Project Report

On

Restaurant

**Submitted by: Goreesh Shingari**

***TABLE OF CONTENTS***

1. **Introduction**
2. **Project Description**
3. Data Base Tables
4. **Coding**
5. **Snapshots**
6. **Testing**

* Introduction to Testing
* Test Strategy
* Test Cases

1. **Implementation**
2. **Documentation**
3. **Maintenance**
4. **Project Legacy**
5. **Bibliography**

***Project Title: River View Restaurant***

****

***1. INTRODUCTION:***

Project is constituent of seven words and I have tried my best to give its definition analyzing each word in the following manner:

P- Perfect Planning

R- Resources

O- Organization

J- Joint Efforts

E- Engineering Skill

C- Communication

T- Technique

* **River View Restaurant:**

It is a website where people pay to sit and eat meals that are cooked and served on the premises. It is also a place where people go to enjoy the time and to eat a meal.

**Objective:-**

Our mission is to deliver the best food items and services by using this website. There is no need to go out to the market, say sitting at home, you can order or book a table in our river view restaurant. You can easily visit our website sitting at home and know what the best foods we have are and if you like then order from home too or you can book your table in our restaurant through our website.

We also have different types of dishes like Chinese, traditional, continental etc these all are describe in this website.

**Our vision is to develop a website which covers all the tips related to the Restaurant.**

***2. PROJECT DESCRIPTION:***

**Modules of Project**

There are various modules associate with project. These modules are working in their specific area to lead and complete the project.

**User Modules of Restaurant:**

* **Login:**

The login is used to sign in the application. The login activity contains username and password text field and login button for login process. It also contain link for forgot password.

* **Register:**

Register activity used to create new account for new user. It contains Name, E-mail id, Password and Co-Password fields. It contains Register button for register process in last.

* **Home:**

After login user directly moved to home screen. Home contains modules, information about Who We Are. What We Do, We Specialize In, customer feedback and our chef’s details.

* **Menu:**

Here user can access the menu type i.e. Special, Indian, Chinese and Punjabi. In these category user can view different food items like Besan Puda, Yogurt Rice, and Chickpea Pullav etc. User can also view item details, item name, unit price and add to cart option.

* **Book Table:**

Here user can book the table by entering their details i.e. Name, Amount of Guests, Number, Date, Time Duration, Special Request and Book button for booking confirmation.

* **Gallery:**

In Gallery you can view the different dishes with their name that we have in our restaurant like Continental, Chinese, and Traditional.

* **Contact us:**

Here the user will able to get in touch with us. In this module all the contact details are given. If any user will want to give us suggestion and feedback then user can able to give there feedback by filling the form available in the contact us module.

**Admin Module**

* **Add Menu Types:**

Here admin can view the menu information and also admin can update or delete the existing menu.

* **Add Menu Details:**

Here admin can add a new Dish with their photo, description, price and the type of dish it is. It shows the dishes we have in the menu and admin can update or delete the dishes information.

* **Table List:**

In this admin can view the bookings of table, with the name of customer, No. of guests, customer’s contact number, date of booking, time from and time to.

* **Dish Book List:**

Here in this module admin can view the total number of bookings, customer number, and total bill of customer.

* **Testimonials:**

Here admin can view the customer’s feedback with their Name, E-mail id and customer’s message that they have wrote

**Features:**

1. **Best food: -** We have all the best food items and soft drinks in our restaurant, we also have buffet system at reasonable prices.
2. **Best customer dealing:-**

Our experienced staff designs custom programs, ensuring success. And if you have any

Query then you asks, we reply you back as soon as possible.

1. **Fulfill the special request of customers:-**

We also have sections where you can tell your special requests and we will try our best to fulfill your request.

1. **Food market:-**

We have also variety of food items; you can easily buy the food according to your needs by sitting at your home.

**BENEFITS OF THE PROJECT**

* Fully works as an online
* Reducing the time
* Site has a solution
* Centralized maintenance of all information.
* Desktop as well as web application.

It involve estimating cost and benefits that can be tangible and intangible because of confusing between the types of costs it is sometimes very difficult to divide the benefits out weight the cost.

**Hardware and Software Requirements**

**Hardware:** Hardware is the physical part of the computer system like mouse, keyboard, monitor etc.

**Hardware Requirements:**

* 20 GB HDD
* Processor equal or higher than 2.0 GHz
* Color monitor
* RAM:2 GB
* CD ROM
* Keyboard and mouse.
* Internet Connection.

**Software:** Software is a set of applications which is used to run the operating system.

**Software Requirements:**

* Sublime text editor
* Programming Languages: HTML, Css, JavaScript, PHP,SQL
* Database: .ein MySQLi
* Browser: Google Chrome or higher, Internet Explorer 7 or higher

**FRONT END**

* **HTML ( Hypertext Markup Language )**

What is HTML?

HTML is an acronym which stands for **Hyper Text Markup Language** which is used for creating web pages and web applications. Let's see what is meant by Hypertext Markup Language, and Web page.

**Hyper Text:** Hypertext simply means "Text within Text." A text has a link within it, is a hypertext. Whenever you click on a link which brings you to a new webpage, you have clicked on a hypertext. Hypertext is a way to link two or more web pages (HTML documents) with each other.

**Markup language:** A markup language is a computer language that is used to apply layout and formatting conventions to a text document. Markup language makes text more interactive and dynamic. It can turn text into images, tables, links, etc.

**Web Page:** A web page is a document which is commonly written in HTML and translated by a web browser. A web page can be identified by entering an URL. A Web page can be of the static or dynamic type. **With the help of HTML only, we can create static web pages**.

Hence, HTML is a markup language which is used for creating attractive web pages with the help of styling, and which looks in a nice format on a web browser. An HTML document is made of many HTML tags and each HTML tag contains different content.

Description

**<! DOCTYPE>:** It defines the document type or it instructs the browser about the version of HTML.

**<html>**: This tag informs the browser that it is an HTML document. Text between html tags describes the web document. It is a container for all other elements of HTML except <! DOCTYPE>

**<head>:** It should be the first element inside the <html> element, which contains the metadata(information about the document). It must be closed before the body tag opens.

**<title>:** As its name suggested, it is used to add title of that HTML page which appears at the top of the browser window. It must be placed inside the head tag and should close immediately. (Optional)

**<body>**: Text between body tag describes the body content of the page that is visible to the end user. This tag contains the main content of the HTML document.

**<h1>**: Text between <h1> tag describes the first level heading of the webpage.

**<p>:** Text between <p> tag describes the paragraph of the webpage.

Brief History of HTML

In the late 1980’s, a physicist, Tim Berners-Lee who was a contractor at CERN, proposed a system for CERN researchers. In 1989, he wrote a memo proposing an internet based hypertext system.

**Tim Berners-Lee** is known as the father of HTML. The first available description of HTML was a document called "HTML Tags" proposed by Tim in late 1991. The latest version of HTML is HTML5, which we will learn later in this tutorial.

HTML Versions

Since the time HTML was invented there are lots of HTML versions in market, the brief introduction about the HTML version is given below:

**HTML 1.0:** The first version of HTML was 1.0, which was the barebones version of HTML language, and it was released in1991.

**HTML 2.0:** This was the next version which was released in 1995, and it was standard language version for website design. HTML 2.0 was able to support extra features such as form-based file upload, form elements such as text box, option button, etc.

**HTML 3.2:** HTML 3.2 version was published by W3C in early 1997. This version was capable of creating tables and providing support for extra options for form elements. It can also support a web page with complex mathematical equations. It became an official standard for any browser till January 1997. Today it is practically supported by most of the browsers.

**HTML 4.01:** HTML 4.01 version was released on December 1999, and it is a very stable version of HTML language. This version is the current official standard, and it provides added support for style sheets (CSS) and scripting ability for various multimedia elements.

**HTML 5 :** HTML5 is the newest version of Hypertext Mark-up language. The first draft of this version was announced in January 2008. There are two major organizations one is W3C (World Wide Web Consortium), and another one is WHATWG( Web Hypertext Application Technology Working Group) which are involved in the development of HTML 5 version, and still, it is under development.

Features of HTML

1) It is a very **easy and simple language**. It can be easily understood and modified.

2) It is very easy to make an **effective presentation** with HTML because it has a lot of formatting tags.

3) It is a **markup language**, so it provides a flexible way to design web pages along with the text.

4) It facilitates programmers to add a **link** on the web pages (by html anchor tag), so it enhances the interest of browsing of the user.

5) It is **platform-independent** because it can be displayed on any platform like Windows, Linux, and Macintosh, etc.

6) It facilitates the programmer to add **Graphics, Videos, and Sound** to the web pages which makes it more attractive and interactive.

7) HTML is a case-insensitive language, which means we can use tags either in lower-case or upper-case.

**\* Css (Cascading Style Sheets)**

CSS stands for Cascading Style Sheets. It is a style sheet language which is used to describe the look and formatting of a document written in markup language. It provides an additional feature to HTML. It is generally used with HTML to change the style of web pages and user interfaces. It can also be used with any kind of XML documents including plain XML, SVG and XUL.

CSS is used along with HTML and JavaScript in most websites to create user interfaces for web applications and user interfaces for many mobile applications.

What does CSS DO?

You can add new looks to your old HTML documents.

You can completely change the look of your website with only a few changes in CSS code.

Why use CSS

These are the three major benefits of CSS:

1) Solves a big problem

Before CSS, tags like font, colour, background style, element alignments, border and size had to be repeated on every web page. This was a very long process. For example: If you are developing a large website where fonts and colour information are added on every single page, it will be become a long and expensive process. CSS was created to solve this problem. It was a W3C recommendation.

2) Saves a lot of time

CSS style definitions are saved in external CSS files so it is possible to change the entire website by changing just one file.

3) Provide more attributes

CSS provides more detailed attributes than plain HTML to define the look and feel of the website.

**J S (JAVA SCRIPT)**

What is JavaScript?

JavaScript (js) is a light-weight object-oriented programming language which is used by several websites for scripting the WebPages. It is an interpreted, full-fledged programming language that enables dynamic interactivity on websites when applied to an HTML document. It was introduced in the year 1995 for adding programs to the webpage in the Netscape Navigator browser. Since then, it has been adopted by all other graphical web browsers. With JavaScript, users can build modern web applications to interact directly without reloading the page every time. The traditional website uses JavaScript to provide several forms of interactivity and simplicity.

Although, JavaScript has no connectivity with Java programming language. The name was suggested and provided in the times when Java was gaining popularity in the market. In addition to web browsers, databases such as CouchDB and MongoDB use JavaScript as their scripting and query language.

Features of JavaScript

There are following features of JavaScript:

All popular web browsers support JavaScript as they provide built-in execution environments.

JavaScript follows the syntax and structure of the C programming language. Thus, it is a structured programming language.

JavaScript is a weakly typed language, where certain types are implicitly cast (depending on the operation).

JavaScript is an object-oriented programming language that uses prototypes rather than using classes for inheritance.

It is a light-weighted and interpreted language.

It is a case-sensitive language.

JavaScript is supportable in several operating systems including, Windows, macOS, etc.

It provides good control to the users over the web browsers.

History of JavaScript

In 1993, **Mosaic**, the first popular web browser, came into existence. In the **year 1994**, **Netscape** was founded by **Marc Andreessen**. He realized that the web needed to become more dynamic. Thus, a 'glue language' was believed to be provided to HTML to make web designing easy for designers and part-time programmers. Consequently, in 1995, the company recruited **Brendan Each** intending to implement and embed Scheme programming language to the browser. But, before Brendan could start, the company merged with **Sun Microsystems** for adding Java into its Navigator so that it could compete with Microsoft over the web technologies and platforms. Now, two languages were there: Java and the scripting language. Further, Netscape decided to give a similar name to the scripting language as Java's. It led to 'JavaScript'. Finally, in May 1995, Marc Andreessen coined the first code of JavaScript named '**Mocha**'. Later, the marketing team replaced the name with '**Live Script**'. But, due to trademark reasons and certain other reasons, in December 1995, the language was finally renamed to 'JavaScript'. From then, JavaScript came into existence.

Application of JavaScript

JavaScript is used to create interactive websites. It is mainly used for:

Client-side validation,

Dynamic drop-down menus,

Displaying date and time,

Displaying pop-up windows and dialog boxes (like an alert dialog box, confirm dialog box and prompt dialog box),

Displaying clocks etc.

**BACK END**

**MySQLi (MYSQL Improved)**

* MySQLi is a database server
* MySQLi is ideal for both small and large applications
* MySQLi supports standard SQL
* MySQLi compiles on a number of platforms
* MySQLi is free to download and use
* MySQLi also supports prepared statements which protect from SQL Injection.
* MySQLi is object-oriented

**1.2.1Advantages of MySQLi in database development**

MySQLi is one of the top databases available in the market. It is the improved version of MYSQL. **MySQLi is a relational database** with many advanced features and options. Over time, MySQLi has proved itself to be a fast, reliable and cost effective competitor to the other more expensive databases like MS SQL Server and Oracle. Here are a few of the advantages of using MySQLi.

**Open Source**

MySQLi is an open source database system which means that anyone can use it for free. Developers can amend its code to suit their requirements which means that MySQLi is highly customizable as well. Another edge of using MySQLi over other database systems is that; it is available widely in the market with no ownership cost.

**Fast Development**

A lot of people around the globe are continuously developing new modules for integration with MySQLi. This means that it has a wider and faster development circle. Patches, upgrades and fixes are developed fast and become available in forums, blogs and developer sites on the internet.

**Better for Small Businesses**

This **relational database system** is free so it reduces the cost of overall database solution for small businesses and companies. This database is relatively easy to learn and operate, so operational cost is reduced substantially which is again an important factor in classifying MySQLi as an applicable and practical tool for small businesses.

**Cross Platform Operability**

MySQLi is easily installable and operable on different platforms including Windows, Linux, OS2 and Solaris. Cross platform operability makes it a favorable choice for development companies. MySQLi Database System also contains APIs for integration with C, C++, PHP, Java, Perl, Python, Tcl, and Ruby etc. You can connect it easily with different development platforms so you can actually integrate applications developed in different OS and development platforms.

**Security**

MySQLi as a relational database is secure as all access passwords are stored in an encrypted format restricting any unauthorized access to the system. It also encrypts the transactions so eavesdroppers and data harvest tools cannot replicate or regenerate the database transactions once they are processed. MySQLi also supports prepared statements which protect from SQL Injection.

**Connectivity**

MySQLi clients can access this relational database through standard TCP/IP sockets, named pipes, UNIX sockets and many more. Standard ODBC 2.5 and above functions and commands are also supported in MySQLi.

In short, MySQLi is a free and favored relational database system which is serving the web and application development communities.

**PHP(Hypertext Pre-processor)**

What is PHP?

PHP is an open-source, interpreted, and object-oriented scripting language that can be executed at the server-side. PHP is well suited for web development. Therefore, it is used to develop web applications (an application that executes on the server and generates the dynamic page.).

PHP was created by **Rasmus Lerdorf in 1994** but appeared in the market in 1995. **PHP 7.4.0** is the latest version of PHP, which was released on **28 November**. Some important points need to be noticed about PHP are as followed:

PHP stands for Hypertext Pre-processor.

PHP is an interpreted language, i.e., there is no need for compilation.

PHP is faster than other scripting languages, for example, ASP and JSP.

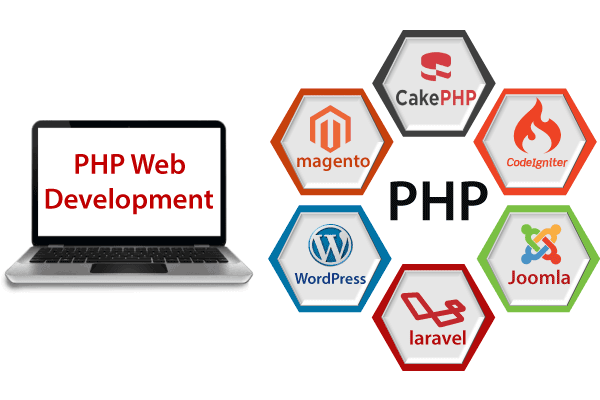
PHP is a server-side scripting language, which is used to manage the dynamic content of the website.

PHP can be embedded into HTML.

PHP is an object-oriented language.

PHP is an open-source scripting language.

PHP is simple and easy to learn language.



Why we use PHP

PHP is a server-side scripting language, which is used to design the dynamic web applications with MySQLi database.

It handles dynamic content, database as well as session tracking for the website.

You can create sessions in PHP.

It can access cookies variable and also set cookies.

It helps to encrypt the data and apply validation.

PHP supports several protocols such as HTTP, POP3, SNMP, LDAP, IMAP, and many more.

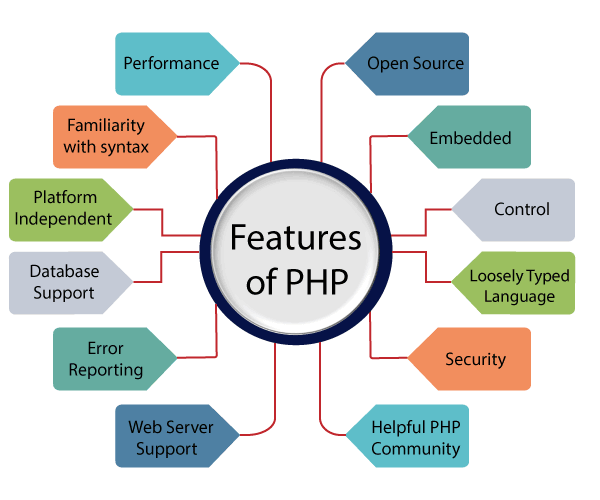
Using PHP language, you can control the user to access some pages of your website.

As PHP is easy to install and set up, this is the main reason why PHP is the best language to learn.

PHP can handle the forms, such as - collect the data from users using forms, save it into the database, and return useful information to the user. **For example** - Registration form.

PHP Features

PHP is very popular language because of its simplicity and open source. There are some important features of PHP given below:



**Performance:**

PHP script is executed much faster than those scripts which are written in other languages such as JSP and ASP. PHP uses its own memory, so the server workload and loading time is automatically reduced, which results in faster processing speed and better performance.

**Open Source:**

PHP source code and software are freely available on the web. You can develop all the versions of PHP according to your requirement without paying any cost. All its components are free to download and use.

**Familiarity with syntax:**

PHP has easily understandable syntax. Programmers are comfortable coding with it.

**Embedded:**

PHP code can be easily embedded within HTML tags and script.

**Platform Independent:**

PHP is available for WINDOWS, MAC, LINUX and UNIX operating system. A PHP application developed in one OS can be easily executed in other OS also.

**Database Support:**

PHP supports all the leading databases such as MySQL, SQLite, ODBC, etc.

**Error Reporting -**

PHP has predefined error reporting constants to generate an error notice or warning at runtime. E.g., E\_ERROR, E\_WARNING, E\_STRICT, E\_PARSE.

**Loosely Typed Language:**

PHP allows us to use a variable without declaring its data type. It will be taken automatically at the time of execution based on the type of data it contains on its value.

**Web servers Support:**

PHP is compatible with almost all local servers used today like Apache, Netscape, Microsoft IIS, etc.

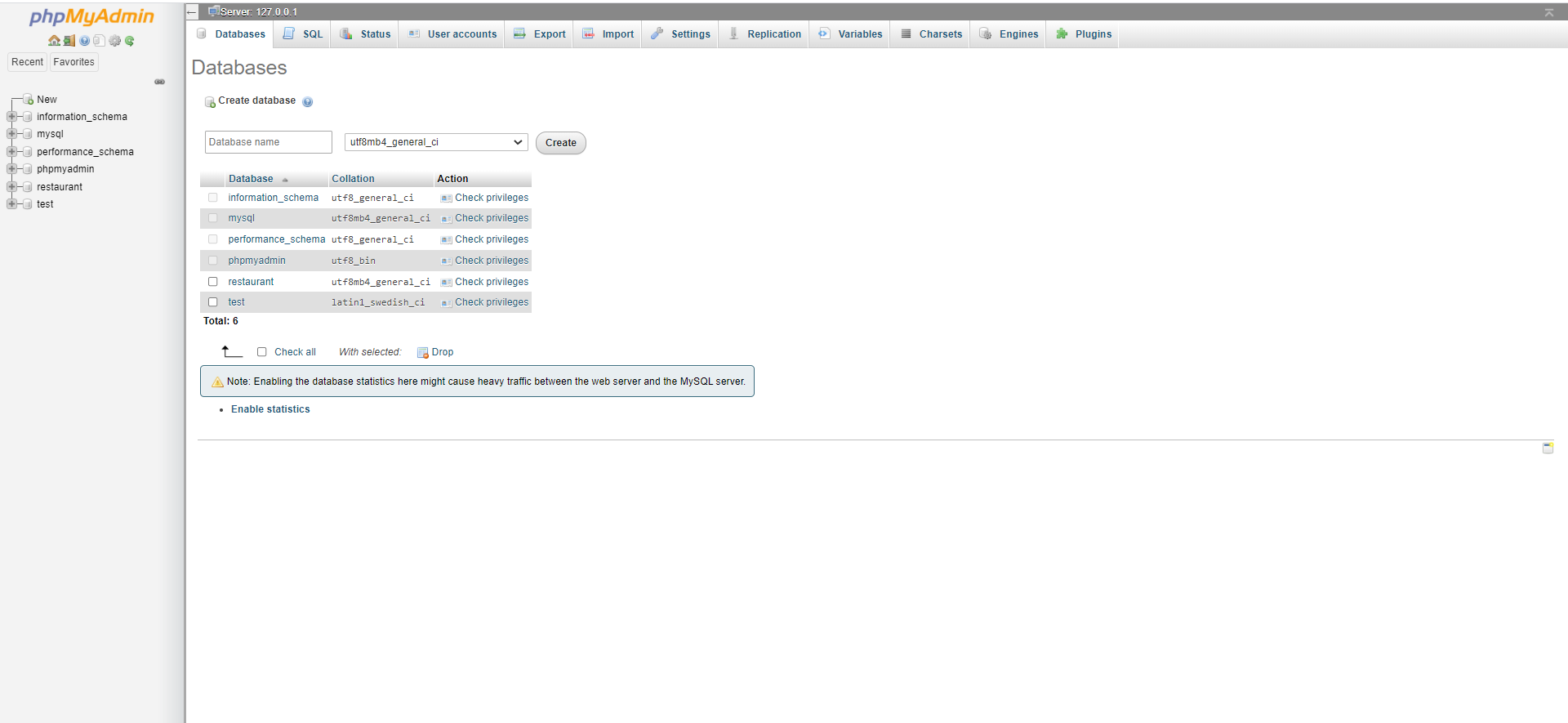
**Security:**

PHP is a secure language to develop the website. It consists of multiple layers of security to prevent threads and malicious attacks.

**Control:**

Different programming languages require long script or code, whereas PHP can do the same work in a few lines of code. It has maximum control over the websites like you can make changes easily whenever you want.

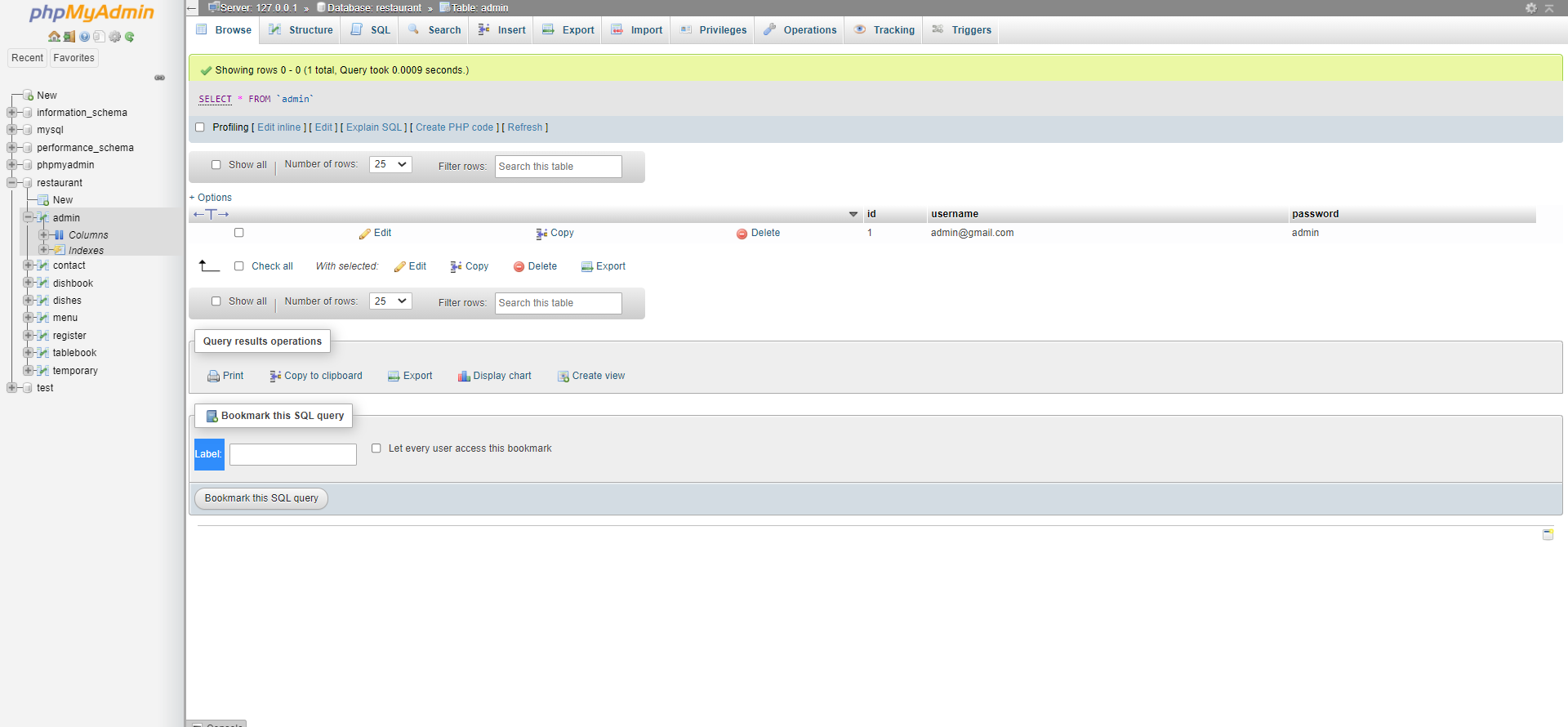
**Database Tables :**

****

**Admin Table:**

In the table information of the admin will store. In this table following details will shown :

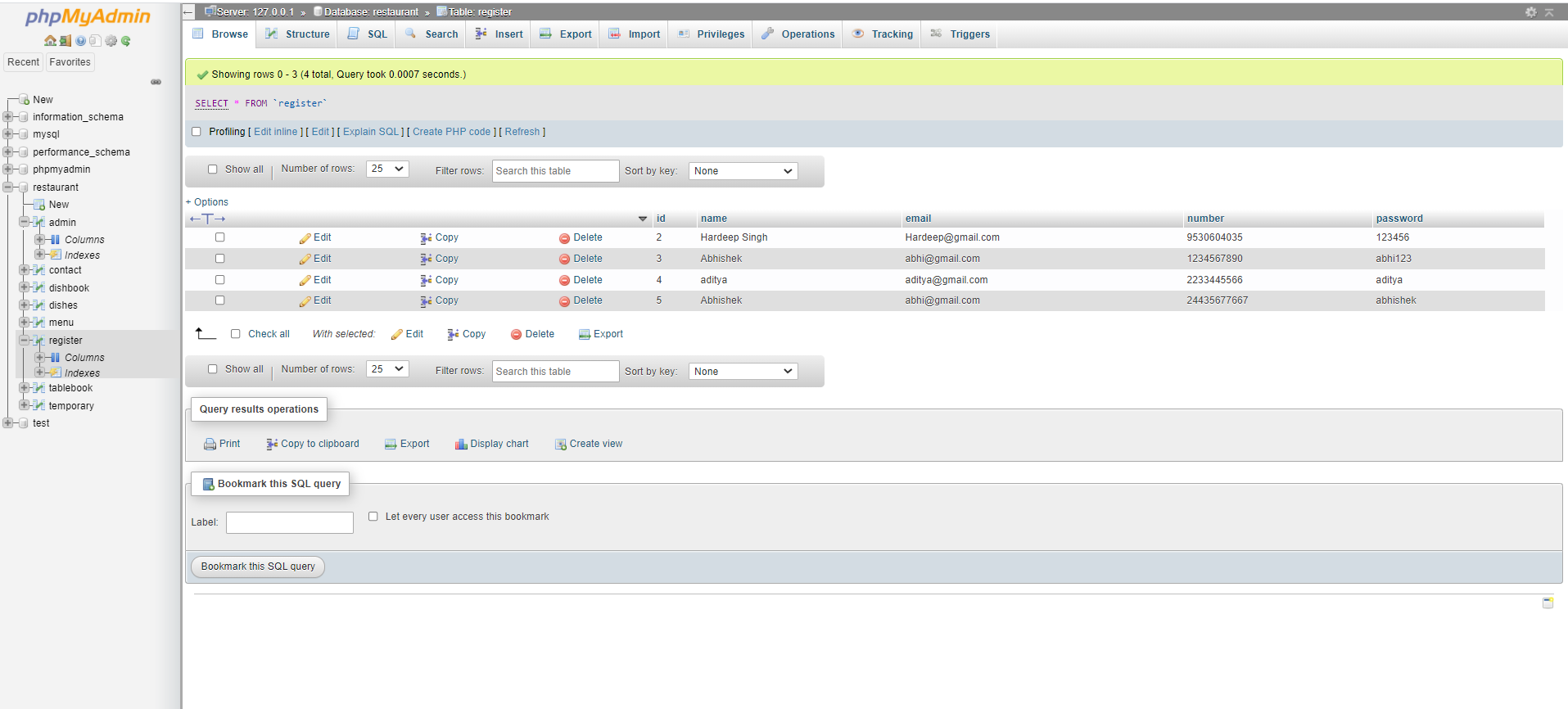
* Email of the admin.
* Password of the admin.



**Registration Table:**

In registration table all the information of the users will store. When the user log in with us, user should fill the following details:

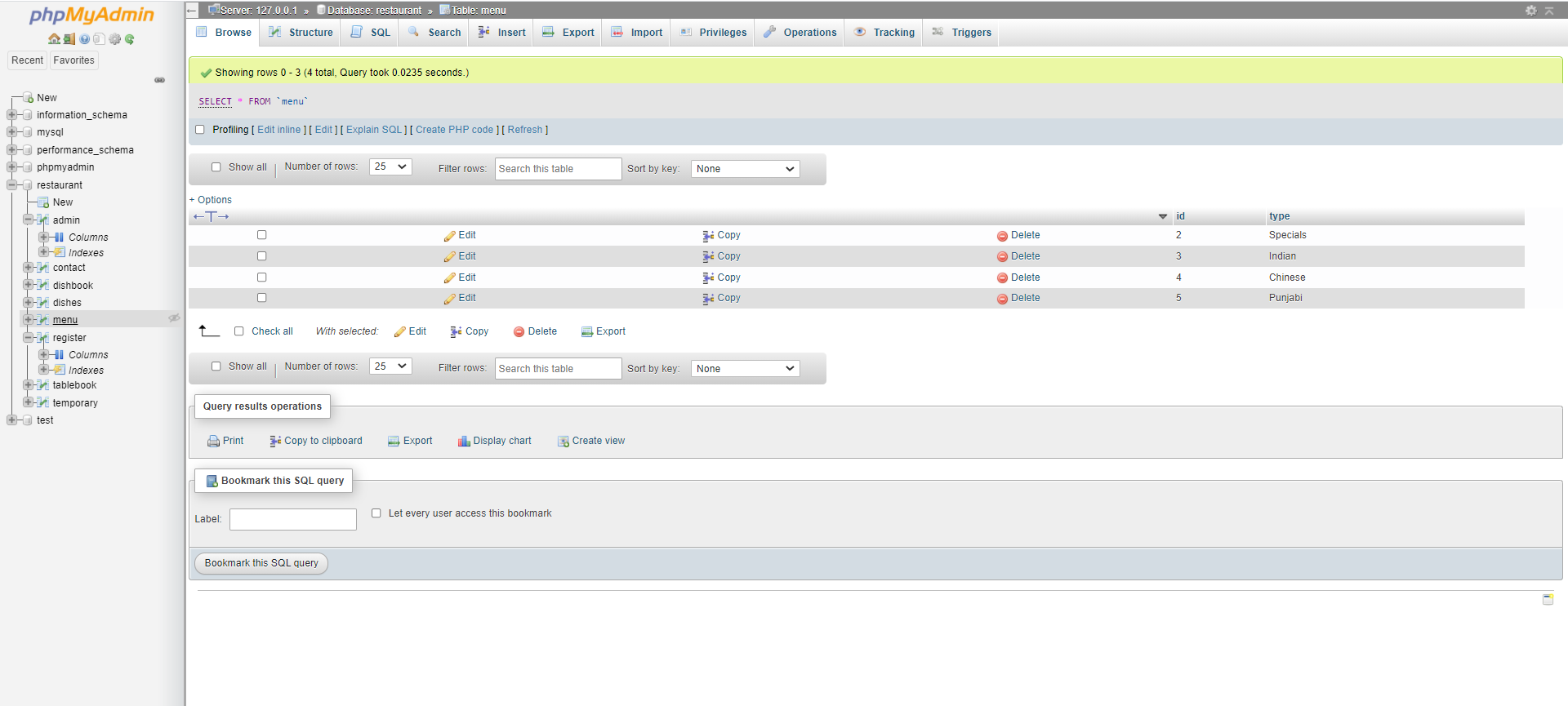
* Name
* E-mail
* Phone
* Password
* Co-password



**Menu Table :**

Here in this table we store the information about the type of menu. Table data given below:

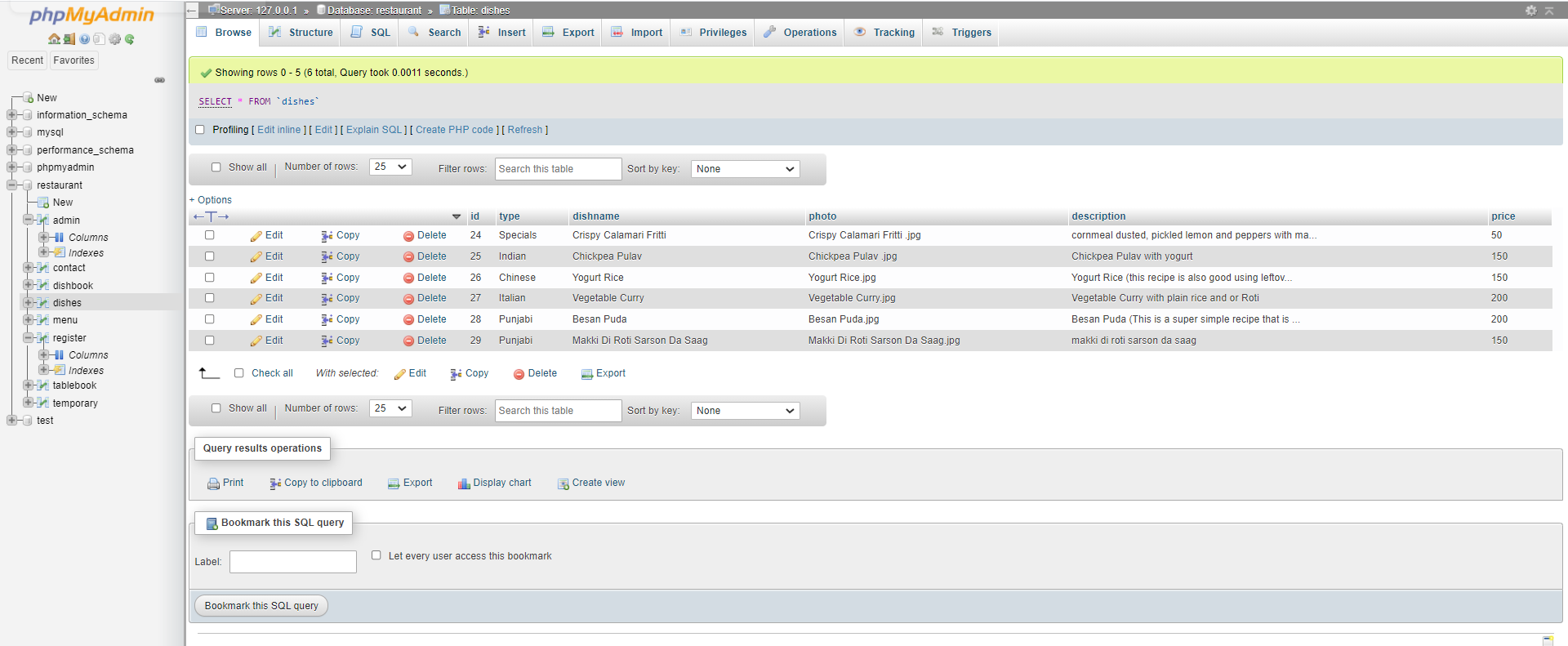
* Menu Type
* Menu id
* Product image



**Dishes Table:**

Here the admin enter all the details about the Dishes. Details enter by the admin will be shown to the user. By view all the details of the Dishes user can easily choose the Dish according to their requirements. Dish tablet able show the following details about the dishes:

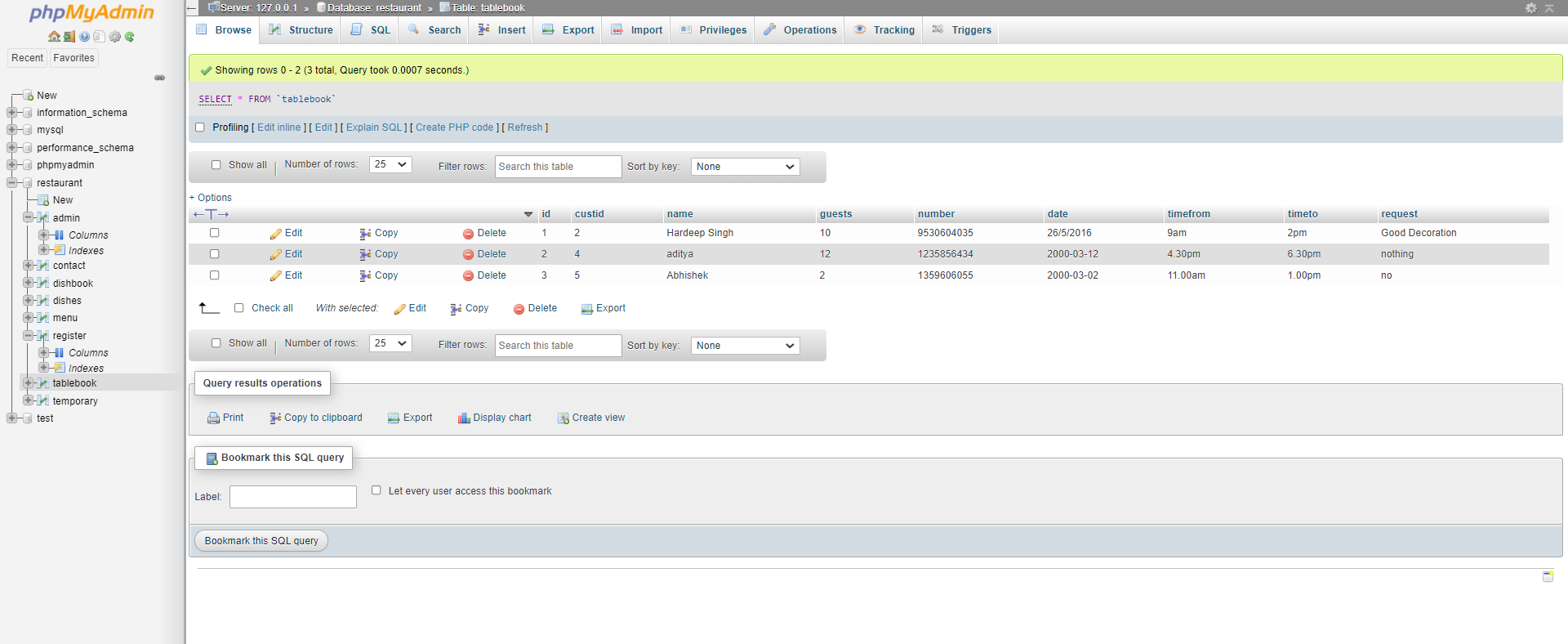
* Dish id
* Dish type
* Dish Name
* Photo
* Description
* Price



**Book Table :**

Book table is used for table booking and to store details of the user. In booking table following details will show in the table:

* Customer id
* Name
* Guests
* Number
* Date
* Time from
* Time to
* Request

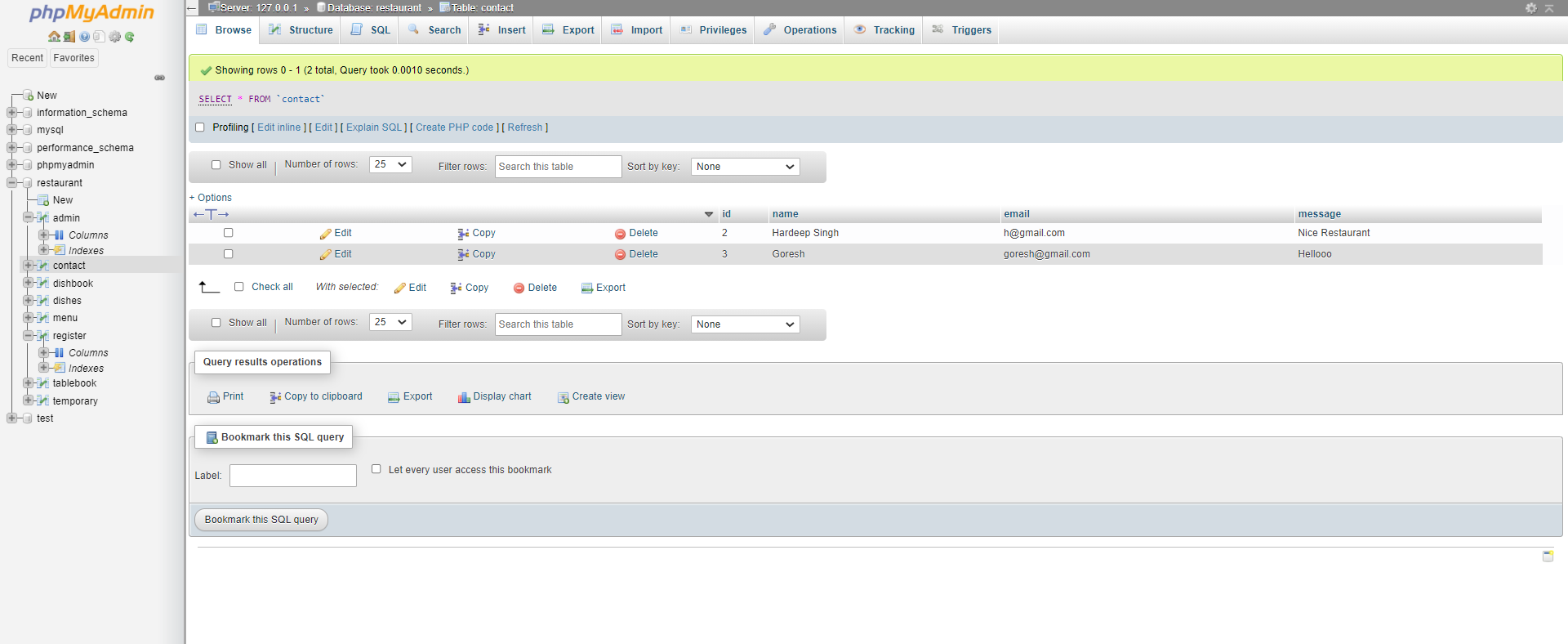
****

**Contact Table**:

In the contact table details of the user who want to contact with us will be stored.

In this table we store the following data:

* User Id
* Name
* E-mail
* Message



1. ***CODING:***

***Master Page code:***

<?php

session\_start();

@$id=$\_SESSION['userid'];

?>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />

<title>River View Restaurant</title>

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

<link rel="stylesheet" type="text/css" href="css/isotope.css" media="screen" />

<link rel="stylesheet" href="js/fancybox/jquery.fancybox.css" type="text/css" media="screen" />

<link href="css/animate.css" rel="stylesheet" media="screen">

<link href="flexslider/flexslider.css" rel="stylesheet" />

<link href="js/owl-carousel/owl.carousel.css" rel="stylesheet">

<link rel="stylesheet" type="text/css" href="css/styles.css"/>

<!-- Font Awesome -->

<link href="font/css/font-awesome.min.css" rel="stylesheet">

</head>

<body>

<header class="header">

<div class="container">

<nav class="navbar navbar-inverse" role="navigation">

<div class="navbar-header">

<button type="button" id="nav-toggle" class="navbar-toggle" data-toggle="collapse" data-target="#main-nav"> <span class="sr-only">Toggle navigation</span> <span class="icon-bar"></span> <span class="icon-bar"></span> <span class="icon-bar"></span> </button>

<a href="index.php" class="navbar-brand scroll-top logo animated bounceInLeft"><b><i>River View Restaurant</i></b></a> </div>

<!--/.navbar-header-->

<div id="main-nav" class="collapse navbar-collapse">

<ul class="nav navbar-nav" id="mainNav">

<li><a href="index.php" class="scroll-link">Home</a></li>

<li><a href="menu.php" class="scroll-link">Menu</a></li>

<li><a href="booktable.php" class="scroll-link">Book Table</a></li>

<li><a href="Gallery.php" class="scroll-link">Gallery</a></li>

<li><a href="contact.php" class="scroll-link">Contact Us</a></li>

<?php

if($id=="")

{?>

<li><a href="Login.php" class="scroll-link">Login</a></li>

<li><a href="Register.php" class="scroll-link">Register</a></li>

<?php

}else

{

?>

<li><a href="updatepass.php" class="scroll-link">Update Password</a></li>

<li><a href="logout.php" class="scroll-link">Logout</a></li>

<?php

}?>

</ul>

</div>

<!--/.navbar-collapse-->

</nav>

<!--/.navbar-->

</div>

<!--/.container-->

</header>

<!--/.header-->

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

***Login Code*:**

<?php

require("Master/header.php");

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

{

require("config.php");

$sql = mysqli\_query($con,"SELECT \* from register where email = '".$\_POST['email']."' and password = '".$\_POST['pass']."'");

if(mysqli\_num\_rows($sql)>0)

{

$row = mysqli\_fetch\_array($sql);

session\_start();

$\_SESSION['userid']=$row['id'];

$\_SESSION['userno']=$row['number'];

header("location:index.php");

}

else

{

echo "<script>alert('Please Register Yourself')</script>";

}

}

?>

<br /><br />

<div class="container">

<div class="col-md-10 col-md-offset-1">

<br /><br />

***Admin login Code:***

<?php

require("config.php");

if(isset($\_POST["submit"]))

{

$query=mysqli\_query($con,"select \* from admin where username='".$\_POST['email']."' and password='".$\_POST['pass']."'");

if(mysqli\_num\_rows($query)>0)

{

$row=mysqli\_fetch\_array($query);

session\_start();

$\_SESSION['id']=$row['id'];

header("location: home.php");

}

else

{

echo mysqli\_error($con);

echo "<script>alert('You enter wrong email/password')</script>";

}

}

?>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />

<title>Admin</title>

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

<link rel="stylesheet" type="text/css" href="css/font-awesome.css"/>

<link rel="stylesheet" type="text/css" href="css/web.css"/>

<script type="text/javascript" src="js/jquery.js"></script>

<script type="text/javascript" src="js/bootstrap.min.js"></script>

<script type="text/javascript" src="js/back-to-top.js"></script>

<script type="text/javascript" src="js/bootstrap-hover-dropdown.min.js"></script>

</head>

<body>

<div class="container">

<div class="col-md-8 col-md-offset-2" style="margin-top:20px;">

<h2>Admin Login</h2><br /><br />

<div class="panel panel-primary">

<div class="panel-heading">

<div class="panel-title">Admin Login</div>

</div>

<div class="panel-body">

<div class="row">

<form class="form-horizontal" method="post" action="adminlogin.php">

<div class="form-group">

<label class="control-label col-md-3" for="email">Email ID</label>

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

<input type="email" class="form-control" id="email" name="email" placeholder="Enter Your Email\_ID" required="required" autofocus="autofocus" />

</div>

</div>

<div class="form-group">

<label class="control-label col-md-3" for="pass">Passwrod</label>

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

<input type="password" class="form-control" id="pass" name="pass" placeholder="Enter Your Password" required="required"/>

</div>

</div>

<div class="form-group">

<div class="col-md-8 col-md-offset-3">

<button type="submit" name="submit" id="submit" class="btn btn-primary">Login</button>

</div>

</div>

</form>

</div>

</div>

</div>

</div>

</div>

</body>

</html>

***Registration Code:***

<h2>Registration Form</h2>

<br /><br />

<div class="panel panel-primary">

<div class="panel-heading">

<div class="panel-title">User Register</div>

</div>

<div class="panel-body">

<div class="row">

<form class="form-horizontal" method="post" action="Register.php">

<div class="form-group">

<label class="control-label col-md-3" for="email">Name</label>

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

<input type="text" class="form-control" id="name" name="name" placeholder="Enter Your Name" required="required" autofocus="autofocus" />

</div>

</div>

<div class="form-group">

<label class="control-label col-md-3" for="email">Email ID</label>

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

<input type="email" class="form-control" id="email" name="email" placeholder="Enter Your Email\_ID" required="required" />

</div>

</div>

<div class="form-group">

<label class="control-label col-md-3" for="email">Number</label>

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

<input type="number" class="form-control" id="number" name="number" placeholder="Enter Your Number" required="required" autofocus="autofocus" />

</div>

</div>

<div class="form-group">

<label class="control-label col-md-3" for="pass">Passwrod</label>

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

<input type="password" class="form-control" id="pass" name="pass" placeholder="Enter Your Password" required="required"/>

</div>

</div>

<div class="form-group">

<label class="control-label col-md-3" for="pass">Co-Passwrod</label>

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

<input type="password" class="form-control" id="repass" name="repass" placeholder="Enter Your Conform Password" required="required"/>

</div>

</div>

<div class="form-group">

<div class="col-md-8 col-md-offset-3">

<button type="submit" name="submit" id="submit" class="btn btn-primary">Register</button>

</div>

</div>

</form>

</div>

</div>

</div>

</div>

</div>

<br /><br />

<?php

require("Master/footer.php");

?>

***Change Password Code:***

<?php

require("config.php");

include("Master/header.php");

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

{

$pass1=" '".$\_POST['newpass']."' ";

$pass2=" '".$\_POST['conewpass']."' ";

if($pass1!=$pass2)

{

echo "<script>alert('password not match')</script>";

}

else

{

$sql=mysqli\_query($con,"select \* from register where id='".$\_SESSION['userid']."' and password='".$\_POST['oldpass']."'");

if(!$sql)

{

die(mysqli\_error($con));

}

else

{

if($row=mysqli\_fetch\_array($sql))

{

$sql=mysqli\_query($con,"update register set password='".$\_POST['newpass']."' where id='".$\_SESSION['userid']."'");

if(!$sql)

{

die(mysqli\_error($con));

}

else

{

echo "<script>alert('password change Successfully')</script>";

}

}

}

}

}

?>

<br><br>

<div class="container">

<div class="col-md-9 col-md-offset-2">

<h1>Change Password</h1>

<br /><br />

<div class="panel panel-primary">

<div class="panel-heading">

<div class="panel-title">Change Password</div>

</div>

<div class="panel-body">

<form class="form-horizontal" action="updatepass.php" method="post">

<div class="form-group">

<label class="control-label col-md-3">Old Password\*</label>

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

<input type="password" class="form-control" name="oldpass" placeholder="Enter Old Password" required="required" />

</div>

</div>

<div class="form-group">

<label class="control-label col-md-3">New Password\*</label>

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

<input type="password" class="form-control" name="newpass" placeholder="Enter New Password" required="required" />

</div>

</div>

<div class="form-group">

<label class="control-label col-md-3">Co-Password\*</label>

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

<input type="password" class="form-control" name="conewpass" placeholder="Retype New Password" required="required" />

</div>

</div>

<div class="form-group">

<div class="col-md-offset-3 col-md-8">

<button type="submit" class="btn btn-primary" name="submit">Change Password</button>

</div>

</div>

</form>

</div>

</div>

</div>

</div>

<br><br><br>

<?php

include("Master/footer.php");

?>

***User Details Code:***

<?php

include("Master/header.php");

require("config.php");

@$idno=$\_GET['idup'];

@$del=$\_GET['idde'];

@$status="Accept";

@$pending="Pending";

if($del>0)

{

$com = mysqli\_query($con,"delete from tablelist where id='$del'");

}

?>

<br /><br />

<div class="col-md-10 col-md-offset-1">

<table class="table table-responsive table-condensed table-bordered">

<thead>

<h1>Booking Table List</h1>

</thead>

<tbody>

<tr>

<td>Sr no</td>

<td>Name</td>

<td>No. of Guests</td>

<td>Contact No.</td>

<td>Date</td>

<td>Time From :</td>

<td>Time To :</td>

<td>Delete</td>

</tr>

<tr>

<?php

@$idi=0;

$com = mysqli\_query($con,"select \* from tablebook");

if(mysqli\_num\_rows($com)>0)

{

while($row = mysqli\_fetch\_array($com))

{

$idi++;

echo ("<tr><td>$idi</td>");

echo ("<td>$row[name]</td>");

echo ("<td>$row[guests]</td>");

echo ("<td>$row[number]</td>");

echo ("<td>$row[date]</td>");

echo ("<td>$row[timefrom]</td>");

echo ("<td>$row[timeto]</td>");

echo ("<td><a href='tablelist.php?idde=$row[id]'>Delete</a></td></tr>");

}

}

?>

</tr>

</tbody>

</table>

</div>

<?php

require('Master/footer.php');

?>

**Contact Us Code:**

<?php

require("Master/header.php");

if(isset($\_POST["submit"]))

{

require("config.php");

$comm = mysqli\_query($con,"insert into contact(name,email,message) values('".$\_POST['name']."','".$\_POST['email']."','".$\_POST['message']."')");

if($comm)

{

echo "<script>alert('Successfully Submit')</script>";

}

else

{

echo mysqli\_error($con);

}

}

?>

<div class="parlex-back">

<div class="container">

<div class="row">

<div class="heading text-center">

<!-- Heading -->

<h2>Contact Us</h2>

<p>There are many variations of passages of Lorem Ipsum available, but the majority have suffered.</p>

</div>

</div>

<div class="row">

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

<!--NOTE: Update your email Id in "contact\_me.php" file in order to receive emails from your contact form-->

<form action="contact.php" method="post">

<div class="control-group">

<div class="controls">

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

placeholder="Full Name" id="name" name="name" required="required" />

<p class="help-block"></p>

</div>

</div>

<div class="control-group">

<div class="controls">

<input type="email" class="form-control" name="email" placeholder="Email"

id="email" required="required" />

</div>

</div>

<br />

<div class="control-group">

<div class="controls">

<textarea rows="10" cols="100" class="form-control" name="message"

placeholder="Message" id="message" required

data-validation-required-message="Please enter your message" minlength="5"

data-validation-minlength-message="Min 5 characters"

maxlength="999" style="resize:none"></textarea>

</div>

</div>

<br />

<div id="success"> </div> <!-- For success/fail messages -->

<button type="submit" name="submit" class="btn btn-primary pull-right">Send</button><br />

</form>

</div>

<aside class="col-md-5">

<h3>Contact Information</h3>

<!-- <p>At lorem Ipsum available, but the majority have suffered alteration in some.</p> -->

<p>

Jyoti Chowk<br>

Jalandhar 144001<br>

India

</p>

<p>

Phone: 0 123 456 789 0 <br />

Email: info@riverview.com

</p><br />

<h3>Template Info</h3>

<!-- <p>Rockline is a free website template by <a href="http://www.webthemez.com/">webthemez.com</a>.

released and licensed under the <a href="http://creativecommons.org/licenses/by/3.0/" rel="license">

Creative Commons Attribution 3.0 License</a>

<a href="http://www.webthemez.com/">my website</a>. -->

</p>

</aside>

</div>

</div>

<!--/.container-->

</div>

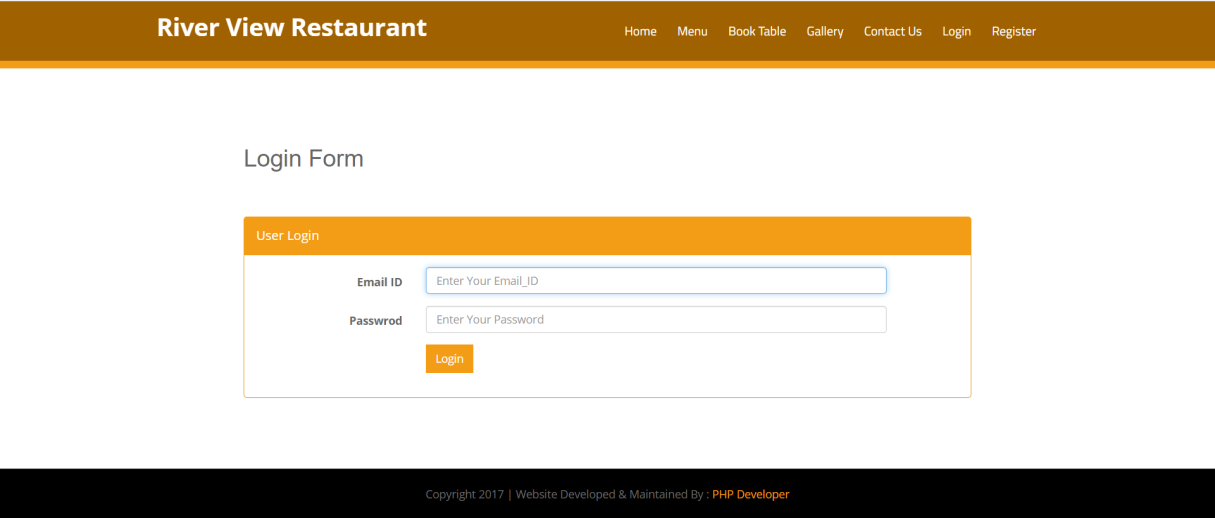
<?php

require("Master/footer.php");

?>

**7. SNAPSHOTS**

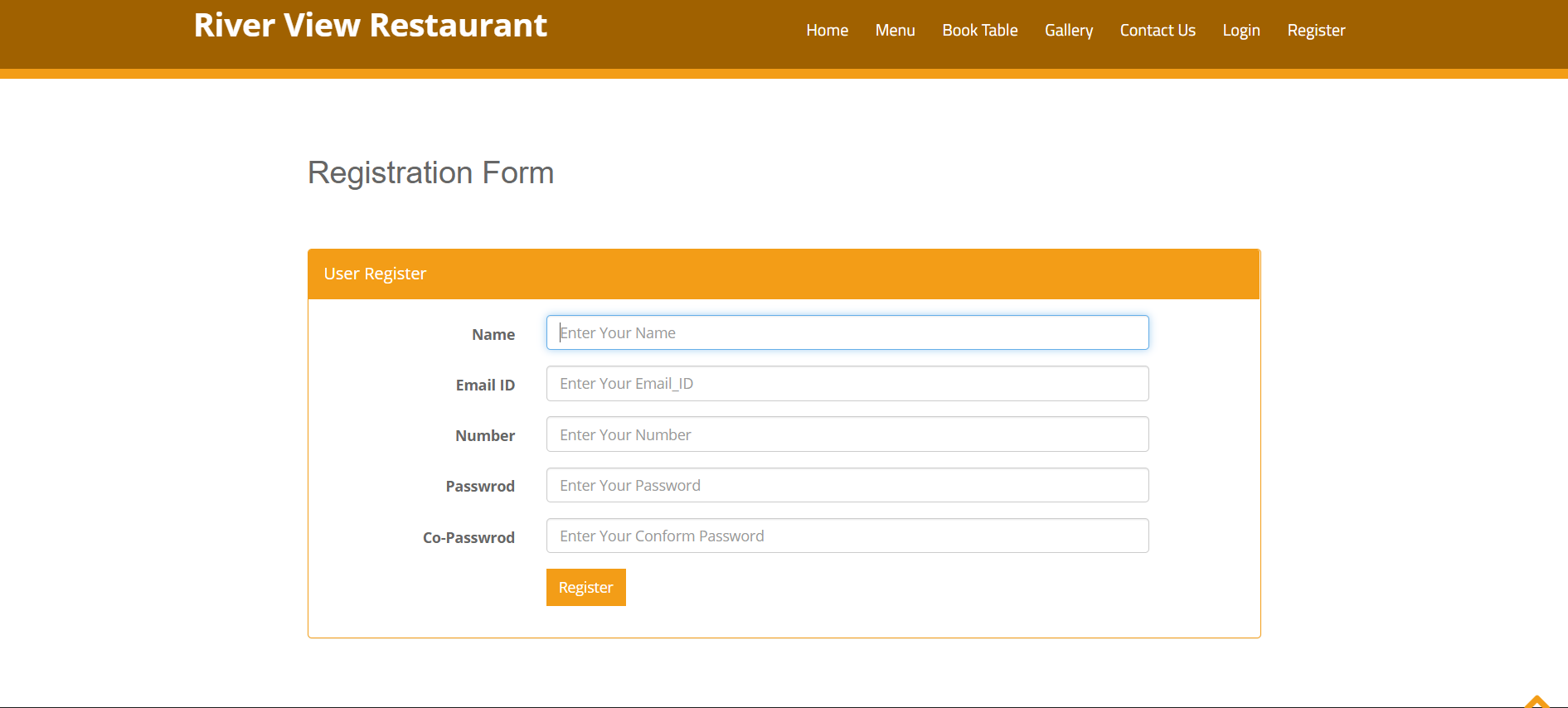
***Login Screen***

****

The above image is the Login Screen of the website. The website made with keep in mind that it is easy and can use as fast as possible.

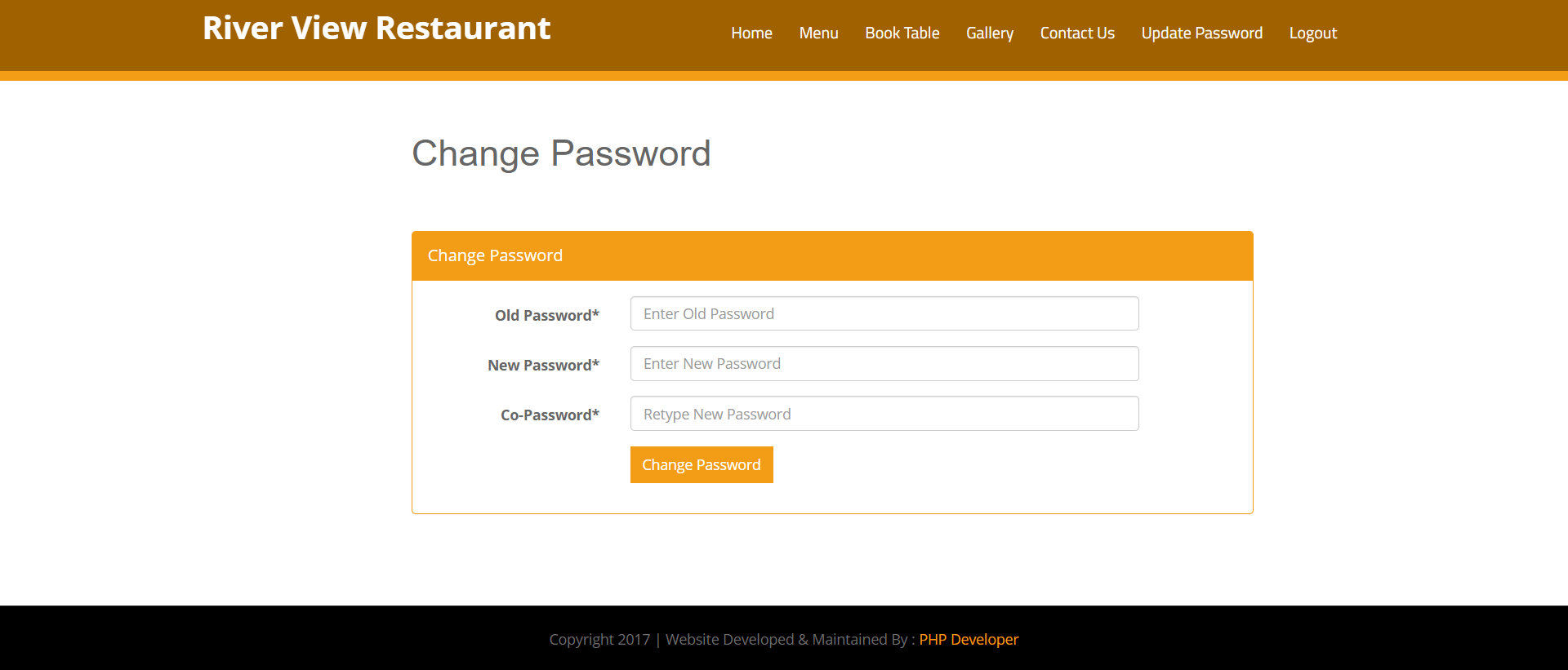
Here User by filling their username and password are able to successfully Login into application.

***Registration form:***



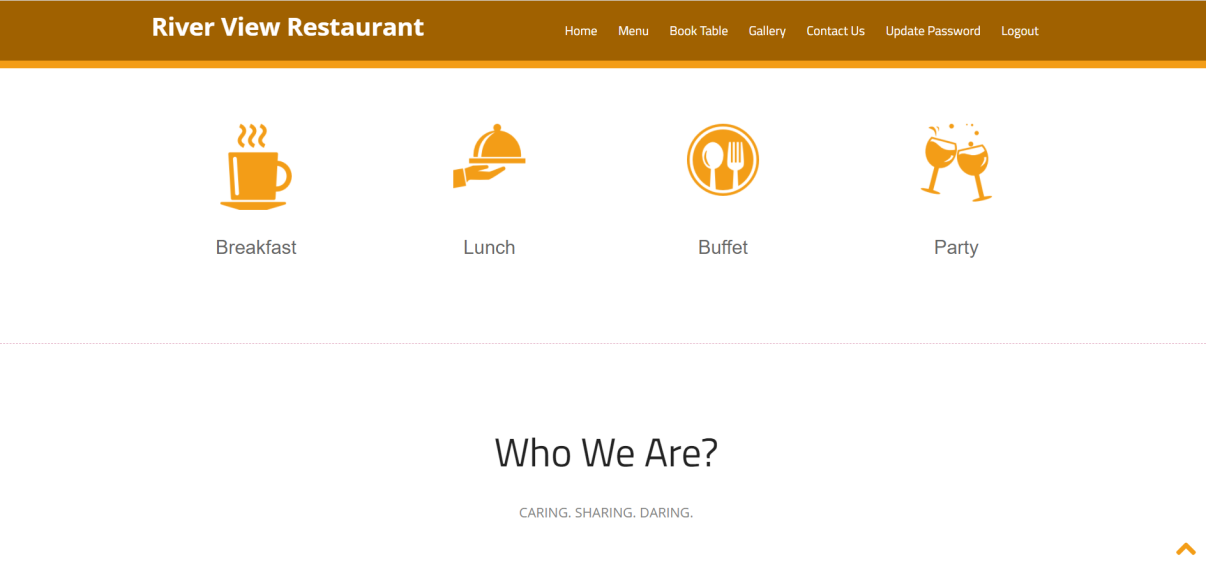
The above image is the register Screen of the application. When we click on Create New Account on Login Screen after that it moves user to New Account screen. On this screen user fill all the required fields that are necessary for sign up. Here user fills Name, Phone, E mail, Password and Co-Password fields for registration. The registration code provided by organization is used to register.

***Change password:***



The above image is the change password Dialog of the application. When user click on change Password on login page then user will redirect in this page. Enter old password and new password and co- password.

***User profile :***

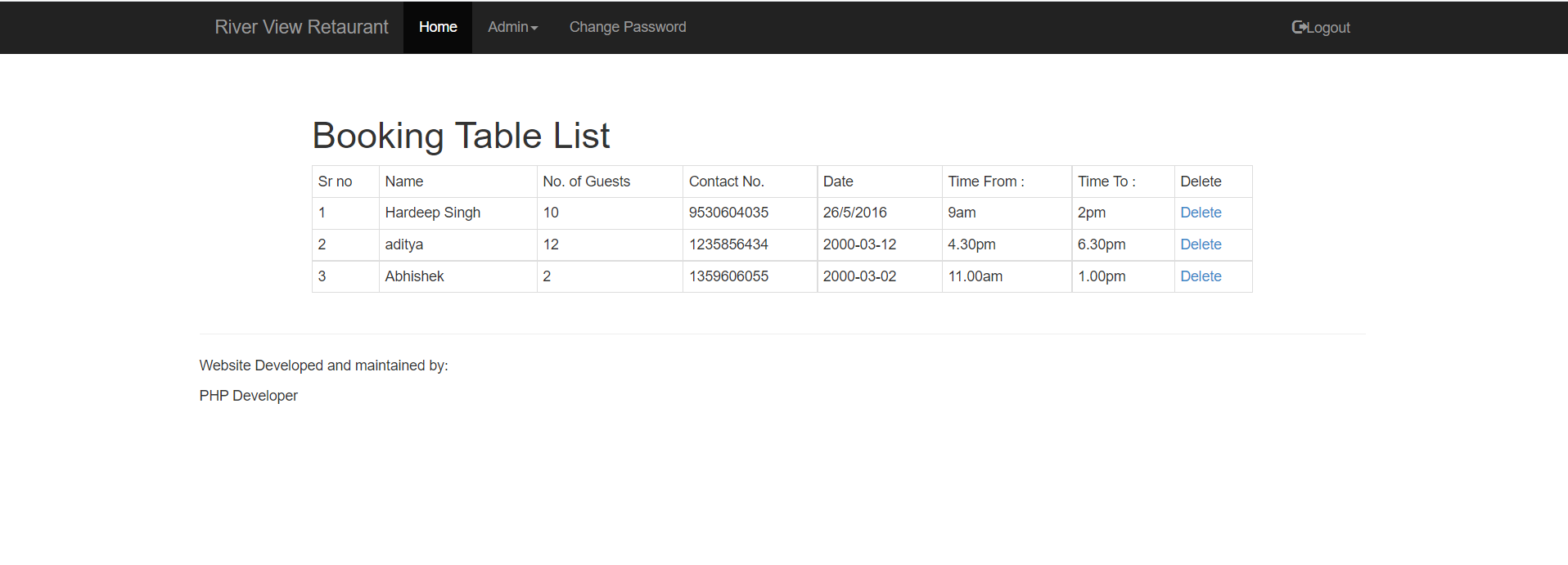
******

The above image is the user profile screen of the website. In this page we have seven links which helps the user. By these seven links user can view home page, menu, book table, gallery, contact us, update password, log out.

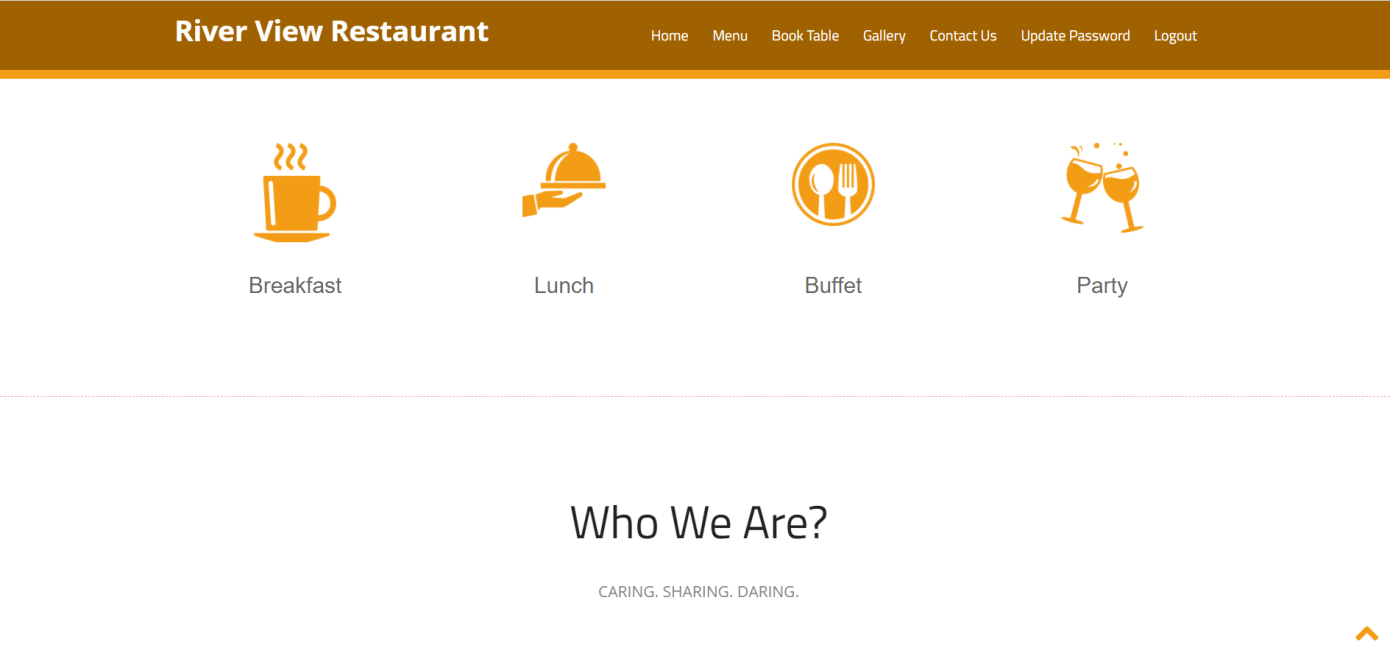
Three links:

* View home page
* Menu
* Book table
* Gallery
* Contact us
* Update password
* Log out

***User Information Screen :***

******

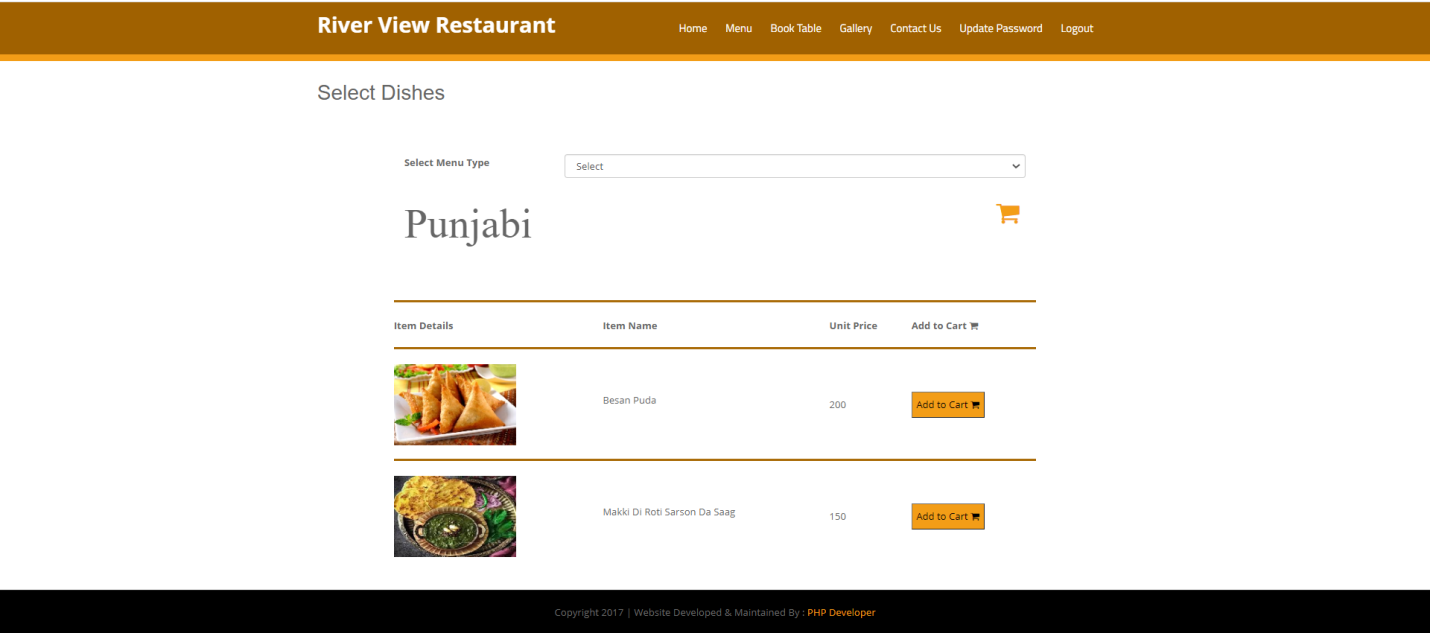
***Home Screen:***

.

The above image is the Home Screen of the website. It contains home, menu, gallery, etc.

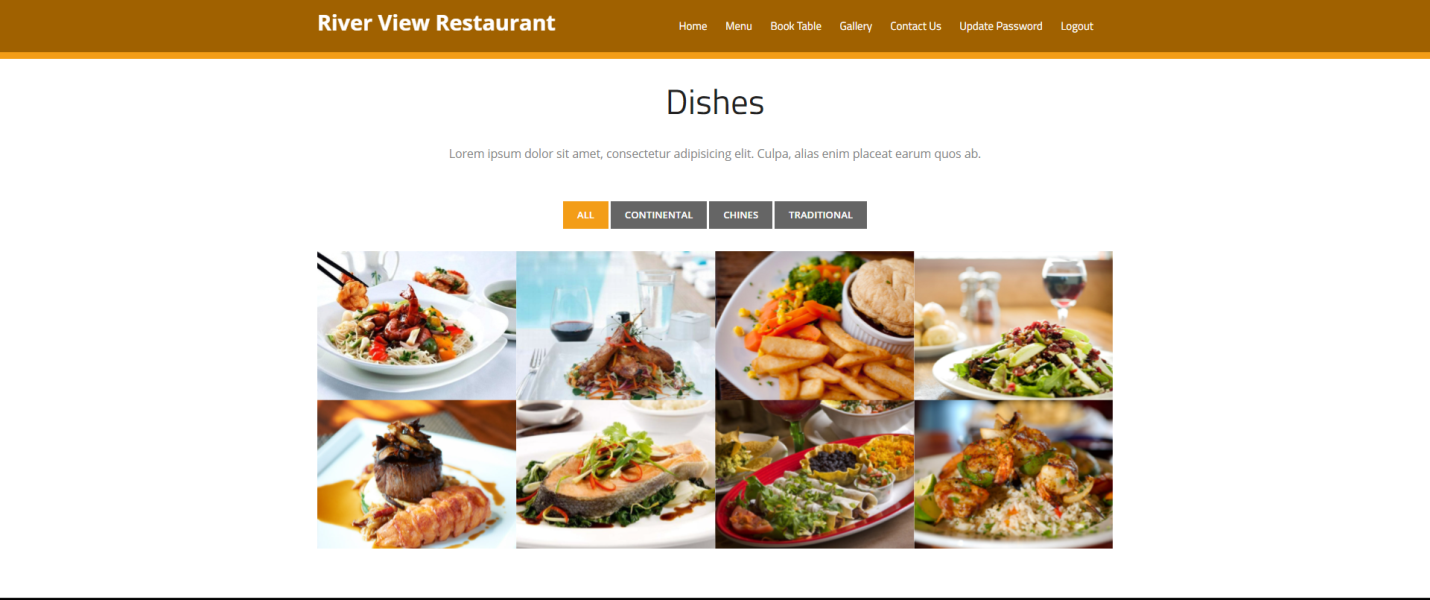
* Home
* Menu
* Gallery

***Menu:***

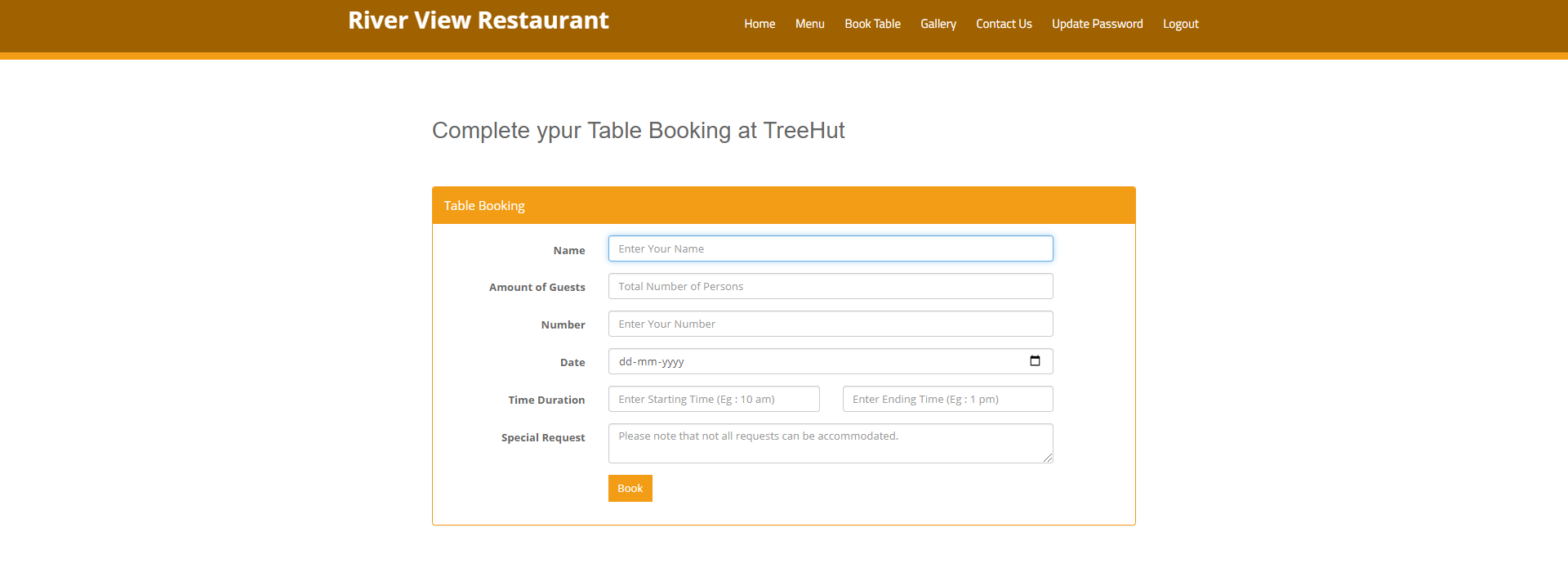
******

In menu page you can select different dishes.

***Gallery:***

****

***Table booking:***

****

**8. TESTING**

**TESTING**

Testing is the major quality control measure employed during software development. Testing is the process of executing a program with the intent of finding an error. No piece of code is completely ready unless it has been fully tested. This stage is very important as at this stage it is verified whether the code developed meet the requirement specifications or not. Moreover, all validations are also checked in the testing stage.

Testing is a process of executing a program with the intent of finding an error. A good test case is the one that has a probability of finding an as yet undiscovered error. If testing is conducted successfully (according to the objective stated) it will uncover error in the software. As secondary benefit, testing demonstrates that software function appears to be working according to the specification that performance requirement appears to have been met.

Testing is the set of activities that can be planned in advance and conducted systematically. It is an integral part of program development. It is in this stage, which we check that the program, that has been coded, Perform according to the requirements. The purpose of doing test is not to demonstrate that there are no errors in the program but to detect any bugs that may still exist.

In the testing stage, the main aim is to look for errors that unknowingly have been occurred. It is common misconception that the purpose of testing is to prove that a program is working correctly. This is dangerous myth because it can lead insufficient testing, and program with hidden fault. Because the actual result and expected result may differ in the field of reality and it can be hazardous for a program.

The importance of software testing and its implications with respect to software quality cannot be over emphasized. Software testing is a crucial element of software quality and represents the ultimate review of specification design and coding.

The increasing visibility of motivating forces for well planned thorough testing. It is not unusual for software development organization to expend 40% of total project effort on testing.

**TEST STRATEGY**

Implemented System is tested using Basic level of Testing that are:

1. UNIT TESTING.
2. INTEGRATION TESTING.
3. SYSTEM TESTING.
4. ACCEPTANCE TESTING.

These different levels of testing attempt to detect different types of faults. The relation of the faults introduced in different phases, and the different levels of testing are shown:

1. **UNIT TESTING**

The first level of testing is unit testing. In this different modules are tested against the specifications produced during design for the modules. Unit testing is essential for verification of the code produced during the coding phase and hence the goal is to test the internal logic of the modules

1. **INTEGRATION TESTING**

The next level of testing is often called integration testing. In this many tested modules are combined into sub-systems, which are then tested the goal here is to see if the modules can be integrated properly, the emphasis being on testing interfaces between modules. This activity can be considered as testing the design, and hence the emphasis on testing module interactions.

1. **SYSTEM TESTING**

The next level of testing is system testing. Here the entire software system is tested. The reference document for this process is requirement document, and the goal is to see if the software meets its requirements. This is essentially a validation exercise. And it was found that they all are working well to meet the Owners requirements.

1. **ACCEPTANCE TESTING**

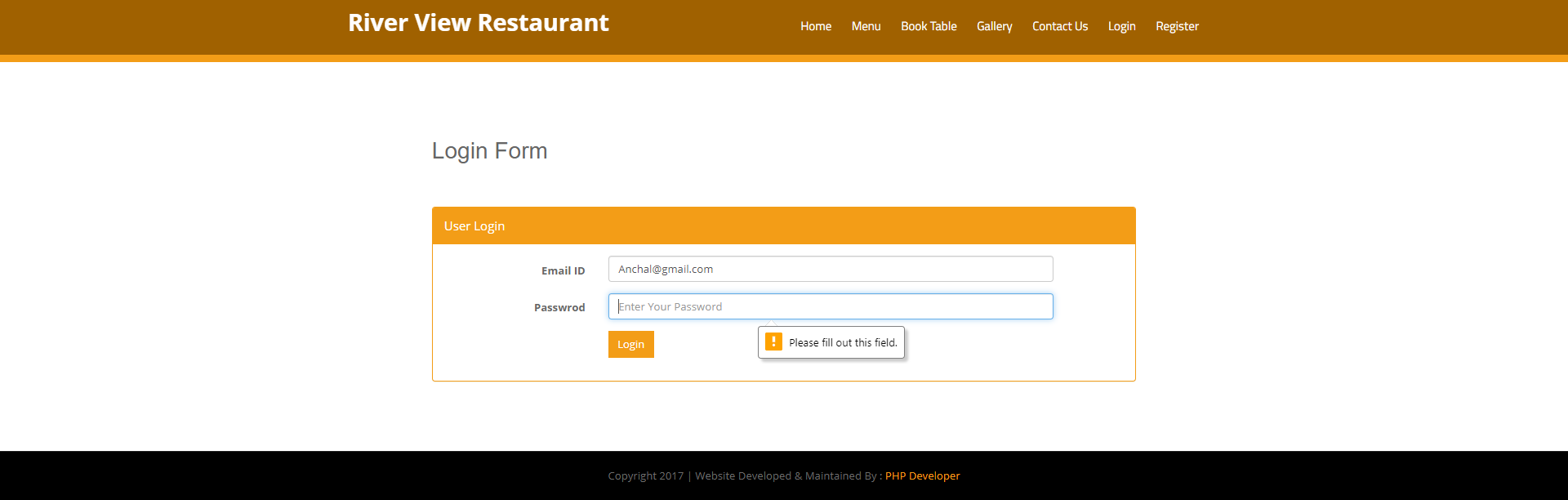
The last level of testing is acceptance testing. Acceptance testing is performed with realistic data of the client to demonstrate that the software is working satisfactorily. Testing here focuses on the external behavior of the system; the internal logic of the program is not emphasized.

**TEST CASES**

For testing to be successful, proper selection of test cases is essential. There are two different approaches to selecting cases - functional testing and structural testing.

* In **functional testing** the software or the module to be tested is treated as a black box, and the test cases are decided based on the specifications of the system or the module. For this reason this type of testing is also called "black box testing" the focus here is on testing the external behavior of the system.
* In **structural testing** the test cases are decided based on the logic of the module to be tested. A common approach here is to achieve some type of coverage of the statements in the code. One common coverage criterion is statement coverage, which requires that test cases be selected so that together they execute each statement exactly once.

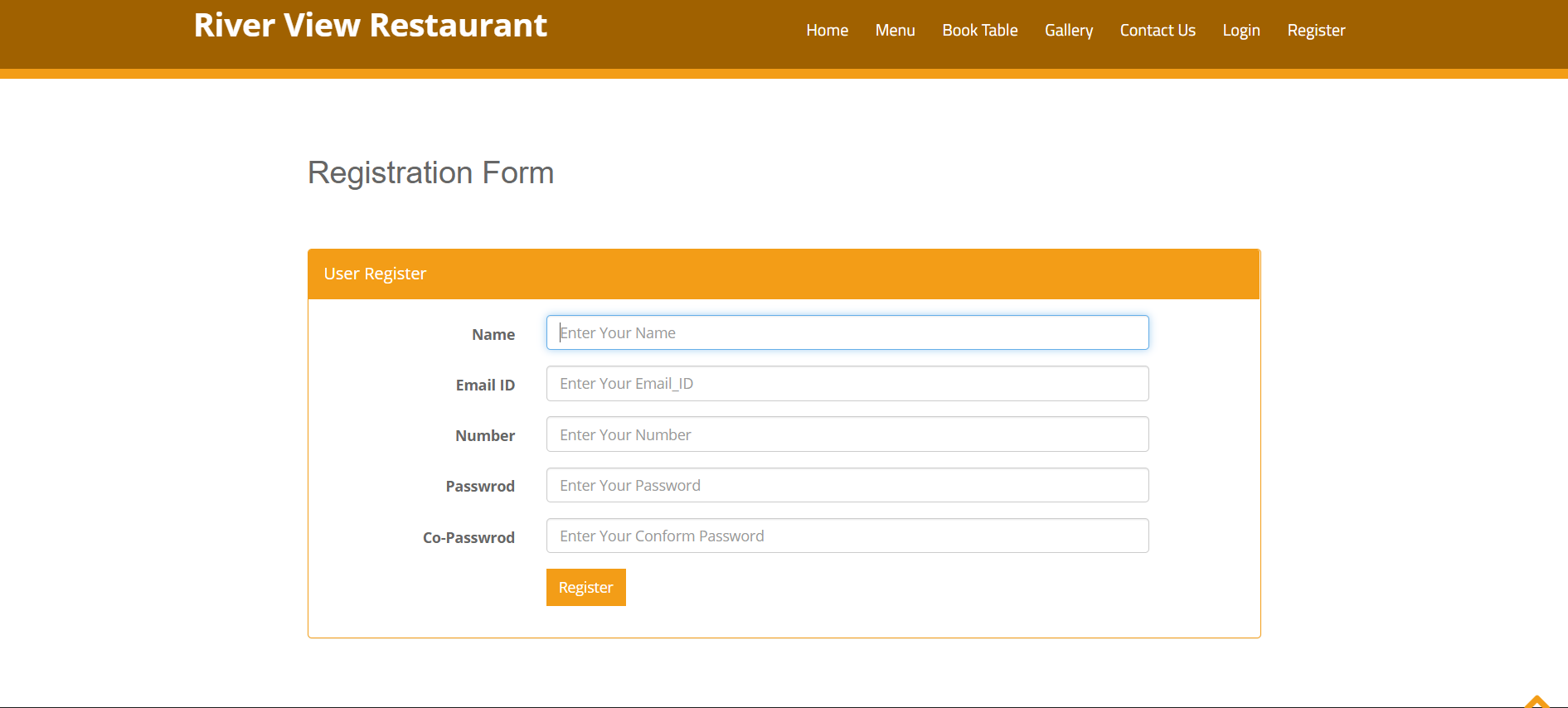
**Test Case 1**

****

|  |  |
| --- | --- |
| Test Case Identification | Login Screen |
| Expected Results | It should display the message invalid login parameters. |
| Actual Results | It displays the error message invalid login parameters. |
| Remarks | Pass |

When User accidently enters a wrong username and password combination, then error message will display invalid username or password.

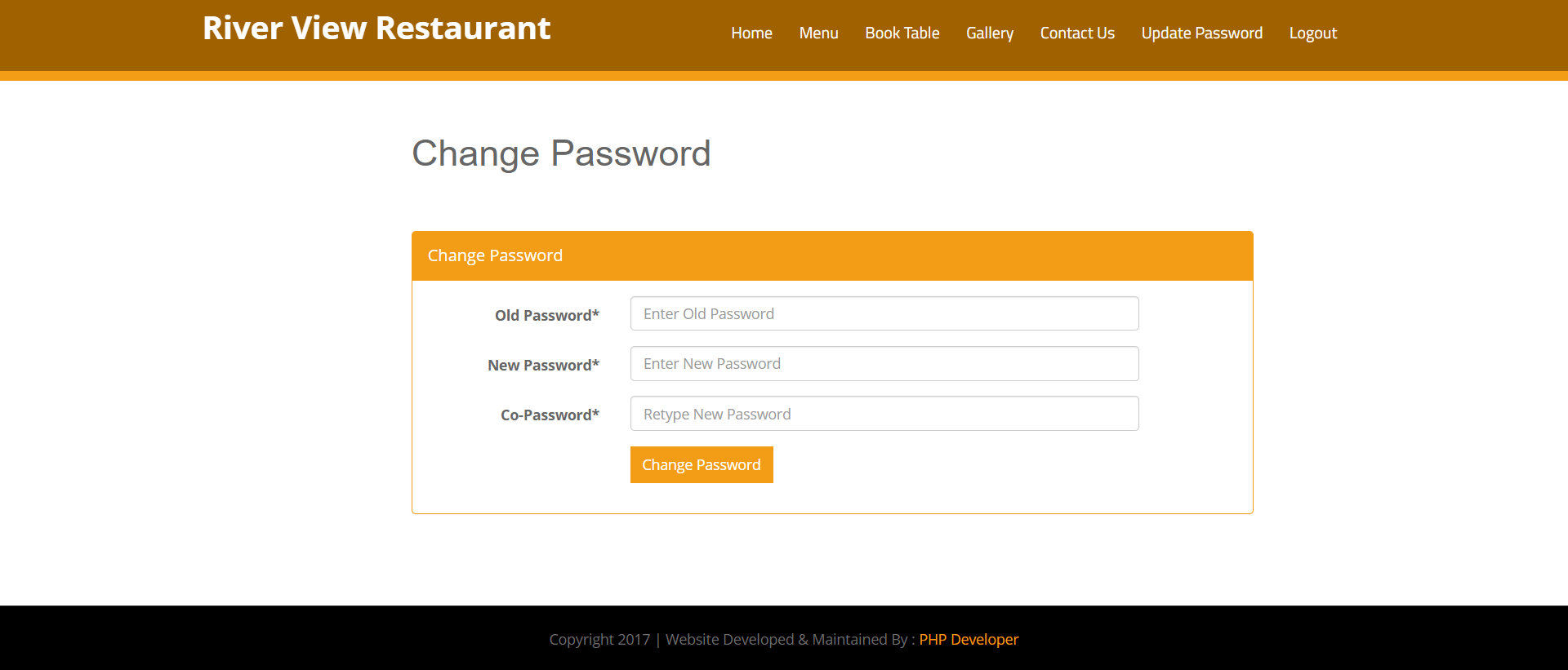
**Test Case 2**



|  |  |
| --- | --- |
| Test Case Identification | New Account Screen |
| Expected Results | It should display the message for the fields which is required to fill. |
| Actual Results | It displays the error message Please enter your name, Please enter your Phone Number etc. |
| Remarks | Pass |

When any User accidently submits the data without filling full details, then error message will display.

**Test Case 3**



|  |  |
| --- | --- |
| Test Case Identification | Change Password Screen |
| Expected Results | It should display the message Password length must be between 6 to 10 characters. |
| Actual Results | It displays the error message Password length must be between 6 to 10 characters. |
| Remarks | Pass |

When User not enters password or enter password less than 6 characters, then error message will display Password length must be between 6 to 10 characters.

***9. IMPLEMENTATION:***

System implementation generally benefits from high levels of user involvement and management support. User participation in the design and operation of information systems has several positive results. First, if users are heavily involved in systems design, they move opportunities to mould the system according to their priorities and business requirements, and more opportunities to control the outcome. Second, they are more likely to react positively to the change process. Incorporating user knowledge and expertise leads to better solutions.

The relationship between users and information systems specialists has traditionally been a problem area for information systems implementation efforts. Users and information systems specialists tend to have different backgrounds, interests, and priorities. This is referred to as the **user-designer communications gap**. These differences lead to divergent organizational loyalties, approaches to problem solving, and vocabularies. Examples of these differences or concerns are below:

**User Concerns**

* Will the system deliver the information I need for my work?
* How quickly can I access the data?
* How easily can I retrieve the data?
* How much clerical support will I need to enter data into the system?
* How will the operation of the system fit into my daily business schedule?

**Designer Concerns**

* How much disk storage space will the master file consume?
* How many lines of program code will it take to perform this function?
* How can we cut down on CPU time when we run the system?
* What are the most efficient ways of storing this data?
* What database management system should we use

***10. Documentation:***

**Project Name** – **River View Restaurant**

**Frontend** – Html, Css, JavaScript

**Procedure to start project**

1. Firstly to debug our project River View Restaurant

2. Click login page to run our project

3 Then user panel will open.

4. User will choose the six options as his/her requirements.

* 1. Menu
  2. Book table
  3. Gallery
  4. Contact us
  5. Login
  6. Register

**11.MAINTENANCE**

**Introduction to Software Maintenance**

Software maintenance denotes any changes made to a softwrae product after it has been delivered to the customer. Maintenance is inevitable for almost any kind of product. It is practically impossible to make the software Completely error free because the input domain of most software products is very large and it is not practical to test the software exhaustively with respect to each value that the input data may assume. Maintenance is also needed to enhance the features of the software to add more functionality to it and to port to new platforms etc.

**Types of Software Maintenance**

Maintenance is fixing or enhancing an system. Many different types of maintenance must be performed on the system to ensure it continues to operate as expected. These include:

* **Adaptive maintenance** - making changes to increase system functionality to meet new requirements.
* **Corrective maintenance** - making changes to repair system defects and bugs observed while the system is in use.
* **Perfective maintenance** - making changes to enhance the system and improve such things as processing performance and usability.
* **Preventive maintenance** - making changes to reduce the chance of future system failures.

**12. PROJECT LEGACY**

**CURRENT STATUS OF THE PROJECT**

The system is now completed and meets the system requirement specifications. So that the system is implemented and is now ready for installation. The current system is performing all functions which were provided in the project.

The Project is currently complete and satisfying all the requirements decided in requirement Analysis. The various functions are:-

1. It allows users to easily create new account and login using username and password.
2. It allows users to search according to their need.
3. It allows Users to view cart, view profile with updation button.
4. It allows Users to buy different kinds of supplements according to your needs.
5. It allows Users to access nutrition meals for gain muscle and lose fat.
6. It allows Users to view different kinds of exercise with different categories of men , women and yoga .

**CONCLUSION AND OBJECTIVES ACHIEVED**

In conclusion the main aim was to develop a website, with the help of which the Users are able to access any type of Fitness Exercises, Nutrition Diet and Buy Supplements. This website is designed as per the Requirements and objectives set in Analysis Phase. This project is beneficial for any type of organization and society. This project gives the best way to access Fitness Tips. This will provide accuracy and easiness to the end user. These Website overcome the many problem of the users, user can easily buy any supplements by sitting at home and access different kinds of exercises. This gives the best experience and facilities to the users. So other maintenance will improve it more and provide more accuracy, efficiency, speed and easiness to the end user.

**13. BIBLIOGRAPHY**

**Websites:**

* [www.google.com](http://www.google.com)
* [www.javatpoint.com](http://www.javatpoint.com)
* [www.intechnic.com](http://www.intechnic.com)