VISVESVARAYA TECHNOLOGICAL UNIVERSITY

BELAGAVI-590018, KARNATAKA

A MINI PROJECT REPORT
ON

“FIFA 18 PLAYER MANAGEMENT SYSTEM”

Submitted in partial fulfillment of requirements for the award of 5th semester,

BACHELOR OF ENGINEERING
IN
COMPUTER SCIENCE & ENGINEERING

Submitted By:
Praveen kumar
(USN: 1MJ16CS105)

Under the Guidance of
Mrs. G . Sivagama Sundari
Associate Professor,
Department of Computer Science & Engineering

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
MVJ COLLEGE OF ENGINEERING
BENGALURU-67

2018-19

MVJ COLLEGE OF ENGINEERING

Channasandra, Near ITPB, Bangalore-67

Department of Computer Science and Engineering

Certificate

This is to certify that the mini project entitled **"FIFA 18 Player Management System”** is a bona fide work carried out by **Praveen Kumar (1MJ16CS105),** a bona fide student of MVJ College of Engineering, in partial fulfillment for the award of degree of Bachelor of Engineering in Computer Science & Engineering of the Visvesvaraya Technological University, Belagavi during the year 2018-19. It is certified that all the corrections/suggestions indicated for Internal Assessment have been incorporated in the Report. The mini Project Report has been approved as it satisfies the academic requirements in respect of Project work prescribed for the said degree.

Signature of the Guide **Signature of the HOD**

Signature of the Examiners

....

Internal **External**

ABSTRACT

Managing the ever increasing numbers of players in different parts of the world is a huge task. This project is aimed at developing a desktop-based application named ‘FIFA 18 player management system’ for managing players using a robust database at the backend and a Web based GUI at the frontend.

The application will allow users to track complete details about a player starting from his personal details, going through club and nationality information to right down to his technicalities at each position in footballing world. The software also allows users to view the whole list of players, teams and footballing statistics at once, thereby helping them build their perspective. Users have the privilege to add new players to a particular team, and to modify their records when the player decides to retire. FIFA 18 player management system also allows users to access players based on their rating other than their preferential position of playing thus guiding managers to build a strong positional team by selecting best rated player at each position. In conclusion, this application will come extremely handy in maintaining player spread across different teams and nations.

ACKNOWLEDGEMENT

The satisfaction and euphoria that accompany the successful completion of any task would be incomplete without the mention of the people who made it possible, whose constant guidance and encouragement crowned our effort with success.

I express my sincere gratitude to our Principal **Dr. Nagaraj Sitaram**, & Vice Principal **Mrs. M. Brindha**, MVJ College of Engineering for providing facilities.

I wish to place on record my grateful thanks to **Mrs. I Manimozhi**, Head of the Department, Computer Science and Engineering, MVJ College of Engineering, Bangalore for providing encouragement and guidance.

I consider it a privilege and honour to express my sincere gratitude to my guide **Mrs. G Sivagama Sundari,** Associate Professor, Department of Computer Science & Engineering for their valuable guidance throughout the tenure of this seminar work and whose support and encouragement made this work possible.

I wish to thank the faculty of Computer Science and Engineering department whose suggestions have enabled me to surpass many of the seemingly impossible hurdles.

Thank you.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1

INTRODUCTION

The project titled "**FIFA 18 Player management system**" is player management software for monitoring and accessing players based on their FIFA 18 PC/XBOX Game ratings. This project is developed using HTML, CSS and JavaScript for front-end and PHP, MySQL for back-end, which focuses on basic operation like adding a new player, new statistics, searching players with detailed information and edit as they grow their skills.

This project is a web based application designed and developed to help user's access players and organize teams. This software is easy to use, and it features a familiar and well-thought-out attractive user interface, combined with strong searching, insertion, and deletion with procedure capabilities.

Analysing players have been a huge task performed by professional scouting agents who are spread around the world. From personal details to football technicalities, FIFA 18 Management system allows easy maintenance record of such skilled youth talent.

Chapter 2

SYSTEM REQUIREMENT

One of the most difficult tasks is that, the selection of the software, once system requirement is known is determining whether a software package fits the requirements. After initial selection further security is needed to determine the desirability of software compared with other candidates. This section first summarizes the application requirement question and then suggests more detailed comparisons.

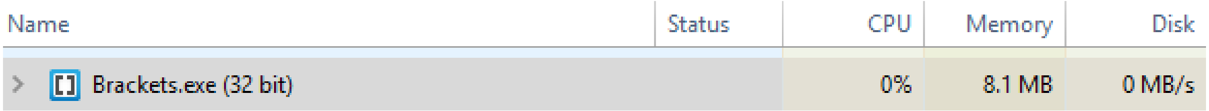
Hardware Requirement

1. 32/64-bit processor
2. i3 or greater intel processor chip
3. 1.7 or more GHz processor

Software Requirement

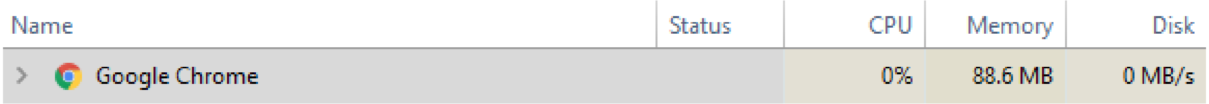
1. Windows 7 or higher version OS
2. Google chrome v70.0.3538 or greater
3. WAMPP web server
4. Brackets web editor

2.1 Software and Executables memory size:



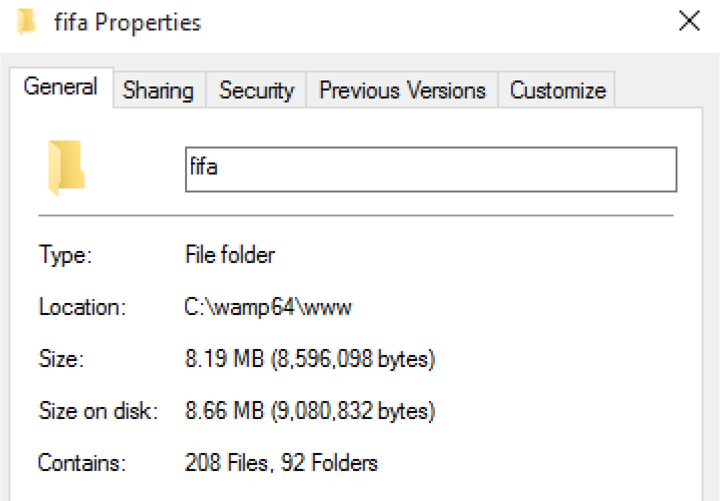
Name	Status	CPU	Memory	Disk
Brackets.exe (32 bit)		0%	8.1 MB	0 MB/s

Fig 2.1 Memory consumption by brackets web editor



Name	Status	CPU	Memory	Disk
Google Chrome		0%	88.6 MB	0 MB/s

Fig 2.2 Memory consumption by chrome

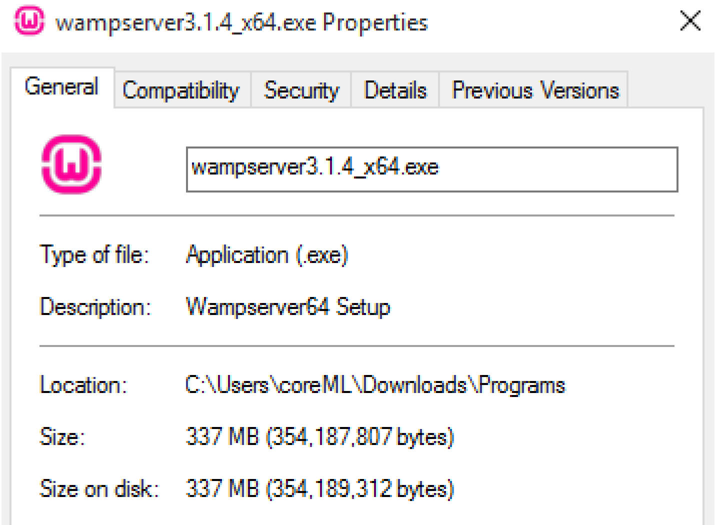


fifa Properties

fifa

Type:	File folder
Location:	C:\wamp64\www
Size:	8.19 MB (8,596,098 bytes)
Size on disk:	8.66 MB (9,080,832 bytes)
Contains:	208 Files, 92 Folders

Fig 2.3 Project size on disk

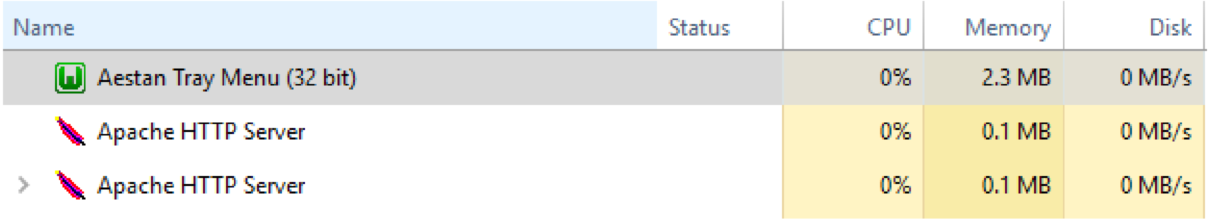


wampserver3.1.4_x64.exe Properties

wampserver3.1.4_x64.exe

Type of file:	Application (.exe)
Description:	Wampserver64 Setup
Location:	C:\Users\coreML\Downloads\Programs
Size:	337 MB (354,187,807 bytes)
Size on disk:	337 MB (354,189,312 bytes)

Fig 2.4 Wamp web server size on disk



Name	Status	CPU	Memory	Disk
Aestan Tray Menu (32 bit)		0%	2.3 MB	0 MB/s
Apache HTTP Server		0%	0.1 MB	0 MB/s
Apache HTTP Server		0%	0.1 MB	0 MB/s

Fig 2.5 Memory consumption by local server (Wamp)

Chapter 3

DESIGN OF THE PROJECT

To analyse and access players on a daily basics can be hectic and huge effort demanding task. To help users, managers, staffs and scouting agents in football world, FIFA 18 player management system provides effortless player management system to help users to analyse, improve, train and access plays on a daily basis.

Description of the Project:

This project consists of player details, which describes about player biodata such as age and nationality. It also consists of player stats which describes about players technical skills. It also consists of tables containing details such as player earnings, club information and preferred position of playing. It also provide a strong searching, updating, deleting and inserting operations with a user friendly web based UI.

The project also helps the users to keep track of the player details in a computerized way without any trouble. The project contains **7 stored procedures** and 3 triggers per table. Stored procedures are used in search engine. Every time the user searches through the database, a procedure is called and the results is collected and displayed for the user in a structured manner. It also has 3 trigger namely "**Insert, Delete and Update**" triggers assigned separately to each table. Whenever operations such as insert or delete or update is performed on any table, these triggers are automatically called, and the logs are captured into 3 separate tables, individually for each trigger. Hence use of triggers provides users to trace back all the latest as well as the oldest changes into any table at any point of time.

This project is a simple prototype of managing larger numbers of players across different nations with different skill sets and attributes. It helps to access players and thus aids in building a strong positional team. It also helps in monitoring player growth.

3.1 ER Diagram:

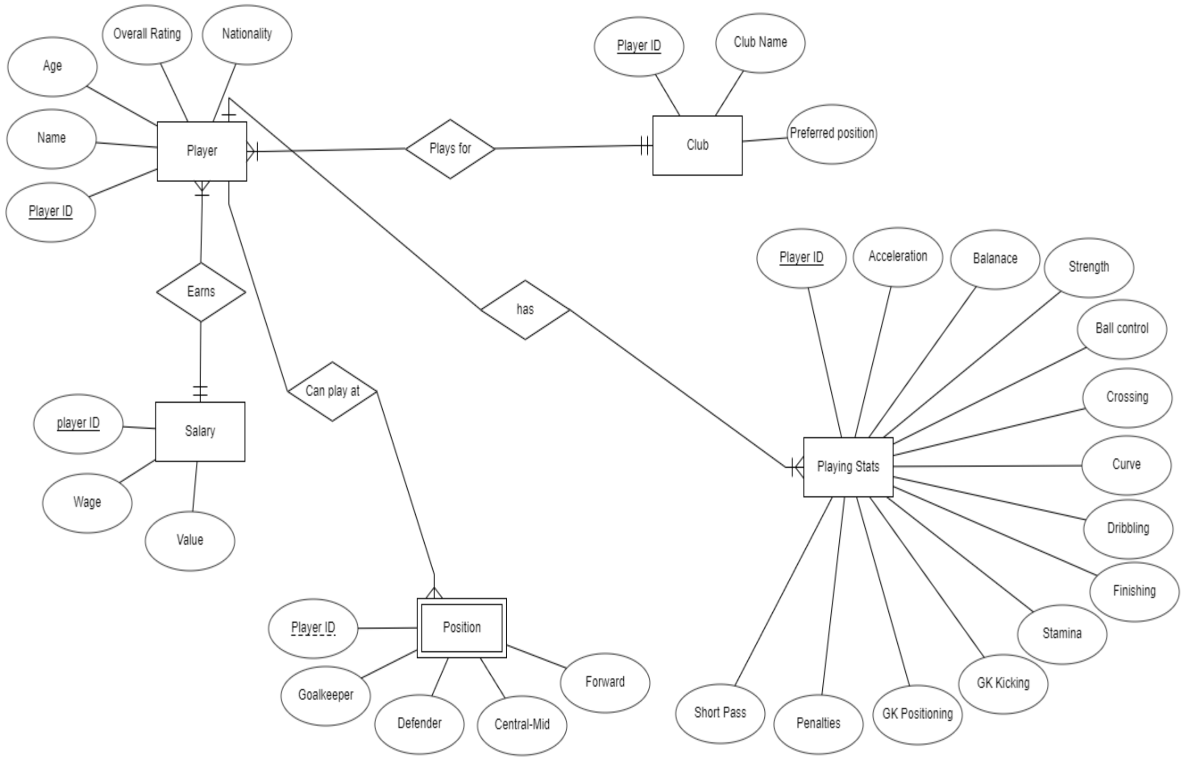
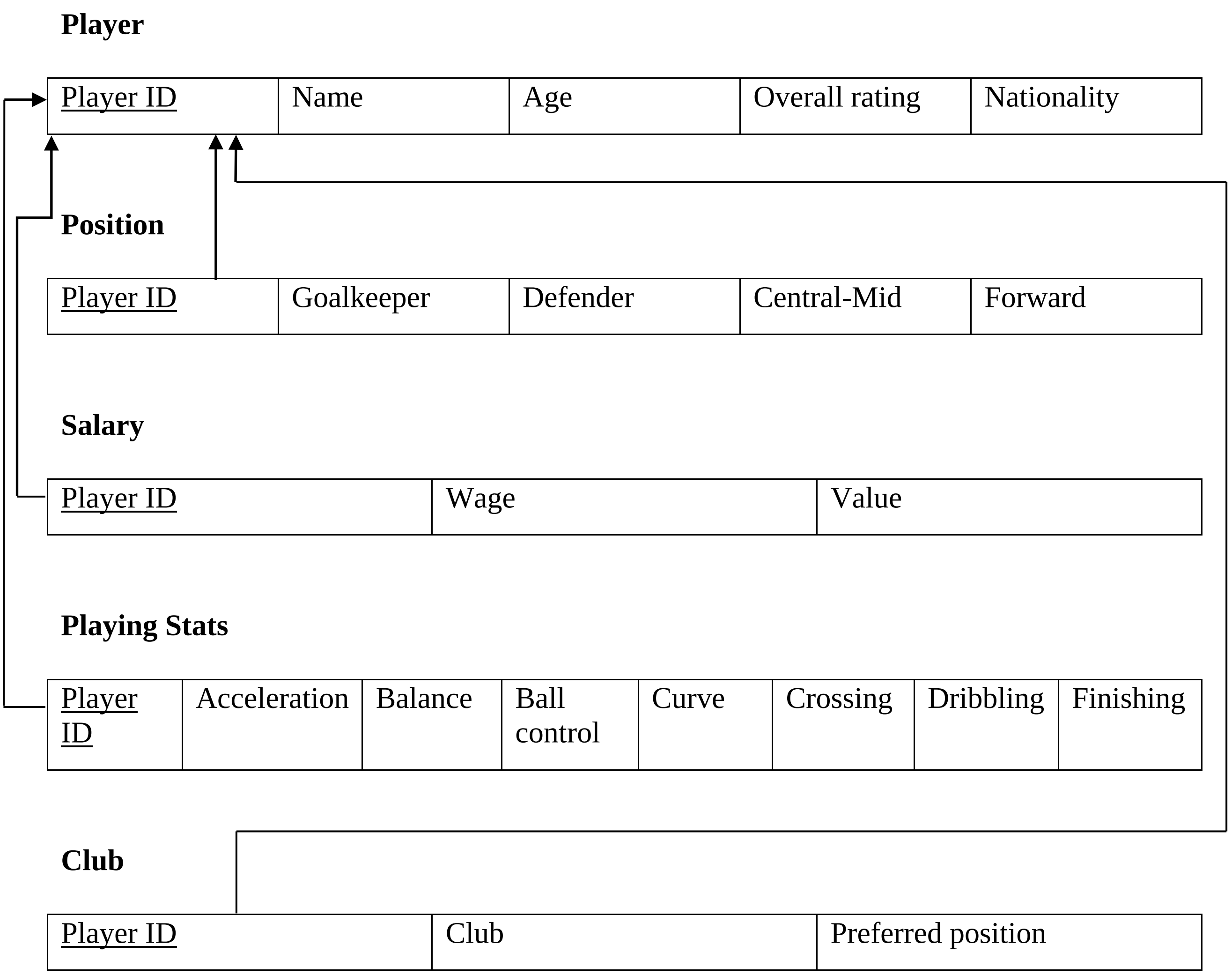


Fig 3.1 ER Diagram

3.2 Schema Diagram:



Player

Player ID	Name	Age	Overall rating	Nationality

Position

Player ID	Goalkeeper	Defender	Central-Mid	Forward

Salary

Player ID	Wage	Value

Playing Stats

Player ID	Acceleration	Balance	Ball control	Curve	Crossing	Dribbling	Finishing

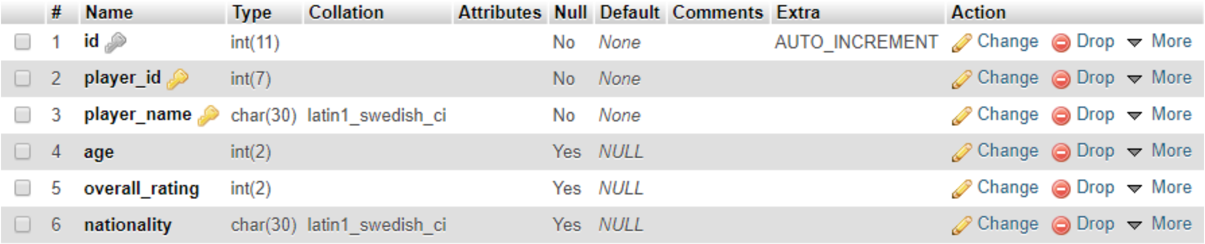
Club

Player ID	Club	Preferred position

Fig 3.2 Relational schema

3.3 Table Structures:

3.3.1 Player details

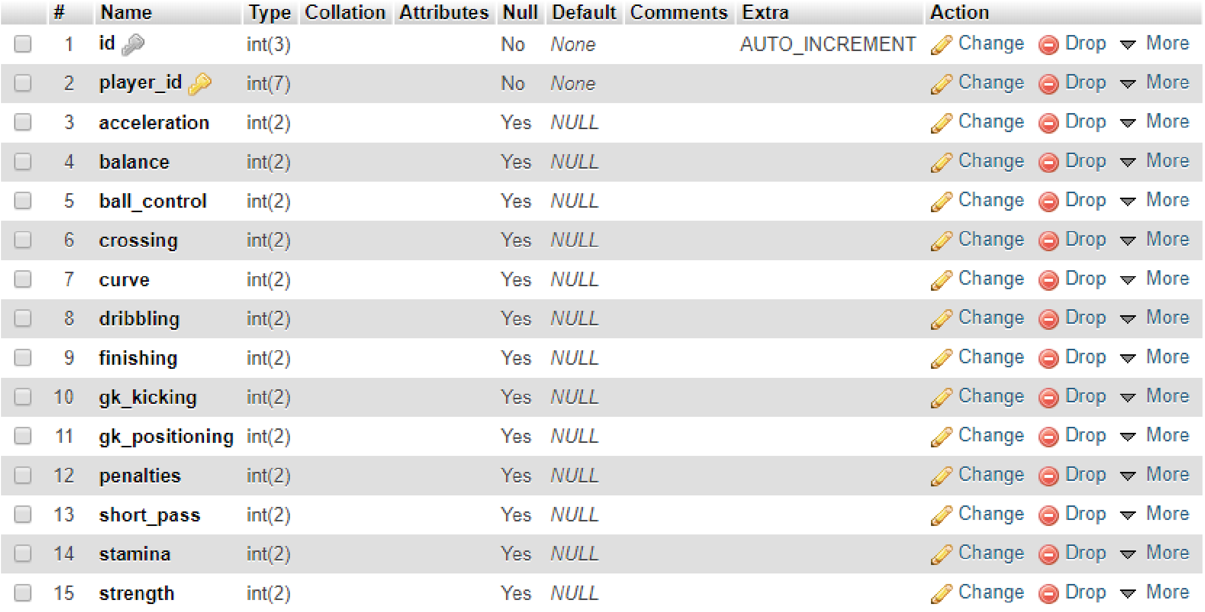


	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
□	1	**id** 🔑	int(11)			No	*None*		AUTO_INCREMENT	✎ Change ⊖ Drop ▾ More
□	2	**player_id** 🔑	int(7)			No	*None*			✎ Change ⊖ Drop ▾ More
□	3	**player_name** 🔑	char(30)	latin1_swedish_ci		No	*None*			✎ Change ⊖ Drop ▾ More
□	4	**age**	int(2)			Yes	*NULL*			✎ Change ⊖ Drop ▾ More
□	5	**overall_rating**	int(2)			Yes	*NULL*			✎ Change ⊖ Drop ▾ More
□	6	**nationality**	char(30)	latin1_swedish_ci		Yes	*NULL*			✎ Change ⊖ Drop ▾ More

Fig 3.3 Player table structure

The player table consists of 6 columns. Player_id and player_name are primary keys and player_id have references of other tables as well. To insert data, player_id should exist in this table before inserting into other tables.

3.3.2 Stats details

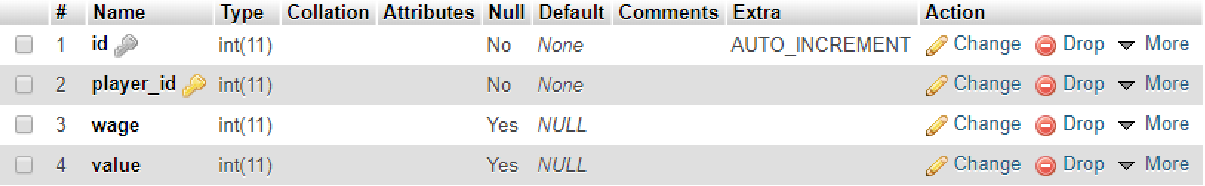


	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
□	1	**id** 🔑	int(3)			No	*None*		AUTO_INCREMENT	✎ Change ⊖ Drop ▾ More
□	2	**player_id** 🔑	int(7)			No	*None*			✎ Change ⊖ Drop ▾ More
□	3	**acceleration**	int(2)			Yes	*NULL*			✎ Change ⊖ Drop ▾ More
□	4	**balance**	int(2)			Yes	*NULL*			✎ Change ⊖ Drop ▾ More
□	5	**ball_control**	int(2)			Yes	*NULL*			✎ Change ⊖ Drop ▾ More
□	6	**crossing**	int(2)			Yes	*NULL*			✎ Change ⊖ Drop ▾ More
□	7	**curve**	int(2)			Yes	*NULL*			✎ Change ⊖ Drop ▾ More
□	8	**dribbling**	int(2)			Yes	*NULL*			✎ Change ⊖ Drop ▾ More
□	9	**finishing**	int(2)			Yes	*NULL*			✎ Change ⊖ Drop ▾ More
□	10	**gk_kicking**	int(2)			Yes	*NULL*			✎ Change ⊖ Drop ▾ More
□	11	**gk_positioning**	int(2)			Yes	*NULL*			✎ Change ⊖ Drop ▾ More
□	12	**penalties**	int(2)			Yes	*NULL*			✎ Change ⊖ Drop ▾ More
□	13	**short_pass**	int(2)			Yes	*NULL*			✎ Change ⊖ Drop ▾ More
□	14	**stamina**	int(2)			Yes	*NULL*			✎ Change ⊖ Drop ▾ More
□	15	**strength**	int(2)			Yes	*NULL*			✎ Change ⊖ Drop ▾ More

Fig 3.4 Player stats table structure

Player stats table consists of 14 attributes, among which, player_id is primary key and also has a foreign key reference to "player" table. It is designed to contain all the football technicalities of a player.

3.3.3 Salary details

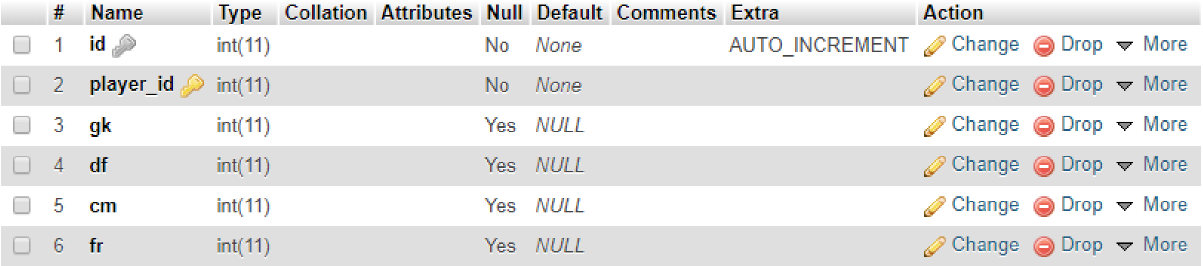


	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
□	1	**id**	int(11)			No	*None*		AUTO_INCREMENT	Change Drop ▾ More
□	2	**player_id**	int(11)			No	*None*			Change Drop ▾ More
□	3	**wage**	int(11)			Yes	*NULL*			Change Drop ▾ More
□	4	**value**	int(11)			Yes	*NULL*			Change Drop ▾ More

Fig 3.5 Salary table structure

The salary table consists player_id as primary key and also have a foreign key reference to "player" table. It is designed to store player weekly wage and his current value in the market.

3.3.4 Position details

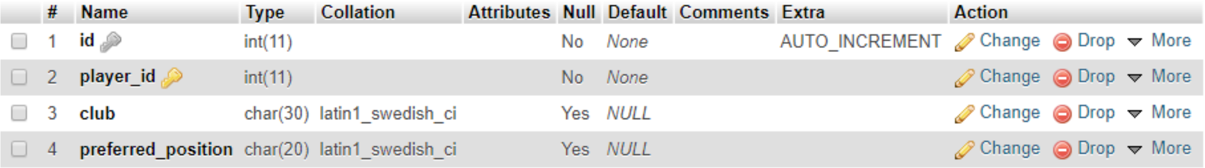


	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
□	1	**id**	int(11)			No	*None*		AUTO_INCREMENT	Change Drop ▾ More
□	2	**player_id**	int(11)			No	*None*			Change Drop ▾ More
□	3	**gk**	int(11)			Yes	*NULL*			Change Drop ▾ More
□	4	**df**	int(11)			Yes	*NULL*			Change Drop ▾ More
□	5	**cm**	int(11)			Yes	*NULL*			Change Drop ▾ More
□	6	**fr**	int(11)			Yes	*NULL*			Change Drop ▾ More

Fig 3.6 Position table structure

The position table also have player_id as primary key and a foreign key reference to "player" table. It is designed to store the positions a player can play, if so, then how well he does on a scale of rating from 0 to 99. It helps user to access player based on positional play and decide the best position for a player.

3.3.5 Club details

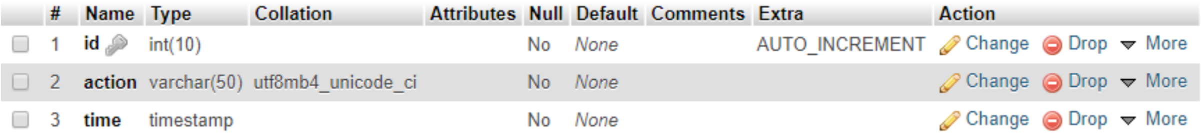


	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
□	1	**id**	int(11)			No	*None*		AUTO_INCREMENT	Change Drop ▾ More
□	2	**player_id**	int(11)			No	*None*			Change Drop ▾ More
□	3	**club**	char(30)	latin1_swedish_ci		Yes	*NULL*			Change Drop ▾ More
□	4	**preferred_position**	char(20)	latin1_swedish_ci		Yes	*NULL*			Change Drop ▾ More

Fig 3.7 Club table structure

The club details table has club information and the preferred position of a player at that club. It also have player_id as primary key and also a foreign key reference on "player" table.

3.3.6 Delete logs

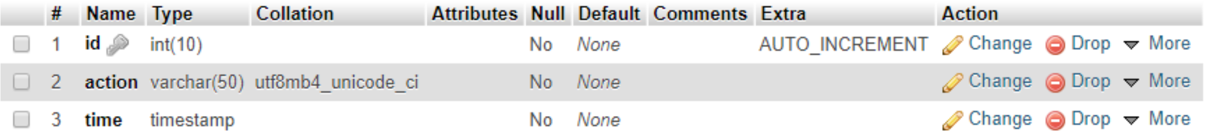


	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
□	1	**id**	int(10)			No	*None*		AUTO_INCREMENT	Change Drop More
□	2	**action**	varchar(50)	utf8mb4_unicode_ci		No	*None*			Change Drop More
□	3	**time**	timestamp			No	*None*			Change Drop More

Fig 3.8 Delete logs trigger structure

The delete logs table consists of 3 columns. ID column is unique and set to auto increment. Action column contain the action along with table name. Time column contains the time at which the trigger was automatically invoked based on the action.

3.3.7 Update logs

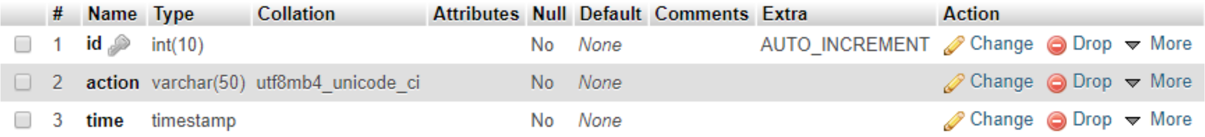


	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
□	1	**id**	int(10)			No	*None*		AUTO_INCREMENT	Change Drop More
□	2	**action**	varchar(50)	utf8mb4_unicode_ci		No	*None*			Change Drop More
□	3	**time**	timestamp			No	*None*			Change Drop More

Fig 3.9 Update logs trigger structure

The Update logs table consists of 3 columns. ID column is unique and set to auto increment. Action column contain the action along with table name. Time column contains the time at which the trigger was automatically invoked based on the action.

3.3.8 Insert logs

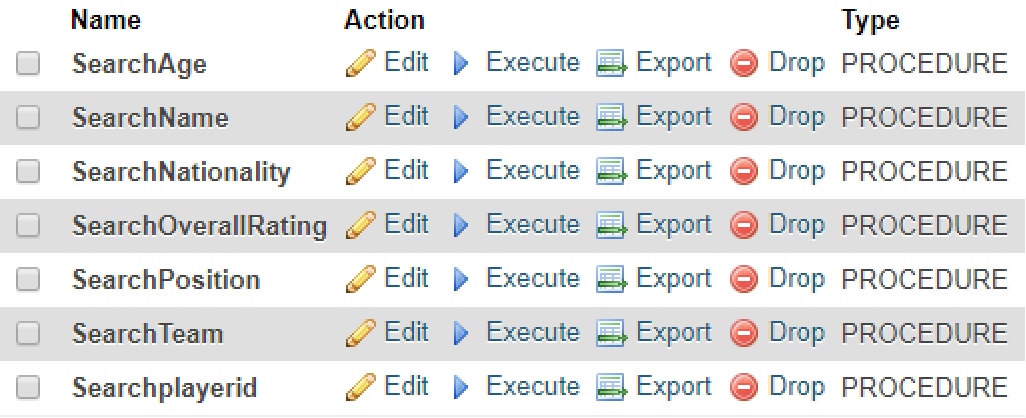


	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
□	1	**id**	int(10)			No	*None*		AUTO_INCREMENT	Change Drop More
□	2	**action**	varchar(50)	utf8mb4_unicode_ci		No	*None*			Change Drop More
□	3	**time**	timestamp			No	*None*			Change Drop More

Fig 3.10 Insert logs trigger structure

The Insert logs table consists of 3 columns. ID column is unique and set to auto increment. Action column contain the action along with table name. Time column contains the time at which the trigger was automatically invoked based on the action.

3.3.9 Stored procedures



	Name	Action				Type
□	SearchAge	Edit	Execute	Export	Drop	PROCEDURE
□	SearchName	Edit	Execute	Export	Drop	PROCEDURE
□	SearchNationality	Edit	Execute	Export	Drop	PROCEDURE
□	SearchOverallRating	Edit	Execute	Export	Drop	PROCEDURE
□	SearchPosition	Edit	Execute	Export	Drop	PROCEDURE
□	SearchTeam	Edit	Execute	Export	Drop	PROCEDURE
□	Searchplayerid	Edit	Execute	Export	Drop	PROCEDURE

Fig 3.11 Stored procedures structure

There are 7 stored procedures present inside search page on the web application. These are called whenever any search instance occur on the web page. The results of the stored procedures are then displayed on the UI in a tabular structure.

Chapter 4

IMPLEMENTATION

4.1 Stored procedures:

The Stored procedures are executed using PHP and MySQL in the application file. The result of stored procedure is shown in Fig 5.2.

Use: Use of stored procedures allows modular programming, reduces network traffic, faster execution, and can be used as security mechanism.

Given below is a code snippet of implementation of all the 7 stored procedure present inside the web application.

```
<section>
<ul class="menu cf">
   <li><a href="../../../../INDEX.html">Home</a></li>
   <li><a href="../../../../search_player/player_search.html">Search</a> </li>
   <li><a href="../../../../update_player/update_player.html">Update</a></li>
   <li><a href="../../../../insert_player/insert_new_player.html">Insert</a></li>
   <li><a href="../../../../database/database.php">Database</a></li>
   <li><a href="../../../../report/project_report.html">Report</a></li>
   <li><a href="../../../procedures.html">Procedures</a></li>
   <li><a href="../../../../about/about.html">About</a></li>
</ul>


<?php
$servername = "localhost";
$username = "root";
$password = "";
$dbname = "fifa";
```

```
// Create connection
$conn = new mysqli($servername, $username, $password, $dbname);
// Check connection
if ($conn->connect_error) {
die("Connection failed: " . $conn->connect_error);
}


$choices = $_POST['choices-single-defaul'];
$input_name = $_POST['input_search_keyword'];



#stored procedures


$spforage = mysqli_query($conn, "CREATE DEFINER=`root`@`localhost` PROCEDURE
`SearchAge`(IN `page` INT(11)) NOT DETERMINISTIC CONTAINS SQL SQL
SECURITY DEFINER SELECT player_name,age,overall_rating,nationality FROM
personal_details WHERE personal_details.age = page;");


$spfornationality = mysqli_query($conn, "CREATE DEFINER=`root`@`localhost`
PROCEDURE `SearchNationality`(IN `page` VARCHAR(30)) NOT DETERMINISTIC
CONTAINS SQL SQL SECURITY DEFINER SELECT * FROM personal_details WHERE
personal_details.nationality=page;");


$spforoverallrating = mysqli_query($conn, "CREATE DEFINER=`root`@`localhost`
PROCEDURE `SearchOverallRating`(IN `page` INT(11)) NOT DETERMINISTIC
CONTAINS SQL SQL SECURITY DEFINER SELECT * FROM personal_details WHERE
personal_details.overall_rating = page;");


$spforteam = mysqli_query($conn, "CREATE DEFINER=`root`@`localhost` PROCEDURE
`SearchTeam`(IN `page` VARCHAR(30)) NOT DETERMINISTIC CONTAINS SQL SQL
SECURITY DEFINER SELECT
pd.player_name,pd.overall_rating,pd.age,pd.nationality,od.club FROM personal_details
pd,other_details od WHERE od.club = page AND pd.player_id = od.player_id ORDER BY
pd.player_id;");
```

```
$spforname = mysqli_query($conn, "CREATE DEFINER=`root`@`localhost` PROCEDURE `SearchName`(IN `page` VARCHAR(30)) NOT DETERMINISTIC CONTAINS SQL SQL SECURITY DEFINER SELECT * FROM personal_details WHERE player_name = page");


$spforplayerid = mysqli_query($conn, "CREATE DEFINER=`root`@`localhost` PROCEDURE `Searchplayerid`(IN `page` INT(11)) NOT DETERMINISTIC CONTAINS SQL SQL SECURITY DEFINER SELECT * FROM personal_details WHERE player_id = page;");


$spforposition = mysqli_query($conn, "CREATE DEFINER=`root`@`localhost` PROCEDURE `SearchPosition`(IN `page` VARCHAR(11)) NOT DETERMINISTIC CONTAINS SQL SQL SECURITY DEFINER SELECT pd.player_name, pd.overall_rating, od.preferred_position, p.gk, p.df, p.cm, p.fr FROM personal_details pd, other_details od, position p WHERE od.preferred_position = page AND p.player_id = od.player_id AND p.player_id = pd.player_id GROUP BY pd.player_id;");



if ($choices == 'AGE' && ctype_digit(strval($input_name))) {
  $call = "CALL SearchAge('$input_name')";
?>
<div class="tbl-header">
<table cellpadding="0" cellspacing="0" border="0">
<thead>
  <tr>
    <th>NAME</th>
    <th>AGE</th>
    <th>OVERALL RATING</th>
    <th>NATIONALITY</th>
  </tr>
</thead>
</table>
</div>
<div class="tbl-content">
<table cellpadding="0" cellspacing="0" border="0">
```

```
<tbody>


<?php
$result = mysqli_query($conn, $call) ;
if ($result) {
// output data of each row
   while($row = $result->fetch_assoc()) {
         echo "<tr>";
         echo("<td>" . $row["player_name"] . "</td> <td>" . $row["age"] . "</td> <td>" .
$row["overall_rating"] . "</td><td>" .$row["nationality"]. "</td>");
         echo "</tr>";
      }
   }
}


else if ($choices == 'NATIONALITY' && !ctype_digit(strval($input_name))) {
   $call = "CALL SearchNationality('$input_name')";
?>
</tbody>
</table>
</div>


<div class="tbl-header">
<table cellpadding="0" cellspacing="0" border="0">
<thead>
   <tr>
      <th>NAME</th>
      <th>AGE</th>
      <th>OVERALL RATING</th>
      <th>NATIONALITY</th>
   </tr>
```

```
</thead>
</table>
</div>

<div class="tbl-content">
<table cellpadding="0" cellspacing="0" border="0">
<tbody>
<?php
$result = mysqli_query($conn, $call) ;
if ($result) {
   while($row = $result->fetch_assoc()) {
      echo "<tr>";
      echo("<td>" . $row["player_name"] . "</td> <td>" . $row["age"] . "</td> <td>" .
$row["overall_rating"] . "</td><td>" .$row["nationality"]. "</td>");
      echo "</tr>";
      }
   }
}

else if ($choices == 'OVERALL RATING' && ctype_digit(strval($input_name))) {
   $call = "CALL SearchOverallRating('$input_name')";
?>
</tbody>
</table>
</div>
<div class="tbl-header">
<table cellpadding="0" cellspacing="0" border="0">
<thead>
   <tr>
      <th>NAME</th>
      <th>AGE</th>
```

```
        <th>OVERALL RATING</th>
        <th>NATIONALITY</th>
    </tr>
</thead>
</table>
</div>
<div class="tbl-content">
<table cellpadding="0" cellspacing="0" border="0">
<tbody>
<?php
$result = mysqli_query($conn, $call) ;
if ($result) {
    while($row = $result->fetch_assoc()) {
        echo "<tr>";
        echo("<td>" . $row["player_name"] . "</td> <td>" . $row["age"] . "</td> <td>" .
$row["overall_rating"] . "</td><td>" .$row["nationality"]. "</td>");
        echo "</tr>";
    }
}
}

else if ($choices == 'PLAYER ID' && ctype_digit(strval($input_name))) {
    $call = "CALL Searchplayerid('$input_name')";
?>
</tbody>
</table>
</div>
<div class="tbl-header">
<table cellpadding="0" cellspacing="0" border="0">
<thead>
    <tr>
```

```
        <th>NAME</th>
        <th>AGE</th>
        <th>OVERALL RATING</th>
        <th>NATIONALITY</th>
    </tr>
</thead>
</table>
</div>
<div class="tbl-content">
<table cellpadding="0" cellspacing="0" border="0">
<tbody>
<?php
$result = mysqli_query($conn, $call) ;
if ($result) {
    while($row = $result->fetch_assoc()) {
        echo "<tr>";
        echo("<td>" . $row["player_name"] . "</td> <td>" . $row["age"] . "</td> <td>" .
$row["overall_rating"] . "</td><td>" .$row["nationality"]. "</td>");
        echo "</tr>";
    }
}
}

else if ($choices == 'PLAYER NAME' && !ctype_digit(strval($input_name))) {
    $call = "CALL SearchName('$input_name')";
?>
</tbody>
</table>
</div>
<div class="tbl-header">
<table cellpadding="0" cellspacing="0" border="0">
```

```
<thead>
   <tr>
      <th>NAME</th>
      <th>AGE</th>
      <th>OVERALL RATING</th>
      <th>NATIONALITY</th>
   </tr>
</thead>
</table>
</div>
<div class="tbl-content">
<table cellpadding="0" cellspacing="0" border="0">
<tbody>
<?php
$result = mysqli_query($conn, $call) ;
if ($result) {
   while($row = $result->fetch_assoc()) {
      echo "<tr>";
      echo("<td>" . $row["player_name"] . "</td> <td>" . $row["age"] . "</td> <td>" .
$row["overall_rating"] . "</td><td>" .$row["nationality"]. "</td>");
      echo "</tr>";
   }
}
}

else if ($choices == 'TEAM' && !ctype_digit(strval($input_name))) {
$call = "CALL SearchTeam('$input_name')";
?>
</tbody>
</table>
</div>
```

```
<div class="tbl-header">
<table cellpadding="0" cellspacing="0" border="0">
<thead>
  <tr>
    <th>NAME</th>
    <th>AGE</th>
    <th>OVERALL RATING</th>
    <th>NATIONALITY</th>
    <th>TEAM</th>
  </tr>
</thead>

</table>
</div>
<div class="tbl-content">
<table cellpadding="0" cellspacing="0" border="0">
<tbody>
<?php
$result = mysqli_query($conn, $call) ;
if ($result) {
  while($row = $result->fetch_assoc()) {
    echo "<tr>";
    echo("<td>" . $row["player_name"] . "</td> <td>" . $row["age"] . "</td> <td>" .
$row["overall_rating"] . "</td><td>" .$row["nationality"]. "</td><td>" . $row["club"] .
"</td>");
    echo "</tr>";
  }
}
}

else if ($choices == 'PLAYING POSITION' && !ctype_digit(strval($input_name))) {
    $call = "CALL SearchPosition('$input_name')";
?>
</tbody>
</table>
</div>
<div class="tbl-header">
<table cellpadding="0" cellspacing="0" border="0">
<thead>
    <tr>
        <th>NAME</th>
        <th>PLAYING POSITION</th>
        <th>OVERALL RATING</th>
        <th colspan="4">RATING AT OTHER POSITIONS</th>
    </tr>
    <tr>
        <th></th>
        <th></th>
        <th></th>
        <th>GOALKEEPER</th>
        <th>DEFENDER</th>
        <th>CENTER-MID</th>
        <th>FORWARD</th>
    </tr>
</thead>
</table>
</div>
<div class="tbl-content">
<table cellpadding="0" cellspacing="0" border="0">
<tbody>
<?php
```

```
$result = mysqli_query($conn, $call) ;
if ($result) {
// output data of each row
   while($row = $result->fetch_assoc()) {
       echo "<tr>";
       echo("<td>" . $row["player_name"] . "</td> <td>" . $row["preferred_position"] . "</td>
<td>" . $row["overall_rating"] . "</td><td>" .$row["gk"]. "</td><td>" . $row["df"] .
"</td><td>" . $row["cm"] . "</td><td>" . $row["fr"] . "</td>");
       echo "</tr>";
       }
   }
}
   else {
       header("Location:index.html");
}

$conn->close();
?>
```

4.2 Inserting new records:

The insert page allows users to select the table to insert values into [Fig 5.4]. It then asks users to input required data columns for the particular table and upon successful insertion, a new page is displayed [Fig 5.5] with appropriate message. Upon failure, another page is displayed with appropriate message and a possible solution.

Given below is the code snippet of the insert page which is execute in the application using PHP and MySQL.

```
<ul class="menu cf">
<li><a href="../../INDEX.html">Home</a></li>
<li><a href="../../search_player/player_search.html">Search</a> </li>
<li><a href="../../update_player/update_player.html">Update</a></li>
```

```
<li><a href="../insert_new_player.html">Insert</a></li>

<li><a href="../../database/database.php">Database</a></li>

<li><a href="../../report/project_report.html">Report</a></li>

<li><a href="../../procedures/procedures.html">Procedures</a></li>

<li><a href="../../about/about.html">About</a></li>

</ul>


<?php

$servername = "localhost";

$username = "root";

$password = "";


// Create connection

$conn = new mysqli($servername, $username, $password);


// Check connection

if ($conn->connect_error) {

die("Connection failed: " . $conn->connect_error);

}


$choices = $_POST["choices-single-defaul"];

?>


<?php
```

```
if($choices=='PERSONAL DETAILS'){
echo "<form action=\"php_submit/into_personal_details.php\" method=\"post\">";
echo "<input id=\"input-1s\" type=\"text\" placeholder=\"PlayerID\" required autofocus name=\"pid\"/>";
echo "<label for=\"input-1\">";
echo "<span class=\"label-text\">Player ID</span>";
echo "<span class=\"nav-dot\"></span>";
echo "</label>";
echo "<input id=\"input-2\" type=\"text\" placeholder=\"Player Name\" required name=\"pname\"/>";
echo "<label for=\"input-2\">";
echo "<span class=\"label-text\">Player Name</span>";
echo "<span class=\"nav-dot\"></span>";
echo "</label>";
echo "<input id=\"input-3\" type=\"text\" placeholder=\"Age\" required name=\"page\"/>";
echo "<label for=\"input-3\">";
echo " <span class=\"label-text\">Player Age</span>";
echo "  <span class=\"nav-dot\"></span>";
echo "</label>";
echo "<input id=\"input-4\" type=\"text\" placeholder=\"Overall Rating\" required name=\"poverallrating\"/>";
echo "<label for=\"input-4\">";
echo "  <span class=\"label-text\">Overall Rating</span>";
echo "  <span class=\"nav-dot\"></span>";
echo "</label>";
echo "<input id=\"input-5\" type=\"text\" placeholder=\"Nationality\" required name=\"pnationality\"/>";

echo "<label for=\"input-5\">";

echo "  <span class=\"label-text\">Nationality</span>";

echo "  <span class=\"nav-dot\"></span>";

echo "</label>";

echo "<button type=\"submit\" onclick=\"php_submit/into_personal_details.php\" class=\"signup-button\">INSERT</button>";

echo "<p class=\"tip\">Press Tab</p>";

echo "</form>";

}


else  if($choices=='PLAYER CLUB'){


echo "<form action=\"php_submit/into_other_details.php\" method=\"post\">";

echo "<input id=\"input-1s\" type=\"text\" placeholder=\"PlayerID\" required autofocus name=\"pid\"/>";

echo "<label for=\"input-1\">";

echo "<span class=\"label-text\">Player ID</span>";

echo "<span class=\"nav-dot\"></span>";

echo "</label>";

echo "<input id=\"input-2\" type=\"text\" placeholder=\"Player Club\" required name=\"pclub\"/>";

echo "<label for=\"input-2\">";

echo "<span class=\"label-text\">Player Club</span>";
echo "<span class=\"nav-dot\"></span>";

echo "</label>";

echo "<input id=\"input-3\" type=\"text\" placeholder=\"Preferred Position\" required name=\"ppreferredposition\"/>";

echo "<label for=\"input-3\">";

echo " <span class=\"label-text\">Preferred Position</span>";

echo "  <span class=\"nav-dot\"></span>";

echo "</label>";

echo "<button type=\"submit\" onclick=\"php_submit/into_other_details.php\" class=\"signup-button\">INSERT</button>";

echo "<p class=\"tip\">Press Tab</p>";

echo "</form>";


}


else  if($choices=='PLAYER POSITION'){


echo "<form action=\"php_submit/into_position.php\" method=\"post\">";

echo "<input id=\"input-1s\" type=\"text\" placeholder=\"PlayerID\" required autofocus name=\"pid\"/>";

echo "<label for=\"input-1\">";

echo "<span class=\"label-text\">Player ID</span>";

echo "<span class=\"nav-dot\"></span>";

echo "</label>";
echo "<input id=\"input-2\" type=\"text\" placeholder=\"Player@GoalKeeper\" required name=\"pgk\"/>";

echo "<label for=\"input-2\">";

echo "<span class=\"label-text\">Player@Goalkeeper Rating</span>";

echo "<span class=\"nav-dot\"></span>";

echo "</label>";

echo "<input id=\"input-3\" type=\"text\" placeholder=\"Player@Defender\" required name=\"pdf\"/>";

echo "<label for=\"input-3\">";

echo " <span class=\"label-text\">Player@Defender Rating</span>";

echo "  <span class=\"nav-dot\"></span>";

echo "</label>";

echo "<input id=\"input-4\" type=\"text\" placeholder=\"Player@Central-Mid\" required name=\"pcm\"/>";

echo "<label for=\"input-4\">";

echo "  <span class=\"label-text\">Player@Central-mid Rating</span>";

echo "  <span class=\"nav-dot\"></span>";

echo "</label>";

echo "<input id=\"input-5\" type=\"text\" placeholder=\"Player@Forward\" required name=\"pfr\"/>";

echo "<label for=\"input-5\">";

echo "  <span class=\"label-text\">Player@Forward Rating</span>";

echo "  <span class=\"nav-dot\"></span>";

echo "</label>";

echo "<button type=\"submit\" onclick=\"php_submit/into_position.php\" class=\"signup-button\">INSERT</button>";
echo "<p class=\"tip\">Press Tab</p>";

echo "</form>";

}


else  if($choices=='PLAYER STATS'){


echo "<form action=\"php_submit/into_player_stats.php\" method=\"post\">";

echo "<input id=\"input-1s\" type=\"text\" placeholder=\"PlayerID\" required autofocus name=\"pid\"/>";

echo "<label for=\"input-1\">";

echo "<span class=\"label-text\">Player ID</span>";

echo "<span class=\"nav-dot\"></span>";

echo "</label>";

echo "<input id=\"input-2\" type=\"text\" placeholder=\"Player Acceleration\" required name=\"pacceleration\"/>";

echo "<label for=\"input-2\">";

echo "<span class=\"label-text\">Acceleration</span>";

echo "<span class=\"nav-dot\"></span>";

echo "</label>";

echo "<input id=\"input-3\" type=\"text\" placeholder=\"Player Balance\" required name=\"pbalance\" />";

echo "<label for=\"input-3\">";

echo " <span class=\"label-text\">Balance</span>";

echo "  <span class=\"nav-dot\"></span>";

echo "</label>";
echo "<input id=\"input-4\" type=\"text\" placeholder=\"Player Ball Control\" required name=\"pballcontrol\"/>";

echo "<label for=\"input-4\">";

echo "  <span class=\"label-text\">Ball Control</span>";

echo "  <span class=\"nav-dot\"></span>";

echo "</label>";

echo "<input id=\"input-5\" type=\"text\" placeholder=\"Player Crossing\" required name=\"pcrossing\"/>";

echo "<label for=\"input-5\">";

echo "  <span class=\"label-text\">Crossing</span>";

echo "  <span class=\"nav-dot\"></span>";

echo "</label>";

echo "<input id=\"input-5\" type=\"text\" placeholder=\"Player Curve\" required name=\"pcurve\"/>";

echo "<label for=\"input-5\">";

echo "  <span class=\"label-text\">Curve</span>";

echo "  <span class=\"nav-dot\"></span>";

echo "</label>";

echo "<input id=\"input-5\" type=\"text\" placeholder=\"Player Dribbling\" required name=\"pdribbling\"/>";

echo "<label for=\"input-5\">";

echo "  <span class=\"label-text\">Dribbling</span>";

echo "  <span class=\"nav-dot\"></span>";

echo "</label>";

echo "<input id=\"input-5\" type=\"text\" placeholder=\"Player Finishing\" required name=\"pfinishing\"/>";
echo "<label for=\"input-5\">";
echo "  <span class=\"label-text\">Finishing</span>";
echo "  <span class=\"nav-dot\"></span>";
echo "</label>";
echo "<input id=\"input-5\" type=\"text\" placeholder=\"Player GK Kicking\" required name=\"pgkk\"/>";
echo "<label for=\"input-5\">";
echo "  <span class=\"label-text\">GK kicking</span>";
echo "  <span class=\"nav-dot\"></span>";
echo "</label>";
echo "<input id=\"input-5\" type=\"text\" placeholder=\"Player GK Positioning\" required name=\"pgkp\"/>";
echo "<label for=\"input-5\">";
echo "  <span class=\"label-text\">GK Positioning</span>";
echo "  <span class=\"nav-dot\"></span>";
echo "</label>";
echo "<input id=\"input-5\" type=\"text\" placeholder=\"Player Penalties\" required name=\"ppenalties\"/>";
echo "<label for=\"input-5\">";
echo "  <span class=\"label-text\">Penalties</span>";
echo "  <span class=\"nav-dot\"></span>";
echo "</label>";
echo "<input id=\"input-5\" type=\"text\" placeholder=\"Player Short Pass\" required name=\"pshortpass\"/>";
echo "<label for=\"input-5\">";
echo "  <span class=\"label-text\">Short pass</span>";

echo "  <span class=\"nav-dot\"></span>";

echo "</label>";

echo "<input id=\"input-5\" type=\"text\" placeholder=\"Player Stamina\" required name=\"pstamina\"/>";

echo "<label for=\"input-5\">";

echo "  <span class=\"label-text\">Stamina</span>";

echo "  <span class=\"nav-dot\"></span>";

echo "</label>";

echo "<input id=\"input-5\" type=\"text\" placeholder=\"Player Strength\" required name=\"pstrength\"/>";

echo "<label for=\"input-5\">";

echo "  <span class=\"label-text\">Strength</span>";

echo "  <span class=\"nav-dot\"></span>";

echo "</label>";

echo "<button type=\"submit\" onclick=\"php_submit/into_player_stats.php\" class=\"signup-button\">INSERT</button>";

echo "<p class=\"tip\">Press Tab</p>";

echo "</form>";


}

else if($choices=='PLAYER EARNINGS'){


echo "<form action=\"php_submit/into_salary.php\" method=\"post\">";
echo "<input id=\"input-1s\" type=\"text\" placeholder=\"PlayerID\" required autofocus name=\"pid\" />";

echo "<label for=\"input-1\">";

echo "<span class=\"label-text\">Player ID</span>";

echo "<span class=\"nav-dot\"></span>";

echo "</label>";

echo "<input id=\"input-2\" type=\"text\" placeholder=\"Player Wages\" required name=\"pwage\"/>";

echo "<label for=\"input-2\">";

echo "<span class=\"label-text\">Player Wage</span>";

echo "<span class=\"nav-dot\"></span>";

echo "</label>";

echo "<input id=\"input-3\" type=\"text\" placeholder=\"Player Value\" required name=\"pvalue\"/>";

echo "<label for=\"input-3\">";

echo " <span class=\"label-text\">Player Value</span>";

echo "  <span class=\"nav-dot\"></span>";

echo "</label>";

echo "<button type=\"submit\" onclick=\"php_submit/into_salary.php\" class=\"signup-button\">INSERT</button>";

echo "<p class=\"tip\">Press Tab</p>";

echo "</form>";


}

else {
```

```
header("Location:index.html");

}

?>
```

(Inside into_personal_details.php)

```
<?php

$servername = "localhost";

$username = "root";

$password = "";

$dbname = "fifa";


// Create connection

$conn = new mysqli($servername, $username, $password, $dbname);

// Check connection

if ($conn->connect_error) {

die("Connection failed: " . $conn->connect_error);

}


$spid = $_POST["pid"];

$spname = $_POST["pname"];

$spage = $_POST["page"];

$spoverallrating = $_POST["poverallrating"];

$spnationality = $_POST["pnationality"];

?>

<ul class="menu cf">
```

```
<li><a href="../../../INDEX.html">Home</a></li>
<li><a href="../../../search_player/player_search.html">Search</a> </li>
<li><a href="../../../update_player/update_player.html">Update</a></li>
<li><a href="../../insert_new_player.html">Insert</a></li>
<li><a href="../../../database/database.php">Database</a></li>
<li><a href="../../../report/project_report.html">Report</a></li>
<li><a href="../../../procedures/procedures.html">Procedures</a></li>
<li><a href="../../../about/about.html">About</a></li>
</ul>


<?php
$sql = "INSERT INTO $dbname.personal_details (player_id, player_name,
age,overall_rating,nationality) VALUES
('$spid','$spname','$spage','$spoverallrating','$spnationality')";


if ($conn->query($sql) === TRUE) {
echo "<h3 id=\"result\" style=\"color:white; text-align:center; font-family:cursive;\">New
record inserted successfully ! Go ahead to <a
href=\"../../../database/database.php\">DATABASE</a> to see for yourself.</h3>";
} else {
//   header("Location:index.html");
echo "Error: " . $sql . "<br>" . $conn->error;
}
$conn->close();
?>
```

(Inside into_other_details.php)

```
<?php

$servername = "localhost";

$username = "root";

$password = "";

$dbname = "fifa";


// Create connection

$conn = new mysqli($servername, $username, $password, $dbname);

// Check connection

if ($conn->connect_error) {

die("Connection failed: " . $conn->connect_error);

}


$spid = $_POST["pid"];

$spclub = $_POST["pclub"];

$sppreferredposition = $_POST["ppreferredposition"];

?>

<ul class="menu cf">

<li><a href="../../../INDEX.html">Home</a></li>

<li><a href="../../../search_player/player_search.html">Search</a> </li>

<li><a href="../../../update_player/update_player.html">Update</a></li>

<li><a href="../../insert_new_player.html">Insert</a></li>

<li><a href="../../../database/database.php">Database</a></li>
```

```
<li><a href="../../../report/project_report.html">Report</a></li>

<li><a href="../../../procedures/procedures.html">Procedures</a></li>

<li><a href="../../../about/about.html">About</a></li>

</ul>



<?php

$sql = "INSERT INTO $dbname.other_details (player_id, club, preferred_position) VALUES ('$spid','$spclub','$sppreferredposition')";



if ($conn->query($sql) === TRUE) {

echo "<h3 style=\"color:white;text-align:center; font-family:cursive;\">New record inserted successfully ! Go ahead to <a href=\"../../../database/database.php\">DATABASE</a> to see for yourself.</h3>";

} else {

header("Location:index.html");

//		echo "Error: " . $sql . "<br>" . $conn->error;

}

$conn->close();

?>
```

4.3 Updating existing records:

The update page allows users to look at a selected table and edit the table live on frontend. The user first selects the table he/she wants to modify [Fig 5.6]. Upon selection, a new page is created with respective table. This table consists of editable rows that can be modified and result can be seen real time [Fig 5.7].

Given below is a code snippet of implementation of update page. It is created using PHP, MySQL and JavaScript. Use of JavaScript makes the UI more flexible by providing real time data interaction.

```
<?php
include('db-connect.php');
$row1 = mysqli_query($con,"SELECT * FROM personal_details");
?>
<!DOCTYPE html>
<html>
<head>
<script src="jquery.tabledit.js"></script>
<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/meyer-
reset/2.0/reset.min.css">
<link href="css/menu.css" rel="stylesheet" type="text/css">
<script type="text/javascript">


$(document).ready(function(){



$('#example1').Tabledit({
url: 'logic-edit-delete.php',
columns: {
identifier: [0, 'player_id'],
editable: [[1, 'player_name'], [2, 'age'],[3, 'overall_rating'],[4, 'nationality']]
},
onDraw: function() {
```

```
console.log('onDraw()');

},

onSuccess: function(data, textStatus, jqXHR) {

console.log('onSuccess(data, textStatus, jqXHR)');

console.log(data);

console.log(textStatus);

console.log(jqXHR);

},

onFail: function(jqXHR, textStatus, errorThrown) {

console.log('onFail(jqXHR, textStatus, errorThrown)');

console.log(jqXHR);

console.log(textStatus);

console.log(errorThrown);

},

onAlways: function() {

console.log('onAlways()');

},

onAjax: function(action, serialize) {

console.log('onAjax(action, serialize)');

console.log(action);

console.log(serialize);

}

});

});
```

```
</script>


<body>

<ul class="menu cf">

<li><a href="../../../../INDEX.html">Home</a></li>

<li><a href="../../../../search_player/player_search.html">Search</a> </li>

<li><a href="../../../update_player.html">Update</a></li>

<li><a href="../../../../insert_player/insert_new_player.html">Insert</a></li>

<li><a href="../../../../database/database.php">Database</a></li>

<li><a href="../../../../report/project_report.html">Report</a></li>

<li><a href="../../../../procedures/procedures.html">Procedures</a></li>

<li><a href="../../../../about/about.html">About</a></li>

</ul>


<input TYPE="button" onClick="history.go(0)" VALUE="Refresh" class="input_style">

<div class="panel panel-default">

<div class="tbl-header">

<table cellpadding="0" cellspacing="0" border="0" id="example1">

<tr><th>Id</th><th>Name</th><th>Age</th><th>Overall
Rating</th><th>Nationality</th></tr>

<?php while($row = mysqli_fetch_assoc($row1)) {?>

<tr>

<td><?php echo $row['player_id'];?></td>

<td><?php echo $row['player_name']; ?></td>
<td><?php echo $row['age']; ?></td>

<td><?php echo $row['overall_rating']; ?></td>

<td><?php echo $row['nationality']; ?></td>

</tr>

<?php } ?>

</table>

</div>

</div>

$input = filter_input_array(INPUT_POST);


if ($input['action'] === 'edit')

{

            $sql = "UPDATE personal_details SET player_name ='" . $input['player_name'] .
"',age ='" . $input['age'] . "',overall_rating ='" . $input['overall_rating'] . "', nationality='" .
$input['nationality'] . "'" ." WHERE player_id='" . $input['player_id'] . "'";


    mysqli_query($con,$sql);

}


<?php

$servername = "localhost";

$username = "root";

$password = "";

$database = "fifa";
```

```
// Create connection

$conn = new mysqli($servername, $username, $password,$database);


// Check connection

if ($conn->connect_error) {

    die("Connection failed: " . $conn->connect_error);

}


$choices = $_POST["choices-single-defaul"];


if($choices=='PERSONAL DETAILS'){

header("Location:test_personal_details/inline-table-edit.php");

}

else  if($choices=='PLAYER EARNINGS'){

    header("Location:test_player_salary/inline-table-edit.php");

}


else  if($choices=='PLAYER POSITION'){

header("Location:test_player_position/inline-table-edit.php");

}


else  if($choices=='PLAYER CLUB'){

header("Location:test_player_club/inline-table-edit.php");

}
```

```
else if($choices=='PLAYER STATS'){

header("Location:test_player_stats/inline-table-edit.php");


}

else {

     header("Location:index.html");

}

  ?>
```

4.4 Deleting existing records:

The delete page allows users to delete data based on categories like age, nationality, player ID, name and overall rating [Fig 5.8]. Choosing from different categories provides better user-database interface. Upon deletion, a new page is displayed with appropriate message [Fig 5.9] and upon failing, a new page with error information and possible solutions.

Given below is a code snippet of implementation of delete page. It is created using PHP and MySQL.

```
<?php

$servername = "localhost";

$username = "root";

$password = "";


// Create connection

$conn = new mysqli($servername, $username, $password);


// Check connection

if ($conn->connect_error) {
```

```
die("Connection failed: " . $conn->connect_error);

}



$choices = $_POST["choices-single-defaul"];

?>

<?php

if($choices=='BY AGE'){

echo "<form action=\"php_submit_delete/delete_byage.php\" method=\"post\">";

echo "<input id=\"input-1s\" type=\"text\" placeholder=\"PLAYER AGE\" required autofocus name=\"page\"/>";

echo "<label for=\"input-1\">";

echo "<span class=\"label-text\">PLAYER AGE</span>";

echo "<span class=\"nav-dot\"></span>";

echo "</label>";

echo "<button type=\"submit\" onclick=\"php_submit_delete/delete_byage.php\" class=\"signup-button\">DELETE</button>";

echo "<p class=\"tip\">Press Tab</p>";

//    echo "<div class=\"signup-button\">INSERT</div>";

echo "</form>";

}



else  if($choices=='BY NAME'){



echo "<form action=\"php_submit_delete/delete_byname.php\" method=\"post\">";
echo "<input id=\"input-1s\" type=\"text\" placeholder=\"PLAYER NAME\" required autofocus name=\"pname\"/>";

echo "<label for=\"input-1\">";

echo "<span class=\"label-text\">PLAYER NAME</span>";

echo "<span class=\"nav-dot\"></span>";

echo "</label>";

echo "<button type=\"submit\" onclick=\"php_submit_delete/delete_byname.php\" class=\"signup-button\">DELETE</button>";

echo "<p class=\"tip\">Press Tab</p>";

//    echo "<div class=\"signup-button\">INSERT</div>";

echo "</form>";


}


else  if($choices=='BY PLAYER ID'){


echo "<form action=\"php_submit_delete/delete_byplayerid.php\" method=\"post\">";

echo "<input id=\"input-1s\" type=\"text\" placeholder=\"PLAYER ID\" required autofocus name=\"pid\"/>";

echo "<label for=\"input-1\">";

echo "<span class=\"label-text\">PLAYER ID</span>";

echo "<span class=\"nav-dot\"></span>";

echo "</label>";

echo "<button type=\"submit\" onclick=\"php_submit_delete/delete_playerid.php\" class=\"signup-button\">DELETE</button>";
echo "<p class=\"tip\">Press Tab</p>";

//    echo "<div class=\"signup-button\">INSERT</div>";

echo "</form>";

}



else  if($choices=='BY OVERALL RATING'){



echo "<form action=\"php_submit_delete/delete_byoverallrating.php\" method=\"post\">";

echo "<input id=\"input-1s\" type=\"text\" placeholder=\"OVERALL RATING\" required
autofocus name=\"poverall_rating\"/>";

echo "<label for=\"input-1\">";

echo "<span class=\"label-text\">PLAYER OVERALL RATING</span>";

echo "<span class=\"nav-dot\"></span>";

echo "</label>";

echo "<button type=\"submit\" onclick=\"php_submit_delete/delete_byoverallrating.php\"
class=\"signup-button\">DELETE</button>";

echo "<p class=\"tip\">Press Tab</p>";

//    echo "<div class=\"signup-button\">INSERT</div>";

echo "</form>";



}

else if($choices=='BY NATIONALITY'){



echo "<form action=\"php_submit_delete/delete_bynationality.php\" method=\"post\">";
echo "<input id=\"input-1s\" type=\"text\" placeholder=\"PLAYER NATIONALITY\"
required autofocus name=\"pnationality\" />";

echo "<label for=\"input-1\">";

echo "<span class=\"label-text\">PLAYER NATIONALITY</span>";

echo "<span class=\"nav-dot\"></span>";

echo "</label>";

echo "<button type=\"submit\" onclick=\"php_submit_delete/delete_bynationality.php\"
class=\"signup-button\">DELETE</button>";

echo "<p class=\"tip\">Press Tab</p>";

//    echo "<div class=\"signup-button\">INSERT</div>";

echo "</form>";


}

else {

header("Location:index.html");

}

?>

<?php

$servername = "localhost";

$username = "root";

$password = "";

$dbname = "fifa";


// Create connection

$conn = new mysqli($servername, $username, $password, $dbname);
```

```
// Check connection

if ($conn->connect_error) {

die("Connection failed: " . $conn->connect_error);

}


$spname = $_POST["pname"];

$spages = $_POST["page"];

$spnationality = $_POST["pnationality"];

$spid = $_POST["pid"];

$spoverallrating = $_POST["poverall_rating"];

?>

<?php

$sql = "DELETE FROM personal_details WHERE player_name=\"$spname\"";


if ($conn->query($sql) === TRUE) {

echo "<h3 style=\"color:white;text-align:center; font-family:cursive;\"> DELETED
SUCCESSFULLY ! Go ahead to <a
href=\"../../../../../database/database.php\">DATABASE</a> to see for yourself.</h3>";

} else {

header("Location:index.html");

//                      echo "Error: " . $sql . "<br>" . $conn->error;

}


$sql = "DELETE FROM personal_details WHERE age=\"$spages\"";
```

```
if ($conn->query($sql) === TRUE) {
echo "<h3 style=\"color:white;text-align:center; font-family:cursive;\"> DELETED
SUCCESSFULLY ! Go ahead to <a
href=\"../../../../database/database.php\">DATABASE</a> to see for yourself.</h3>";
} else {
header("Location:index.html");
//                      echo "Error: " . $sql . "<br>" . $conn->error;
}


$sql = "DELETE FROM personal_details WHERE nationality=\"$spnationality\"";


if ($conn->query($sql) === TRUE) {
echo "<h3 style=\"color:white;text-align:center; font-family:cursive;\"> DELETED
SUCCESSFULLY ! Go ahead to <a
href=\"../../../../../database/database.php\">DATABASE</a> to see for yourself.</h3>";
} else {
header("Location:index.html");
//                      echo "Error: " . $sql . "<br>" . $conn->error;
}


$sql = "DELETE FROM personal_details WHERE overall_rating=\"$spoverallrating\"";


        if ($conn->query($sql) === TRUE) {
    echo "<h3 style=\"color:white;text-align:center; font-family:cursive;\"> DELETED
SUCCESSFULLY ! Go ahead to <a
href=\"../../../../../database/database.php\">DATABASE</a> to see for yourself.</h3>";
} else {
   header("Location:index.html");
//                     echo "Error: " . $sql . "<br>" . $conn->error;
}


$sql = "DELETE FROM personal_details WHERE player_id=\"$spid\"";


      if ($conn->query($sql) === TRUE) {
   echo "<h3 style=\"color:white;text-align:center; font-family:cursive;\"> DELETED
SUCCESSFULLY ! Go ahead to <a
href=\"../../../../../database/database.php\">DATABASE</a> to see for yourself.</h3>";
} else {
   header("Location:index.html");
//                     echo "Error: " . $sql . "<br>" . $conn->error;
}
$conn->close();
?>
```

4.5 Triggers:

The trigger page allows users to look at the database log of operations such as delete, update and insert.

Use: To improve data integrity, trigger can be used. When an action is performed on data, it is possible to check if the manipulation of the data concurs with the underlying business rules, and thus avoids erroneous entries in a table.

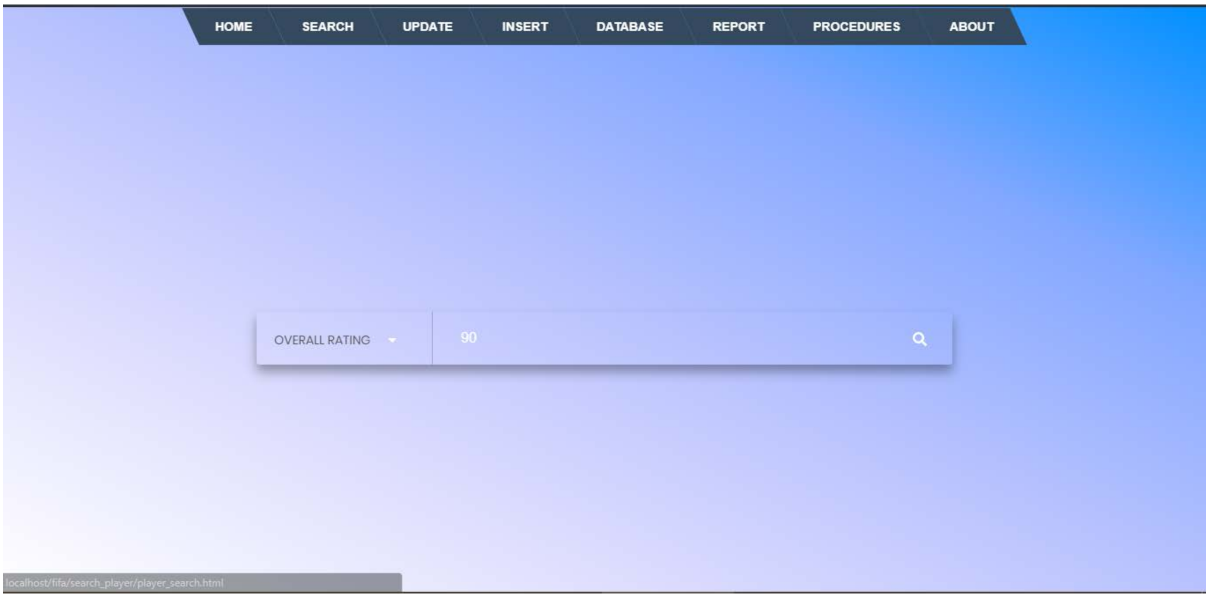
Given below is a code snippet of implementation of trigger page. It is created using PHP, MySQL and HTML. Use of HTML makes the UI show database logs in a structured manner.

```
<?php

if($choices=='TRIGGER PROCEDURE'){

echo "  <h1>INSERT TRIGGERS</h1>";

echo "<div class=\"tbl-header\">";

echo " <table cellpadding=\"0\" cellspacing=\"0\" border=\"0\">";

echo "<thead>";

echo "<tr>";

echo " <th>ID</th>";

echo " <th>ACTION</th>";

echo " <th>TIME</th>";

echo "</tr>";

echo " </thead>";

echo " </table>";

echo "</div>";

echo "<div class=\"tbl-content\">";

echo "<table cellpadding=\"0\" cellspacing=\"0\" border=\"0\">";

$sql = "SELECT * FROM insert_logs ORDER BY id";

$result = $conn->query($sql);

if ($result->num_rows >0 ) {

while($row = $result->fetch_assoc()) {

echo "<tbody>";

echo "<tr>";
echo "<td>" .$row["id"]. "</td>";

echo "<td>" .$row["action"]. "</td>";

echo "<td>" .$row["time"]. "</td>";

echo "</tr>";

echo "</tbody>";

}

}else {

echo "<h3 style=\"text-align:center; font-family:cursive;\">No changes yet !</h3>";

}

echo "</table>";

echo "</div><br>";



echo "  <h1>UPDATE TRIGGERS</h1>";

echo "<div class=\"tbl-header\">";

echo " <table cellpadding=\"0\" cellspacing=\"0\" border=\"0\">";

echo "<thead>";

echo "<tr>";

echo " <th>ID</th>";

echo " <th>ACTION</th>";

echo " <th>TIME</th>";

echo "</tr>";

echo " </thead>";

echo " </table>";
echo "</div>";

echo "<div class=\"tbl-content\">";

echo "<table cellpadding=\"0\" cellspacing=\"0\" border=\"0\">";

$sql = "SELECT * FROM update_logs ORDER BY id";

$result = $conn->query($sql);

if ($result->num_rows >0 ) {

while($row = $result->fetch_assoc()) {

echo "<tbody>";

echo "<tr>";

echo "<td>" .$row["id"]. "</td>";

echo "<td>" .$row["action"]. "</td>";

echo "<td>" .$row["time"]. "</td>";

echo "</tr>";

echo "</tbody>";

}

}else {

echo "<h3 style=\"text-align:center; font-family:cursive;\">No changes yet !</h3>";

}

echo "</table>";

echo "</div><br>";




echo "  <h1>DELETE TRIGGERS</h1>";

echo "<div class=\"tbl-header\">";
echo " <table cellpadding=\"0\" cellspacing=\"0\" border=\"0\">";

echo "<thead>";

echo "<tr>";

echo " <th>ID</th>";

echo " <th>ACTION</th>";

echo " <th>TIME</th>";

echo "</tr>";

echo " </thead>";

echo " </table>";

echo "</div>";

echo "<div class=\"tbl-content\">";

echo "<table cellpadding=\"0\" cellspacing=\"0\" border=\"0\">";

$sql = "SELECT * FROM delete_logs ORDER BY id";

$result = $conn->query($sql);

if ($result->num_rows >0 ) {

while($row = $result->fetch_assoc()) {

echo "<tbody>";

echo "<tr>";

echo "<td>" .$row["id"]. "</td>";

echo "<td>" .$row["action"]. "</td>";

echo "<td>" .$row["time"]. "</td>";

echo "</tr>";

echo "</tbody>";

}
}else {

echo "<h3 style=\"text-align:center; font-family:cursive;\">No changes yet !</h3>";

}


echo "</table>";

echo "</div><br>";

}
```

Chapter 5

SCREENSHOTS

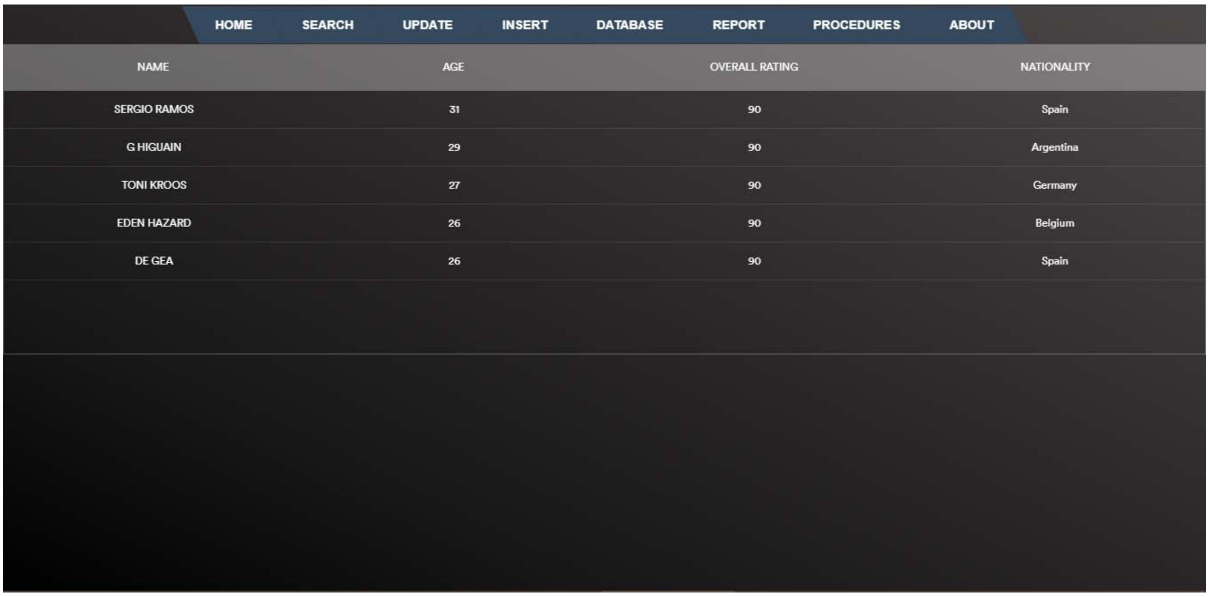
5.1 Stored procedures:



OVERALL RATING 90

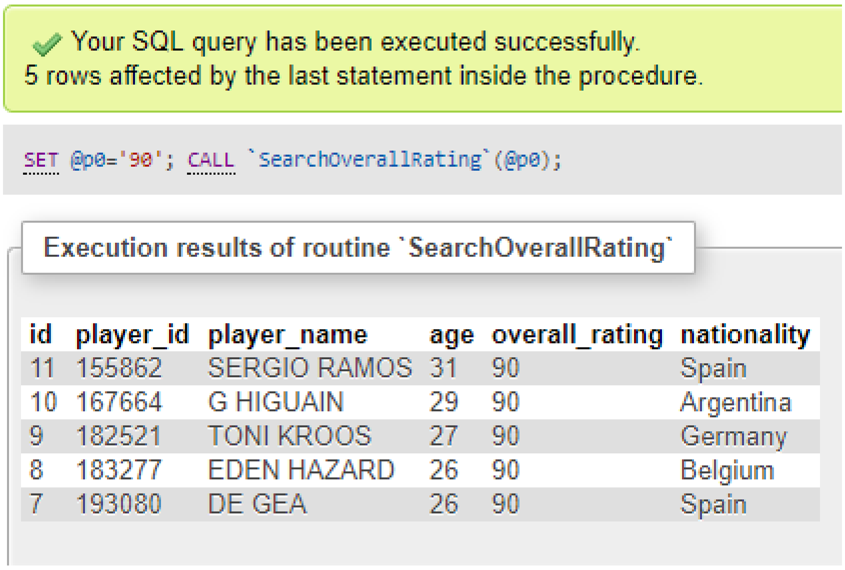
localhost/fifa/search_player/player_search.html

Fig 5.1 Search input for stored procedures.



NAME	AGE	OVERALL RATING	NATIONALITY
SERGIO RAMOS	31	90	Spain
G HIGUAIN	29	90	Argentina
TONI KROOS	27	90	Germany
EDEN HAZARD	26	90	Belgium
DE GEA	26	90	Spain

Fig 5.2 Search result using stored procedure from frontend.



✓ Your SQL query has been executed successfully.
5 rows affected by the last statement inside the procedure.

```
SET @p0='90'; CALL `SearchOverallRating`(@p0);
```

Execution results of routine `SearchOverallRating`

id	player_id	player_name	age	overall_rating	nationality
11	155862	SERGIO RAMOS	31	90	Spain
10	167664	G HIGUAIN	29	90	Argentina
9	182521	TONI KROOS	27	90	Germany
8	183277	EDEN HAZARD	26	90	Belgium
7	193080	DE GEA	26	90	Spain

Fig 5.3 Search result using stored procedure from backend.

5.2 Inserting new records:

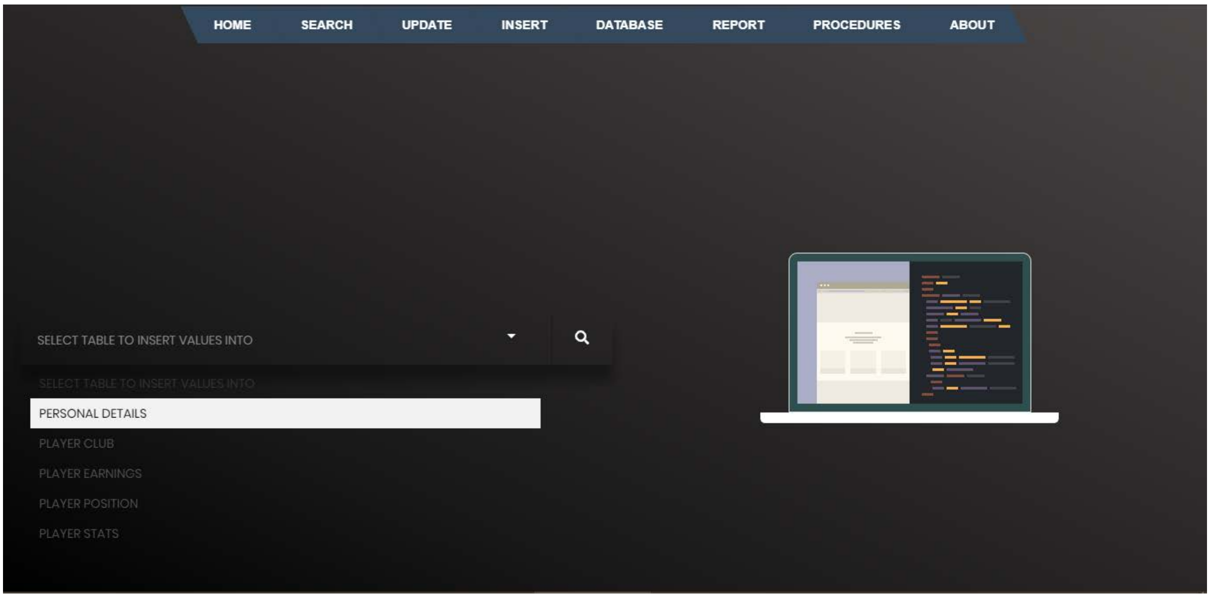
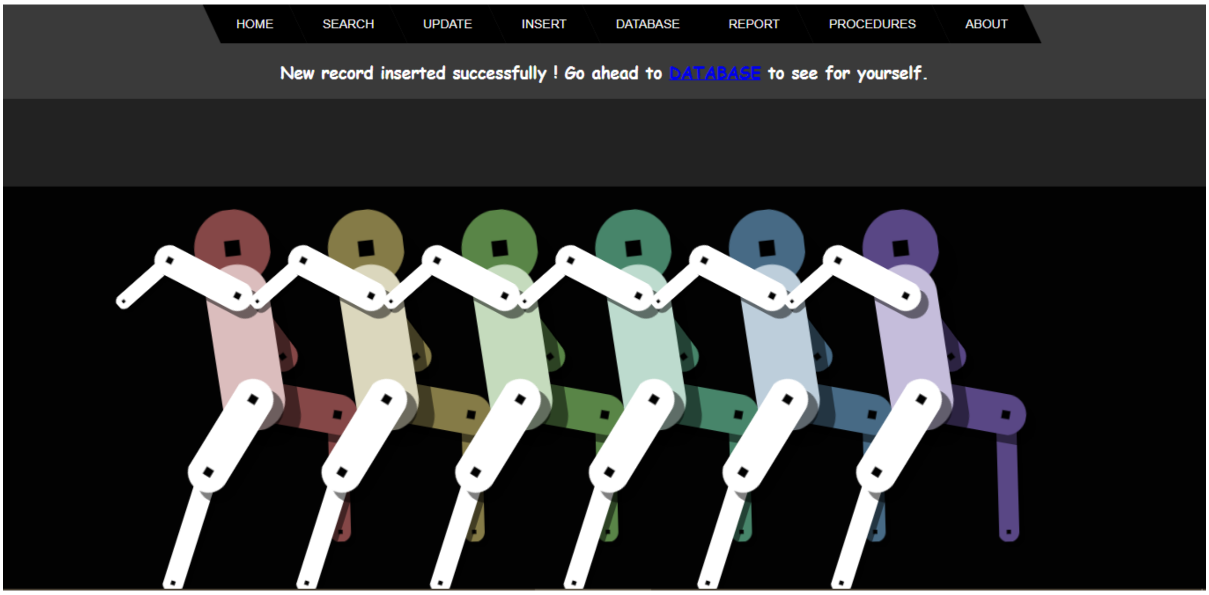


Fig 5.4 Selecting table for insertion



New record inserted successfully ! Go ahead to DATABASE to see for yourself.

Fig 5.5 Successful insert instance page

5.3 Update existing records:

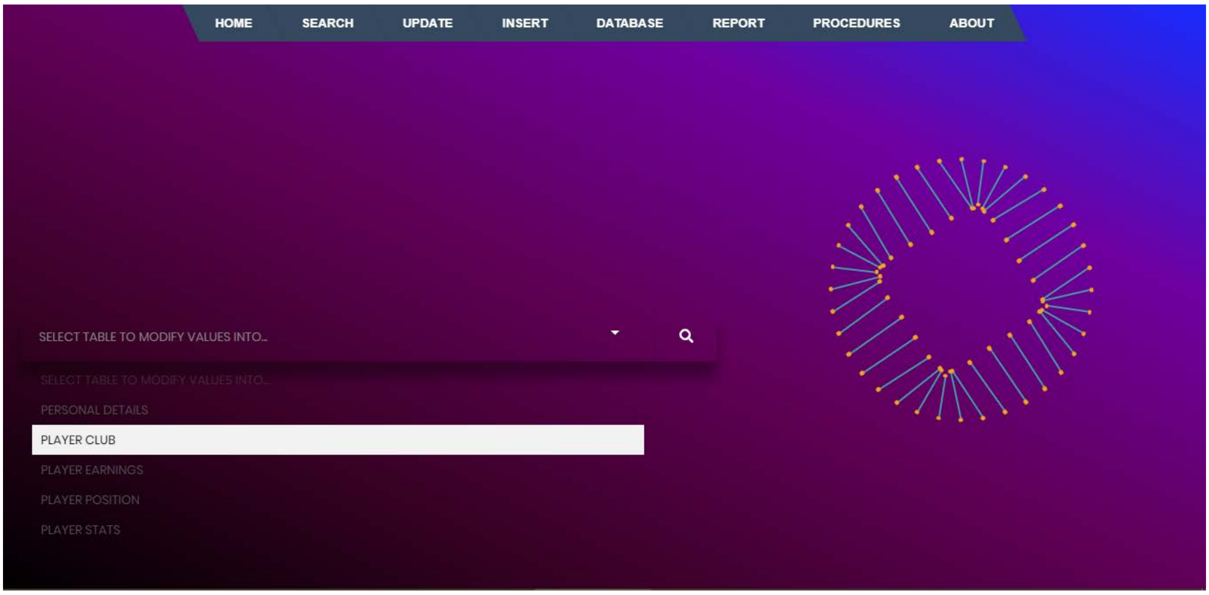
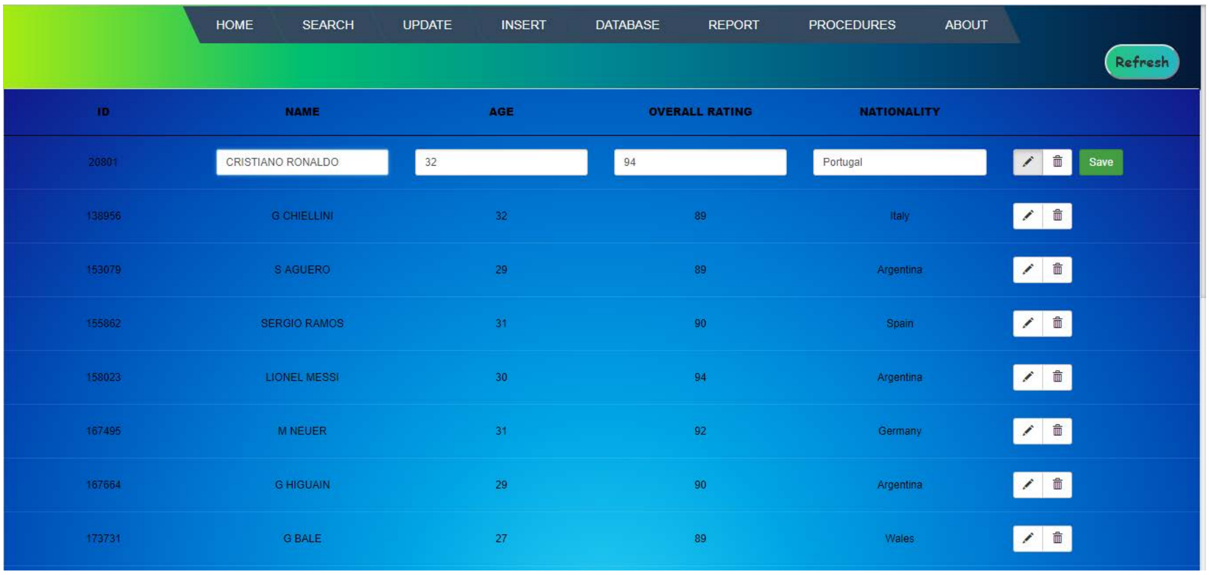


Fig 5.6 Selecting table to modify record



ID	NAME	AGE	OVERALL RATING	NATIONALITY	
20801	CRISTIANO RONALDO	32	94	Portugal	Save
138956	G CHIELLINI	32	89	Italy	
153079	S AGUERO	29	89	Argentina	
155862	SERGIO RAMOS	31	90	Spain	
158023	LIONEL MESSI	30	94	Argentina	
167495	M NEUER	31	92	Germany	
167664	G HIGUAIN	29	90	Argentina	
173731	G BALE	27	89	Wales	

Fig 5.7 Modifying records in real time

5.4 Deleting records:

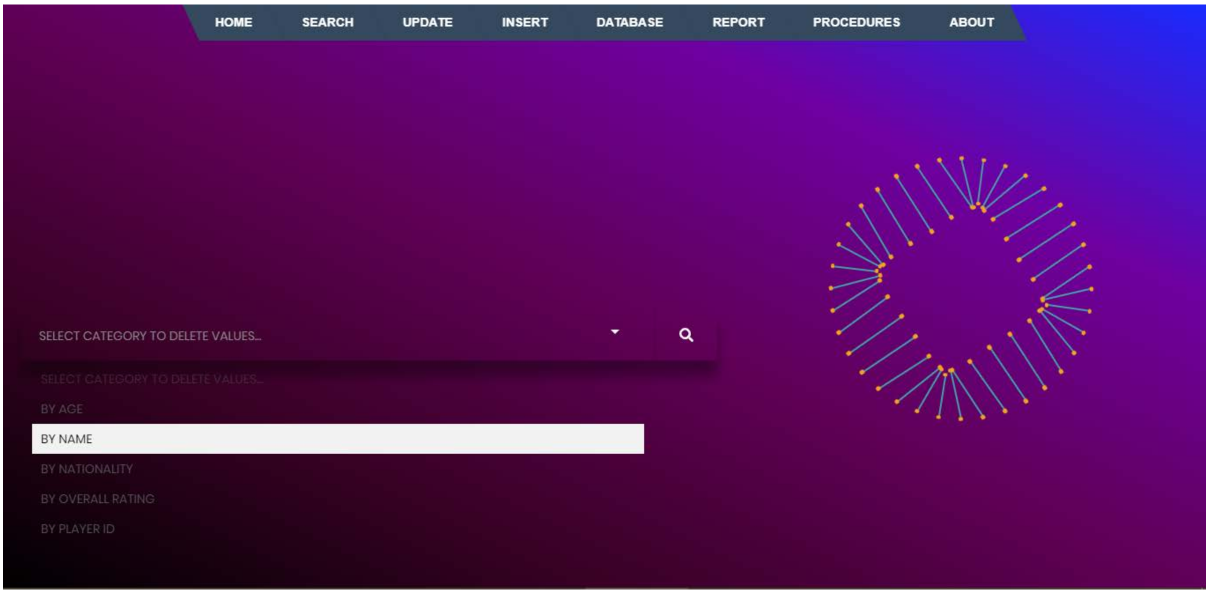
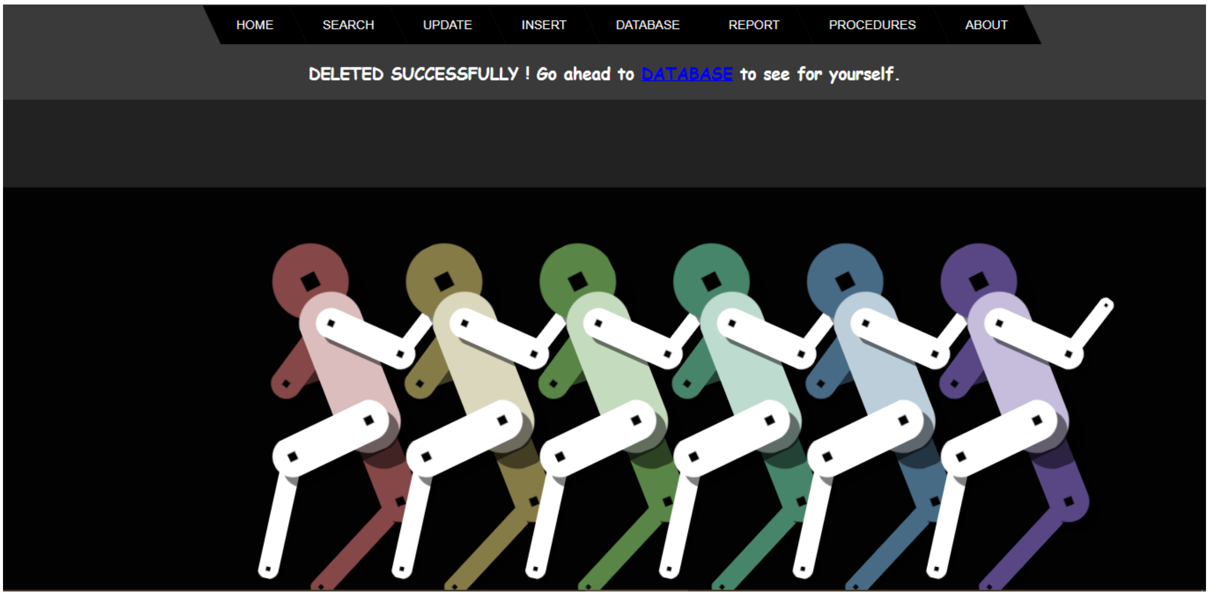


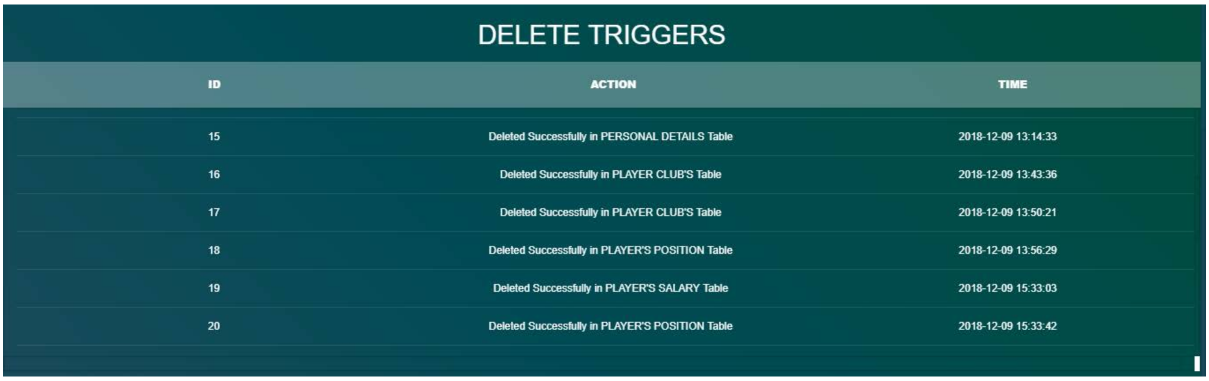
Fig 5.8 Selecting categories for deleting record



DELETED SUCCESSFULLY ! Go ahead to DATABASE to see for yourself.

Fig 5.9 Successful deletion instance

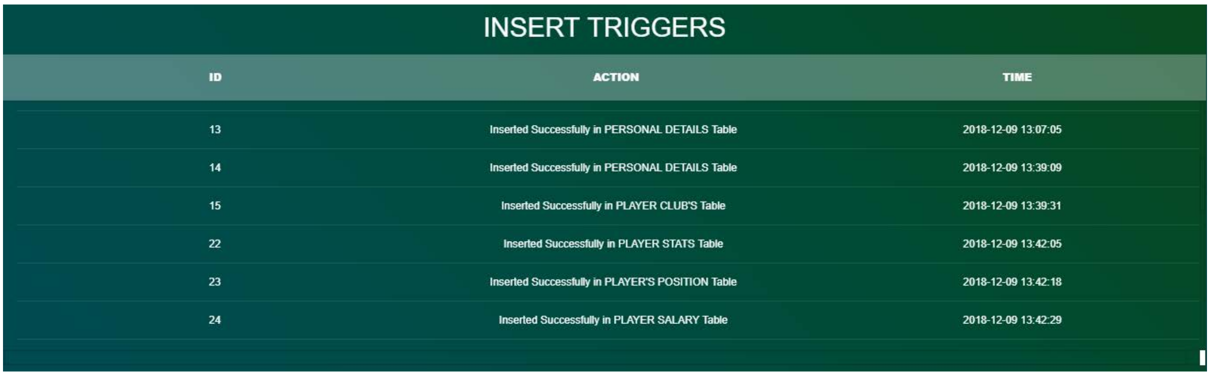
5.5 Triggers:



DELETE TRIGGERS

ID	ACTION	TIME
15	Deleted Successfully in PERSONAL DETAILS Table	2018-12-09 13:14:33
16	Deleted Successfully in PLAYER CLUB'S Table	2018-12-09 13:43:36
17	Deleted Successfully in PLAYER CLUB'S Table	2018-12-09 13:50:21
18	Deleted Successfully in PLAYER'S POSITION Table	2018-12-09 13:56:29
19	Deleted Successfully in PLAYER'S SALARY Table	2018-12-09 15:33:03
20	Deleted Successfully in PLAYER'S POSITION Table	2018-12-09 15:33:42

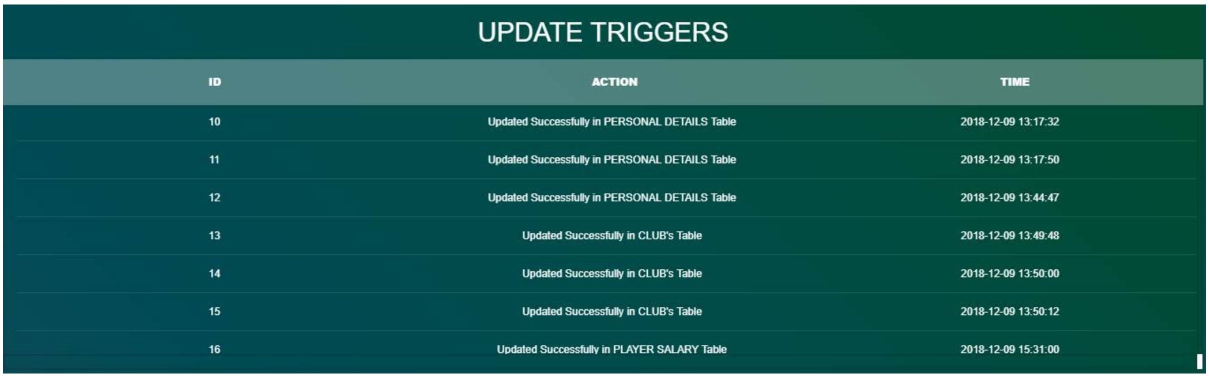
Fig 5.10 Delete triggers table



INSERT TRIGGERS

ID	ACTION	TIME
13	Inserted Successfully in PERSONAL DETAILS Table	2018-12-09 13:07:05
14	Inserted Successfully in PERSONAL DETAILS Table	2018-12-09 13:39:09
15	Inserted Successfully in PLAYER CLUB'S Table	2018-12-09 13:39:31
22	Inserted Successfully in PLAYER STATS Table	2018-12-09 13:42:05
23	Inserted Successfully in PLAYER'S POSITION Table	2018-12-09 13:42:18
24	Inserted Successfully in PLAYER SALARY Table	2018-12-09 13:42:29

Fig 5.11 Insert triggers table

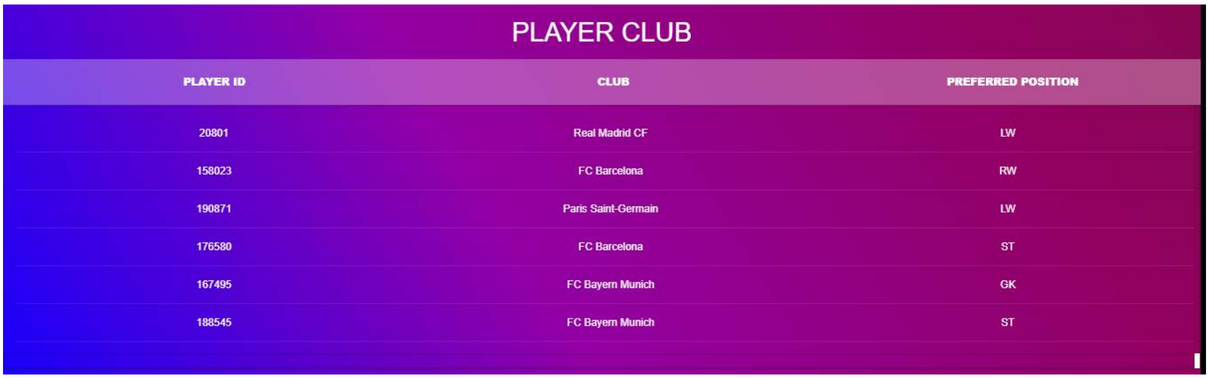


UPDATE TRIGGERS

ID	ACTION	TIME
10	Updated Successfully in PERSONAL DETAILS Table	2018-12-09 13:17:32
11	Updated Successfully in PERSONAL DETAILS Table	2018-12-09 13:17:50
12	Updated Successfully in PERSONAL DETAILS Table	2018-12-09 13:44:47
13	Updated Successfully in CLUB's Table	2018-12-09 13:49:48
14	Updated Successfully in CLUB's Table	2018-12-09 13:50:00
15	Updated Successfully in CLUB's Table	2018-12-09 13:50:12
16	Updated Successfully in PLAYER SALARY Table	2018-12-09 15:31:00

Fig 5.12 Update triggers table

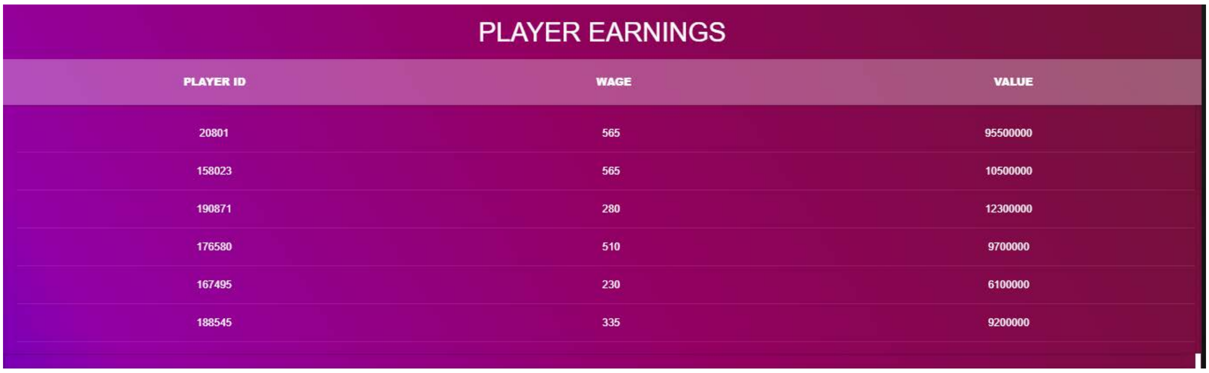
5.6 Database tables:



PLAYER CLUB

PLAYER ID	CLUB	PREFERRED POSITION
20801	Real Madrid CF	LW
158023	FC Barcelona	RW
190871	Paris Saint-Germain	LW
176580	FC Barcelona	ST
167495	FC Bayern Munich	GK
188545	FC Bayern Munich	ST

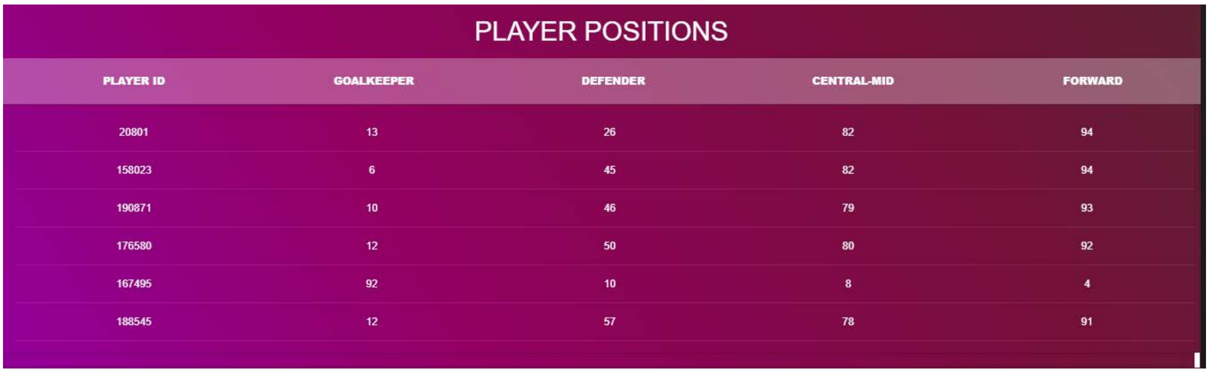
Fig 5.13 Player club table



PLAYER EARNINGS

PLAYER ID	WAGE	VALUE
20801	565	95500000
158023	565	10500000
190871	280	12300000
176580	510	9700000
167495	230	6100000
188545	335	9200000

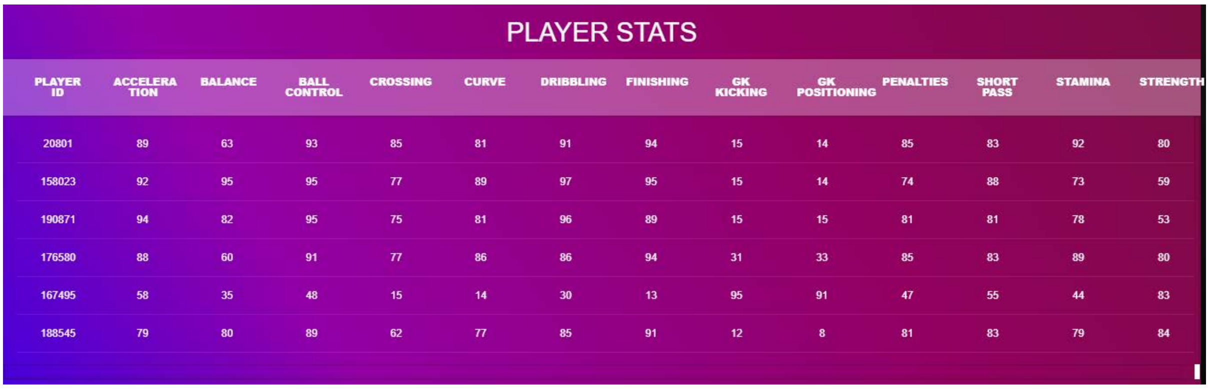
Fig 5.14 Player salary table



PLAYER POSITIONS

PLAYER ID	GOALKEEPER	DEFENDER	CENTRAL-MID	FORWARD
20801	13	26	82	94
158023	6	45	82	94
190871	10	46	79	93
176580	12	50	80	92
167495	92	10	8	4
188545	12	57	78	91

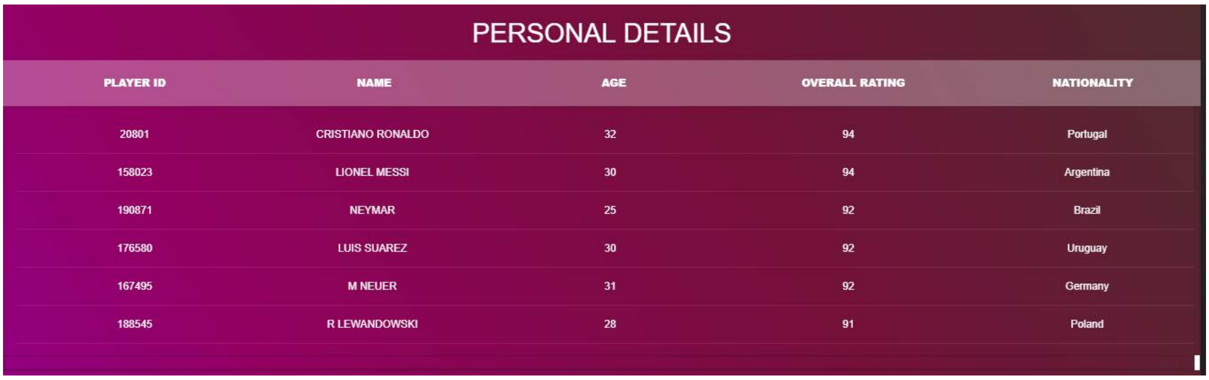
Fig 5.15 Player position table



PLAYER STATS

PLAYER ID	ACCELERA TION	BALANCE	BALL CONTROL	CROSSING	CURVE	DRIBBLING	FINISHING	GK KICKING	GK POSITIONING	PENALTIES	SHORT PASS	STAMINA	STRENGTH
20801	89	63	93	85	81	91	94	15	14	85	83	92	80
158023	92	95	95	77	89	97	95	15	14	74	88	73	59
190871	94	82	95	75	81	96	89	15	15	81	81	78	53
176580	88	60	91	77	86	86	94	31	33	85	83	89	80
167495	58	35	48	15	14	30	13	95	91	47	55	44	83
188545	79	80	89	62	77	85	91	12	8	81	83	79	84

Fig 5.16 Player stats table



PERSONAL DETAILS

PLAYER ID	NAME	AGE	OVERALL RATING	NATIONALITY
20801	CRISTIANO RONALDO	32	94	Portugal
158023	LIONEL MESSI	30	94	Argentina
190871	NEYMAR	25	92	Brazil
176580	LUIS SUAREZ	30	92	Uruguay
167495	M NEUER	31	92	Germany
188545	R LEWANDOWSKI	28	91	Poland

Fig 5.17 Player’s table

CONCLUSION

This project is developed to nurture the needs of a user/scouting agent to monitor players and inspect their technicalities from every aspect on a football field. This is a computerized version of player management system which will benefit the players as well as the staff of a club.

In this entire process one can search player details, add new skilled players, Update ratings and view all the player statistics. The software takes care data and carefully stores all the player information. It provides security and encapsulation by the use of stored procedures.

FUTURE SCOPE

There is a future scope of this project is to help managers and club staffs to get out the best youth talent across the world. Features like predicting players rating based on their current performances and training sessions helps team staffs to judge players according to the club’s needs.

BIBLIOGRAPHY

[1] Database System Model, Languages, Design and Application Programming, Ramez Elmasri and ShamKant B. Navathe, 7th edition, 2017, Pearson.

[2] Database Management System, Ramakrishnan, and Gehrke, 3rd edition, 2014, Mc-GrawHill.

Websites Referred:

- https://stackoverflow.com/search?q=insert+table
- https://stackoverflow.com/search?q=view+table
- https://stackoverflow.com/search?q=search+from+table
- www.quora.com
- https://www.codeproject.com/search.aspx?q=php+code+for+radio+buttons&x=0&y=0&sbo=qa
- https://www.codeproject.com/Articles/38808/Overview-of-SQL-Server-database-Triggers
- https://www.techrepublic.com/article/determine-when-to-use-stored-procedures-vs-sql-in-the-code/
- https://codepen.io/
- https://www.uplabs.com/
- https://www.ibm.com/support/knowledgecenter/en/SS6NHC/com.ibm.swg.im.dashdb.sql.ref.doc/doc/c0004100.html
- https://www.w3schools.com/html/default.asp
- https://www.w3schools.com/css/default.asp
- https://www.w3schools.com/php/default.asp
- https://www.ibm.com/support/knowledgecenter/en/SSEPEK_10.0.0/apsg/src/tpc/db2z_storedprocedure.html