like to thank our mentor **Dr. M J Sudhamani**, Assistant Professor, Dept of CSE, for her continuous guidance and support throughout this project.

We would like to thank all the teaching and non-teaching staff of Department of Computer Science & Engineering, RNSIT, Bengaluru-98 for their constant support and encouragement.

Abstract

Cricket World cup Database is a website which we have developed to show users the statistics and various data points of the ongoing world cup. We have tried to make the website design as simple as possible and also let users navigate easily. We have mainly used MYSQL,PHP as the back end and HTML/CSS as our front end. We have basically added two login pages: Admin Login and User Login

As the name suggests, the admin page is for administrators and they are the superuser of the website.

They are allowed to manipulate the data, like add matches, add player details, and also edit the player information. Admins can also schedule matches, cancel matches and calculate match draws and other data points about the tournament. Admins have the privilege to also control the users of the website and also all the players in the tournament. They can also add various new stadiums in the tournament.

User login is a page created for the users of the website in which they can view various information. Users can view the Upcoming Matches, the player list, and the stadium in which the match will be played in. They can also view player profiles and see more about their favourite player. Our database consists mainly of 5 tables namely: Players Table, Matches Table, Stadiums, Results, and Umpires. These tables are relational and contain all the data.

The way the website works is a user first logs in, and a new web page is opened where he can choose to see information about any player in the database. At the top of the web page are the various tables in which he can choose to view.

Admins on the other hand, have options to add players, matches etc available in the database itself. This information will be updated in our MYSQL database.

The main objective of this project was to allow users to see various data points about the tournament so they have better information about the tournament and the players.

This is a website currently and can also be developed into a web-app or a traditional mobile application later on.

Contents

Acknowledgement	i
Abstract	ii
List of Figures	v
List of Tables	vi
1 Introduction	1
1.1 Overview of DBMS	1
1.2 History of DBMS	2
1.3 Applications of DBMS	3
1.3.1 DBMS Languages	3
1.3.2 Design	4
1.3.3 Relational DBMS	4
1.3.4 DBMS Architecture	4
2 MySQL	5
2.1 MySQL Features	5
2.2 User Interfaces	6
2.2.1 Graphical user interfaces	6
2.2.2 MySQL Workbench	6
2.3 phpMyAdmin	6
2.3.1 Triggers Used	7
2.4 Limitations	7
3 Resource Requirements	8
3.1 Hardware Requirements	8

3.2 Software Requirements	8
4 System Design	9
4.1 Schema Diagram:	10
5 Implementation	12
5.0.1 Creating a Connection to a Database:	12
5.0.2 Closing a Connection:	12
5.0.3 Pseudocode for Major Functionalities	13
6 Testing	17
7 Results & Snapshots	18
7.1 Home Page	18
7.2 Admin Login	18
7.3 user Login	19
7.4 Stadiums	19
7.5 Team Ranks	19
7.6 Player Profile	20
7.7 Add Player	20
7.8 Admin Page	20
7.9 User Page	21
8 Conclusion & Future Enhancements	22
8.1 Conclusion	22
8.2 Future Enhancements:	23
References	24

List of Figures

4.1	ER Diagram	9
4.2	Schema Diagram	10
7.1	Home page	18
7.2	Admin Login	18
7.3	home page	19
7.4	Stadium Page	19
7.5	Team Ranks	19
7.6	Player Profile	20
7.7	Add Player	20
7.8	Admin Page	20
7.9	User Page	21

List of Tables

3.1	Hardware Requirements	8
3.2	Software Requirements	8
6.1	Test Case	17

Chapter 1

Introduction

1.1 Overview of DBMS

A database management system (DBMS) is system software for creating and managing databases. A DBMS makes it possible for end users to create, protect, read, update and delete data in a database. The most prevalent type of data management platform, the DBMS essentially serves as an interface between databases and users or application programs, ensuring that data is consistently organized and remains easily accessible. Database engine allows the data to be:

- Accessed
- Locked
- Modified

The database schema defines the database's logical structure. These three foundational elements help provide concurrency, security, data integrity and uniform data administration procedures. The DBMS supports many typical database administration tasks, including change management, performance monitoring and tuning, security, and backup and recovery. Most database management systems are also responsible for automated rollbacks and restarts as well as logging and auditing of activity in databases and the applications that access them.

The DBMS provides a centralized view of data that can be accessed by multiple users from multiple locations in a controlled manner. A DBMS can limit what data end users see and how they view the data, providing many views of a single database schema. End users and software programs are free from having to understand where the data is physically located or on what type of storage medium it resides because the DBMS handles all requests.

1.2 History of DBMS

Charles Bachman was the first person to develop the Integrated Data Store (IDS) which was based on network data model for which he was inaugurated with the Turing Award (The most prestigious award which is equivalent to Nobel prize in the field of Computer Science.). It was developed in early 1960's. In the late 1960's, IBM (International Business Machines Corporation) developed the Integrated Management Systems which is the standard database system used till date in many places. It was developed based on the hierarchical database model. It was during the year 1970 that the relational database model was developed by Edgar Codd. Many of the database models we use today are relational based. It was considered the standardized database model from then.

The relational model was still in use by many people in the market. Later during the same decade (1980's), IBM developed the Structured Query Language (SQL) as a part of R project. It was declared as a standard language for the queries by ISO and ANSI. The Transaction Management Systems for processing transactions was also developed by James Gray for which he was felicitated the Turing Award.

Further, there were many other models with rich features like complex queries, datatypes to insert images and many others. The Internet Age has perhaps influenced the data models much more. Data models were developed using object oriented programming features, embedding with scripting languages like Hyper Text Markup Language (HTML) for queries. With humongous data being available online, DBMS is gaining more significance day by day.

Object databases were developed in the 1980s to overcome the inconvenience of object-relational impedance mismatch, which led to the coining of the term "post-relational" and also the development of hybrid object-relational databases.

The next generation of post-relational databases in the late 2000s became known as NoSQL databases, introducing fast key-value stores and document-oriented databases. A competing "next generation" known as NewSQL databases attempted new implementations that retained the relational/SQL model while aiming to match the high performance of NoSQL compared to commercially available relational DBMS's.

1.3 Applications of DBMS

The applications of DBMS can be divided into four major areas:

- Human Resource Management
- Credit card exchanges
- Manufacturing
- Education Sector

1.3.1 DBMS Languages

DBMS has various languages namely:

- Data Definition Language (DDL): DDL stands for Data Definition Language. It is used to define database structure or pattern. It is used to create schema, tables, indexes, constraints, etc. in the database. Here are some tasks that come under DDL:
 - Create: It is used to create objects in the database.
 - Alter: It is used to alter the structure of the database.
 - Drop: It is used to delete objects from the database.
 - Truncate: It is used to remove all records from a table.
 - Rename: It is used to rename an object.
 - Comment: It is used to comment on the data dictionary.
- Data Manipulation Language(DML): It is used for accessing and manipulating data in a database. It handles user requests. Here are some tasks that come under DML:
 - Select: It is used to retrieve data from a database.
 - Insert: It is used to insert data into a table.
 - Update: It is used to update existing data within a table.
 - Delete: It is used to delete all records from a table.
 - Merge: It performs UPSERT operation, i.e., insert or update operations.
 - Call: It is used to call a structured query language or a Java subprogram.
 - Lock Table: It controls concurrency.
- Data Control Language(DCL): It is used to retrieve the stored or saved data.

1.3.2 Design

Professions such as engineering and architecture are concerned with design. They start with a set of specification; seek cost-effective solutions that satisfy the specification. Designing is an iterative process. Designer generates a possible design, tests it and then uses the results as the basis for exploring other solutions. The use of interactive graphical tools in PHP MyAdmin pervades the fields including architecture, computer engineering, and the design of large scale database systems.

1.3.3 Relational DBMS

Everything in a relational database is stored in the form of relations. The RDBMS database uses tables to store data. A table is a collection of related data entries and contains rows and columns to store data. Each table represents some real-world objects such as person, place, or event about which information is collected. The organized collection of data into a relational table is known as the logical view of the database. Properties of a Relation:

- Each relation has a unique name by which it is identified in the database.
- The tuples of a relation have no specific order.
- All attributes in a relation are atomic, i.e., each cell of a relation contains exactly one value.

1.3.4 DBMS Architecture

The DBMS design depends upon its architecture. The basic client/server architecture is used to deal with a large number of PCs, web servers, database servers and other components that are connected with networks. The client/server architecture consists of many PCs and a workstation which are connected via the network. DBMS architecture depends upon how users are connected to the database to get their request done.

Chapter 2

MySQL

MySQL is an open-source relational database management system (RDBMS). Its name is a combination of "My", the name of co-founder Michael Widenius's daughter My, and "SQL", the acronym for Structured Query Language. A relational database organizes data into one or more data tables in which data may be related to each other; these relations help structure the data. SQL is a language programmers use to create, modify and extract data from the relational database, as well as control user access to the database. In addition to relational databases and SQL, an RDBMS like MySQL works with an operating system to implement a relational database in a computer's storage system, manages users, allows for network access and facilitates testing database integrity and creation of backups.

MySQL is written in C and C++. Its SQL parser is written in yacc, but it uses a home-brewed lexical analyzer. MySQL works on many system platforms, including AIX, BSDi, FreeBSD, HP-UX, ArcaOS, eComStation, IBM i, IRIX, Linux, macOS, Microsoft Windows, NetBSD, Novell NetWare, OpenBSD, OpenSolaris, OS/2 Warp, QNX, Oracle Solaris, Symbian, SunOS, SCO OpenServer, SCO UnixWare, Sanos and Tru64. A port of MySQL to OpenVMS also exists.

The MySQL server software itself and the client libraries use dual-licensing distribution. They are offered under GPL version 2, or a proprietary license.

2.1 MySQL Features

MySQL is offered under two different editions: the open source MySQL Community Server and the proprietary Enterprise Server. MySQL Enterprise Server is differentiated by a series of proprietary extensions which install as server plugins, but otherwise shares the version numbering system and is built from the same code base.

- A broad subset of ANSI SQL 99, as well as extensions
- Cross-platform support
- Triggers
- Cursors
- Updatable views

OpenGL User Interface Library (GLUI) is a C++ user interface library based on the OpenGL Utility Toolkit (GLUT) which provides controls such as buttons, checkboxes, radio buttons, and spinners to OpenGL applications. It is window and operating system independent, relying on GLUT to handle all system-dependent issues, such as window and mouse management. The OpenGL Utility Library (GLU) is a computer graphics library. It consists of a number of functions that use the base OpenGL library to provide higher-level drawing routines from the more primitive routines that OpenGL provides. It is usually distributed with the base OpenGL package.

2.2 User Interfaces

2.2.1 Graphical user interfaces

A graphical user interface (GUI) is a type of interface that allows users to interact with electronic devices or programs through graphical icons and visual indicators such as secondary notation, as opposed to text-based interfaces, typed command labels or text navigation.

2.2.2 MySQL Workbench

MySQL Workbench is the integrated environment for MySQL. It was developed by MySQL AB, and enables users to graphically administer MySQL databases and visually design database structures.

2.3 phpMyAdmin

We have used PHPMyadmin as our database interface. It is a GUI and each table is a separate entity. We can define the Primary Key and Foreign Key also in this phpMyAdmin. We can also set triggers to invoke procedures when a certain event occurs.

2.3.1 Triggers Used

- Stadium Table Trigger:

Code:

```
CREATE TRIGGER 'default\_date' BEFORE INSERT ON 'stadiums'  
FOR EACH ROW set new.DOI=CURRENT\_DATE()
```

- Players Table Trigger:

Code:

```
CREATE TRIGGER 'update' BEFORE UPDATE ON 'player'  
FOR EACH ROW BEGIN  
IF (new.runs<old.runs) THEN SET new.runs=old.runs;  
IF (new.wickets<old.wickets) THEN SET new.wickets=old.wickets;  
END IF;  
IF (new.no\_of\_matches<old.no\_of\_matches) THEN SET new.no\_of\_matches=old.no\_of\_matches;  
END IF;  
END IF;  
END
```

2.4 Limitations

- MySQL is a great database for small volumes - up to 100 GB.
- Not more than 16 keys in an index.
- Less stable than most other sequential databases

Chapter 3

Resource Requirements

3.1 Hardware Requirements

The Hardware requirements are very minimal and the program can be run on most of the machines. Table 3.1 gives details of hardware requirements.

Table 3.1: Hardware Requirements

Processor	Intel Core i3 processor
Processor Speed	1.70 GHz
RAM	4 GB
Storage Space	40 GB
Monitor Resolution	1024*768 or 1336*768 or 1280*1024

3.2 Software Requirements

The software requirements are description of features and functionalities of the system. Table 3.2 gives details of software requirements.

Table 3.2: Software Requirements

Operating System	Windows 7
IDE	Microsoft Visual Studio with C++ 2022
MySQL Software	MySQL Workbench 8.0 phpMyAdmin 5.2.0

Chapter 4

System Design

ER Diagram: An entity–relationship model is usually the result of systematic analysis to define and describe what is important to processes in an area of a business. An E-R model does not define the business processes; it only presents a business data schema in graphical form.

It is usually drawn in a graphical form as boxes (entities) that are connected by lines (relationships) which express the associations and dependencies between entities. An ER model can also be expressed in a verbal form, for example: one building may be divided into zero or more apartments, but one apartment can only be located in one building. Entities may be characterized not only by relationships, but also by additional properties (attributes), which include identifiers called "primary keys". Diagrams created to represent attributes as well as entities and relationships may be called entity-attribute-relationship diagrams, rather than entity-relationship models. An ER model is typically implemented as a database.

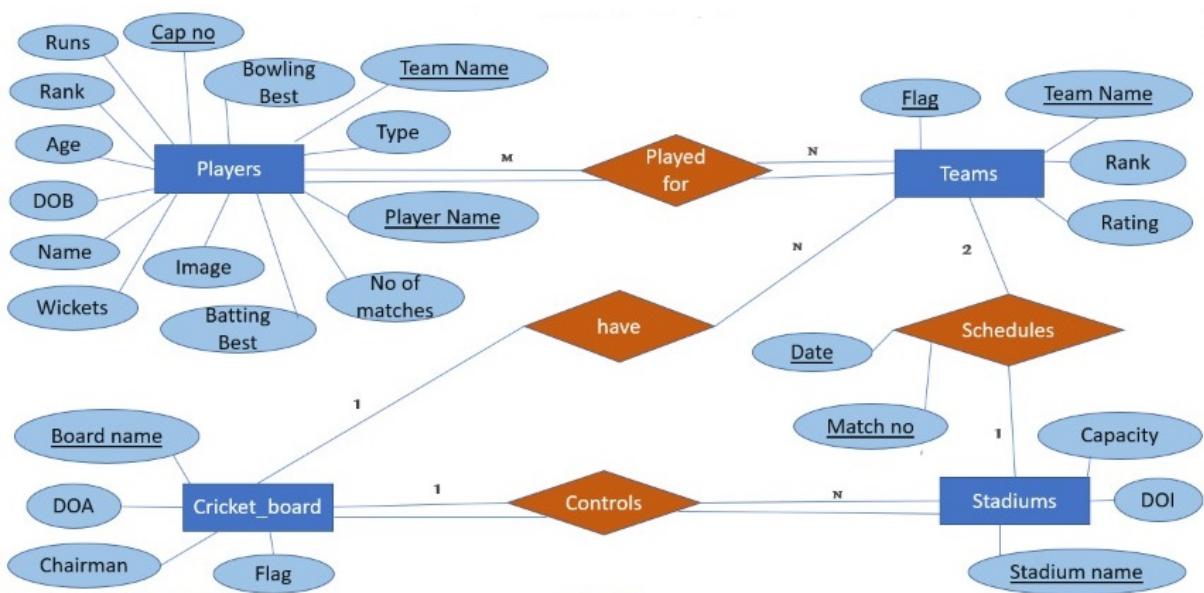


Figure 4.1: ER Diagram

The four main cardinal relationships are:

- One-to-one (1:1):

For example, each customer in a database is associated with one mailing address.

- One-to-many (1: N):

For example, a single customer might place an order for multiple products. The customer is associated with multiple entities, but all those entities have a single connection back to the same customer.

- Many-to-one (N: 1):

One-to-Many relationship in DBMS is a relationship between instances of an entity with more than one instance of another entity.

- Many-to-many (M: N):

A many-to-many relationship occurs when multiple records in a table are associated with multiple records in another table.

4.1 Schema Diagram:

The term "schema" refers to the organization of data as a blueprint of how the database is constructed (divided into database tables in the case of relational databases). The formal definition of a database schema is a set of formulas (sentences) called integrity constraints imposed on a database.

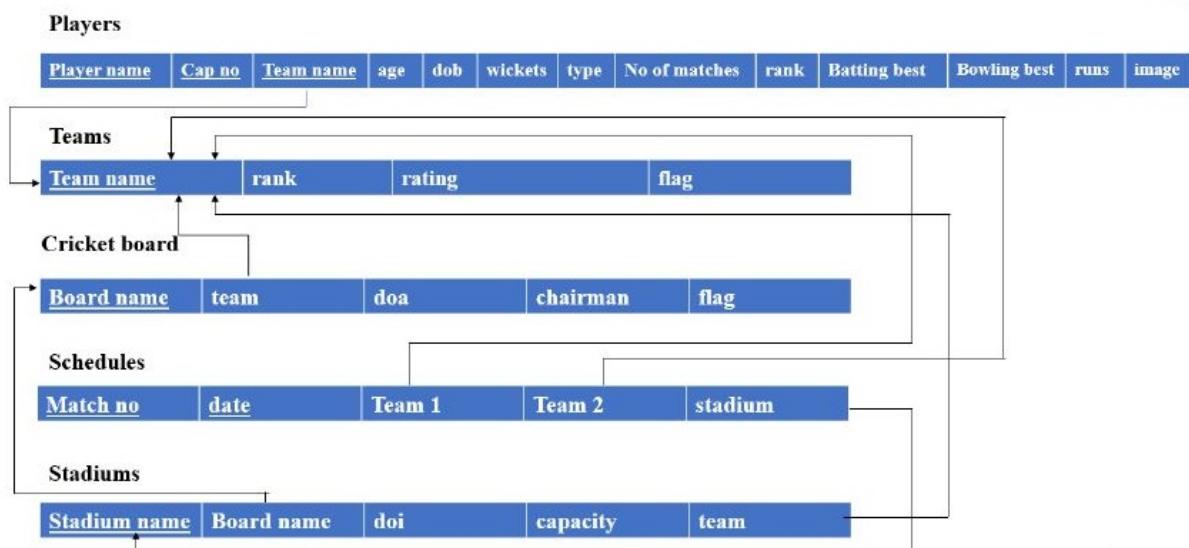


Figure 4.2: Schema Diagram

A relational schema shows references among fields in the database. When a primary key is referenced in another table in the database, it is called a foreign key. This is denoted by an arrow

with the head pointing at the referenced key attribute. A schema diagram helps organize values in the database. It also gives an idea of what order the tables should be created in. The following diagram shows the schema diagram for the database.

Chapter 5

Implementation

Implementation is the stage in the project where the theoretical design is turned into a working system and is giving confidence on the new system for the users that it will work efficiently and effectively. It involves careful planning, investigation of the current system and its constraints on implementation, design of methods to achieve the changeover, an evaluation of change over methods. Implementation is the most important phase. The most critical stage in achieving a successful new system is giving the users confidence that the new system will work and be effective. Any system developed should be secured and protected against possible hazards. Security measures are provided to prevent unauthorized access of the database at various levels. Password protection and simple procedures to prevent the unauthorized access are provided to the users. The system allows the user to enter the system only through proper user name and password.

5.0.1 Creating a Connection to a Database:

Before you can access data in a database, you must create a connection to the database. In PHP, this is done with the `getconnection()` function.

```
<?php $con=MySQL_connect("localhost","root","","cricket");?>
```

5.0.2 Closing a Connection:

The connection will be closed automatically when the script ends. To close the connection before, use the `close()` function:

```
<?php mysqli\_close(\$con);?>
```

5.0.3 Pseudocode for Major Functionalities

Sample for Create Table:

```
CREATE TABLE 'team' (
    'name' varchar(30) NOT NULL,
    'rank' decimal(5,0) NOT NULL,
    'rating' int(11) NOT NULL,
    'flag' varchar(40) NOT NULL
);
```

Insert Functionality:

```
<?php

session_start();

$con = mysqli_connect("localhost", "root", "", "cricket") or die(mysqli_error($con))

$capno=$_POST['cap_no'];
$playername=$_POST['playername'];
$age=$_POST['age'];
$dob=$_POST['dob'];
$runs=$_POST['runs'];
$wickets=$_POST['wickets'];
$type=$_POST['type'];
$num=$_POST['no_of_matches'];
$rank=$_POST['rank'];
$batting=$_POST['batting_best'];
$bowling=$_POST['bowling_best'];
$image=$_POST['image'];
$teamname=$_POST['team'];

$query="insert into player (cap_no,age,dob,runs,wickets,type,no_of_matches,rank,batting,bowling,image,teamname) values ($capno,$age,$dob,$runs,$wickets,$type,$num,$rank,$batting,$bowling,$image,'$teamname')";

if(mysqli_query($con,$query))
{
```

```

echo "<script type='text/javascript'>alert('NEW RECORD CREATED SUCCESSFULLY!!');
header("refresh: 0.01; url=admin1st.html");
}

else
{
echo "<script type='text/javascript'>alert('ERROR');</script>";
header("refresh: 0.01; url=addplayeroption.php");
mysqli_error($con);
}

?>

```

Delete Functionality:

```

<?php
$con = mysqli_connect("localhost", "root", "", "cricket") or die(mysqli_error($con))

$player=$_POST['playername'];
$sql="delete from player where playername like '$player'";
if(mysqli_query($con,$sql)==true){
$sql1 = "SELECT * from player where playername like'$player'";
$res=mysqli_query($con,$sql1);
if(mysqli_num_rows($res)==0)
{
echo "<script type='text/javascript'>alert('player deleted successfully!!');</script>";
header("refresh: 0.01; url=admin1st.html");

}
else{
echo "<script language='javascript'>";
echo "alert('ERROR IN DELETING')";
echo "</script>";
header("refresh: 0.01; url=deleteplayer1.php");
}
}

```

```

}

mysqli_close($con);

?>

```

Update Functionality:

```

<?php

session_start();

$name=$_POST['name'];

$runs=$_POST['runs'];

$wickets=$_POST['wickets'];

$no_of_matches=$_POST['no_of_matches'];

$con = mysqli_connect("localhost", "root", "", "cricket") or die(mysqli_error($con))

$q1="select runs as oldruns , wickets as oldwickwts , no_of_matches as oldmatches

if($result2=mysqli_query($con,$q1))
{
$row=mysqli_fetch_array($result2);

$runs=$runs+$row["oldruns"];
$wickets=$wickets+$row["oldwickwts"];
$no_of_matches=$no_of_matches+$row["oldmatches"];

$q="update player set runs='$runs',wickets='$wickets',no_of_matches='$no_of_matches'

if(mysqli_query($con,$q))
{
header("refresh: 0.01; url=rank.php");
}
}

```

```
}

else

{echo "<script type='text/javascript'>alert('ERROR!!111');</script>";
 header("refresh: 0.01; url=rank.php");
}

}

else

{echo "<script type='text/javascript'>alert('ERROR!!2222');</script>";
 header("refresh: 0.01; url=rank.php");
}

?>
```

Chapter 6

Testing

In unit testing, the program modules that make up the system are tested individually. Unit testing focuses to locate errors in the working modules that are independent to each other. This enables to detect errors in coding and the logic within the module alone. This testing is also used to ensure the integrity of the data stored. The various routines were checked by passing the inputs and the corresponding output is tested. Table 6.1 gives details of validation. Test cases used in the project as follows:

Table 6.1: Test Case

Test Case No.	Metric	Description	Observation
1.	Admin Function	Admin can login by clicking on admin login button from index.html page and then enter valid username and password.	It will redirect to the admin page.
2	User Function	User can login by clicking on user login button from index.html page and then enter valid username and password	It will redirect to the User page.

Chapter 7

Results & Snapshots

7.1 Home Page

This will have the option to choose to login as a user or admin.



Figure 7.1: Home page

7.2 Admin Login

Admin can login from this page.

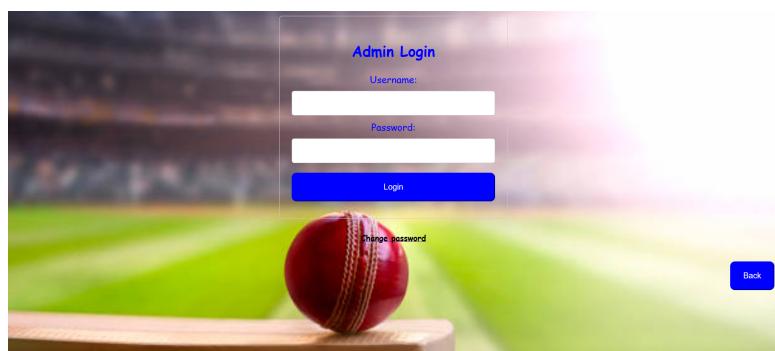


Figure 7.2: Admin Login

7.3 user Login

This is the user login page.

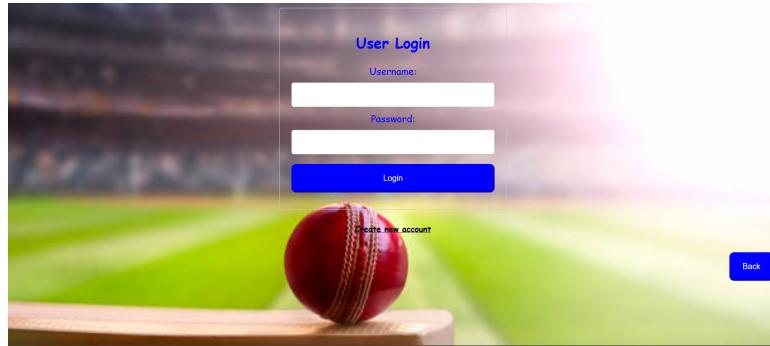


Figure 7.3: home page

7.4 Stadiums

This will show all the various stadiums that the matches will be played at the tournament.

STADIUM NAME	CAPACITY	DOI	BOARD NAME	TEAM'S STADIUM
CHINNASWAMY	45000	2023-01-11	BOARD OF CONTROL FOR CRICKET IN INDIA	INDIA
EDEN GARDEN STADIUM	70000	2023-01-09	BOARD OF CONTROL FOR CRICKET IN INDIA	INDIA
RATEEV GANDHI INTERNATIONAL STADIUM	55000	2023-01-09	BOARD OF CONTROL FOR CRICKET IN INDIA	INDIA
WANKADE	45000	2023-01-09	BOARD OF CONTROL FOR CRICKET IN INDIA	INDIA

Figure 7.4: Stadium Page

7.5 Team Ranks

This page will show the ranks of the team that is participating in the tournament.

RANK	NAME	RATING
1	INDIA	150
2	Pakistan	125
3	Australia	110
4	England	100
5	South Africa	95
6	New Zealand	90
7	Sri Lanka	85

Figure 7.5: Team Ranks

7.6 Player Profile

This section shows us all the details of the player which is selected in the database.



Figure 7.6: Player Profile

7.7 Add Player

The admin can add a player through this page in the database.

A screenshot of an 'Add Player' form. It has fields for Cap Number, Player Name, Age, DOB, Runs, and Wickets. Each field has a placeholder text and a small input field.

Figure 7.7: Add Player

7.8 Admin Page

Admin has all the various options to Add or Delete a player and also add stadiums and alter the rankings of the various teams present.

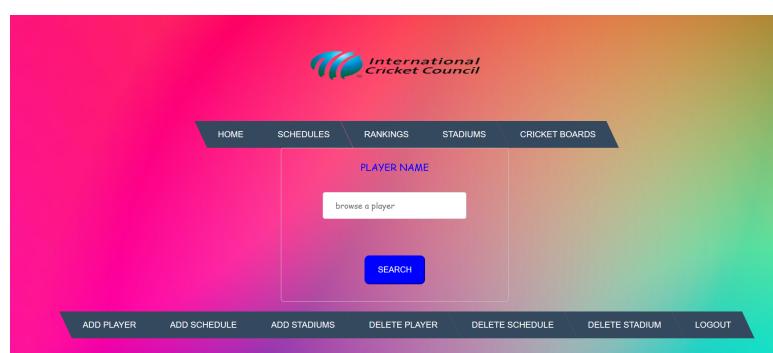


Figure 7.8: Admin Page

7.9 User Page

The user can choose which information to see from this page.



Figure 7.9: User Page

Chapter 8

Conclusion & Future Enhancements

8.1 Conclusion

The project, developed using PHP and MySQL is based on the requirement specification of the user and the analysis of the existing system, with flexibility for future enhancement.

The expanded functionality of today's software requires an appropriate approach towards software development.

This Cricket world database management software is designed for people who want to manage various particulars can be known by recording them in the database. various records and particulars about match got increased rapidly.

Thereby the numbers of matches and there is going to be increased day-by-day. And hence there is a lot of strain on the person who are watching the Cricket world to know about future matches and also to see the records done by various players and getting details in fingertips.

Identification of the drawbacks of the existing system leads to the designing of computerized system that will be compatible to the existing system with the system which is more user friendly and more GUI oriented.

8.2 Future Enhancements:

The current project is just based on taking the information and storing in respective data tables and representing the information in the different required forms and has the ability to search using the attribute. There are some enhancements which can be implemented further. They are as follows:

- Module that automatically gives the information about various cricket boards, stadiums and also rankings of various IPL teams and players by selecting or entering the relevant required item.
- Module that gives information about completely filled and partially filled information on various particulars.
- Can create module such that the user can login and gain information through the window.
- Make this project to the scope for players also.

References

- [1]] Ramez Elmasri and Shamkant B. Navathe, “Database systems Models, Languages, Design and Application Programming”,6th Edition, Pearson,2017.
- [2]] Ramakrishnan and Gehrke, “Database management systems”, 3 rd Edition McGraw Hill 2014
- [3] Silberschatz Korth and Sudharshan, “Database System Concepts”, 6th Edition Mc-Graw Hill, 2013.
- [4] Coronel, Morris, and Rob “Database Principles Fundamentals of Design, Implementation and Management” Cengage Learning 2012.
- [5] <https://stackoverflow.com/questions/1651806/sql-server-trigger>
- [6] https://www.w3schools.com/mysql/mysql_rdbms.asp
- [7] <https://devdocs.io/html/element/article>
- [8] <https://dev.mysql.com/doc/workbench/en/wb-data-modeling.html>