**NLINE BANK MANAGEMENT SYSTEM**

**Submitted by**

**BALAMURUGAN.G (1813101058209)**

**GOKUL.P (1813101058217)**

**2020-2021**

**MINI PROJECT**

**IN**

**COMPUTER SCIENCE**



**DEPARTMENT OF COMPUTER SCIENCE**

**DG VAISHNAV COLLEGE**

**CHENNAI-600106**

**DWARAKA DOSS GOVERDHAN DOSS VAISHNAV COLLEGE (AUTONOMOUS)**

**ARUMBAKKAM, CHENNAI-600 106.**



**DEPARTMENT OF COMPUTER SCIENCE**

**MINI PROJECT**

*Certified that this is a record work done during the academic year 2020 - 2021.*

*By*

**BALAMURUGAN.G (1813101058209)**

**Faculty In-charge Head In-charge**

*Submitted for practical examination held on ……………………… at Dwarak*

*Doss Goverdhan Doss Vaishnav College, Chennai-106.*

**Internal Examiner External Examiner**

**ACKNOWLEDGEMENT**

First we express our gratitude to God Almighty whose grace gave us the mental and physical power for the successful completion of this endeavor.

Our deepest gratitude and thanks to our principal **Dr. S. Santhosh Baboo, M.Sc. Ph.D.,** who has always helped us.

We wish to express our sincere thanks and gratitude to the Head of the Department of Computer Science **Ms. A. Vijayalakshmi, MCA, M.Phil., SET**, who has been a guiding force and constant source of inspiration to us.

Our sincere gratitude and thanks to our beloved Internal Guide **Mrs. P.Revathi Msc., NET** for having extend their fullest cooperation and guidance without which this project would not have been a success.

Our thanks to all other faculty and non-teaching staff members of our department for their support, thanks to parents and peers for having stood by us and helped us to complete this project.

**TABLE OF CONTENTS**

|  |  |  |
| --- | --- | --- |
| **S.NO** | **CONTENTS** | **PG.NO.** |
| **1** | **INTRODUCTION**  ABSTRACT   * 1. Overview   2. Expected Outcome   3. Problem Definition |  |
| **2** | **SYSTEM DESCRIPTION**   * 1. Program Description   2. Modules   3. Existing System   4. Proposed System |  |
| **3** | **DEVELOPMENT ENVIRONMENT**   * 1. Hardware Configuration   2. Software Configuration   3. Technological Specification   4. Language Specification |  |
| **4** | **APPENDIX**  Sample Source Code & Result |  |
| **5** | **CONCLUSION**   * 1. Future Enhancements   2. Bibliography |  |

**ABSTRACT**

**ABOUT THE PROJECT**

The main scope of mini - project titled “**ONLINE BANK MANAGEMENT SYSTEM**” is to provide a system which handles the information of the people coming into the bank and maintaining their accounts.

* Online Banking is one of the most important financial activities which will be carried out by any person who holds a bank account. Once a user logs in he or she can check the bank balance, check bank account transaction history or account summary, add beneficiary accounts, transfer funds to another account, loan information.
* Whenever we deal with a banking system main concern should be the security related to banking transactions and account login activity. In the existing system the transactions are done only manually but in the proposed system all the transactions are done through online banking system.
* The present scenario in the gyms is that the records are kept by writing in a file on the paper. Every management task is done manually. This creates a system unreliable and confusing to keep the correct track of the records.
* The maintenance of the system like this is hardly required until it needs to change any part of the system. The information about the various things contained in the system are like members, trainers, equipment can get by just a few clicks unlike the paper documents required the serious reading for such information.
* This will improve the transparency between the members which is always a good quality in the system. It will also give the layer of security to the administration and the users that on, authorized users can access by their credentials.

## MODULES

* Homepage module
* Admin module
* Customer module
* Deposit module
* withdraw module
* Transaction module
* Grant Loan module
* Pay Loan module
* Payee module
* Fastag module

**Homepage Module:** This module contains two Roles one is for the Admin Role where one can Access the Full information about the Maintenance and Another Role is for the customers where one can access their accounts.

**Admin Module:** This module contains the Full information about the customers.

**Customer Module:** This module contains the information about their own account and transactions

**Deposit Module:** OnlyAdmin can deposit the money in the customer’s account

**Withdraw Module**: OnlyAdmin can Withdrawthe money in the customer’s account

**Transaction Module:** This module shows the transaction history of the customer and can also transfer funds.

**Grant Loan Module:** OnlyAdmin can access and grant loan for their customers**.**

**Pay Loan Module:** Cusomers can pay their debts and view their loans.

**payee Module:** The payee or beneficiary is the person or entity entitled to receive

the claim amount and other benefits upon the death of the benefactor or on the

maturity of the policy. 

**Fastag Module:** This module contains the fastag information of the customer, recharge their fastag account and admins can help the customer to create fastag account.

**SYSTEM ANALSIS**

**INTRODUCTION**

System analysis is the detailed study of the various operations performed by the system and their relationship with in and outside the system. The key question in this phase is: What are all the problems there in the present system and what must be done to solve those problems. The success of the system depends largely on how clearly the problem is defined, thoroughly investigated and properly carried out. The analysis should provide the mechanism of problem understanding and also a framework for its solution. Here the existing system was studied thoroughly by the collection of relevant data about the system. Then the proposed system was analyzed thoroughly by the needs.

### USER PERFORMANCE

Before developing any computerized system it is important to examine the problems in the existing system and the additional information that should be fulfilled in the proposed system. In the present system we are able to insert the customer details, process details and billing can be done. It also used to calculate the total amount of the bill.

**REQUIREMENT SPECIFICATION**

### HARDWARE SPECIFICATION

**Processer :** Intel(R) Core(TM) i5-4570

**Ram :** 4 GB

**Processor speed :** 2Ghz

**Hard Disk :** 1 TB

**Monitor :** 1080 FHD Display

**Keyboard :** 104 Keys

### SOFTWARE SPECIFICATION

**Front End :** HTML, CSS, JAVASCRIPT, BOOTSTRAP

**Back End :** PHP

**Database :** MYSQL

**Interface :** Chrome

**Operating System :** Windows 7 (Or) Higher

# SYSTEM DESIGN

### SYSTEM DESIGN

In this phase, we provide a physical shape to all the models that are created in the analysis phase. The design phase is used to physically create different elements of the project. This phase involves the following sub-phases.

### MODULE DESIGN

In this step, we identify the modules and sub-modules of the project. Module is the functional unit of a project. The modularity of a project depends upon the functional requirements and also the users of the project. In this step, we identify the modules and sub-modules of the project. Module is the functional unit of a project. The modularity of a project depends upon the functional requirements and also the users of the project.

In this project, following functional modules are identified.

* **Administrator:** This module involves all the functionalities meant for the administrator.
* **Customer:** This module involves all the functionalities meant for the customer.
* **Home page:** This module involves all the information meant for the bank.

### DATA DESIGN

Data dictionary is a document, which describes the database design of the project. In this step, we generate the data dictionary using the ER Model of the system. Each entity is represented as a table here. The Data dictionary describes the tables, their fields, data types and constraints on the fields.

The Data Design transforms the information domain model created during analysis into the data structure that will be required to implement the software.

Data Design is the first of three design activities that are conducted during software engineering. The impact of data structure on program structures and procedural complexity cases data design to have a profound influence on software quality. The concept of information hiding and data abstraction provides the foundation for an approach to data design.

The primary activity during data design is to select logical representation of data objects (data dictionary) identified during the requirements definitions and specifications phase.

It is the process of designing database files, which are the key source of information to the system. The files should properly, designed planed for collection, accumulation, editing the required information. The objectives of the file design are to provide affective auxiliary storage and to contribute to the overall efficiency of the computer program components to the system.

###### FIRST NORMAL FORM:

A relation R of this system is said to be in first normal form, if all domains contain atomic values. In this project the Data in each table has atomic values but the data are redundant. So, I consider next normal form i.e., Second normal form.

###### SECOND NORMAL FORM:

A relation R of this system is in second normal form as if every irreducibly dependent on primary key. Coming to child tables we will be having a reference to the primary key of master table. In this project I normalized the data tables up to second normal form.

### INTERFACE DESIGN

In this step, we design all the interfaces for the application. Interface is the medium between the user and the system. We consider all the client specifications while designing the screens and reports.

Every user interface has been designed so that the system provides all the characteristics like easy to use, less number of keystrokes to complete an action, less usage of left hand, easy to learn, easy to navigate, consistent, error-free and functional.

The entire interface that has designed with the features listed below:

* Every action is provided with a prompting message. So that it provides a better communication.
* Every interface maintained consistent navigation, menus, icons, colors, shape and layout.

Every page has been designed so that it optimizes the user‟s work efficiency.

**SOFTWARE**

**DESCRIPTION**

### ABOUT FRONT END

**PHP:**PHP started out as a small open source project that evolved as more and more people found out how useful it was. Rasmus Lerdorf unleashed the first version of PHP way back in 1994.

* PHP is a recursive acronym for "PHP: Hypertext Preprocessor".
* PHP is a server side scripting language that is embedded in HTML. It is used to manage dynamic content, databases, session tracking, even build entire e- commerce sites.
* It is integrated with a number of popular databases, including MySQL, PostgreSQL, Oracle, Sybase, Informix, and Microsoft SQL Server.

**Characteristics of PHP**

Five important characteristics make PHP's practical nature possible

* Simplicity
* Efficiency
* Security
* Flexibility
* Familiarity

### DEVELOPMENT LANGUAGES, SCRIPTS AND LIBRARIES

**HTML:** HTML, or Hypertext Markup Language, is used to create a page. Site authors use HTML to format text as titles and headings, to arrange graphics on a webpage, to link to different pages within a website, and to link to different websites.

HTML is a set of codes that a website author inserts into a plain text file to format the content. The author inserts HTML tags, or commands, before and after words or phrases to indicate their format and location on the page. HTML tags are also used to add tables, lists, images, music, and other elements to a webpage.

**CSS:** CSS is the language for describing the presentation of Web pages, including colors, layout, and fonts. It allows one to adapt the presentation to different types of devices, such as large screens, small screens, or printers.

CSS is independent of HTML and can be used with any XML-based markup language. The separation of HTML from CSS makes it easier to maintain sites, share style sheets across pages, and tailor pages to different environments.

**BOOTSTRAP:** Bootstrap is a free and open-source CSS framework directed at responsive, mobile-first front-end web development. It contains CSS- and (optionally) JavaScript-based design templates for typography, forms, buttons, navigation, and other interface components.

Bootstrap is a web framework that focuses on simplifying the development of informative web pages (as opposed to web apps). The primary purpose of adding it to a web project is to apply Bootstrap's choices of color, size, font and layout to that project. As such, the primary factor is whether the developers in charge find those choices to their liking. Once added to a project, Bootstrap provides basic style definitions for all HTML elements. The result is a uniform appearance for prose, tables and form elements across web browsers. In addition, developers can take advantage of CSS classes defined in Bootstrap to further customize the appearance of their contents. For example, Bootstrap has provisioned for light- and dark- colored tables, page headings, more prominent pull quotes, and text with a highlight.

Bootstrap also comes with several JavaScript components in the form of jQuery plugins. They provide additional user interface elements such as dialog box, tooltips, and carousels.

### ABOUT BACK END

**MYSQL:** XAMPP is a software distribution which provides the Apache web server, MySQL database in one package for managing your MySQL databases. The data objects are stored as separate documents inside a collection instead of storing the data into the columns and rows of a traditional relational database is to implement a data store that provides high performance, high availability, and automatic scaling is extremely simple to install and implement.

* MySQL is released under an open-source license. So you have nothing to pay to use it.
* MySQL is a very powerful program in its own right. It handles a large subset of the functionality of the most expensive and powerful database packages.
* MySQL uses a standard form of the well-known SQL data language.
* MySQL works on many operating systems and with many languages including PHP, PERL, C, C++, JAVA, etc.
* MySQL is very friendly to PHP, the most appreciated language for web development.

### ADMINISTRATIVE MYSQL COMMAND

Here is the list of the important MySQL commands, which you will use time to time to work with MySQL database −

**1**. **SELECT** — extracts data from a database **SELECT** column\_name **FROM** table\_name; **SELECT** statements fetch data from a database.

**2**. **UPDATE** — updates data in a database **UPDATE** table\_name **SET** some\_column = some\_value **WHERE** some\_column = some\_value; **UPDATE** statements allow us to edit rows in a table.

**3**. **DELETE** — deletes data from a database **DELETE** **FROM** table\_name **WHERE** some\_column = some\_value; **DELETE** statements remove rows from a table.

**4.** **INSERT** **INTO** — inserts new data into a database **INSERT** **INTO** table\_name (column\_1, column\_2, column\_3) **VALUES** (value\_1, ‘value\_2’, value\_3);

**INSERT** statements add a new row to a table.

**5**. **CREATE** **DATABASE** — creates a new database **CREATE** **DATABASE** databasename; **CREATE** **DATABASE** statements create a new SQL database.

**6.** **ALTER** **DATABASE** — modifies a database **ALTER** **DATABASE** database\_name [COLLATE collation\_name ]

**ALTER** **DATABASE** statements change the characteristics of a database.

**SYSTEM TESTING**

**AND**

**IMPLEMENTATION**

### SYSTEM TESTING

Testing is the process of confirming that a program or system does what it is proposed off, Testing is the only way to assure the quality of s/w and it is an umbrella activity rather that a separate phase. This is an activity to be performed in parallel with the s/w efforts and one that consists of its own phase of analysis, design, implementation, execution and maintenance.

### TESTING STRATEGY

###### Unit Testing:

This testing method considers a module as single unit and checks the unit at interfaces and communities with other modules rather than getting into details at statement level. Here the module will be treated as BLACKBOX, which will take some inputs and generate output. Outputs for a given set of input combination are pre calculated and are generated by the module.

###### Integration Testing:

Here all the pre-tested individual modules will be assembled to create a larger system and tests are carried out at system level to make sure that all modules are working in sync vehicle with each other. This testing methodology helps in making sure that all modules which are running perfectly when checked individually and are also running cohesion with other modules. For this testing we create test-cases to check all modules once and then a generated test combination of test paths to vehicle throughout the system to make sure that no path is making its way into chaos.

###### Validation Testing:

Testing is major quality control measure employed during vehicle development. Its basic function is to detect errors. Sub functions when combined may not produce than it is desired. Global data structures can represent the problems. Integrated testing is a systematic technique for constructing the program structure while conducting the tests. To uncover errors that are associated with interfacing the objective is to make test modules and built a program structure that has detected by design.

In a non-incremental integration all the modules are combined in advance and the program is tested as a whole. Here error will appear in an end-less loop function.. In incremental testing the program is constructed and tested in small segments where the errors are isolated and corrected.

Different incremental integration strategies are

1. Top-Down integration
2. Bottom-Up integration
3. Regression integration

Testing means quality test. Testing is a process of executing a program with the intent of finding error. A good test case is one that has a high probability of finding an as yet un discovered error.

Objective should be to design test that systematically uncover different classes of error and to do with a minimum amount of time and effort. Testing cannot show the absence of defects, it can only show that s\w defects are present. It is important to keep this statement in a mind as testing is being conducted.

Any engineering product can be tested in one of the two ways.

Knowing the specific function that a product has been designed to perform, test can be conducted that demonstrates each function is fully operational. This approach is called „BLACK BOX TEXTING‟.

Knowing the internal working of the product, test can be conducted to ensure that “all gears mesh”, that is, that internal operation of the product performs according to specification and all internal components have been adequately exercised. This approach is called “WHITE BOX TESTING”.

These approaches provide a mechanism that can help to ensure the completeness of tests and provide the highest likelihood for uncovering errors in s/w.

The goals of verification and validation are to access and improve the quality of work products generated during development and modification of s/w.

These are 2 types of verification namely.

1. Life-cycle verification
2. Formal verification.

Validation is the process of evaluating s/w at the end of s/w development

process.

Quality assurance is a planned and systematic pattern of action necessary to provide adequate confirms to the technical requirement.

Walktvehicleoughs are sessions where the material being examined is examined is presented by a review and evaluated by a team of reviewers.

Inspection involves assessing the s/w life cycle and improving the quality of work products.

Life-cycle verification is the process of determining the degree to which the work products of a given phase of the development cycle fulfill the specification established during prior phases.

Formal verification is a rigorous mathematical demonstration that source code confirms to its specification.

High quality can be achieved tvehicleough testing of source code alone.

Although a program should be totally free of errors, this seldom the case for large s/w products.

There are 3 major categories of s/w error.

1. Requirement errors
2. Design errors
3. Implementation errors

Quality assurance defines the objective of the project and reviews the overall activities so that the errors are corrected early in the development process.

During analysis and design, an s/w verification plan and acceptance test plan is prepared. The verification plan describes the methods to be used in verifying that the requirements are satisfied by the design documents and that the source is consistent with the requirements specification and design documents. The acceptance test plan includes test cases, outcomes and capabilities demonstrated by each test case. Following completion of the verification plan and acceptance plan, an s/w verification

review is held to evaluate the adequacy of the plans.

During product evolution, in-process audits are conducted to verify consistency and completeness of the work products. Items to be audited for consistency include interface specification for hardware and vehicle and people: internal design verses functional requirements verses test descriptions.

Prior to product delivery, a functional audit and a physical audit performed. The functional audit reconfirms that all the requirements have been met. The physical audit verifies that the source code and all associated documents are complete, consistent with one another and ready to deliver. An s/w verification summary is prepared to describe the results of all reviews.

System testing involves two kinds of activities-integration testing and acceptance testing.

### ACCEPTANCE TESTING

It involves planning and execution of functional tests and stress tests in order to demonstrate that the implemented system satisfies its requirements.

### ALPHA AND BETA TESTING

If s/w is developed as product to be used by many customers, it is impractical to perform formal acceptance test with each one. So, one most developers use Alpha and Beta testing to uncover that only the end user seems able to find.

Alpha testing is conducted by the customer in the presence of many project leaders and recorded the errors and usage problems what they faced.

Beta testing is conducted at customer site by the end users of the s/w, the customer recorded that encountered during beta testing and sent those problems to us regular intervals. Then we made the modification and released to the entire customer base.

### IMPLEMENTATION

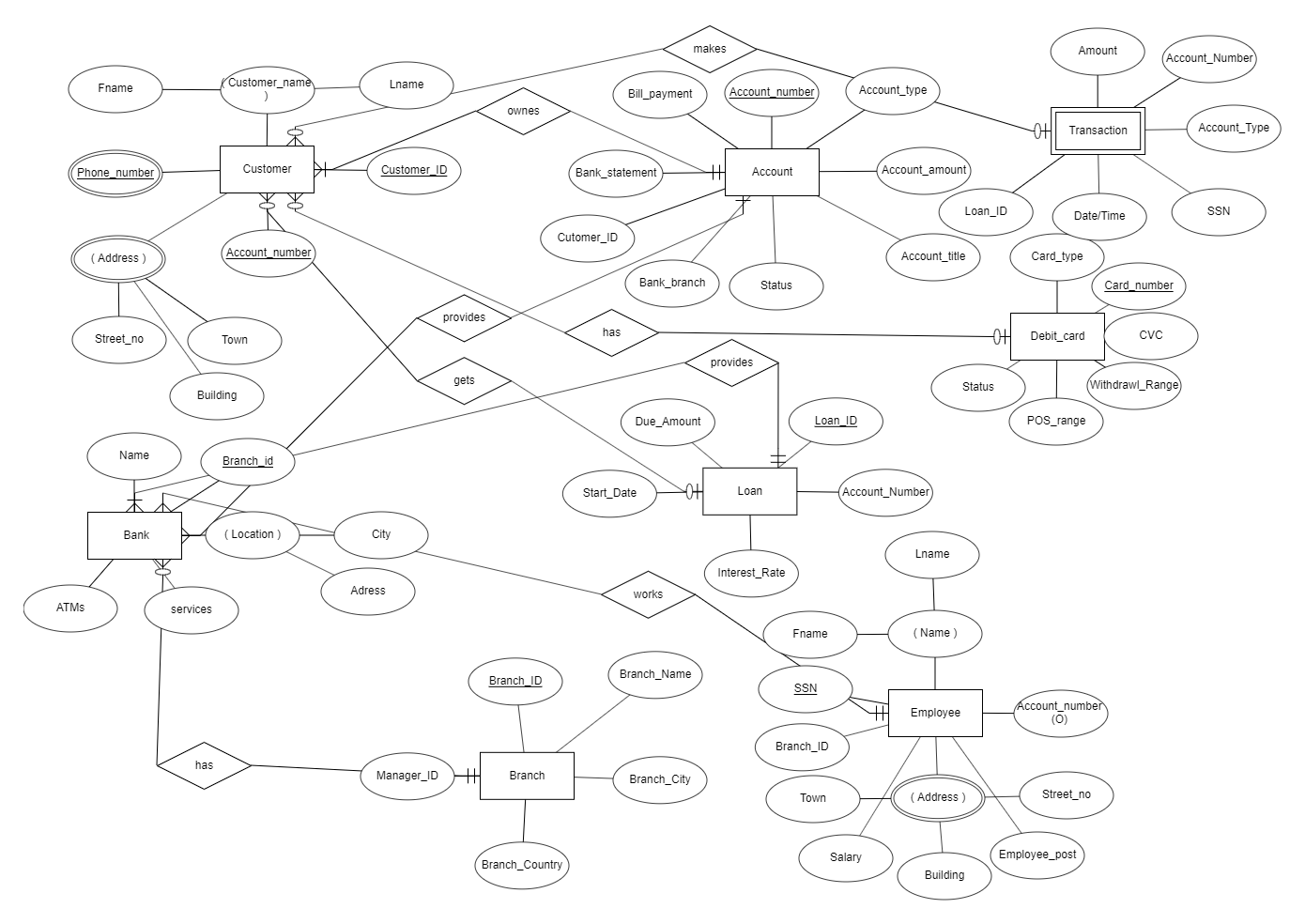
The crucial phase in the system life cycle is successful completion of the new system design. Implementation simply means converting a new system design in to operation. This involves creating files, training the operation staff and installing the hardware and telecommunication network before the system is up and running.

A Crucial factor in the conversion is not disrupting the functioning of the organization. In system implementation user training is essential.

The user manual is prepaid reflexively because it is an item that must accompany every system. The manual is necessary when the user uses geographically, remote from the project team or what they cannot attend all training sessions.

**ER DIAGRAM**

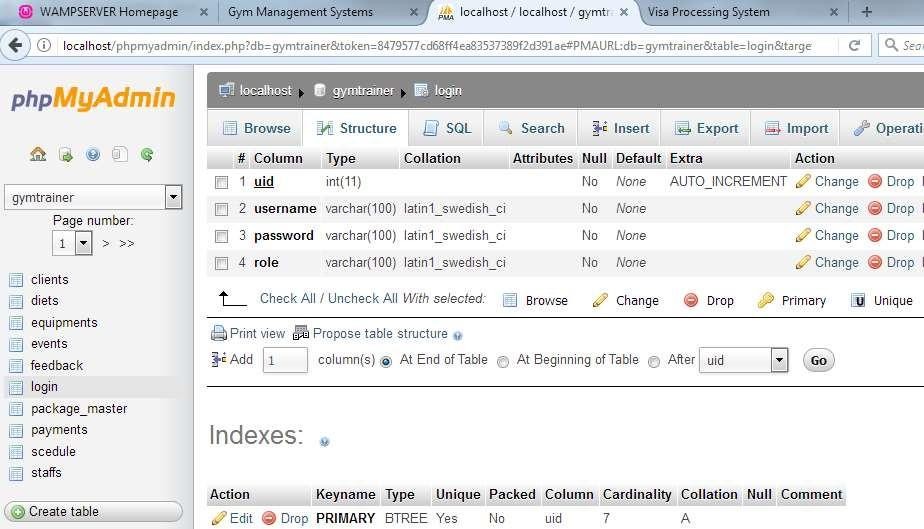
### ER Diagram:



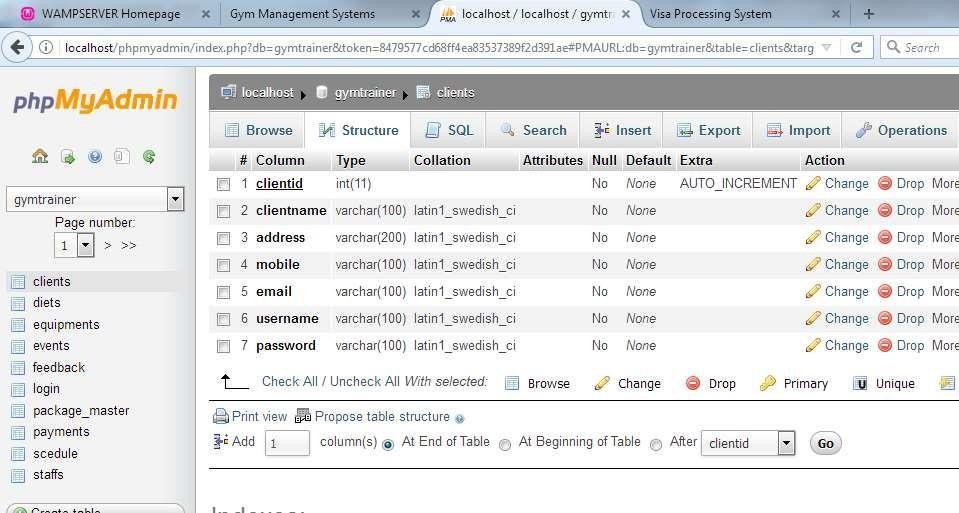
**TABLE STRUCTURE**

**TABLE STRUCTURE**

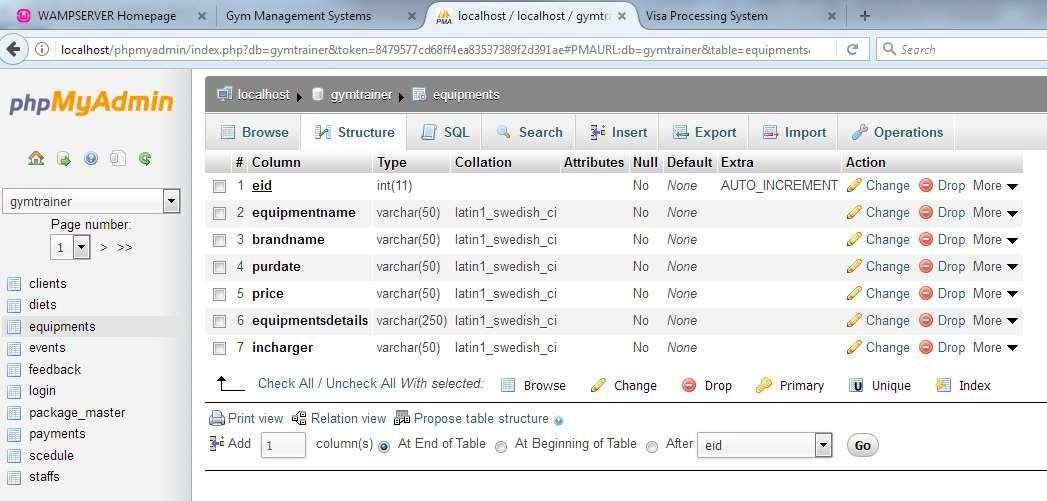
##### LOGIN TABLE



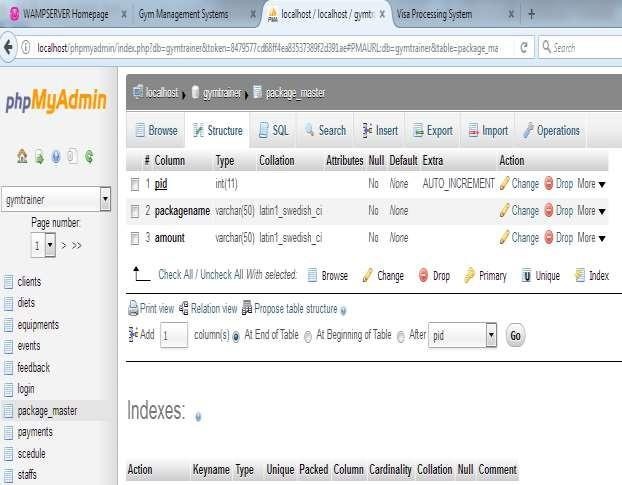
CLIENTS TABLE



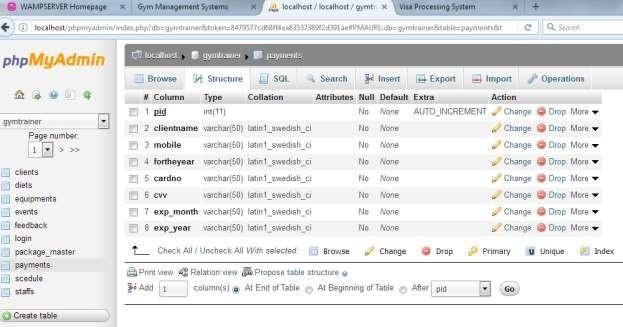
EQUIPMENTS



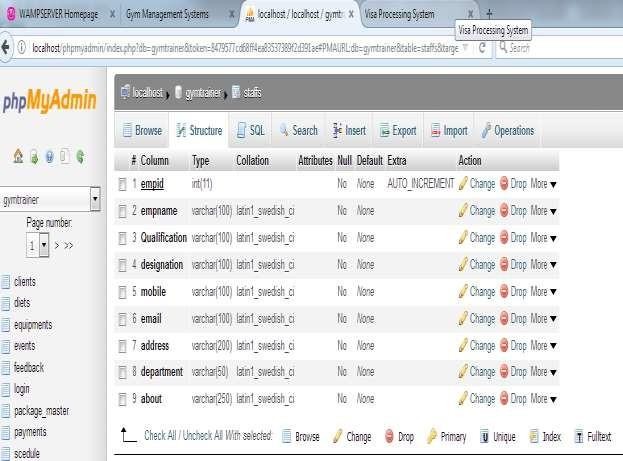
PACKAGE MASTER



PAYMENTS

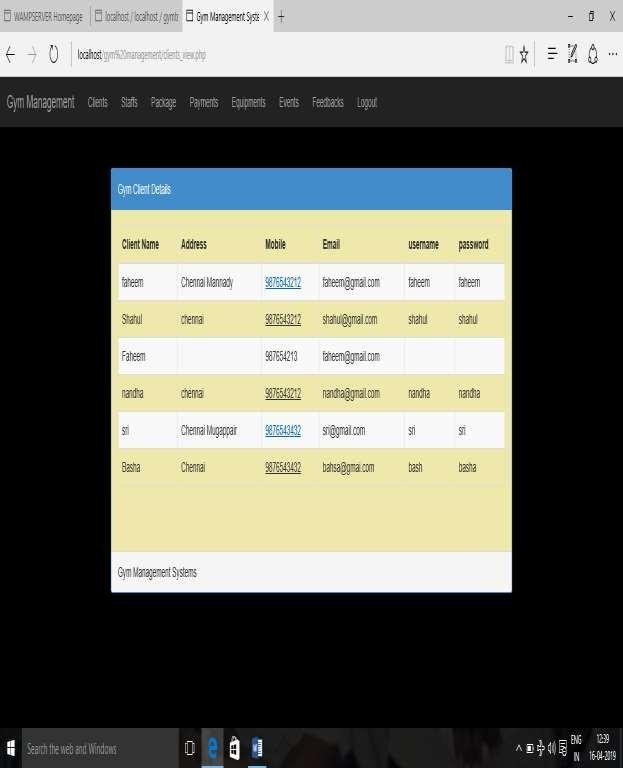


STAFFS TABLE

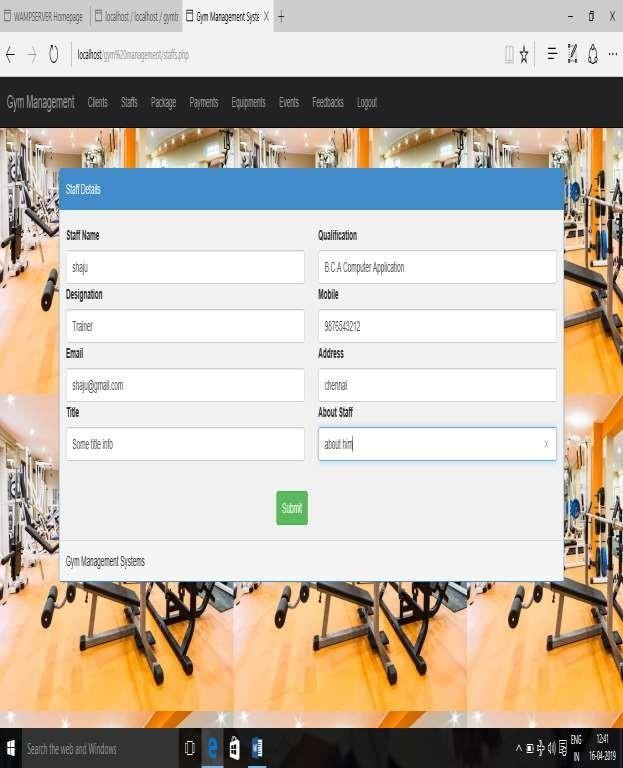


**FORM LAYOUT**

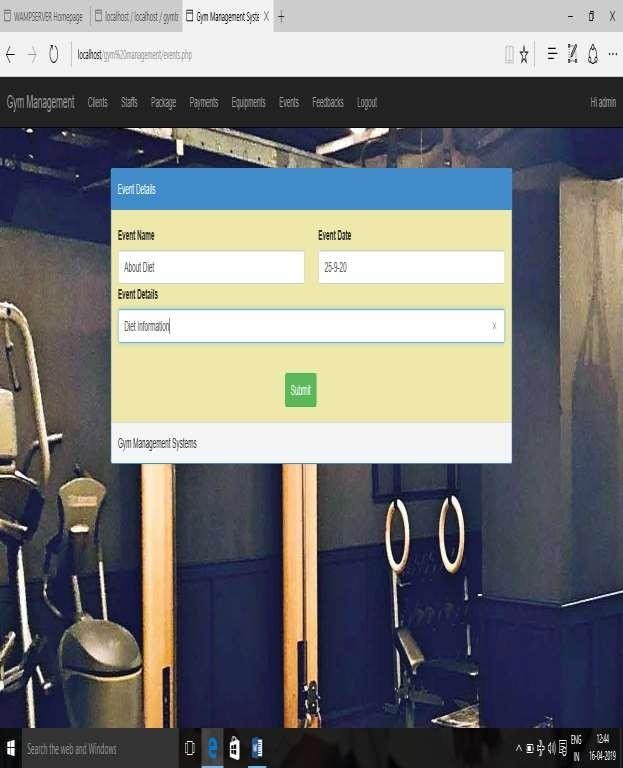
**CLIENTS FORM**



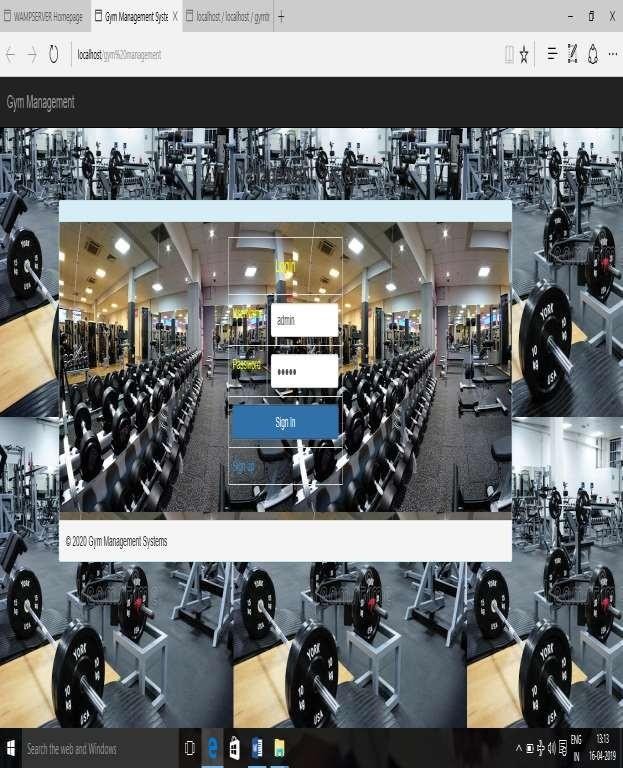
**STAFF PAGE**



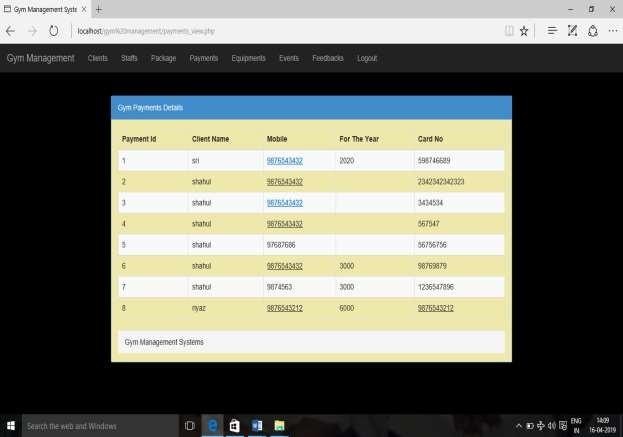
**EVENTS DETAILS**



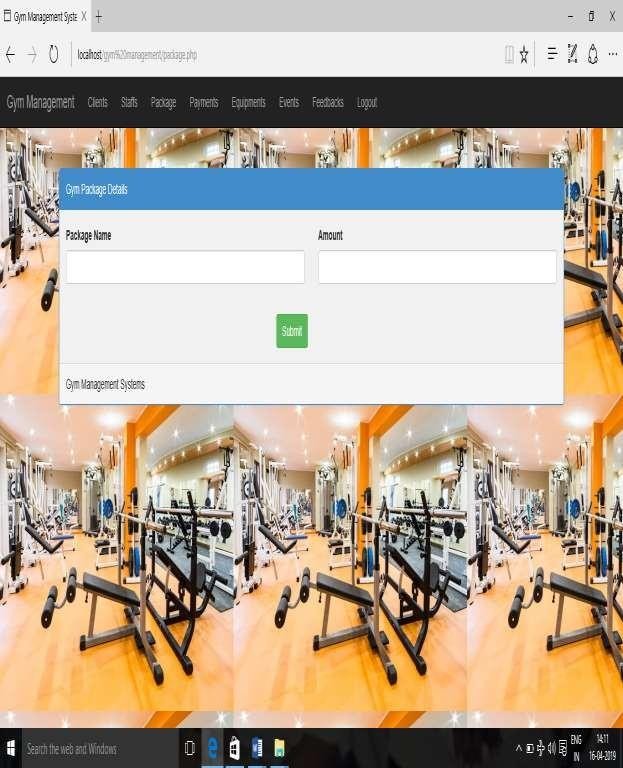
**LOGIN FORM**



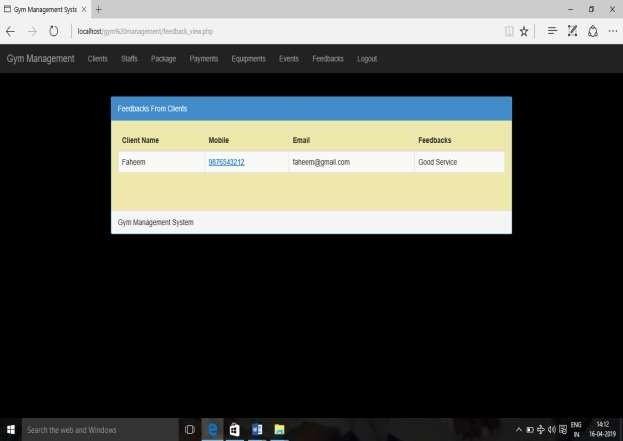
**PAYMENT FORM**



**PACKAGES FORM**



**FEEDBACK FORM**



**SOURCECODE**

**Clients Form:**

<?php

require "Connection.php"; function fill\_Id($con)

{

$sql = "select max(clientNo)+1 from clients";

$result=mysql\_query($sql);

while($data = mysql\_fetch\_row($result))

{

echo $data[0];

}

}

function fill\_department($con)

{

$output='';

$sql = "select department from subjectmaster";

$result=mysql\_query($sql);

while($row = mysql\_fetch\_array($result))

{

$output .= '<option value="'.$row["department"].'">'.$row["department"].'</option>';

}

return $output;

}

if(isset($\_POST['btncst']))

{

$txtclientname= $\_POST['txtclientname'];

$txtaddress = $\_POST['txtaddress'];

$txtmobile = $\_POST['txtmobile'];

$txtemail = $\_POST['txtemail'];

$txtusername = $\_POST['txtusername'];

$txtpassword = $\_POST['txtpassword'];

mysql\_query("insert into clients set

Working 2 : '.mysql\_error());

clientname = '$txtclientname', address = '$txtaddress',

mobile = '$txtmobile',

email = '$txtemail',

username = '$txtusername',

password = '$txtpassword'")or die('Query Not

$clients="User";

mysql\_query("insert into login(username,password,role) values('$txtusername','$txtpassword','$clients')");

echo "Clients Added Successfully"; header("location:index.php");

}

?>

<!DOCTYPE html>

<html>

<head>

<meta charset="UTF-8">

<title>Gym Management Systems</title>

<link rel="stylesheet" href="css/bootstrap.min.css"/>

<script src="js/jquery-2.2.2.min.js"></script>

<script src="js/bootstrap.min.js"></script>

<script src="js/main.js"></script>

</head>

<body style="background-image:url(images/10.jpg);">

<div class="navbar navbar-inverse navbar-fixed-top">

<div class="container-fluid">

<div class="navbar-header">

<a href="index.php" class="navbar-brand">Fitness Management</a>

</div>

<ul class="nav navbar-nav">

<li><a href="index.php">Back</a></li>

</div>

</div>

</ul>

<p><br></p>

<p><br></p>

<p><br></p>

<div class="container-fluid">

<div class="row">

<div class="col-md-12" id="signup\_msg">

</div>

color:#EEE8AA;">

<div class="col-md-2"></div>

<div class="col-md-8">

<div class="panel panel-primary">

<div class="panel-heading">Client Details</div>

<div class="panel-body" style="background-

<form method="post">

<div class="row">

name="txtclientname" class="form-control">

<div class="col-md-6">

<label for="Name">Client Name</label>

<input type="text" id="txtclientname"

</div>

<div class="col-md-6">

<label for="Name">Addres</label>

<input type="text" id="txtaddress"

name="txtaddress" class="form-control">

id="txtmobile" name="txtmobile">

</div>

</div>

<div class="row">

<div class="col-md-6">

<label for="Name">Mobile</label>

<input type="text" class="form-control"

id="txtemail" name="txtemail">

</div>

<div class="col-md-6">

<label for="Name">Email</label>

<input type="text" class="form-control"

</div>

</div>

<div class="row">

id="txtusername" name="txtusername">

<div class="col-md-6">

<label for="Name">Username</label>

<input type="text" class="form-control"

control" id="txtpassword" name="txtpassword">

</div>

<div class="col-md-6">

<label for="Name">Password</label>

<input type="password" class="form-

</div>

</div>

<div class="row">

</div>

<p></br></p>

<div class="row">

<div class="col-md-5"></div>

<div class="col-md-7">

class="btn btn-success btn-lg">Signup</button>

</div>

</div>

<button id="btncst" name="btncst"

</div>

</div>

</div>

</form>

<div class="panel-footer">Fitness Management Systems</div>

</div>

</body>

</html>

**Clients view**

<?php

</div>

<div class="col-md-2"></div>

require "Connection.php";

$strclients="select \* from clients";

?>

<!DOCTYPE html>

<html>

<head>

<meta charset="UTF-8">

<title>Gym Management Systems</title>

<link rel="stylesheet" href="css/bootstrap.min.css"/>

<script src="js/jquery-2.2.2.min.js"></script>

<script src="js/bootstrap.min.js"></script>

<script src="js/main.js"></script>

</head>

<body style="background-color:#000000;">

<div class="navbar navbar-inverse navbar-fixed-top">

<div class="container-fluid">

<div class="navbar-header">

<a href="index.php" class="navbar-brand">Gym Management</a>

</div>

<ul class="nav navbar-nav">

<li><a href="clients\_view.php">Clients</a></li>

</div>

</div>

</ul>

<li><a href="staffs.php">Staffs</a></li>

<li><a href="package.php">Package</a></li>

<li><a href="payments\_view.php">Payments</a></li>

<li><a href="equipments.php">Equipments</a></li>

<li><a href="events.php">Events</a></li>

<li><a href="feedback\_view.php">Feedbacks</a></li>

<li><a href="logout.php">Logout</a></li>

<p><br></p>

<p><br></p>

<p><br></p>

<div class="container-fluid">

<div class="row">

<div class="col-md-12" id="signup\_msg">

</div>

color:#EEE8AA;">

<div class="col-md-2"></div>

<div class="col-md-8">

<div class="panel panel-primary">

<div class="panel-heading">Gym Client Details</div>

<div class="panel-body" style="background-

<form method="post">

<table class="table table-striped table-hover table-bordered">

<thead class="thead-dark">

<tr>

</tr>

</thead>

<tbody>

<?php

<th>Client Name</th>

<th>Address</th>

<th>Mobile</th>

<th>Email</th>

<th>username</th>

<th>password</th>

$results = mysqli\_query($con,"select \* from clients"); while ($row=mysqli\_fetch\_array($results))

{

?></td>

?></td>

?></td>

?>

<tr>

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

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

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

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

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

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

</tr>

<?php

}

?>

</tbody>

</table>

<p></br></p>

</div>

</div>

</div>

</form>

<div class="panel-footer">Gym Management Systems</div>

</div>

</body>

</html>

**Equipments**

<?php

</div>

<div class="col-md-2"></div>

session\_start();

require "Connection.php";

function fill\_staffs($con)

{

$output='';

$sql = "select distinct(empname) from staffs";

$result=mysql\_query($sql);

while($row = mysql\_fetch\_array($result))

{

$output .= '<option value="'.$row["empname"].'">'.$row["empname"].'</option>';

}

return $output;

}

if(isset($\_POST['btncst']))

{

$txtname = $\_POST['txtname'];

$txtbrand = $\_POST['txtbrand'];

$txtpurdate = $\_POST['txtpurdate'];

$txtprice = $\_POST['txtprice'];

$txtequipments = $\_POST['txtequipments'];

$txtstaffname = $\_POST['txtstaffname'];

mysqli\_query($con,"insert into equipments set

equipmentname = '$txtname',

brandname = '$txtbrand', purdate = '$txtpurdate',

Working 2 : '.mysqli\_error());

price = '$txtprice',

equipmentsdetails = '$txtequipments', incharger = '$txtstaffname'")or die('Query Not

echo "

label='close'>&times;</a>

<div class='alert alert-success'>

<p><br></p><p><br></p>

<a href='#' class='close' data-dismiss='alert' aria-

<b>Equipments Added Successfully</b>

</div> ";

}

?>

<!DOCTYPE html>

<html>

<head>

<meta charset="UTF-8">

<title>Gym Management Systems</title>

<link rel="stylesheet" href="css/bootstrap.min.css"/>

<script src="js/jquery-2.2.2.min.js"></script>

<script src="js/bootstrap.min.js"></script>

<script src="js/main.js"></script>

</head>

<body style="background-image:url(images/10.jpg)">

<div class="navbar navbar-inverse navbar-fixed-top">

<div class="container-fluid">

<div class="navbar-header">

<a href="MasterAdmin.php" class="navbar-brand">Gym

Management</a>

</div>

<ul class="nav navbar-nav">

<li><a href="clients\_view.php">Clients</a></li>

<li><a href="staffs.php">Staffs</a></li>

<li><a href="package.php">Package</a></li>

<li><a href="payments\_view.php">Payments</a></li>

<li><a href="equipments.php">Equipments</a></li>

<li><a href="events.php">Events</a></li>

<li><a href="feedback\_view.php">Feedbacks</a></li>

<li><a href="logout.php">Logout</a></li>

</ul>

<ul class="nav navbar-nav navbar-right">

<li><a href="#"><?php echo "Hi ".$\_SESSION["uname"]; ?></a>

</li>

</ul>

</div>

</div>

<p><br></p>

<p><br></p>

<p><br></p>

<div class="container-fluid">

<div class="row">

<div class="col-md-12" id="signup\_msg">

</div>

color:#EEE8AA;">

<div class="col-md-2"></div>

<div class="col-md-8">

<div class="panel panel-primary">

<div class="panel-heading">Equipments Details</div>

<div class="panel-body" style="background-

<form method="post">

<div class="row">

Name</label>

name="txtname" class="form-control">

<div class="col-md-6">

<label for="Name">Equipment

<input type="text" id="txtname"

id="txtbrand" name="txtbrand">

</div>

<div class="col-md-6">

<label for="Name">Brand Name</label>

<input type="text" class="form-control"

</div>

</div>

<div class="row">

Date</label>

name="txtpurdate" class="form-control">

<div class="col-md-6">

<label for="Name">Purchase

<input type="text" id="txtpurdate"

id="txtprice" name="txtprice">

</div>

<div class="col-md-6">

<label for="Name">Price</label>

<input type="text" class="form-control"

</div>

</div>

<div class="row">

Details</label>

name="txtequipments" class="form-control">

<div class="col-md-6">

<label for="Name">Equipment

<input type="text" id="txtequipments"

id="txtstaffname" class="form-control">

</div>

<div class="col-md-6">

<label for="Name">Incharger</label>

<select name="txtstaffname"

<option

value="">Select Staff Name</option> fill\_staffs($con); ?>

</div>

</div>

<?php echo

</select>

<div class="row">

</div>

<p></br></p>

<div class="row">

<div class="col-md-5"></div>

<div class="col-md-7">

class="btn btn-success btn-md">Submit</button>

</div>

</div>

<button id="btncst" name="btncst"

</div>

</div>

</div>

</form>

<div class="panel-footer">Gym Management Systems</div>

</div>

</body>

</html>

</div>

<div class="col-md-2"></div>

**Events Source Code**

<?php session\_start();

require "Connection.php";

if(isset($\_POST['btncst']))

{

$txtname = $\_POST['txtname'];

$txteventdate = $\_POST['txteventdate'];

$txtdetails = $\_POST['txtdetails'];

mysqli\_query($con,"insert into events set

|  |  |  |
| --- | --- | --- |
| eventname | = | '$txtname', |
| eventdate | = | '$txteventdate', |

Working 2 : '.mysqli\_error()); echo "

label='close'>&times;</a>

eventdetails = '$txtdetails'")or die('Query Not

<div class='alert alert-success'>

<p><br></p><p><br></p>

<a href='#' class='close' data-dismiss='alert' aria-

<b>Events Added Successfully</b>

</div> ";

}

?>

<!DOCTYPE html>

<html>

<head>

<meta charset="UTF-8">

<title>Gym Management Systems</title>

<link rel="stylesheet" href="css/bootstrap.min.css"/>

<script src="js/jquery-2.2.2.min.js"></script>

<script src="js/bootstrap.min.js"></script>

<script src="js/main.js"></script>

</head>

<body style="background-image:url(images/10.jpg)">

<div class="navbar navbar-inverse navbar-fixed-top">

<div class="container-fluid">

<div class="navbar-header">

<a href="MasterAdmin.php" class="navbar-brand">Gym

Management</a>

</div>

<ul class="nav navbar-nav">

<li><a href="clients\_view.php">Clients</a></li>

<li><a href="staffs.php">Staffs</a></li>

<li><a href="package.php">Package</a></li>

<li><a href="payments\_view.php">Payments</a></li>

<li><a href="equipments.php">Equipments</a></li>

<li><a href="events.php">Events</a></li>

<li><a href="feedback\_view.php">Feedbacks</a></li>

<li><a href="logout.php">Logout</a></li>

</ul>

<ul class="nav navbar-nav navbar-right">

<li><a href="#"><?php echo "Hi ".$\_SESSION["uname"]; ?></a>

</li>

</ul>

</div>

</div>

<p><br></p>

<p><br></p>

<p><br></p>

<div class="container-fluid">

<div class="row">

<div class="col-md-12" id="signup\_msg">

</div>

color:#EEE8AA;">

<div class="col-md-2"></div>

<div class="col-md-8">

<div class="panel panel-primary">

<div class="panel-heading">Event Details</div>

<div class="panel-body" style="background-

<form method="post">

<div class="row">

name="txtname" class="form-control">

<div class="col-md-6">

<label for="Name">Event Name</label>

<input type="text" id="txtname"

id="txteventdate" name="txteventdate">

</div>

<div class="col-md-6">

<label for="Name">Event Date</label>

<input type="text" class="form-control"

</div>

</div>

<div class="row">

Details</label>

name="txtdetails" class="form-control">

<div class="col-md-12">

<label for="Name">Event

<input type="text" id="txtdetails"

</div>

</div>

<p></br></p>

<div class="row">

<div class="col-md-5"></div>

<div class="col-md-7">

class="btn btn-success btn-md">Submit</button>

</div>

</div>

<button id="btncst" name="btncst"

</div>

</div>

</div>

</form>

<div class="panel-footer">Gym Management Systems</div>

</div>

</body>

</html>

**Feedbacks**

<?php

</div>

<div class="col-md-2"></div>

require "Connection.php";

if(isset($\_POST['btncst']))

{

$txtclientname= $\_POST['txtclientname'];

$txtmobile = $\_POST['txtmobile'];

$txtemail = $\_POST['txtemail'];

$txtfeedback = $\_POST['txtfeedback'];

mysqli\_query($con,"insert into feedback set

clientname = '$txtclientname', mobile = '$txtmobile',

email = '$txtemail',

Working 2 : '.mysqli\_error()); echo "

label='close'>&times;</a>

feedbacks = '$txtfeedback'")or die('Query Not

<div class='alert alert-success'>

<p><br></p><p><br></p>

<a href='#' class='close' data-dismiss='alert' aria-

<b>Feedback Sent Successfully</b>

</div> ";

}

?>

<!DOCTYPE html>

<html>

<head>

<meta charset="UTF-8">

<title>Gym Management Systems</title>

<link rel="stylesheet" href="css/bootstrap.min.css"/>

<script src="js/jquery-2.2.2.min.js"></script>

<script src="js/bootstrap.min.js"></script>

<script src="js/main.js"></script>

</head>

<body style="background-color:#000000;">

<div class="navbar navbar-inverse navbar-fixed-top">

<div class="container-fluid">

<div class="navbar-header">

<a href="index.php" class="navbar-brand">Gym Management</a>

</div>

<ul class="nav navbar-nav">

<li><a href="coachers\_view.php">Coachers</a></li>

<li><a href="events\_view.php">Events</a></li>

<li><a href="equipments\_view.php">Equipments</a></li>

<li><a href="payments.php">Payments</a></li>

<li><a href="feedbacks.php">Feedbacks</a></li>

<li><a href="logout.php">Logout</a></li>

</div>

</div>

</ul>

<p><br></p>

<p><br></p>

<p><br></p>

<div class="container-fluid">

<div class="row">

<div class="col-md-12" id="signup\_msg">

</div>

From Users</div> color:#EEE8AA;">

<div class="col-md-2"></div>

<div class="col-md-8">

<div class="panel panel-primary">

<div class="panel-heading">Gym Management Feedbacks

<div class="panel-body" style="background-

<form method="post">

<div class="row">

name="txtclientname" class="form-control">

<div class="col-md-6">

<label for="Name">Client Name</label>

<input type="text" id="txtclientname"

</div>

<div class="col-md-6">

<label for="Name">Mobile</label>

id="txtmobile" name="txtmobile">

</div>

</div>

<input type="text" class="form-control"

<div class="row">

id="txtemail" name="txtemail">

<div class="col-md-6">

<label for="Name">Email</label>

<input type="text" class="form-control"

id="txtfeedback" name="txtfeedback">

</div>

<div class="col-md-6">

<label for="Name">Feedbacks</label>

<input class="form-control"

</div>

</div>

<div class="row">

</div>

<div class="row">

</div>

<p></br></p>

<div class="row">

<div class="col-md-5"></div>

<div class="col-md-7">

class="btn btn-success btn-md">Submit</button>

</div>

</div>

<button id="btncst" name="btncst"

</div>

</div>

</div>

</form>

<div class="panel-footer">Gym Management Systems</div>

</div>

</div>

<div class="col-md-2"></div>

</body>

</html>

**Payments**

<?php session\_start();

require "Connection.php"; function fill\_package($con)

{

$output='';

$sql = "select amount,packagename from package\_master";

$result=mysqli\_query($con,$sql); while($row = mysqli\_fetch\_array($result))

{

$output .= '<option value="'.$row["amount"].'">'.$row["packagename"].'</option>';

}

return $output;

}

function fill\_Id($con)

{

$sql = "select max(clientNo)+1 from clients";

$result=mysqli\_query($con,$sql); while($data = mysqli\_fetch\_row($result))

{

echo $data[0];

}

}

function fill\_department($con)

{

$output='';

$sql = "select department from subjectmaster";

$result=mysqli\_query($con,$sql); while($row = mysqli\_fetch\_array($result))

{

$output .= '<option value="'.$row["department"].'">'.$row["department"].'</option>';

}

return $output;

}

if(isset($\_POST['btncst']))

{

$txtclientname= $\_POST['txtclientname'];

$txtyear = $\_POST['txtyear'];

$txtmobile = $\_POST['txtmobile'];

$txtcardno = $\_POST['txtcardno'];

$txtcvv= $\_POST['txtcvv'];

$txtexpyear = $\_POST['txtexpyear'];

$txtmonth = $\_POST['txtmonth'];

mysqli\_query($con,"insert into payments set

Working 2 : '.mysqli\_error());

echo "

label='close'>&times;</a>

clientname = '$txtclientname', mobile = '$txtmobile',

fortheyear = '$txtyear', cardno = '$txtcardno',

cvv = '$txtcvv', exp\_month ='$txtmonth',

exp\_year = '$txtexpyear'")or die('Query Not

<div class='alert alert-success'>

<p><br></p><p><br></p>

<a href='#' class='close' data-dismiss='alert' aria-

<b>Payments Made Successfully</b>

}

?>

<!DOCTYPE html>

<html>

</div> ";

<head>

<meta charset="UTF-8">

<title>Gym Management Systems</title>

<link rel="stylesheet" href="css/bootstrap.min.css"/>

<script src="js/jquery-2.2.2.min.js"></script>

<script src="js/bootstrap.min.js"></script>

<script src="js/main.js"></script>

</head>

<body style="background-color:#000000;">

<div class="navbar navbar-inverse navbar-fixed-top">

<div class="container-fluid">

<div class="navbar-header">

<a href="index.php" class="navbar-brand">Gym Management</a>

</div>

<ul class="nav navbar-nav">

<li><a href="coachers\_view.php">Coachers</a></li>

<li><a href="events\_view.php">Events</a></li>

<li><a href="equipments\_view.php">Equipments</a></li>

<li><a href="payments.php">Payments</a></li>

<li><a href="feedbacks.php">Feedbacks</a></li>

<li><a href="logout.php">Logout</a></li>

</div>

</div>

</ul>

<p><br></p>

<p><br></p>

<p><br></p>

<div class="container-fluid">

<div class="row">

<div class="col-md-12" id="signup\_msg">

</div>

color:#EEE8AA;">

<div class="col-md-2"></div>

<div class="col-md-8">

<div class="panel panel-primary">

<div class="panel-heading">Payments Details</div>

<div class="panel-body" style="background-

<form method="post">

<div class="row">

<div class="col-md-4">

<label for="Name">Client Name</label>

<input type="text" id="txtclientname"

name="txtclientname" class="form-control" value="<?php echo $\_SESSION["uname"]; ?>" readonly >

</div>

name="txtmobile" class="form-control">

<div class="col-md-4">

<label for="Name">Mobile</label>

<input type="text" id="txtmobile"

Year</label>

</div>

<div class="col-md-4">

<label for="Name">For The

class="form-control"> value="">Select Package</option> fill\_package($con); ?>

</div>

</div>

<select name="txtyear" id="txtyear"

<option

<?php echo

</select>

name="txtcardno" class="form-control">

<div class="row">

<div class="col-md-4">

<label for="Name">Card No</label>

<input type="text" id="txtcardno"

</div>

<div class="col-md-4">

<label for="Name">CVV No</label>

name="txtcvv" class="form-control">

</div>

<input type="text" id="txtcvv"

name="txtmonth" class="form-control">

<div class="col-md-2">

<label for="Name">Exp Month</label>

<input type="text" id="txtmonth"

name="txtexpyear" class="form-control">

</div>

<div class="col-md-2">

<label for="Name">Exp Year</label>

<input type="text" id="txtexpyear"

</div>

</div>

<p><br></p>

<div class="row">

<div class="col-md-5"></div>

<div class="col-md-4">

<button id="btncst"

name="btncst" class="btn btn-success btn-md">Submit</button>

</div>

<p><br></p>

</form>

</div>

</div>

</div>

<div class="panel-footer">Gym Management Systems</div>

</div>

</body>

</html>

</div>

<div class="col-md-2"></div>

# Conclusion

#### CONCLUSION:

The package was designed in such a way that future modifications can be done easily. The following conclusion can be deduced from the development of the project.

* + Automation of the entire system improves the efficiency
  + It provides a friendly graphical user interface which proves to be better when compared to the existing system.
  + It gives appropriate access to the authorized users depending on their permissions.
  + It effectively overcomes the delay in communications.
  + Updating of information becomes so easier.
  + System security, data security and reliability are the striking features.
  + The System has adequate scope for modification in future if it is necessary.

**FutuRE ENHANCEMENt**

#### FUTURE ENHANCEMENT:

This application avoids the manual work and the problems concern with it. It is an easy way to obtain the information regarding the various travel services that are present in our System.

Well I have worked hard in order to present an improved website better than the existing one‟s regarding the information about the various activities. Still, we found out that the project can be done in a better way. Primarily, In this system user login and then go to customize. By using this user will send request for customize to the site. User will set the date for customize appointments. After that admin see his appointments and see the customization, modification also done.

The next enhancement, we will develop online services. That mean, if user have any problems he can send his problem to the admin through internet from his home then admin will send reply to him. In this user have some login name and password.

**BIBLIOGRAPHY**

#### BIBLIOGRAPHY

* Web programming Charles Petzoid
* PHP Unleashed Stephen Walther
* PHP JQUERY Black Book Steven Holzer
* Professional PHP Alex Homer, Dave Sussman
* MYSQL Server Essential Reference Sharon Dooley