



(Company No. 101067-P)

الجامعة الإسلامية العالمية ماليزيا
INTERNATIONAL ISLAMIC UNIVERSITY MALAYSIA
يُونَيْتِي اِسْلَامُ اِنْتَارَا بَغْسِيَا مِلْدِيَا

Garden of Knowledge and Virtue

PREMIER INTERNATIONAL ISLAMIC RESEARCH UNIVERSITY

ECIE 4348 : DATABASE SYSTEMS

Project: IIUM Sport Facilities Booking

SEM 1, 2020/2021

Lecturer: Dr. Malik Arman Morshidi

Group 6:

<u>No.</u>	<u>Name</u>	<u>Matric Number</u>
1	Syed Zarif Farhannudin Bin Syed Zainal Abidin	1624665
2	Muhammad Farhan Bin Mohamedon	1620041
3	Muhamad Azam Asri Bin Azman	1622503
4	Nik Muhammad Shahmi Bin Nik Mat Kamil	1620395
5	Nur Irsya Aiman Bin Mohd Isa	1627705

Table of Contents

1.0 Introduction.....	2
1.1 Project Summary	2
1.2 Objectives.....	2
1.3 Project Scope.....	2
2.0 Analysis and Discussion	3
2.1 User Interface (End User).	3
2.2 Database	7
2.3 End User (Code for Reservation)	13
2.4 Admin (Log In)	18
2.5 Admin Home Page	19
2.6 Admin (Accounts)	20
2.7 Admin (Booking Confirmation).....	26
2.8 Admin (Facility).....	30
3.0 Conclusion	33

1.0 Introduction

1.1 Project Summary

This project is about a booking system of available sport facilities in IIUM. The main purpose is to replace the outdated manual booking system and to alleviate the booking process. The current system which the guests need to walk into the sport complex and fill in form to book the facilities is inconvenient and quite troublesome. In response to this problem, we came out with a booking system using PHP, SQL and HTML that includes interactive queries to store data in the database. The webpage is simple and user friendly to make sure the system meets the user needs.

1.2 Objectives

The aims and objectives of this project are as follows:

- To implement the knowledge and techniques we have learned in Web Database Programming using PHP and MySQL.
- To create an interactive and user-friendly booking system for sport facilities in IIUM.
- To create a database to store the booking and user data.

1.3 Project Scope

The booking system will be used not only by the IIUM community, the visitors can also book the sports facilities easily if they wish to use one.

2.0 Analysis and Discussion

2.1 User Interface (End User).

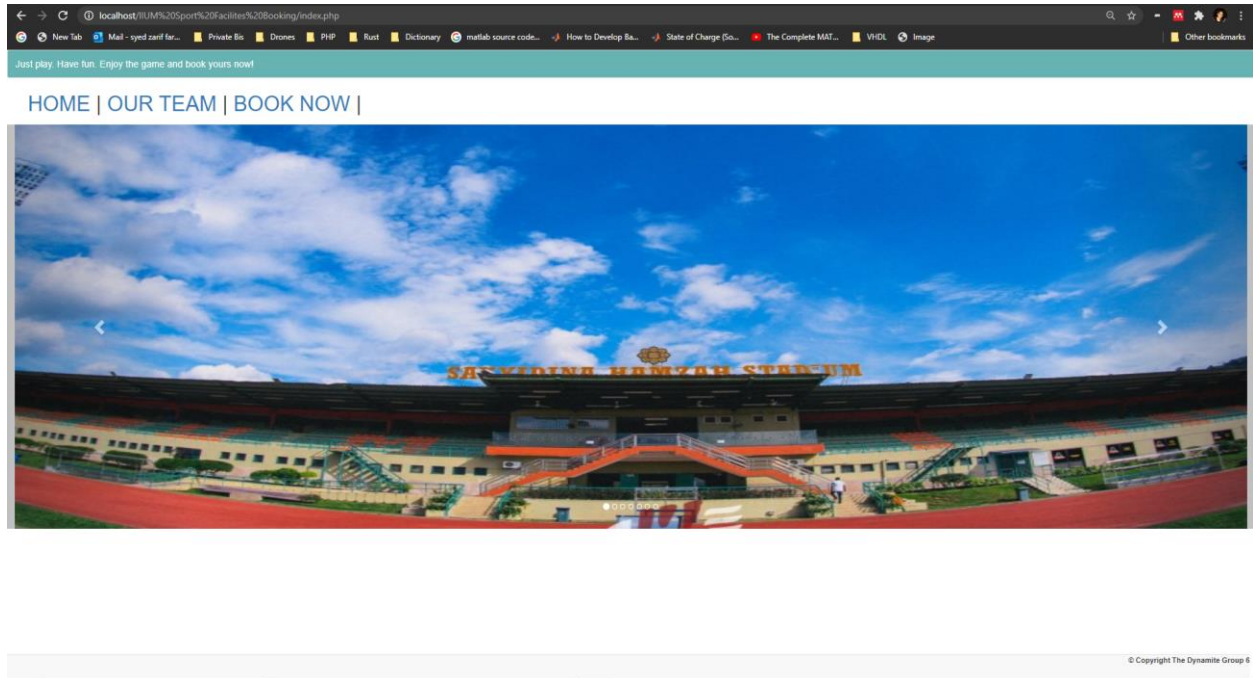


Figure 2.0

This is the homepage for our booking system. The user will be greeted with a slideshow of cover photos. It has a navigation bar with options of 'Home', 'Our Team' and 'Book Now'. The 'Home' will take the user back to the homepage, 'Our Team' will bring the user to the about us page, and 'Book Now' is the page which the user will book the facilities.

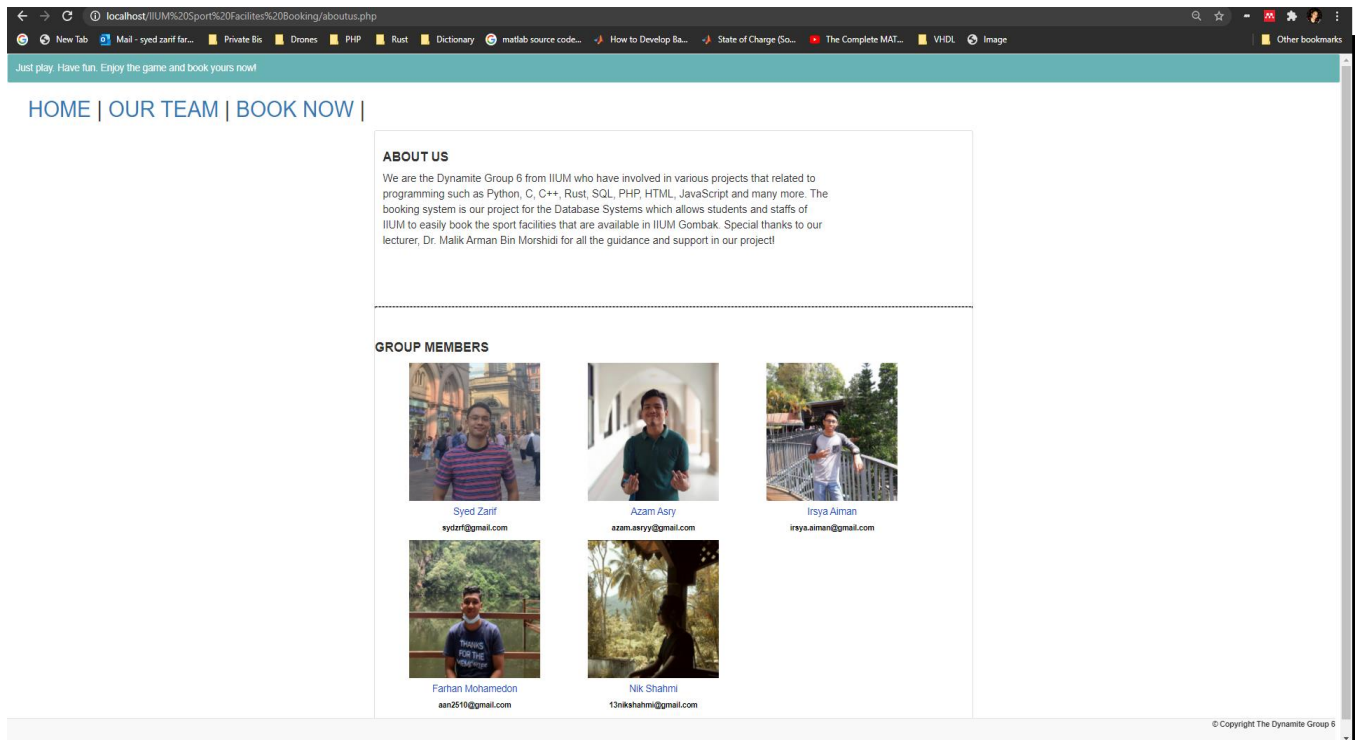


Figure 2.1

This is the ‘Our Team’ page. It features our team members and the details about what the system is about as well as our group members.

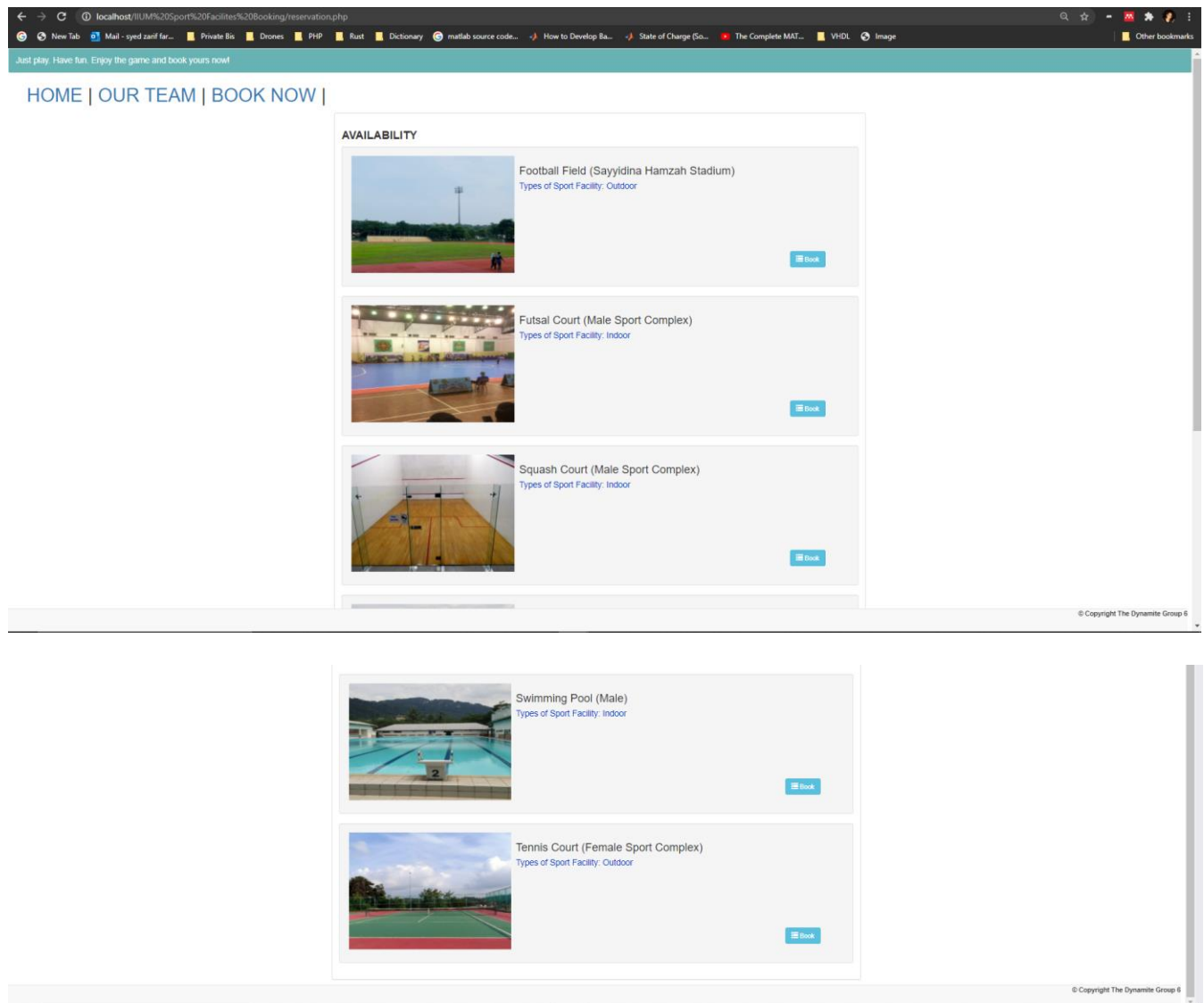


Figure 2.2

Figure above is the most important part of the booking system, the ‘Book Now’ page. This is where the users will do their bookings. The page shows several facilities with their details that can be booked by IIUM community and visitors.

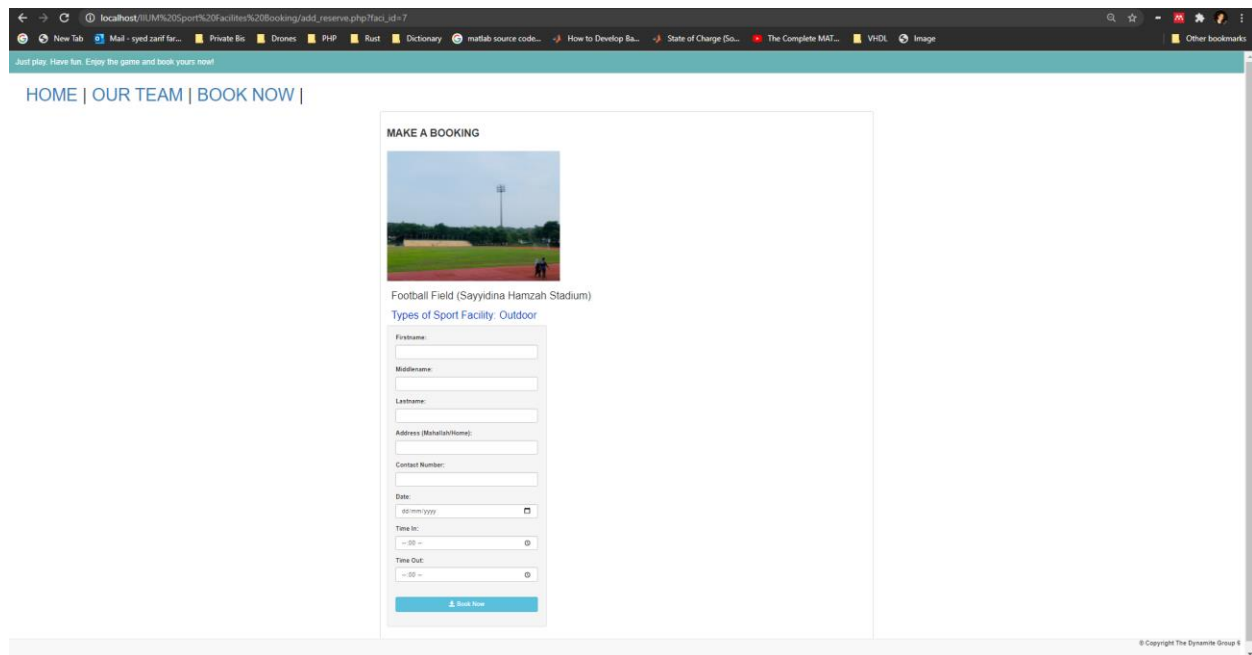


Figure 2.3

Once they click their desired facilities, they need to fill in the required details for the booking, which includes Name, Contact Number, Date and Time.

2.2 Database

```
51 CREATE TABLE `guest` (  
52   `guest_id` int(11) NOT NULL,  
53   `firstname` varchar(50) NOT NULL,  
54   `middlename` varchar(30) NOT NULL,  
55   `lastname` varchar(40) NOT NULL,  
56   `address` varchar(50) NOT NULL,  
57   `contactno` varchar(13) NOT NULL  
58 ) ENGINE=InnoDB DEFAULT CHARSET=latin1;  
59  
60 -----  
61  
62 --  
63 -- Table structure for table `faci`  
64 --  
65  
66 CREATE TABLE `faci` (  
67   `faci_id` int(11) NOT NULL,  
68   `faci_sport` varchar(50) NOT NULL,  
69   `faci_type` varchar(11) NOT NULL,  
70   `photo` varchar(100) NOT NULL  
71 ) ENGINE=InnoDB DEFAULT CHARSET=latin1;  
72  
73 --  
74 -- Dumping data for table `faci`  
75 --  
76  
77 INSERT INTO `faci` (`faci_id`, `faci_sport`, `faci_type`, `photo`) VALUES  
78 (1, 'Futsal Court (Male Sport Complex)', 'Indoor', '1futsal.jpg'),  
79 (2, 'Tennis Court (Female Sport Complex)', 'Outdoor', '3tennis.jpg'),  
80 (3, 'Swimming Pool (Male)', 'Indoor', '4swim.jpg'),  
81 (4, 'Squash Court (Male Sport Complex)', 'Indoor', '5squash.jpg'),  
82 (5, 'Football Field (Sayyidina Hamzah Stadium)', 'Outdoor', '6ball.jpg');  
83  
84 -----  
85  
86 --  
87 -- Table structure for table `booking`  
88 --  
89  
90 CREATE TABLE `booking` (  
91   `booking_id` int(11) NOT NULL,  
92   `guest_id` int(11) NOT NULL,  
93   `faci_id` int(11) NOT NULL,  
94   `faci_no` int(5) NOT NULL,  
95   `status` varchar(20) NOT NULL,  
96   `date_in` date NOT NULL,  
97   `checkin_time` time NOT NULL,  
98   `checkout_time` time NOT NULL  
99 ) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```



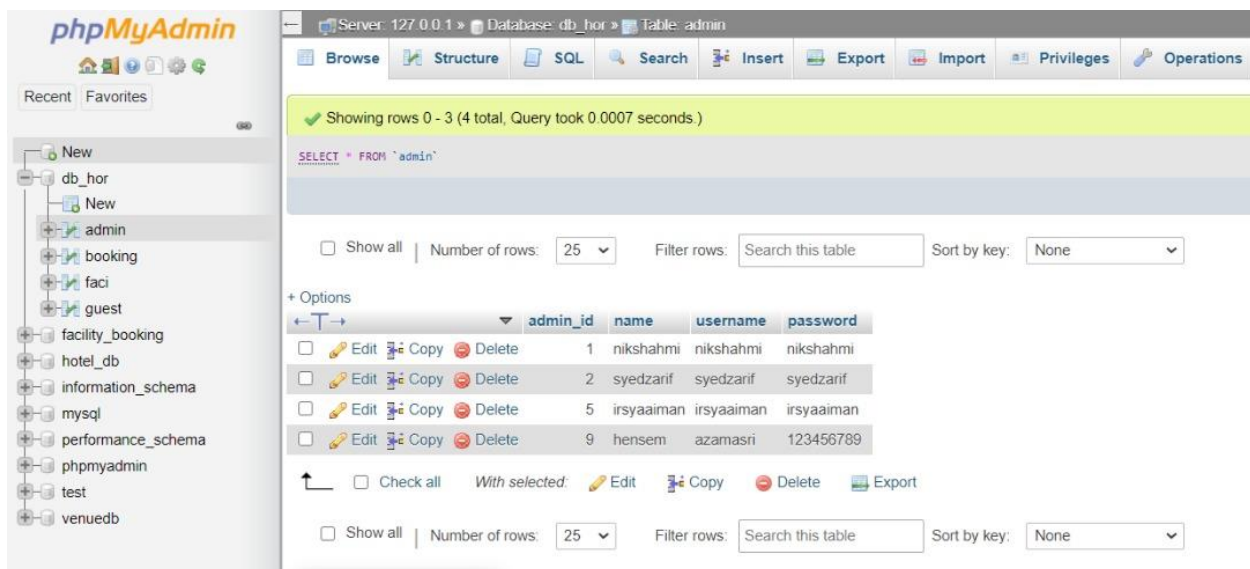
```

108 ALTER TABLE `admin`
109     ADD PRIMARY KEY (`admin_id`);
110
111 --
112 -- Indexes for table `guest`
113 --
114 ALTER TABLE `guest`
115     ADD PRIMARY KEY (`guest_id`);
116
117 --
118 -- Indexes for table `faci`
119 --
120 ALTER TABLE `faci`
121     ADD PRIMARY KEY (`faci_id`);
122
123 --
124 -- Indexes for table `booking`
125 --
126 ALTER TABLE `booking`
127     ADD PRIMARY KEY (`booking_id`);
128
129 --
130 -- AUTO_INCREMENT for dumped tables
131 --
132
133 --
134 -- AUTO_INCREMENT for table `admin`
135 --
136 ALTER TABLE `admin`
137     MODIFY `admin_id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=2;
138
139 --
140 -- AUTO_INCREMENT for table `guest`
141 --
142 ALTER TABLE `guest`
143     MODIFY `guest_id` int(11) NOT NULL AUTO_INCREMENT;
144
145 --
146 -- AUTO_INCREMENT for table `faci`
147 --
148 ALTER TABLE `faci`
149     MODIFY `faci_id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=6;
150
151 --
152 -- AUTO_INCREMENT for table `transaction`
153 --
154 ALTER TABLE `booking`
155     MODIFY `booking_id` int(11) NOT NULL AUTO_INCREMENT;COMMIT;
156
157

```

In the database SQL file, there are a total of 4 tables created. The first one is an 'admin' table which is used to store admin related information which include admin id, admin name, username and password. Next is the 'guest' table which specifically uses information related to the guest. The information includes the guest id, guest first, middle and last name, address and contact number. The third table is 'faci' table which is used to store facility related information like facility id, facility name, type of facility and the picture of facility. Lastly, the 'booking' table is used to store booking id, guest id, facility id, facility number, status of booking, date in and out, and also check in and out time.

All the tables will be altered overtime when new data is inserted into the table. For example, when a new admin has been registered into the system, the table will be modified, and new admin related data will be stored in the table. Variables like admin id, guest id and facility id will be auto increment in the process of entering new data.



The screenshot shows the phpMyAdmin interface for a database named 'db_hor'. The 'admin' table is selected, and its structure and data are displayed. The table has four columns: 'admin_id', 'name', 'username', and 'password'. The data is as follows:

admin_id	name	username	password
1	nikshahmi	nikshahmi	nikshahmi
2	syedzarif	syedzarif	syedzarif
5	irsyaaiman	irsyaaiman	irsyaaiman
9	hensem	azamasri	123456789

Figure 2.3

Currently there are four administrators stored in the ‘admin’ database table. Through using phpMyAdmin software, we can edit, copy or delete the administrator related information in the database. There are a total of 4 keys in the table and the primary key is the admin_id while the others are secondary key.

	booking_id	guest_id	faci_id	faci_no	status	date_in	checkin_time	checkout_time
<input type="checkbox"/>	15	21	7	3	Booked	2021-01-08	18:00:00	19:00:00
<input type="checkbox"/>	13	19	1	1	Booked	2021-01-28	15:00:00	16:00:00
<input type="checkbox"/>	11	16	1	2	Booked	2021-01-28	14:00:00	15:00:00
<input type="checkbox"/>	18	24	7	0	Pending	2021-01-12	14:00:00	16:00:00
<input type="checkbox"/>	19	24	7	0	Pending	2021-01-12	14:00:00	16:00:00
<input type="checkbox"/>	17	23	2	0	Pending	2021-01-08	13:00:00	14:00:00
<input type="checkbox"/>	16	22	7	0	Pending	2021-01-07	11:00:00	12:00:00

Figure 2.4

In the ‘booking’ database table, there are currently 7 guests who have booked a facility. The primary keys are booking id, and faci_id and guest_id are foreign keys. While the others are secondary keys. The database can also be edited, copied or deleted.

Server: 127.0.0.1 » Database: db_hor » Table: guest

Options: Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

	guest_id	firstname	middlename	lastname	address	contactno
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	4	Syed Zarif Farhannudin Bin	sef	Syed Zainal Abidin	NO. 6, JALAN PP 56, TAMAN PINGGIRAN PUTRA	0192770609
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	5	Syed Zarif Farhannudin Bin	sef	Syed Zainal Abidin	NO. 6, JALAN PP 56, TAMAN PINGGIRAN PUTRA	0192770609
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	6	Syed Zarif Farhannudin Bin	sef	Syed Zainal Abidin	NO. 6, JALAN PP 56, TAMAN PINGGIRAN PUTRA	0192770609
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	7	Syed Zarif Farhannudin Bin	sef	Syed Zainal Abidin	NO. 6, JALAN PP 56, TAMAN PINGGIRAN PUTRA	0192770609
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	8	Syed Zarif Farhannudin Bin	sef	Syed Zainal Abidin	NO. 6, JALAN PP 56, TAMAN PINGGIRAN PUTRA	0192770609
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	9	Syed Zarif Farhannudin Bin	12	Syed Zainal Abidin	NO. 6, JALAN PP 56, TAMAN PINGGIRAN PUTRA	0192770609
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	10	Syed	Zarif	Farhannudin	ALI E1.6	0192770609
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	11	Syed Zarif Farhannudin Bin	Zarif	Syed Zainal Abidin	NO. 6, JALAN PP 56, TAMAN PINGGIRAN PUTRA	0192770609
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	12	Syed Zarif Farhannudin Bin	Zarif	Syed Zainal Abidin	NO. 6, JALAN PP 56, TAMAN PINGGIRAN PUTRA	0192770609
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	13	Siti	Aminah	Rashid	ASIAH B2.2	011117458
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	14	Syed Zarif Farhannudin Bin	Zarif	Syed Zainal Abidin	NO. 6, JALAN PP 56, TAMAN PINGGIRAN PUTRA	0192770609
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	15	Muhd	Azam	Asri	Mahallah Ali	0193075660
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	16	Syed	Zarif	Farhannudin	ALI E1.6	013252562
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	17	zam	asry	az	Mahallah Zubair	011122548745
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	18	zam	asri	aish	mahallah safiyyah	0125896585
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	19	hey	ho	hi	mahallah safiyyah	0123658596
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	20	carry	the	zarep	Mahallah Ali E1.6	0124567898
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	21	Azam	Asri	Azman	Mahallah Ali E1.5	0132232528
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	22	Nur Insyia Aiman	Bin	Mohd Isa	Zubair Block G	01120812990
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	23	Ali	Bin	Abu	Mahallah Ali	01120922990
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	24	Muhammad Farhan	bin	Mohamedon	Mahallah Zubair	0123456789
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	25	Muhammad Farhan	bin	Mohamedon	Mahallah Zubair	0123456789

Figure 2.5

The 'guest' database table will store the details like name, address, contact number, date & time that are queried from the booking form. The primary key is the guest_id, the foreign keys is faci_id, while the rest are secondary keys.

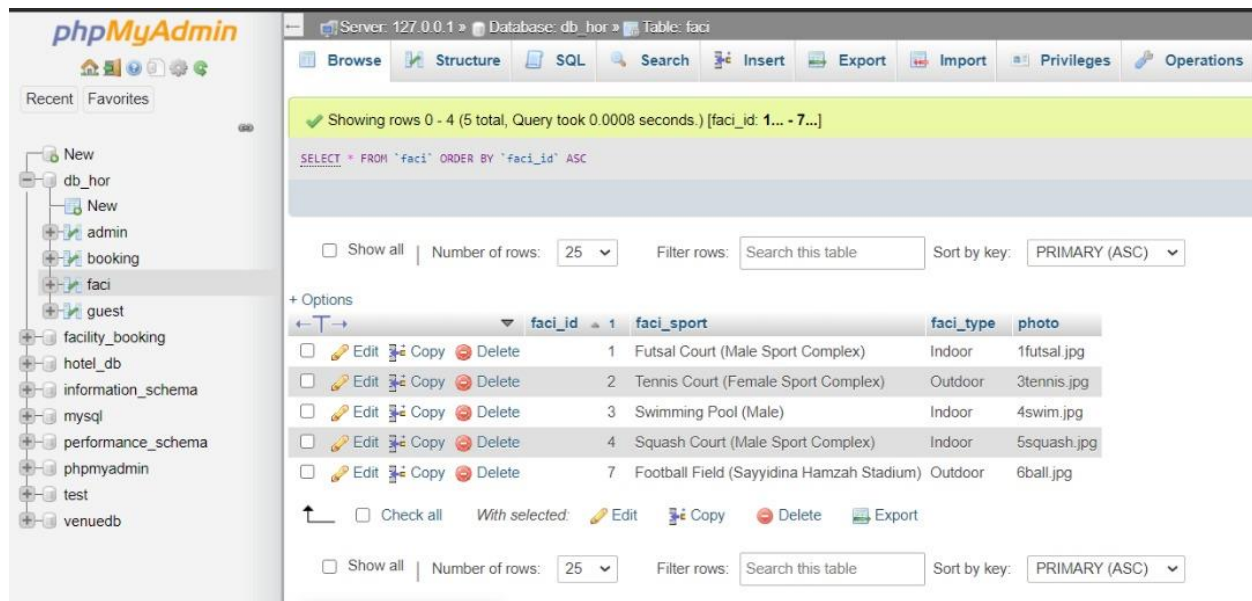


Figure 2.6

As for the facilities provided, the system provides five options to book the available facilities listed as Futsal Court (Male Sport Complex), Tennis Court (Female Sport Complex), Swimming Pool (Male), Squash Court (Male Sport Complex) and Football Field (Sayyidina Hamzah Stadium) with `faci_id` as 1, 2, 3, 4 and 7 respectively. The primary key here is '`faci_id`'. The '`faci_type`' attribute describes the type of the facility whether it is indoor or outdoor. Each facility is also provided with the picture of the facility in providing better reference and display for users.

2.3 End User (Code for Reservation)



Figure 2.7

```
23 <div style = "margin-left:50%;" class = "container">
24 <div class = "panel panel-default">
25 <div class = "panel-body">
26 <strong><h3><b>AVAILABILITY</b></h3></strong>
27 <?php
28 require_once 'admin/connect.php';
29 $query = $conn->query("SELECT * FROM 'faci' ORDER BY 'faci_sport' ASC") or die(mysql_error());
30 while($fetch = $query->fetch_array()){
31
32 }
33 <div class = "well" style = "height:300px; width:100%;">
34 <div style = "float:left;">
35 <img src = "photo/<?php echo $fetch['photo']?>" height = "250" width = "350"/>
36 </div>
37 <div style = "float:left; margin-left:10px;">
38 <h3><?php echo $fetch['faci_sport']?></h3>
39 <h4 style = "color:#0033cc;"><?php echo "Types of Sport Facility: ".$fetch['faci_type']. ""?></h4>
40 <br /><br /><br /><br /><br />
41 <a style = "margin-left:580px;" href = "add_reserve.php?faci_id=<?php echo $fetch['faci_id']?>" class = "btn btn-info"><i class = "glyphicon glyphicon-list"></i> Book</a>
42 </div>
43 </div>
44 <?php
45 }
46 </div>
```

This reservation.php code will be for the Book button. When the user clicks the button, it will bring the user to the booking details page. In line 29, the code will query from the 'faci' table, and it will show the facilities name (faci_sport) in ascending order as shown in Figure 2.2. Once the user clicks, the code will fetch the 'faci_id' and will be referred to add_reserve.php.

```

23 <div style = "margin-left:50;" class = "container">
24 <div class = "panel panel-default">
25 <div class = "panel-body">
26 <strong>h3<b>MAKE A BOOKING</b></h3></strong>
27 <br />
28 <?php
29     require_once 'admin/connect.php';
30     $query = $conn->query("SELECT * FROM 'faci' WHERE 'faci_id' = '$_REQUEST[faci_id]') or die(mysql_error());
31     $fetch = $query->fetch_array();
32 >?>
33 <div style = "height:300px; width:800px;">
34 <div style = "float:left;">
35 <img src = "photo/<?php echo $fetch['photo']?>" height = "300px" width = "400px">
36 </div>
37 <br style = "clear:both;" />
38
39 <div style = "float:left; margin-left:10px;">
40 <h3><?php echo $fetch['faci_sport']?></h3>
41 <h3 style = "color:#0033cc;"><?php echo "Types of Sport Facility: ".$fetch['faci_type'].","?></h3>
42 </div>
43 </div>
44 <br style = "clear:both;" />
45 <div class = "well col-md-4">
46 <form method = "POST" enctype = "multipart/form-data">
47 <div class = "form-group">
48 <label>Firstname:</label>
49 <input type = "text" class = "form-control" name = "firstname" required = "required" />
50 </div>
51 <div class = "form-group">
52 <label>Middlename:</label>
53 <input type = "text" class = "form-control" name = "middlename" required = "required" />
54 </div>
55 <div class = "form-group">
56 <label>Lastname:</label>
57 <input type = "text" class = "form-control" name = "lastname" required = "required" />
58 </div>
59 <div class = "form-group">
60 <label>Address (Mahallah/Home):</label>
61 <input type = "text" class = "form-control" name = "address" required = "required" />
62 </div>
63 <div class = "form-group">
64 <label>Contact Number:</label>
65 <input type = "text" class = "form-control" name = "contactno" required = "required" />
66 </div>
67 <div class = "form-group">
68 <label>Date:</label>
69 <input type = "date" class = "form-control" name = "datein" required = "required" />
70 </div>
71
72 <div class = "form-group">
73 <label>Time In:</label>
74 <input type = "time" step="3600" min="08:00" max="22:00" required class = "form-control" name = "timein" required = "required" />
75 </div>
76 <div class = "form-group">
77 <label>Time Out:</label>
78 <input type = "time" step="3600" min="08:00" max="22:00" required class = "form-control" name = "timeout" required = "required" />
79 </div>
80
81 <br />
82 <div class = "form-group">
83 <button class = "btn btn-info form-control" name = "add_guest"><i class = "glyphicon glyphicon-save"></i> Book Now</button>
84 </div>
85 </form>
86 </div>
87 </div>
88 <div class = "col-md-4"></div>
89 <?php require_once 'add_query_reserve.php'?>
90 </div>
91 </div>
92 <br />
93 <br />
94 <div style = "text-align:right; margin-right:10px;" class = "navbar navbar-default navbar-fixed-bottom">
95 <label>©copy; Copyright The Dynamite Group 6</label>
96 </div>
97 </body>
98 <script src = "js/jquery.js"></script>
99 <script src = "js/bootstrap.js"></script>
100
101 </html>

```

This is where the user will reserve their facilities. The code will inquire the user to input their details for the booking. When the user clicks Book Now, the code will refer to add_query_reserve.php.


```

1 <?php
2 require_once 'admin/connect.php';
3 if(isset($_POST['add_guest'])){
4     $firstname = $_POST['firstname'];
5     $middlename = $_POST['middlename'];
6     $lastname = $_POST['lastname'];
7     $address = $_POST['address'];
8     $contactno = $_POST['contactno'];
9     $date_in = $_POST['datein'];
10    $checkin_time = $_POST['timein'];
11    $checkout_time = $_POST['timeout'];
12
13    $conn->query("INSERT INTO 'guest' (firstname, middlename, lastname, address, contactno) VALUES('$firstname', '$middlename', '$lastname', '$address', '$contactno')") or die(mysqli_error());
14    $query = $conn->query("SELECT * FROM 'guest' WHERE 'firstname' = '$firstname' && 'lastname' = '$lastname' && 'contactno' = '$contactno'") or die(mysqli_error());
15    $fetch = $query->fetch_array();
16    $query2 = $conn->query("SELECT * FROM 'booking' WHERE 'date_in' = '$date' && 'booking_id' = '$REQUEST[booking_id]' &&
17    'status' = 'Pending' && 'checkin_time' = 'timein' && 'checkout_time' = 'timeout'") or die(mysqli_error());
18    $row = $query2->num_rows;
19    if($date_in < date("Y-m-d", strtotime("+8 HOURS"))){
20        echo "<script>alert('Must be present date!')</script>";
21    }else{
22        if($row > 0){
23            echo "<div class = 'col-md-4'>
24                <label style = 'color:#ff0000;'>Not Available Date</label><br />";
25            $q_date = $conn->query("SELECT * FROM 'booking' WHERE 'status' = 'Pending'") or die(mysqli_error());
26            while($q_date = $q_date->fetch_array()){
27                echo "<ul>
28                    <li>
29                        <label class = 'alert-danger;'>".date("M d, Y", strtotime($q_date['date_in']."+8HOURS'))."</label>
30                    </li>
31                </ul>";
32            }
33        }
34    }else{
35        if($guest_id = $fetch['guest_id']){
36            $faci_id = $_REQUEST['faci_id'];
37            $conn->query("INSERT INTO 'booking' (guest_id, faci_id, status, date_in, checkin_time, checkout_time)
38            VALUES('$guest_id', '$faci_id', 'Pending', '$date_in', '$checkin_time', '$checkout_time'") or die(mysqli_error());
39            header("location:reply_reserve.php");
40        }else{
41            echo "<script>alert('Error Javascript Exception!')</script>";
42        }
43    }

```

This code will query the database about the user's detail as shown in the Figure 2.5 below.

For some of the query, we have put restrictions like for the user Name, it should not contain numbers, Contact Number should not contain alphabets and Date should be a later date. For Time, the value must be between 8AM to 10PM and the user can only book by hours because the time input has been set its step to 3600. All the queries fetched will be inserted into the database and waiting for admin confirmation. The user then will be referred to reply_reserve.php.

Types of Sport Facility: Outdoor

Firstname:

Muhammad Farhan

Middlename:

bin

Lastname:

Mohamedon

Address (Mahallah/Home):

Mahallah Zubair

Contact Number:

0123456789

Date:

12/01/2021



Time In:

02:00 PM



Time Out:

04:00 PM



 Book Now

7

Figure 2.8

This will be the last page for the user, `reply_reserve.php`. Basically, it only shows the reservation has been made and thanks the user. The user can return to the ‘Book Now’ page (Figure 2.8) by clicking the button as shown in line 35, where it refers to the `reservation.php`.

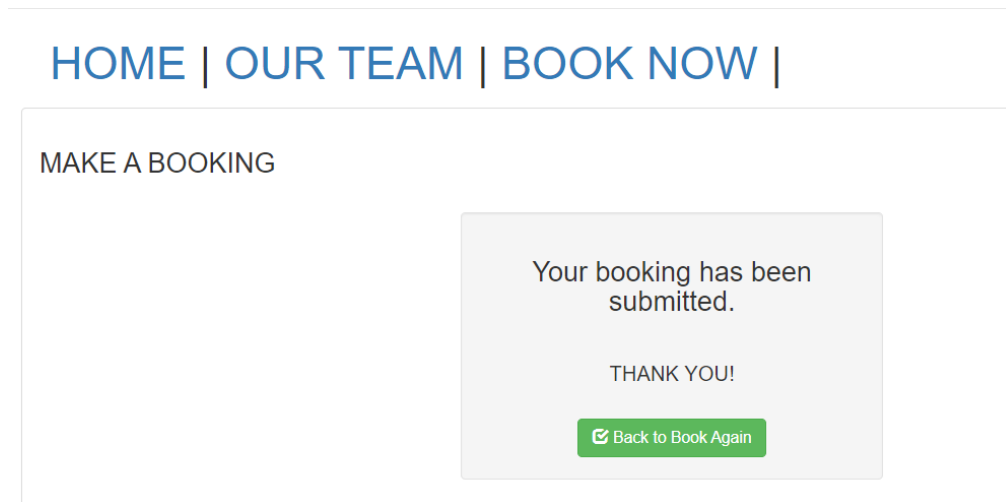


Figure 2.9

```
18 <ul id = "menu">
19 <li><a href = "index.php">HOME</a></li> |
20 <li><a href = "aboutus.php">OUR TEAM</a></li> |
21 <li><a href = "reservation.php">BOOK NOW</a></li> |
22
23 </ul>
24 <div style = "margin-left:0;" class = "container">
25 <div class = "panel panel-default">
26 <div class = "panel-body">
27 <strong><h3>MAKE A BOOKING</h3></strong>
28 <br />
29 <div class = "col-md-4"></div>
30 <div class = "well col-md-4">
31 <center><h3>Your booking has been submitted.</h3></center>
32 <br />
33 <center><h4>THANK YOU!</h4></center>
34 <br />
35 <center><a href = "reservation.php" class = "btn btn-success"><i class = "glyphicon glyphicon-check"></i> Back to Book Again</a></center>
36 </div>
37 <div class = "col-md-4"></div>
38 </div>
39 </div>
40 <br />
41 <br />
42 <br />
43 <div style = "text-align:right; margin-right:10px;" class = "navbar navbar-default navbar-fixed-bottom">
44 <label><small>©copy; Copyright The Dynamite Group 6</small></label>
45 </div>
46 </body>
47 <script src = "js/jquery.js"></script>
48 <script src = "js/bootstrap.js"></script>
49 </html>
```

2.4 Admin (Log In)

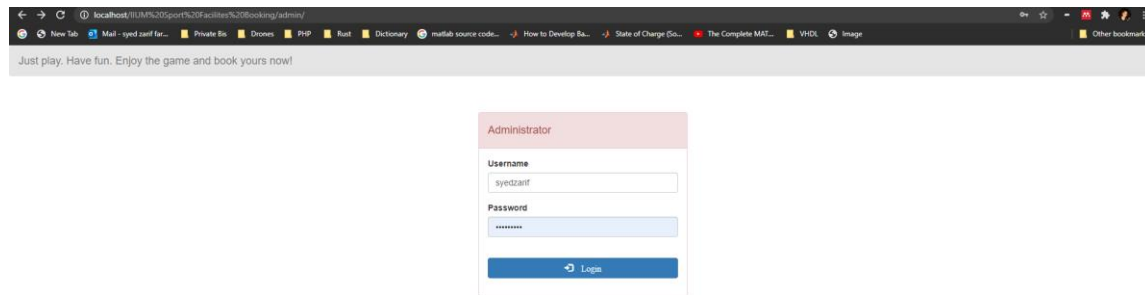


Figure 2.10

Admin is the one who manages user's booking. The database has been added with 5 admins which are the team members of this group. At first, admin will be prompted to the login page where login credentials must be inserted first to login.

```
1 <?php
2 if(isset($_POST['login'])){
3     $username = $_POST['username'];
4     $password = $_POST['password'];
5     $query = $conn->query("SELECT * FROM `admin` WHERE `username` = '$username' && `password` = '$password'") or die(mysqli_error());
6     $fetch = $query->fetch_array();
7     $row = $query->num_rows;
8
9     if($row > 0){
10         session_start();
11         $_SESSION['admin_id'] = $fetch['admin_id'];
12         header('location:home.php');
13     }else{
14         echo "<center><label style = 'color:red;'>Invalid username or password</label></center>";
15     }
16 }
17 >
```

The above code displays the backend of the login page. It is saved as login.php. When the admin inserts his username and password, line 5 in the code will query from the database in a table named as admin. The code will fetch the records of username and password attributes. If what has been inserted by the admin matches what is in the database, the admin's session will start, else it will echo "Invalid username or password."

2.5 Admin Home Page

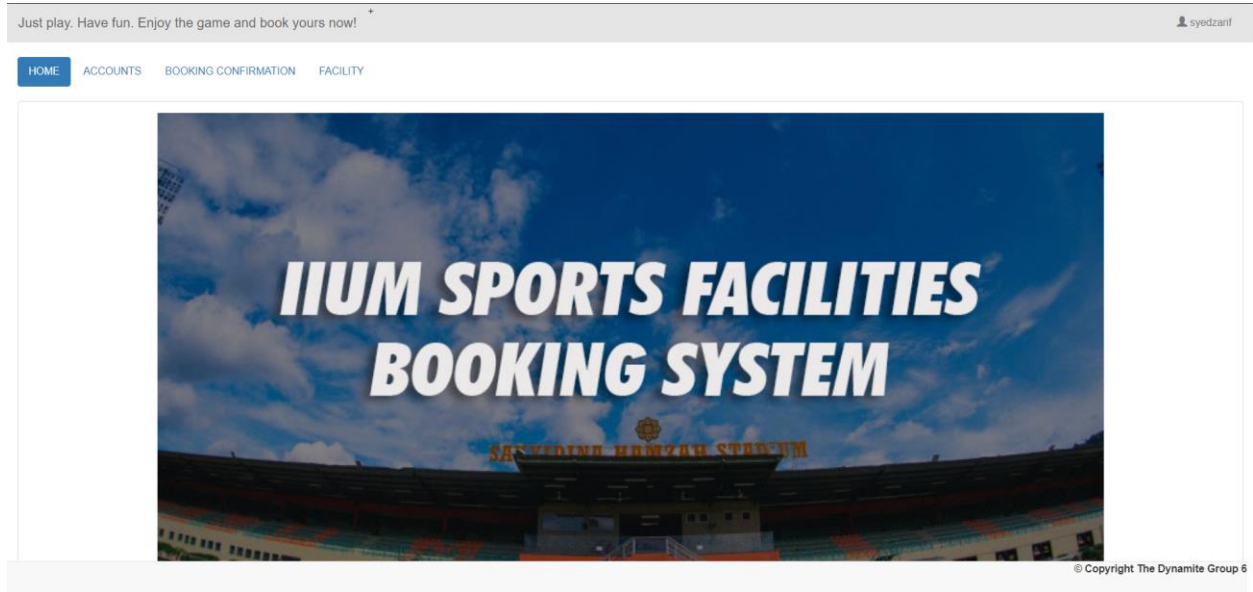


Figure 2.11

Figure 2.11 above shows what is displayed once the admin session starts. In the home page of admin, there are three main functions as you can see on the navigation bar which are the accounts, booking confirmation and facility.

2.6 Admin (Accounts)

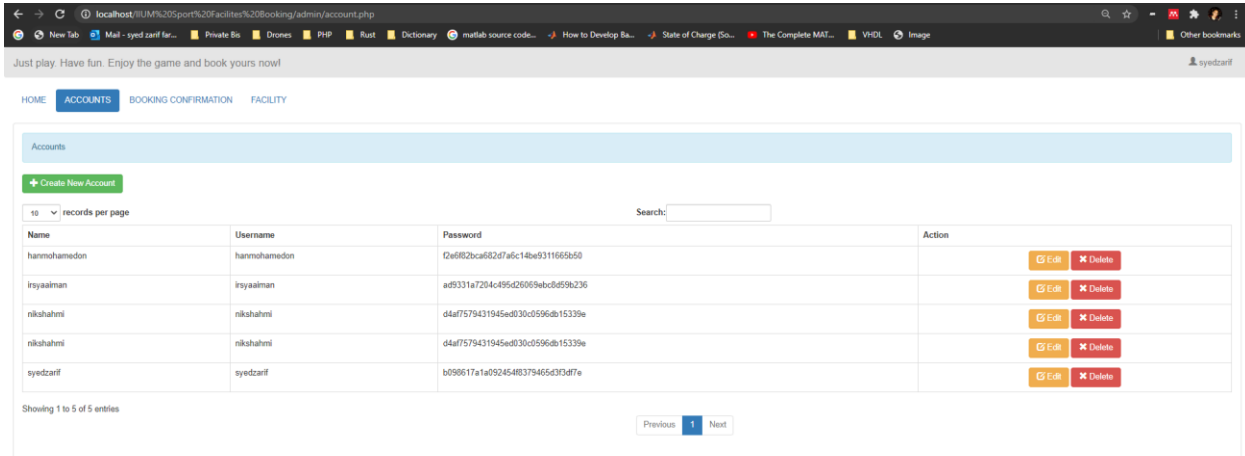


Figure 2.12

When the admin clicks on 'account' on the navigation bar, he will be prompted to the list of admins with their credentials. The admin is able to insert or add new accounts, update or edit any accounts, and delete any accounts with buttons with each featured name respectively.

```
39 <div class = "container-fluid">
40 <div class = "panel panel-default">
41 <div class = "panel-body">
42 <div class = "alert alert-info">Accounts</div>
43 <a class = "btn btn-success" href = "add_account.php"><i class = "glyphicon glyphicon-plus"></i> Create New Account</a>
44 <br />
45 <br />
46 <table id = "table" class = "table table-bordered">
47 <thead>
48 <tr>
49 <th>Name</th>
50 <th>Username</th>
51 <th>Password</th>
52 <th>Action</th>
53 </tr>
54 </thead>
55 <tbody>
56 <?php
57 $query = $conn->query("SELECT * FROM 'admin'" or die(mysqli_error());
58 while($fetch = $query->fetch_array()){
59 <?>
60 <tr>
61 <td><?php echo $fetch['name']?></td>
62 <td><?php echo $fetch['username']?></td>
63 <td><?php echo md5($fetch['password'])?></td>
64 <td><center><a class = "btn btn-warning" href = "edit_account.php?admin_id=<?php echo $fetch['admin_id']?>"><i class = "glyphicon glyphicon-edit"></i> Edit</a>
65 <a class = "btn btn-danger" onclick = "confirmationDelete(this); return false;" href = "delete_account.php?admin_id=<?php echo $fetch['admin_id']?>">
66 <i class = "glyphicon glyphicon-remove"></i> Delete</a></center></td>
67 </tr>
68 <?php
69 }
```

Code above is saved in account.php which is the backend of the account page above.

Clicking on create a new account will prompt the admin to insert a new account for a new admin.

At line 43, the “Create New Account” is linked with `add_account.php` with `href` syntax. Same goes to the “Edit” and “Delete” button where each of them is linked with `edit_account.php` and `delete_account.php` respectively.

Account / Create Account

Name

Username

Password


 Saved

Figure 2.13

Admin must insert the name, username, and password for the new admin to be added to the database in the admin table. The records and attributes will be added with the new credentials.

```

39 <div class = "container-fluid">
40 <div class = "panel panel-default">
41 <div class = "panel-body">
42 <div class = "alert alert-info">Account / Create Account</div>
43 <br />
44 <div class = "col-md-4">
45 <form method = "POST">
46 <div class = "form-group">
47 <label>Name </label>
48 <input type = "text" class = "form-control" name = "name" />
49 </div>
50 <div class = "form-group">
51 <label>Username </label>
52 <input type = "text" class = "form-control" name = "username" />
53 </div>
54 <div class = "form-group">
55 <label>Password </label>
56 <input type = "password" class = "form-control" name = "password" />
57 </div>
58 <br />
59 <div class = "form-group">
60 <button name = "add_account" class = "btn btn-info form-control"><i class = "glyphicon glyphicon-save"></i> Saved</button>
61 </div>
62 </form>
63 <?php require_once 'add_query_account.php'?>
64 </div>
65 </div>
66 </div>
67 </div>

```

Code above is the backend of the add new admin page which is saved as add_account.php. Once the admin clicks on “saved”, add_query_account.php will take part to add the new admin’s credentials into the database. Below is the code in add_query_account.php.

```
1 <?php
2 if(isset($_POST['add_account'])) {
3     $name = $_POST['name'];
4     $username = $_POST['username'];
5     $password = $_POST['password'];
6     $query = $conn->query("SELECT * FROM `admin` WHERE `username` = '$username'" ) or die(mysqli_error());
7     $valid = $conn->num_rows;
8     if($valid > 0) {
9         echo "<center><label style = 'color:red;'>Username already taken</center></label>";
10    }
11    else {
12        $conn->query("INSERT INTO `admin` (name, username, password) VALUES('$name', '$username', '$password')") or die(mysqli_error());
13        header("location:account.php");
14    }
15 }
```

Above code shows record in attribute of “username” is “query” from table “admin”. If the username used is matched with any existing username, it will echo “Username already taken.”, if not, INSERT command is used to add the new admin’s credential into the “admin” table in a tuple.

For edit feature, admin must click on the “edit” button and he must edit the name, username, and password of the edited admin before clicking save changes. Figure below shows the edit page for admin.

Account / Change Account

Name

hanmohamedon

Username

hanmohamedon

Password


 Save Changes

Figure 2.14

```

39 <div class = "container-fluid">
40 <div class = "panel panel-default">
41 <div class = "panel-body">
42 <div class = "alert alert-info">Account / Change Account</div>
43 <?php
44 $query = $conn->query("SELECT * FROM 'admin' WHERE 'admin_id' = '$_REQUEST[admin_id]'" or die(mysqli_error());
45 $fetch = $query->fetch_array();
46 ?>
47 <br />
48 <div class = "col-md-4">
49 <form method = "POST" action = "edit_query_account.php?admin_id=<?php echo $fetch['admin_id']?>">
50 <div class = "form-group">
51 <label>Name </label>
52 <input type = "text" class = "form-control" value = "<?php echo $fetch['name']?>" name = "name" />
53 </div>
54 <div class = "form-group">
55 <label>Username </label>
56 <input type = "text" class = "form-control" value = "<?php echo $fetch['username']?>" name = "username" />
57 </div>
58 <div class = "form-group">
59 <label>Password </label>
60 <input type = "password" class = "form-control" value = "<?php echo $fetch['password']?>" name = "password" />
61 </div>
62 <br />
63 <div class = "form-group">
64 <button name = "edit_account" class = "btn btn-warning form-control"><i class = "glyphicon glyphicon-edit"></i> Save Changes</button>
65 </div>
66 </form>
67 </div>
68 </div>
69 </div>
70 </div>

```

The backend of the page is coded quite the same as per add_account.php but the “Save Changes” button is linked to edit_query_account.php. The file is saved as edit_account.php. The code is shown below.

First, the code query or request data from table “admin”, attribute “table_id” to match the edited admin. Once the admin has been edited and “Save Changes” is clicked, edit_query_account.php will query data from the database and update the name, username, and password with the matching admin_id. Below is the code of edit_query_account.php.

```
1 <?php
2 require_once 'connect.php';
3 if(isset($_POST['edit_account'])){
4     $name = $_POST['name'];
5     $username = $_POST['username'];
6     $password = $_POST['password'];
7     $conn->query("UPDATE 'admin' SET 'name' = '$name', 'username' = '$username', 'password' = '$password' WHERE 'admin_id' = '$_REQUEST[admin_id]'" or die(mysql_error());
8     header("location:account.php");
9 }
10
```

For delete feature, once “Delete” is clicked, it will prompt the admin to confirm the deletion, click on OK and the selected admin account will be deleted. Below is the code for the delete feature which is saved as delete_account.php. The code just query from table “admin” and DELETE from it the matching “admin_id”.

```
1 <?php
2 require_once 'connect.php';
3 $conn->query("DELETE FROM 'admin' WHERE 'admin_id' = '$_REQUEST[admin_id]'" or die(mysql_error());
4 header("location: account.php");
```

Thus, INSERT, UPDATE, and DELETE features can be displayed here in adding new admin, edit the existing admin accounts, and delete the accounts.

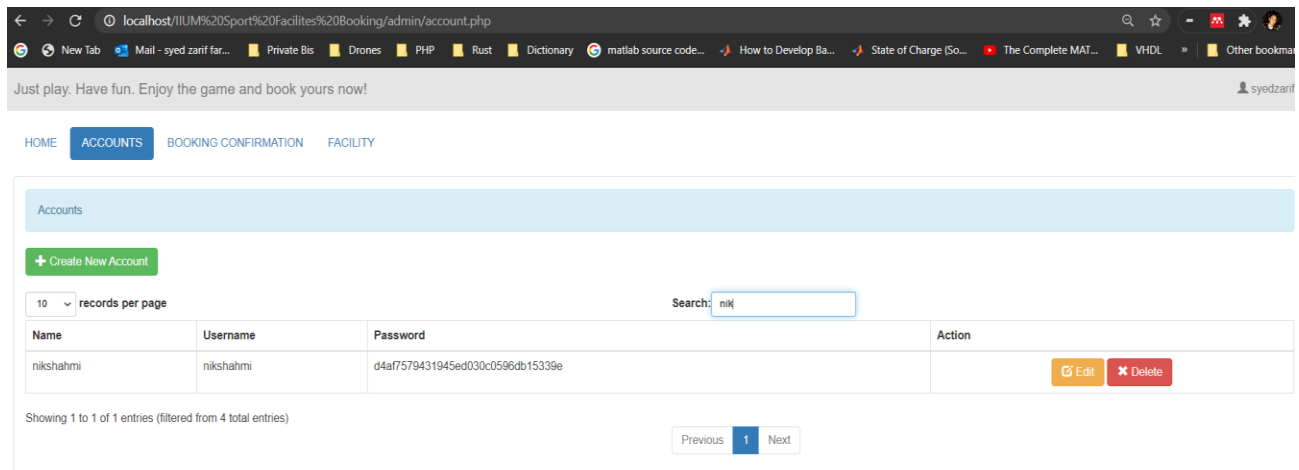


Figure 2.15

```

95 <script type = "text/javascript">
96   $(document).ready(function() {
97     $('#table').DataTable();
98   });
99 </script>

```

The search bar in every page which is the accounts, booking confirmation and facility is always included to find any string that matches the data inside the database. In order for this search bar to work, a single function from line 95 to 99 is added to each page that contains the data table. Figure 2.11 shows the example when admin searches 'nik' and the table will immediately show the data from the admin table that matches with 'nik'.

2.7 Admin (Booking Confirmation)

Once users book any facility, admins are needed to handle the approval of the books to avoid any redundant booking. This is done at “Booking Confirmation” at the navigation bar as shown below.

Name	Contact No	Facility	Reserved Date	Time In	Time Out	Status	Action
Ali Abu	01120922990	Tennis Court (Female Sport Complex)	2021-01-08	13:00:00	14:00:00	Pending	Book Discard
Muhammad Farhan Mohamedon	0123456789	Football Field (Sayyidina Hamzah Stadium)	2021-01-12	14:00:00	16:00:00	Pending	Book Discard
Muhammad Farhan Mohamedon	0123456789	Football Field (Sayyidina Hamzah Stadium)	2021-01-12	14:00:00	16:00:00	Pending	Book Discard
Nur Irsya Aiman Mohd Isa	01120812990	Football Field (Sayyidina Hamzah Stadium)	2021-01-07	11:00:00	12:00:00	Pending	Book Discard

Figure 2.16

Admin can click “Book” which is approving the book or “Discard” which is delete or disapprove the book. Below the navigation bar, the tab reserved shows the reserved bookings which are yet to be approved and the booked tab is the confirmed bookings. Below is the Booked tab page.

Name	Contact No	Facility	Reserved Date	Court Number	Time In	Time Out	Status
Azam Azman	0132232528	Football Field (Sayyidina Hamzah Stadium)	2021-01-08	3	18:00:00	19:00:00	Booked
hey hi	0123658596	Futsal Court (Male Sport Complex)	2021-01-28	1	15:00:00	16:00:00	Booked
Syed Farhannudin	013252562	Futsal Court (Male Sport Complex)	2021-01-28	2	14:00:00	15:00:00	Booked

Figure 2.17

```

40 <div class = "panel panel-default">
41
42 <?php
43     $q_p = $conn->query("SELECT COUNT(*) as total FROM 'booking' WHERE 'status' = 'Pending'") or die(mysql_error());
44     $f_p = $q_p->fetch_array();
45     $q_ci = $conn->query("SELECT COUNT(*) as total FROM 'booking' WHERE 'status' = 'Booked'") or die(mysql_error());
46     $f_ci = $q_ci->fetch_array();
47 >?>
48 <div class = "panel-body">
49
50 <a class = "btn btn-success disabled"><span class = "badge"><?php echo $f_p['total']?></span>Reserved</a>
51 <a class = "btn btn-info" href = "checkin.php"><span class = "badge"><?php echo $f_ci['total']?></span>Booked</a>
52 <br />
53 <br />
54 <table id = "table" class = "table table-bordered">
55 <thead>
56 <tr>
57 <th>Name</th>
58 <th>Contact No</th>
59 <th>Facility</th>
60 <th>Reserved Date</th>
61 <th>Time In</th>
62 <th>Time Out</th>
63 <th>Status</th>
64 <th>Action</th>
65 </tr>
66 </thead>
67 <tbody>
68 <?php
69 $query = $conn->query("SELECT * FROM 'booking' NATURAL JOIN 'guest' NATURAL JOIN 'faci' WHERE 'status' = 'Pending'") or die(mysql_error());
70 while($fetch = $query->fetch_array()){
71 >?>
72 <tr>
73 <td><?php echo $fetch['firstname']." ".$fetch['lastname']?></td>
74 <td><?php echo $fetch['contactno']?></td>
75 <td><?php echo $fetch['faci_sport']?></td>
76 <td><?php echo $fetch['date_in']?></td>
77 <td><?php echo $fetch['checkin_time']?></td>
78 <td><?php echo $fetch['checkout_time']?></td>
79 <td><?php echo $fetch['status']?></td>
80 <td><center><a class = "btn btn-success" href = "confirm_reserve.php?booking_id=<?php echo $fetch['booking_id']?>"><i class = "glyphicon glyphicon-check"></i> Book </a>
81 <a class = "btn btn-danger" onclick = "confirmationDelete(); return false;" href = "delete_pending.php?booking_id=<?php echo $fetch['booking_id']?>">
82 <i class = "glyphicon glyphicon-trash"></i> Discard</a></td>
83 </tr>
84 </tbody>
85 </table>
86 <?php

```

The above code is the backend of the booking confirmation named reserve.php. In line 47, it will query the database from the booking table and faci table where the status is in pending by using the 'NATURAL JOIN' and 'WHERE 'status' = 'Pending'' to get the unique column as well as the secondary column. Then, when the admin confirms the reservation by clicking the book button, it will then fetch the booking_id and call the confirm_reserve.php.

```

43 <?php
44 $query = $conn->query("SELECT * FROM `booking` NATURAL JOIN `guest` NATURAL JOIN `faci` WHERE `booking_id` = '$_REQUEST[booking_id]'" or die(mysqli_error());
45 $fetch = $query->fetch_array();
46 ?>
47 <br />
48 <form method = "POST" enctype = "multipart/form-data" action = "save_form.php?booking_id=<?php echo $fetch['booking_id']?>">
49     <div class = "form-inline" style = "float:left;">
50         <label>Firstname</label>
51         <br />
52         <input type = "text" value = "<?php echo $fetch['firstname']?>" class = "form-control" size = "40" disabled = "disabled"/>
53     </div>
54     <div class = "form-inline" style = "float:left; margin-left:20px;">
55         <label>Middlename</label>
56         <br />
57         <input type = "text" value = "<?php echo $fetch['middlename']?>" class = "form-control" size = "40" disabled = "disabled"/>
58     </div>
59     <div class = "form-inline" style = "float:left; margin-left:20px;">
60         <label>Lastname</label>
61         <br />
62         <input type = "text" value = "<?php echo $fetch['lastname']?>" class = "form-control" size = "40" disabled = "disabled"/>
63     </div>
64     <br style = "clear:both:"/>
65     <br />
66     <div class = "form-inline" style = "float:left;">
67         <label>Court Type</label>
68         <br />
69         <input type = "text" value = "<?php echo $fetch['faci_type']?>" class = "form-control" size = "20" disabled = "disabled"/>
70     </div>
71     <div class = "form-inline" style = "float:left; margin-left:20px;">
72         <label>Court Number</label>
73         <br />
74         <input type = "number" min = "0" max = "5" name = "faci_no" class = "form-control" required = "required"/>
75     </div>
76     <br style = "clear:both:"/>
77     <br />
78     <button name = "add_form" class = "btn btn-primary"><i class = "glyphicon glyphicon-save"></i> Submit</button>
79 </form>

```

```

1 <?php
2 require_once 'connect.php';
3 if (ISSET($_POST['add_form'])) {
4     $faci_no = $_POST['faci_no'];
5     $query = $conn->query("SELECT * FROM `booking` WHERE `faci_no` = '$faci_no' && `status` = 'Booked'" or die(mysqli_error());
6     $row = $query->num_rows;
7     $time = date("H:i:s", strtotime("+8 HOURS"));
8
9     $query2 = $conn->query("SELECT * FROM `booking` NATURAL JOIN `guest` NATURAL JOIN `faci` WHERE `booking_id` = '$_REQUEST[booking_id]'" or die(mysqli_error());
10    $fetch2 = $query2->fetch_array();
11
12
13    $conn->query("UPDATE `booking` SET `faci_no` = '$faci_no' , `status` = 'Booked' WHERE `booking_id` = '$_REQUEST[booking_id]'" or die(mysqli_error());
14    header("location:checkin.php");
15
16 }
17 ?>

```

Next in confirm_reserve.php page, the admin will then set which court number for the user and then click submit to confirm. This will set the button add_form to be true and call the save_form.php page to update the booking table based on the specified faci_no by setting the status from pending to booked.

```

66 <?php
67 $query = $conn->query("SELECT * FROM 'booking' NATURAL JOIN 'guest' NATURAL JOIN 'faci' WHERE 'status' = 'Booked'") or die(mysqli_query());
68 while($fetch = $query->fetch_array()){
69     ?>
70 <tr>
71     <td><?php echo $fetch['firstname']. " ".$fetch['lastname']?></td>
72     <td><?php echo $fetch['contactno']?></td>
73     <td><?php echo $fetch['faci_sport']?></td>
74     <td><?php echo $fetch['date_in']?></td>
75     <td><?php echo $fetch['faci_no']?></td>
76     <td><?php echo $fetch['checkin_time']?></td>
77     <td><?php echo $fetch['checkout_time']?></td>
78     <td><?php echo $fetch['status']?></td>
79 </tr>

```

Then, this code is the backend for the list of bookings where the user has been confirmed to book the facility. In line 67, it queries the database from the booking table and faci table but only with the booked status.

2.8 Admin (Facility)

Under FACILITY on the navigation bar, admin is able to add new facilities, and edit or delete existing facilities. The features are done with the same method as the admin accounts before. But INSERT, UPDATE, DELETE here is added with a photo where admins can upload, change, or delete photos of facilities. Below is the page of FACILITY with existing facilities.

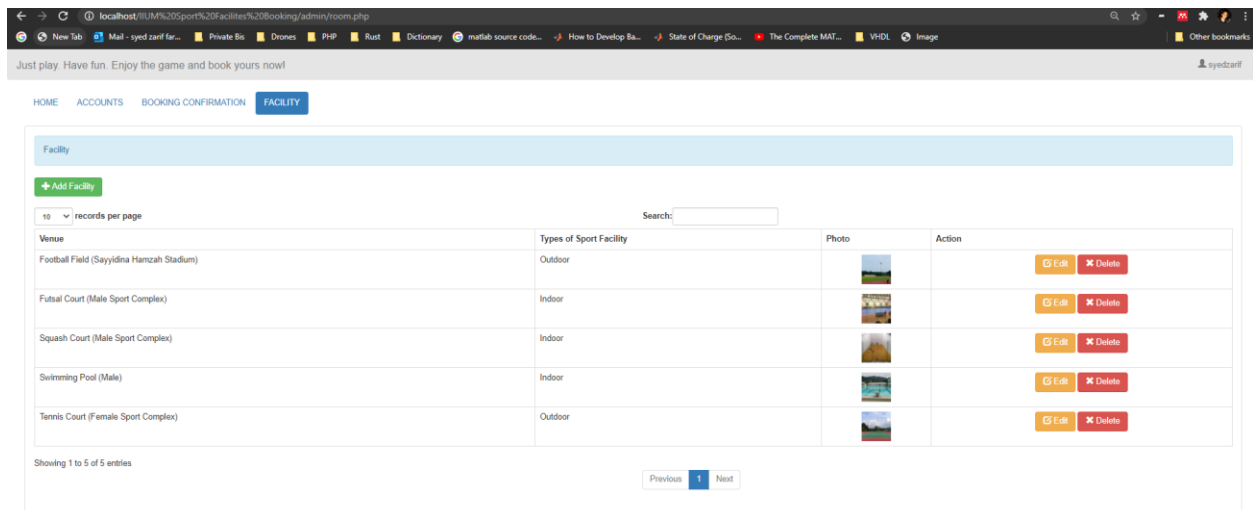


Figure 2.18

Admin can click on the Add Facility button to add a new facility, Edit button to change any records of the facility, Delete button to delete any facility. All of these features are coded the same way as admin accounts previously. Code below displays the Edit and Delete button fetch data from the faci table in the database. Edit button is linked with edit_room.php and Delete button is linked with delete_room.php where each of them carries out the UPDATE and DELETE feature respectively. Below is an example of the code to Add a new facility as well its page.

```

1  f?php
2  if(isset($_POST['add_room'])){
3      $faci_sport = $_POST['faci_sport'];
4      $faci_type = $_POST['faci_type'];
5
6      $photo = addslashes(file_get_contents($_FILES['photo']['tmp_name']));
7      $photo_name = addslashes($_FILES['photo']['name']);
8      $photo_size = getimagesize($_FILES['photo']['tmp_name']);
9      move_uploaded_file($_FILES['photo']['tmp_name'], "../photo/" . $_FILES['photo']['name']);
10     $conn->query("INSERT INTO `faci` (`faci_sport`,`faci_type`, `photo`) VALUES('$_faci_sport','$_faci_type', '$photo_name')") or die(mysqli_error());
11     header("location:room.php");
12 }
13

```

```

39  <div class = "container-fluid">
40      <div class = "panel panel-default">
41          <div class = "panel-body">
42              <div class = "alert alert-info">Facility/ Change Facility</div>
43              <br />
44              <div class = "col-md-4">
45                  <form method = "POST" enctype = "multipart/form-data">
46                      <div class = "form-group">
47                          <br /><label>Facility</label>
48                          <input type = "form-control" class = "form-control" name = "faci_sport" required = "required" />
49                      </div>
50                      <div class = "form-group">
51                          <br /><label>Facility Type</label>
52                          <select class = "form-control" required = required name = "faci_type">
53                              <option value = "">Choose an option</option>
54                              <option value = "Indoor" >Indoor</option>
55                              <option value = "Outdoor" >Outdoor</option>
56                          </select>
57                      </div>
58
59                      <div class = "form-group">
60                          <br /><label>Photo </label>
61                          <div id = "preview" style = "width:150px; height :150px; border:1px solid #000;">
62                              <center id = "lbl">[Photo]</center>
63                          </div>
64                          <input type = "file" required = "required" id = "photo" name = "photo" />
65                      </div>
66                      <br />
67                      <div class = "form-group">
68                          <button name = "add_room" class = "btn btn-info form-control"><i class = "glyphicon glyphicon-save"></i> Add</button>
69                      </div>
70                  </form>
71                  <?php require_once 'add_query_room.php'?>
72              </div>
73          </div>
74      </div>
75  </div>

```


What is different here is Admin can add a photo of the facility. Once the Admin puts in the name of the facility, its type, and its photo, the database will update table “faci” which contains the records and attributes of facilities. Code above is the code where the photo is INSERT into the database, in the table faci, under attribute photo.

HOME ACCOUNTS BOOKING CONFIRMATION FACILITY

Facility/ Change Facility/div>

Facility

Facility Type

Choose an option

Photo

[Photo]

Choose File No file chosen

Add

Figure 2.19

3.0 Conclusion

To conclude, the objectives of this project are achieved successfully. The booking system for sport facilities in IIUM is developed using the SQL, HTML and PHP web database programming in providing the foundation of the system's logic. Several tests are made on the booking system have shown its interactivity and functionalities for both user and admin side. Four basic functions of database storage are also implemented which are create, read, update and delete (CRUD). These can be seen by the INSERT, SELECT, UPDATE and DELETE commands used in the development. With these functions, the booking facilities system is in the correct implementation as a database web application.