# CHAPTER 1

**INTRODUCTION**

The project titled as “Online Shopping Portal” is developed in PHP as front end and SQL as back end. The main aim of the project is to make a complete solution for Online Shopping. This project has Admin login module, User Signup and User login module.

### Module

* Admin Login
* User Sign up
* User Login
* Home Page

**Admin Login**

The admin can login into the Online Shopping Portal by giving the credentials. It contains following sub modules:

* **Order Management**- In this admin can manage Today’s order, pending orders and Delivered orders.
* **Manage Users**-The admin can view registered user’s details.
* **Create Category and Sub Category**- The admin can add new category and sub category and category details.
* **Insert Product**- The admin can add new product and its details into the selected category or sub category.
* **Manage Product**- The admin can add a new product or remove the product by selecting the existing category.
* **Logout**- The admin can logout from the admin page.
* **Change Password**- The admin can change his/her password.

**User Signup–**

A new user can signup into the system by giving their full name, mobile number, email id and set a new password to register. After the registration, the user can login to the portal using their credentials.

**User Login**

**Home**

The registered user can login into the Online Shopping Portal by giving the credentials.

In this we have list of categories available and user can search the products and add to the cart and add to the Wish list.

* **Cart-** The customers can select, reserve, and purchase product in this form.
* **Wish list-** In this customer can save the collection of desired products to their user account.
* **Search-** User can search the available products.
* **Track Order**- Customers can track their order by providing order id and registered email.

### 1.2 System Analysis

**1.2.1 Existing System**

The existing system is not user friendly. The records are made in paper and the book can be managed by huge number of peoples. The records have to be stored in a very large spacious room. The records cannot be making many copies.

Limitations of Existing system

* + - * Time Consuming
      * Human error is liable
      * Difficulty faced at the time of organizing the whole data
      * The records have to be recorded in note book
      * Note books can be altered by any persons at any time
    1. **Proposed System**

The proposed system is going to replace the existing system toll with its disadvantages removed along with the improved performance. The proposed system is designed such that it will determine proper solution for the problems in the existing system. The proposed system is GUI(Graphical User Interface) oriented. It contains many database activities like adding user data, deleting user data, viewing user data and reporting user data.

**Features of Proposed System**

* + - * All user details are recorded in the database easily. The records can be managed by the administrator.
      * The proposed systems don’t need huge human power to manage the system.
      * There ports can be easily searched using the proposed system.

# CHAPTER 2

**LITERATURE SURVEY**

#### PHP

PHP started 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 in1994.

* + - PHP is a recursive acronym for "PHP: Hypertext Pre-processor".
    - PHP is a server-side scripting language that is embedded in HTML. It is used to 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.
    - PHP is pleasingly zippy in its execution, especially when compiled as an Apache module on the Unix side. The MySQL server, once started, executes even very complex queries with huge result sets in record-setting time.
    - PHP supports a large number of major protocols such as POP3, IMAP, and LDAP. PHP4 added support for Java and distributed object architectures (COM and CORBA), making n-tier development a possibility for the first time.
    - PHP is forgiving: PHP language tries to be as forgiving as possible.
    - PHP Syntax is C-Like.

##### Common Uses of PHP

* + - PHP performs system functions, i.e., from files on a system it can create, open, read, write, and close them.
    - PHP can handle forms, i.e., gather data from files, save data to a file, through email you can send data, return data to the user.
    - You add, delete, modify elements within your database through PHP.
    - Access cookies variables and set cookies.
    - Using PHP, you can restrict users to access some pages of your website.

#### 2.2 HTML

* + - HTML stands for Hypertext Markup Language, and it is the most widely used language to write Web Pages.
    - Hypertext refers to the way in which Web pages (HTML documents) are linked together. Thus, the link available on a webpage is called Hypertext.
    - As its name suggests, HTML is a Markup Language which means you use HTML to simply "mark-up" a text document with tags that tell a Web browser how to structure it to display.
    - Originally, HTML was developed with the intent of defining the structure of documents like headings, paragraphs, lists, and so forth to facilitate the sharing of scientific information between researchers.
    - Now, HTML is being widely used to format web pages with the help of different tags available in HTML language.

##### Features in HTML5

* + - The DOCTYPE declaration for HTMl5 is very simple, <!DOCTYPE html>
    - The character encoding <metacharset=”UTF-8”>
    - New function for embedding audio(<audio>), video(<video>), graphics(<svg> and<canvas>)
    - Client-side data storage
    - Interactive documents
    - New structural elements <article>, <header>, <footer>, <nav>, <section> and <figure>
    - New from control calendar, date, time, email, URLs, search
    - Java script enhancement
    - HTML Geo location, HTML Drag and Drop, html local storage, HTML Application Cash, HTML Web Workers, HTML SSE.

**Applications of HTML**

HTML has been used widely and effectively. Below is the list of Uses of HTML language.

* **Web pages development:** HTML is heavily used for creating pages that are displayed on the world wide web. Every page contains a set of HTML tags including hyperlinks which are used for connecting to other pages. Every page that we witness, on the world wide web, is written using a version of HTML code.
* **Web document creation:** Document creation on the internet is dominated by HTML and its basic concept via tag and DOM i.e., document object model. HTML tags are inserted before and afterward or phrases to locate their format and location on the page. A web document consists of three sections: title, head, and body. Head includes the information to identify the document, including title and any other important keyword. A title can be seen on the browser’s bar and body section is the main portion of the website visible to the viewer. All the three segments are designed and created by the uses of HTML tags. Every section has their own specific set of tags, which are dedicatedly rendered keeping the head, title and body concepts in a loop.
* **Internet navigation:** This is one of the most important uses of HTML which is revolutionary. This navigation is possible by utilizing the concept of Hypertext. It is basically a text which refers to other web pages or text and when user click on it, would navigate to referenced text or page. HTML is heavily used to embed the hyperlink within the web pages. A user can easily navigate within the web pages and between websites as well, which are located on different servers.
* **Responsive images on web pages:** At the elementary level in applications of HTML, queries can be set to utilize the images, which are responsive in nature. With the src set attribute of img element in HTML, and combining it with picture element, a developer can fully control how the user will render an image. Now different types of an image with size variation can be loaded by using the img element. Rules can be easily set with the picture element, we can declaring element with default source and then for every case, a source can be provided.
* **Data Entry support with HTML:** HTML5 standard and set of APIs can be used to support data entry level of work. As browsers implement new HTML5 standards, developers can simply add the attributes to the tag which indicate required fields, text, data format etc. HTML5 has come up with several new attributes to drive on-screen keyboards, validation, and other data-entry experiences so that end-user can have a better data-entry.
* **Game development usage:** Before the advent of HTML5, game development was an exclusive domain of Flash and Silver light. Since browsers support new specifications for HTML5 including CSS3 and light-fast JavaScript engine to drive a new rich experience, HTML5 can bring the reality of game development possible.

#### 2.3 CSS

Style Sheets, fondly referred to as CSS, is a simple design language intended to simplify the process of making web pages presentable. CSS handles the look and feel part of a web page. Using CSS, you can control the color of the text, the style of fonts, the spacing between paragraphs, how columns are sized and laid out, what background images or colors are used, layout designs, variations in display for different devices and screen sizes as well as a variety of other effects. CSS is easy to learn and understand but it provides powerful control over the presentation of an HTML document. Most commonly, CSS is combined with the mark-up languages HTML or XHTML.

The cascade part of CSS means that more than one style sheet can be attached to a document, and all of them can influence the presentation. For example, a designer can have a global style sheet for the whole site, but a local one for say, controlling the link color and background of a specific page. Or, a user can use her own style sheet if she has problems seeing the page, or if she just prefers a certain look.

##### Advantages of CSS

* **Increases Download Speed:** By taking the appearance mark-up out of the html file you decrease the html file size and significantly decrease the download time of that file.
* **Makes Pages More Accessible:** Your pages become more accessible. By separating the styling (CSS) from the content and structure (HTML), you are well on your way to satisfying Accessibility requirements. This is an important thing to consider, if you are creating sites that might be used by the visually impaired.
* **Multiple Style Sheets Cascade into One:** Style Sheets allow style information to be specified in many ways. Styles can be specified inside a single HTML element, inside the element of an HTML page, or in an external CSS file. Even multiple external Style Sheets can be referenced inside a single HTML document.
* **Reduces Time:** It is much easier to update pages. It is much faster to update a page that uses styles over using tags and the like. With CSS, you can decide how headings should appear, and enter that information once. Every heading in every page that is linked to this style sheet now has that appearance. Want to make every heading of level 3 obviously different from those of level 2? Edit the style sheet, and every such heading now has the altered appearance. Consider how much time you will save. With cascading style sheets, whole organizations can share small number of style sheets, ensuring consistency across the site with no need for constant updating and editing to accommodate changes. How do Style Sheets Work? Style sheets are just text files, or text embedded in the head of an HTML document, that help separate content from appearance. The content of a page goes into an HTML file and the appearance goes into a style sheet. But how does this entire end up as a web page in your browser? Think of a style sheet as a set of instructions, suggesting to a web browser how to draw a page. The style sheet suggests how the browser should display the page based on rules you define in the style sheet.

#### 2.4 MYSQL

* MySQL is an Oracle-backed open source Relational Database Management System (RDBMS) based on Structured Query Language (SQL). MySQL runs on virtually all platforms, including Linux, Unix and Windows. Although it can be used in a wide range of applications, MySQL is most often associated with web applications and online publishing.

MySQL is an important component of an open source enterprise stack called LAMP. LAMP is a web development platform that uses Linux as the operating system, Apache as the web server, MySQL as the relational database management system and PHP as the object- oriented scripting language. (Sometimes Perl or Python is used instead of PHP.)

Originally conceived by the Swedish company MySQL AB, MySQL was acquired by Sun Microsystems in 2008 and then by Oracle when it bought Sun in 2010. Developers can use MySQL under the GNU General Public License (GPL), but enterprises must obtain a commercial license from Oracle. Today, MySQL is the RDBMS behind many of the top websites in the world and countless corporate and consumer-facing web-based applications, including Face book, Twitter and YouTube.

##### How MySQL works?

MySQL is based on a client-server model. The core of MySQL is MySQL server,which handles all of the database instructions (or commands). MySQL server is available as a separate program for use in a client-server networked environment and as a library that can be embedded (or linked) into separate applications. MySQL operates along with several utility programs which support the administration of MySQL databases.

# CHAPTER 3

**SYSTEM ANALYSIS**

#### EXISTING SYSTEM

In an existing system there are users who have their own mechanic database which is very minimal. And also, they have no idea if their vehicles are breaking down or had any mechanical issue in remote locations or any long distant locations from their known mechanic shops. Users with the contacts of people at the particular place may look for a help from them only if they are ready to do. It is not possible to find out the suitable mechanic for the desired service at remote locations. The only way they have is to look for any other transportation at the time of issue and then they need to get a mechanic to the particular location at which they have left their vehicle.

#### 3.1.1 DISADVANTAGES:

* + - 1. Time consuming.
      2. Difficult to find suitable mechanic.

#### 3.2 PROPOSED SYSTEM/SOLUTION

Here the users of On Road Vehicle Breakdown Assistance system can search for list of mechanic at any location or the nearby locations which will help them in an unexpected situation raised by the mechanical issues of their vehicles. Only the licensed mechanics can get listed here while the search. And there are available mechanic who can come and repair the mechanical issues in the user’s vehicle.

#### 3.2.1ADVANTAGES:

* + - 1. Secure registration of user’s and mechanics.
      2. Easy access to the data.
      3. The new system is more user-friendly, reliable and flexible.
      4. Reduced manual work.

# CHAPTER 4

**SYSTEM REQUIREMENTS**

### Hardware Requirements

Machine Name : DELL

Processor : Intel Core i6-3337U CPU

RAM : 4GB

System Type : 64 bit–Operating System

### 4.2 Software Requirements

Front End : Jet Brains Php Storm

Back End : SQL

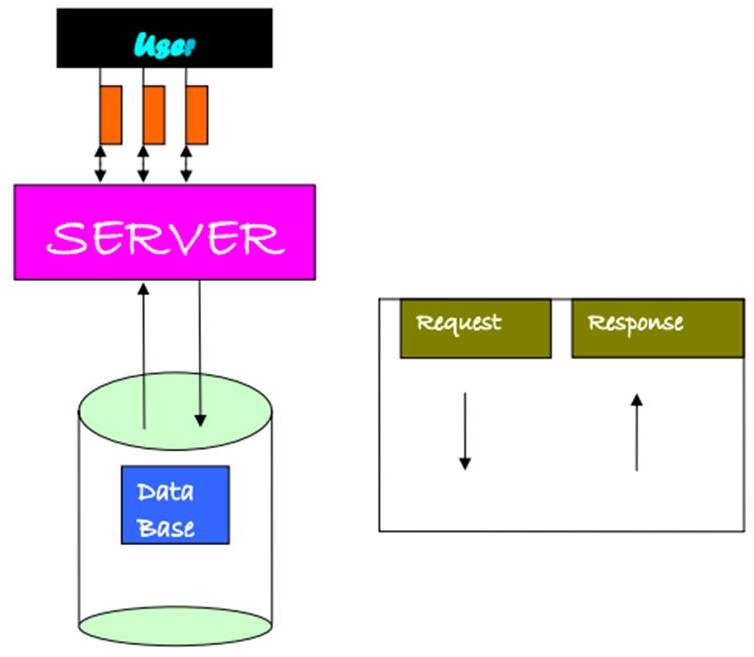
Operating System : Windows10

# CHAPTER 5

**SYSTEM DESIGN**

#### SYSTEM ARCHITECTURE

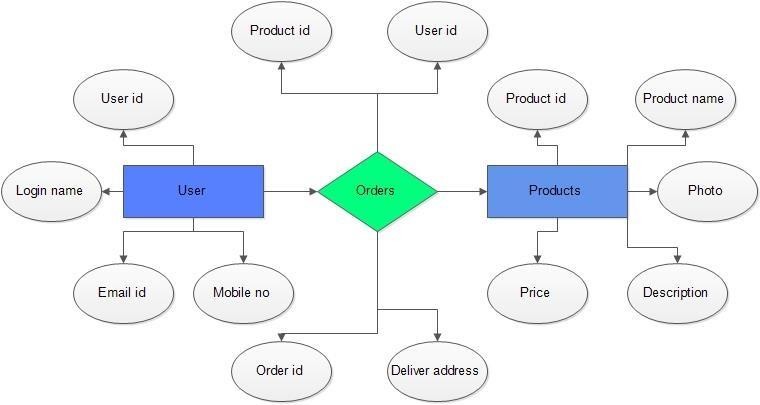
Below architecture diagram represents mainly flow of requests from users to database through servers. In this scenario overall system is designed in three tires separately using three layers called presentation layer, business logic layer and data link layer. This project was developed using 3-tier architecture.



**FIGURE 5.1** SYSTEM ARCHITECTURE

#### 5.2 ER DIAGRAM

An Entity-Relationship model(ER model) describes interrelated things of interest in a specific domain of knowledge . A basic ER model is composed of entity types (which classify the things of interest) and specifies relationships that can exits between instances of those entity types



**ER diagram of Online Shopping Portal**

**5.3 Schema Diagram**

Database schema is described as database connections and constraints. It contains attributes. Every database has a state instances represent current set of databases with values. There are different types of keys in a database schema.

|  |
| --- |
| **CATEGORY** |
| Id |
| Category name |
| Category  description |
| Creation Date |
| Updation Date |

|  |
| --- |
| **SUBCATEGORY** |
| Id |
| Category Id |
| Subcategory |
| Creation Date |
| Updation Date |

|  |
| --- |
| **USER** |
| Id |
| Name |
| Email |
| Contact No |
| Password |
| Shipping Address |
| Shipping City |
| Shipping Pincode |
| Billing Address |
| Billing City |
| Billing Pincode |
| Reg. Date |
| Updation Date |

|  |
| --- |
| **USER LOG** |
| Id |
| User Email |
| Userip |
| Login Time |
| Logout Time |
| Status |

|  |
| --- |
| **ORDERS** |
| Id |
| User id |
| Product id |
| Quantity |
| Order Date |
| Payment Method |
| Order Status |

**Schema diagram of Online Shopping Portal**



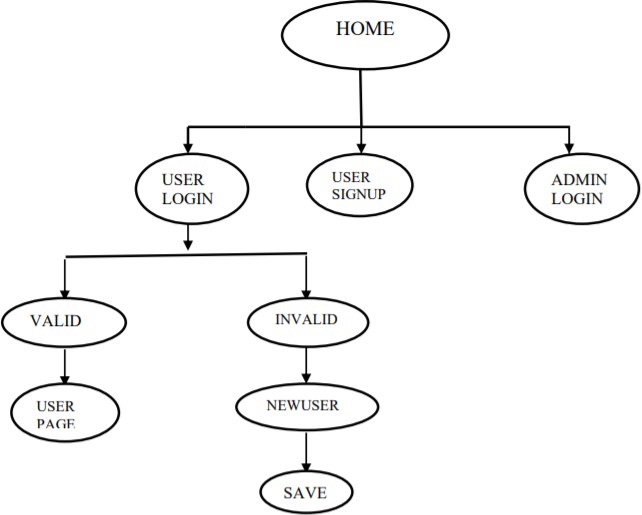
|  |
| --- |
| **WISHLIST** |
| Id |
| User id |
| Product id |
| Posting Date |

|  |
| --- |
| **PRODUCT** |
| Id |
|  |
| Subcategory |
| Product No. |
| Product Company |
| Product Price |
| Product Price before Discount |
| Product Description |
| Product Image 1 |
| Product Image 2 |
| Product Image 3 |
| Shipping Charge |
| Product Availability |
| Posting Date |
| Updation Date |

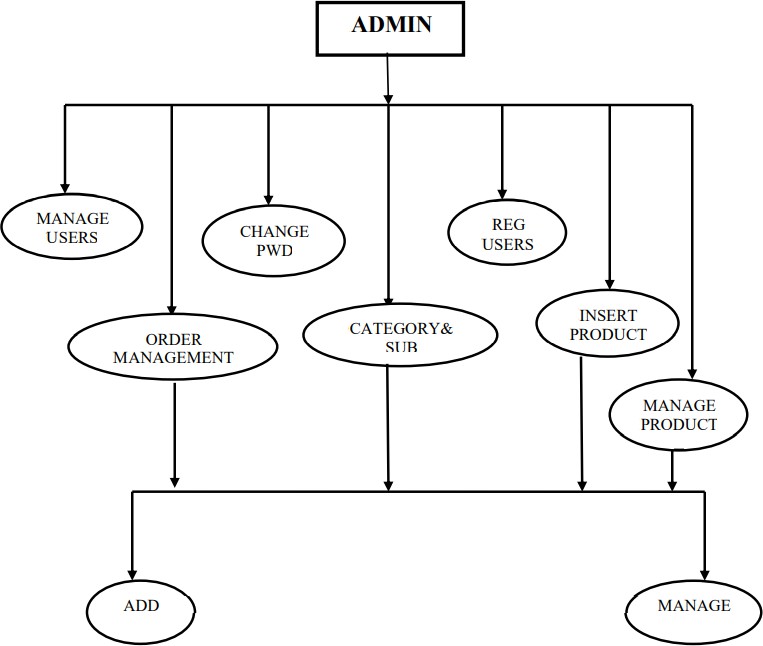
#### DATA FLOW DIAGRAM

A data flow diagram(DFD) is a graphical representation of the "flow" of data through an information system, modeling its process aspects. Often, they are a preliminary step used to create an overview of the system which can later be elaborated. DFDs can also be used for the visualization of data processing (structured design).

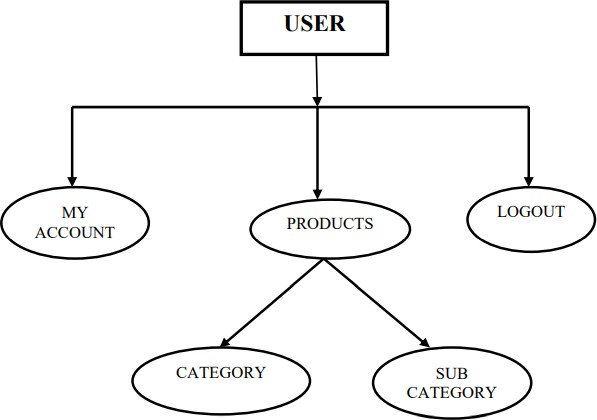
A DFD shows what kinds of information will be input to and output from the system, where the data will come from and go to, and where the data will be stored. It does not show information about the timing of processes, or information about whether processes will operate in sequence or in parallel (which is shown on a flowchart).



**Fig 5.2.1 Home**



**Fig 5.2.2 Admin**



**Fig 5.3 User**

## 5.5 DATABASE DESIGN

**Database Name**: shopping

**Table Name: Admin**

|  |  |  |
| --- | --- | --- |
| **Field Name** | **Field Type** | **Description** |
| Id | Integer | Store the admin id |
| User Name | Varchar | Store the admin name |
| Password | Varchar | Store the admin password |
| Creation Date | Timestamp | Store the current timestamp |
| Updating Date | Varchar | Store the update timestamp |

**Table Name: Category**

|  |  |  |
| --- | --- | --- |
| **Field Name** | **Field Type** | **Description** |
| Id | Integer | Store the product id |
| Category name | Varchar | Store the category name |
| Creation Date | Timestamp | Store the create timestamp |
| Updating Date | Timestamp | Store the update timestamp |

**Table Name: Orders**

|  |  |  |
| --- | --- | --- |
| **Field Name** | **Field Type** | **Description** |
| Id | Integer | Store the order id |
| User Id | Integer | Store the user id |
| Product Id | Integer | Store the product id |
| Quantity | Integer | Store the quantity |
| Order Date | Timestamp | Store the order timestamp |
| Payment method | Varchar | Store the payment method |
| Order status | Varchar | Store the order status |

**Table Name: Order track history**

|  |  |  |
| --- | --- | --- |
| **Field Name** | **Field Type** | **Description** |
| Id | Integer | Store the order id |
| Order id | Integer | Store the order id |
| Status | Varchar | Store the status |
| Remark | Medium text | Store the remark |
| Posting Date | Timestamp | Store the posting timestamp |

**Table Name: User**

|  |  |  |
| --- | --- | --- |
| **Field Name** | **Field Type** | **Description** |
| Id | Integer | Store the user id |
| User Email | Varchar | Store the user email |
| Login Time | Timestamp | Store the login timestamp |
| Logout | Varchar | Store the logout |
| Status | Integer | Store the status |

**Table Name: Product Review**

|  |  |  |
| --- | --- | --- |
| **Field Name** | **Field Type** | **Description** |
| Id | Integer | Store the id |
| Product Id | Integer | Store the product id |
| Quality | Integer | Store the Quality |
| Price | Integer | Store the price |
| Value | Char | Store the product value |
| Name | Varchar | Store the product name |
| Summary | Varchar | Store the summary |
| Review | Long text | Store the review |
| Review Date | Timestamp | Store the review timestamp |

# CHAPTER 6

## IMPLEMENTATION

#### INDEX.PHP(ADMIN)

<?php session\_start(); error\_reporting(0);

include("include/config.php"); if(isset($\_POST['submit']))

{

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

$password=md5($\_POST['password']);

$ret=mysqli\_query($con,"SELECT\*FROMadminWHERE username='$username' and password='$password'");

$num=mysqli\_fetch\_array($ret); if($num>0)

{

$extra="change-password.php";//

$\_SESSION['alogin']=$\_POST['username'];

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

$host=$\_SERVER['HTTP\_HOST'];

$uri=rtrim(dirname($\_SERVER['PHP\_SELF']),'/\\'); header("location:http://$host$uri/$extra");

exit();

}

else

{

$\_SESSION['errmsg']="Invalid username or password";

$extra="index.php";

$host=$\_SERVER['HTTP\_HOST'];

$uri=rtrim(dirname($\_SERVER['PHP\_SELF']),'/\\'); header("location:http://$host$uri/$extra");

exit();

}

}

?>

<!DOCTYPEhtml>

<htmllang="en">

<head>

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

<metaname="viewport"content="width=device-width,initial- scale=1.0">

<title>ShoppingPortal|Adminlogin</title>

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

<linktype="text/css"href="bootstrap/css/bootstrap-responsive.min.css"rel="stylesheet">

<linktype="text/css"href="css/theme.css"rel="stylesheet">

<linktype="text/css"href="images/icons/css/font-awesome.css"rel="stylesheet">

<link type="text/css"href='[http://fonts.googleapis.com/css?family=Open+Sans:400italic,600italic,40](http://fonts.googleapis.com/css?family=Open%2BSans%3A400italic%2C600italic%2C40)0,600' rel='stylesheet'>

</head>

<body>

<divclass="navbarnavbar-fixed-top">

<divclass="navbar-inner">

<divclass="container">

<aclass="btnbtn-navbar"data-toggle="collapse"data-target=".navbar-inverse-collapse">

<iclass="icon-reordershaded"></i>

</a>

<aclass="brand"href="index.html"> Shopping Portal | Admin

</a>

<divclass="nav-collapsecollapsenavbar-inverse-collapse">

<ulclass="navpull-right">

<li><a href="<http://localhost/shopping/>">Back to Portal

</a></li>

</ul>

</div><!--/.nav-collapse-->

</div>

</div><!--/navbar-inner-->

</div><!--/navbar-->

<divclass="wrapper">

<divclass="container">

<divclass="row">

<divclass="modulemodule-loginspan4 offset4">

<formclass="form-vertical" method="post">

<divclass="module-head">

<h3>SignIn</h3>

</div>

<spanstyle="color:red;"><?phpecho htmlentities($\_SESSION['errmsg']); ?><?php echo htmlentities($\_SESSION['errmsg']="");?></span>

<divclass="module-body">

<divclass="control-group">

<divclass="controlsrow-

fluid">

<inputclass="span12"

type="text"id="inputEmail"name="username"placeholder="Username">

</div>

</div>

<divclass="control-group">

<divclass="controlsrow-

fluid">

<inputclass="span12"type="password"

id="inputPassword"name="password"placeholder="Password">

</div>

</div>

</div>

<divclass="module-foot">

<divclass="control-group">

<divclass="controls

clearfix">

<buttontype="submit" class="btn btn-primary pull-right"name="submit">Login</button>

</div>

</div>

</div>

</form>

</div>

</div>

</div>

</div><!--/.wrapper-->

<divclass="footer">

<divclass="container">

<bclass="copyright">&copy;2017ShoppingPortal</b>

All rightsreserved.

</div>

</div>

<scriptsrc="scripts/jquery-1.9.1.min.js"type="text/javascript"></script>

<scriptsrc="scripts/jquery-ui-1.10.1.custom.min.js"type="text/javascript"></script>

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

</body>

##### 6.2 CONFIG.PHP

<?php

$servername="localhost";

$username="root";

$password="root";

$database="breakdown";

//Ctreateconnection

$conn=newmysqli($servername,$username,$password,$database);

if($conn->connect\_error){

die("Connectionfailed:".$conn->connect\_error);

}

?>

##### 6.3 LOFIN.PHP

<!DOCTYPEhtml>

<htmllang="en">

<head>

<!--Meta-->

<metacharset="utf-8">

<metahttp-equiv="Content-Type"content="text/html;charset=UTF-8">

<metaname="viewport"content="width=device-width,initial-scale=1.0,user-scalable=no">

<metaname="description"content="">

<metaname="author"content="">

<metaname="keywords"content="MediaCenter,Template,eCommerce">

<metaname="robots"content="all">

<title>ShoppingPortal|Signi-in|Signup</title>

<!--BootstrapCoreCSS-->

<linkrel="stylesheet"href="assets/css/bootstrap.min.css">

<!--CustomizableCSS-->

<linkrel="stylesheet"href="assets/css/main.css">

<linkrel="stylesheet"href="assets/css/green.css">

<linkrel="stylesheet"href="assets/css/owl.carousel.css">

<linkrel="stylesheet"href="assets/css/owl.transitions.css">

<!--<linkrel="stylesheet"href="assets/css/owl.theme.css">-->

<linkhref="assets/css/lightbox.css"rel="stylesheet">

<linkrel="stylesheet"href="assets/css/animate.min.css">

<linkrel="stylesheet"href="assets/css/rateit.css">

<linkrel="stylesheet"href="assets/css/bootstrap-select.min.css">

<!--DemoPurposeOnly.Shouldberemovedinproduction-->

<linkrel="stylesheet"href="assets/css/config.css">

<linkhref="assets/css/green.css"rel="alternatestylesheet"title="Greencolor">

<linkhref="assets/css/blue.css"rel="alternatestylesheet"title="Bluecolor">

<linkhref="assets/css/red.css"rel="alternatestylesheet"title="Redcolor">

<linkhref="assets/css/orange.css"rel="alternatestylesheet"title="Orangecolor">

<linkhref="assets/css/dark-green.css"rel="alternatestylesheet"title="Darkgreencolor">

<!--DemoPurposeOnly.Shouldberemovedinproduction:END-->

<!--Icons/Glyphs-->

<linkrel="stylesheet"href="assets/css/font-awesome.min.css">

<!--Fonts-->

<linkhref='[http://fonts.googleapis.com/css?family=Roboto:300,400,500,700'](http://fonts.googleapis.com/css?family=Roboto%3A300%2C400%2C500%2C700%27)rel='stylesheet'type='text/css'>

<!--Favicon-->

<linkrel="shortcuticon"href="assets/images/favicon.ico">

<script type="text/javascript">function valid()

{

if(document.register.password.value!=document.register.confirmpassword.value)

{

alert("PasswordandConfirmPasswordFielddonotmatch!!"); document.register.confirmpassword.focus();

returnfalse;

}

returntrue;

}

</script>

<script>

functionuserAvailability(){

$("#loaderIcon").show(); jQuery.ajax({

url: "check\_availability.php", data:'email='+$("#email").val(), type: "POST", success:function(data){

$("#user-availability-status1").html(data);

$("#loaderIcon").hide();

},

error:function(){}

});

}

</script>

</head>

<bodyclass="cnt-home">

<!-- ============================================== HEADER

==============================================-->

<headerclass="header-style-1">

<!-- ============================================== TOP MENU

==============================================-->

<?phpinclude('includes/top-header.php');?>

<!--==============================================TOPMENU:END

==============================================-->

<?phpinclude('includes/main-header.php');?>

<!-- ============================================== NAVBAR

==============================================-->

<?phpinclude('includes/menu-bar.php');?>

<!-- ============================================== NAVBAR : END

==============================================-->

</header>

<!-- ============================================== HEADER : END

==============================================-->

<divclass="breadcrumb">

<divclass="container">

<divclass="breadcrumb-inner">

<ulclass="list-inlinelist-unstyled">

<li><ahref="home.html">Home</a></li>

<liclass='active'>Authentication</li>

</ul>

</div><!--/.breadcrumb-inner-->

</div><!--/.container-->

</div><!--/.breadcrumb-->

<divclass="body-contentouter-top-bd">

<divclass="container">

<divclass="sign-in-pageinner-bottom-sm">

<divclass="row">

<!--Sign-in-->

<divclass="col-md-6col-sm-6sign-in">

<h4class="">signin</h4>

<pclass="">Hello,Welcometoyouraccount.</p>

<formclass="register-formouter-top-xs"method="post">

<spanstyle="color:red;">

<?php

echohtmlentities($\_SESSION['errmsg']);

?>

<?php

echohtmlentities($\_SESSION['errmsg']="");

?>

</span>

<divclass="form-group">

<labelclass="info-title"for="exampleInputEmail1">EmailAddress<span>\*</span></label>

<inputtype="email"name="email"class="form-controlunicase-form-controltext-input" id="exampleInputEmail1">

</div>

<divclass="form-group">

<labelclass="info-title"for="exampleInputPassword1">Password<span>\*</span></label>

<inputtype="password"name="password"class="form-controlunicase-form-controltext- input" id="exampleInputPassword1">

</div>

<divclass="radioouter-xs">

<a href="forgot-password.php" class="forgot-password pull-right">Forgot your Password?</a>

</div>

<button type="submit" class="btn-upper btn btn-primary checkout-page-button"name="login">Login</button>

</form>

</div>

<!--Sign-in-->

<!--createanewaccount-->

<divclass="col-md-6col-sm-6create-new-account">

<h4class="checkout-subtitle">createanewaccount</h4>

<pclass="texttitle-tag-line">CreateyourownShoppingaccount.</p>

<formclass="register-formouter-top-xs"role="form"method="post"name="register" onSubmit="return valid();">

<divclass="form-group">

<labelclass="info-title"for="fullname">FullName<span>\*</span></label>

<inputtype="text"class="form-controlunicase-form-controltext-input"id="fullname" name="fullname" required="required">

</div>

<divclass="form-group">

<labelclass="info-title"for="exampleInputEmail2">EmailAddress<span>\*</span></label>

<input type="email" class="form-control unicase-form-control text-input" id="email"onBlur="userAvailability()" name="emailid" required >

<spanid="user-availability-status1"style="font-size:12px;"></span>

</div>

<divclass="form-group">

<labelclass="info-title"for="contactno">ContactNo.<span>\*</span></label>

<inputtype="text"class="form-controlunicase-form-controltext-input"id="contactno" name="contactno" maxlength="10" required >

</div>

<divclass="form-group">

<labelclass="info-title"for="password">Password.<span>\*</span></label>

<inputtype="password"class="form-controlunicase-form-controltext-input"id="password" name="password"required >

</div>

<divclass="form-group">

<label class="info-title" for="confirmpassword">Confirm Password.

<span>\*</span></label>

<input type="password" class="form-control unicase-form-control text-input"id="confirmpassword" name="confirmpassword" required >

</div>

<buttontype="submit"name="submit"class="btn-upperbtnbtn-primarycheckout-page- button" id="submit">Sign Up</button>

</form>

<spanclass="checkout-subtitleouter-top-xs">SignUpTodayAndYou'llBeAbleTo:

</span>

<divclass="checkbox">

<labelclass="checkbox">

Speedyourwaythroughthecheckout.

</label>

<label class="checkbox">Track your orders easily.

</label>

<labelclass="checkbox">

Keeparecordofallyourpurchases.

</label>

</div>

</div>

<!--createanewaccount--> </div><!--/.row-->

</div>

<?phpinclude('includes/brands-slider.php');?>

</div>

</div>

<?phpinclude('includes/footer.php');?>

<scriptsrc="assets/js/jquery-1.11.1.min.js"></script>

<scriptsrc="assets/js/bootstrap.min.js"></script>

<scriptsrc="assets/js/bootstrap-hover-dropdown.min.js"></script>

<scriptsrc="assets/js/owl.carousel.min.js"></script>

<scriptsrc="assets/js/echo.min.js"></script>

<scriptsrc="assets/js/jquery.easing-1.3.min.js"></script>

<scriptsrc="assets/js/bootstrap-slider.min.js"></script>

<scriptsrc="assets/js/jquery.rateit.min.js"></script>

<scripttype="text/javascript"src="assets/js/lightbox.min.js"></script>

<scriptsrc="assets/js/bootstrap-select.min.js"></script>

<scriptsrc="assets/js/wow.min.js"></script>

<scriptsrc="assets/js/scripts.js"></script>

<!--Fordemopurposes–canberemovedonproduction-->

<scriptsrc="switchstylesheet/switchstylesheet.js"></script>

<script>

$(document).ready(function(){

$(".changecolor").switchstylesheet({seperator:"color"});

$('.show-theme-options').click(function(){

$(this).parent().toggleClass('open'); return false;

});

});

$(window).bind("load",function(){

$('.show-theme-options').delay(2000).trigger('click');

});

</script>

<!--Fordemopurposes–canberemovedonproduction:End-->

</body>

</html>

# CHAPTER 7

**TESTING**

* 1. **Introduction**

Testing is the process of executing a program with the aim of finding errors. To make our software perform well it should be error-free. If testing is done successfully, it will remove all the errors from the software.

**7.2 Types of Testing**

##### Unit Testing

It focuses on the smallest unit of software design. In this, we test an individual unit or group of interrelated units. It is often done by the programmer by using sample input and observing its corresponding outputs.

##### Integration Testing

The objective is to take unit tested components and build a program structure that has been dictated by design. Integration testing is testing in which a group of components is combined to produce output. Integration testing is of four types:

(i)Top-down (ii)Bottom-up (iii)Sandwich (iv)Big-Bang

##### Regression Testing

Everytime a new module is added leads to changes in the program. This type of testing makes sure that the whole component works properly even after adding components to the complete program.

##### Smoke Testing

This test is done to make sure that software under testing is ready or stable for further testing. It is called a smoke test as the testing an initial pass is done to check if it did not catch the fire or smoke in the initial switch on.

##### Alpha Testing

This is a type of validation testing. It is a type of acceptance testing which is done before the product is released to customers. It is typically done by QA people.

##### Beta Testing

The beta test is conducted at one or more customer sites by the end-user of the software.

This version is released for a limited number of users for testing in a real-time environment.

##### System Testing

This software is tested such that it works fine for the different operating systems. It is covered under the black box testing technique. In this, we just focus on the required input and output without focusing on internal working. In this, we have security testing, recovery testing, stress testing, and performance testing.

##### Stress Testing

In this, we give unfavorable conditions to the system and check how they perform in those conditions.

##### Performance Testing

It is designed to test the run-time performance of software within the context of an integrated system. It is used to test the speed and effectiveness of the program. It is also called load testing. In it we check, what is the performance of the system in the given load.

##### Object-Oriented Testing

This testing is a combination of various testing techniques that help to verify and validate object-oriented software. This testing is done in the following manner:

* + Testing of Requirements,
  + Design and Analysis of Testing,
  + Testing of Code,
  + Integration testing.

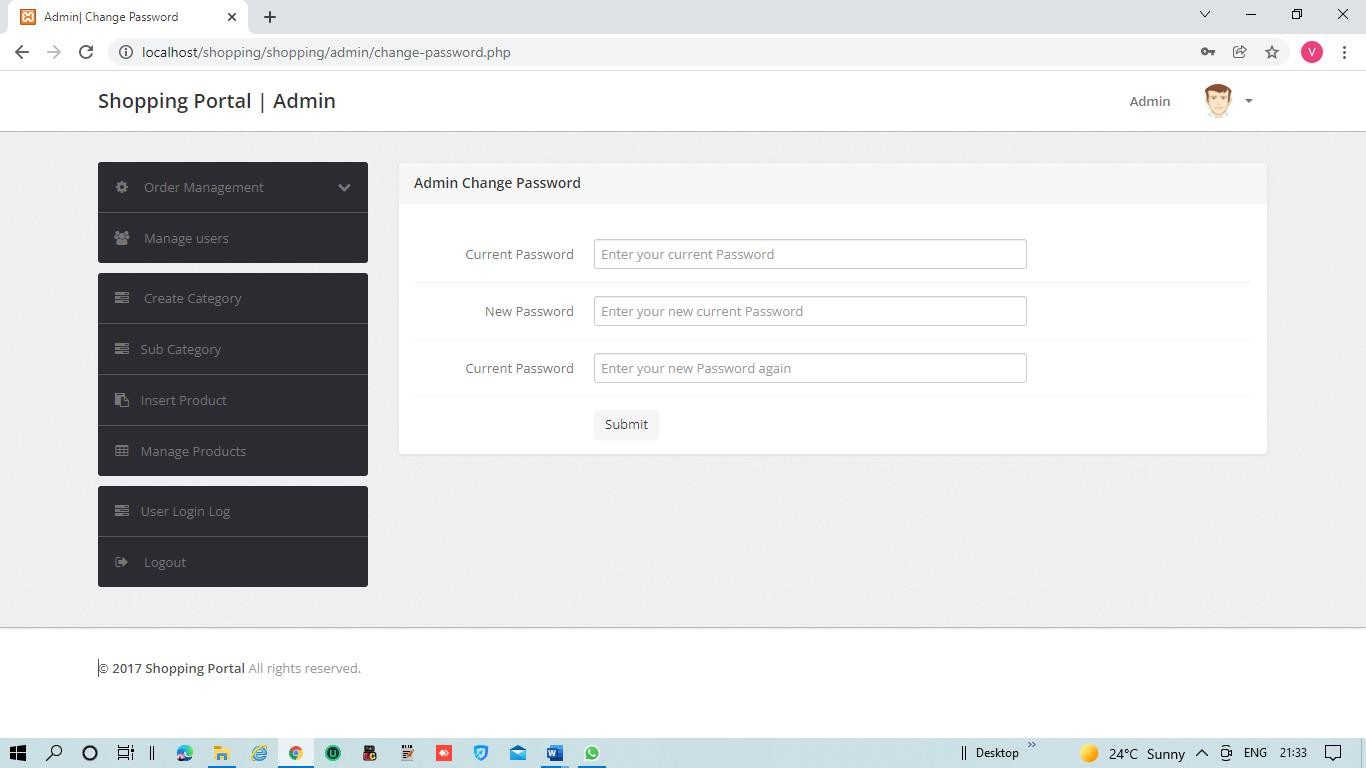
**7.3 Test Cases**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **MODULE** | **GIVENINPUT** | **EXPECTED OUTPUT** | **ACTUALOUTPUT** | **REMARK** |
| Admin Login | Enter valid user name and  password | Login Successfully | Logged in Successfully | Tested OK |
| Admin | Accept Customer  Request | Request accepted and insert into database | Inserted successfully | Tested OK |
| Admin | Accept Customer  Request | Request accepted and insert into database | Inserted successfully | Tested OK |
| Admin | View Reports | View | View | Tested OK |
| customer Login | invalid User name and  password | Login Failure | Login Failure | Tested OK |
| customer Login | valid User name and  password | Login Success | Login Failure | Tested OK |
| User Login | invalid  User name and password | Login Failure | Login Failure | Tested OK |
| Petrol  bunk Login | valid  User name and password | Login Success | Login Failure | Tested OK |

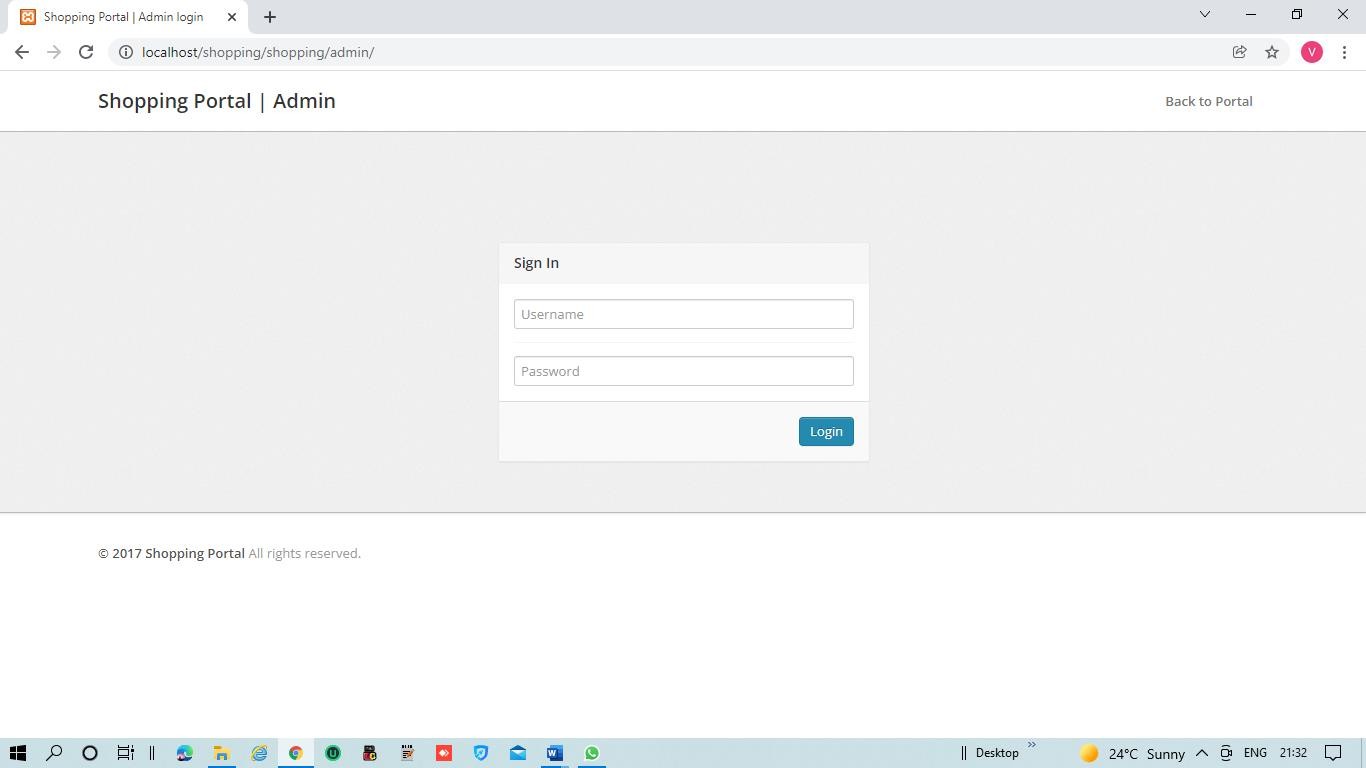
Table 7.2 Test cases

# CHAPTER 8

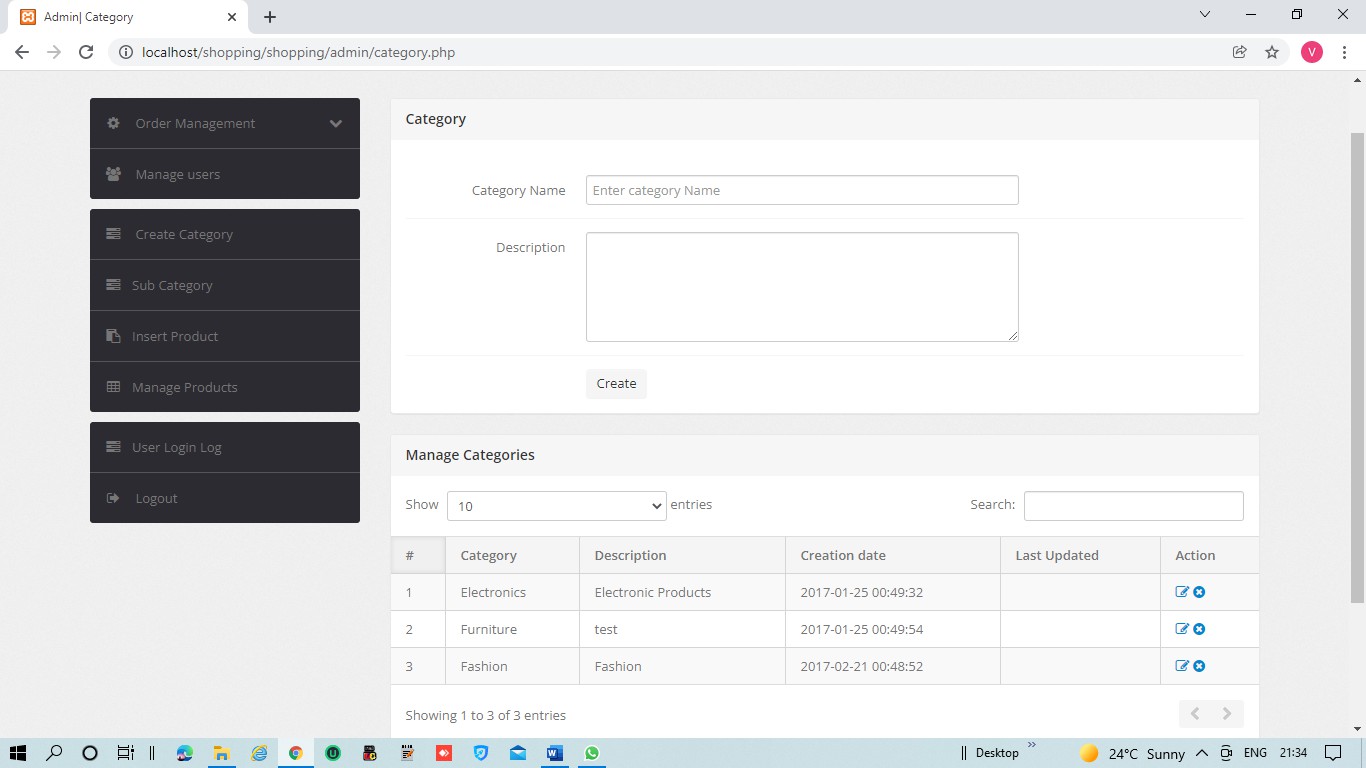
# SCREENSHOTS

**Fig 8.1 Home Page**

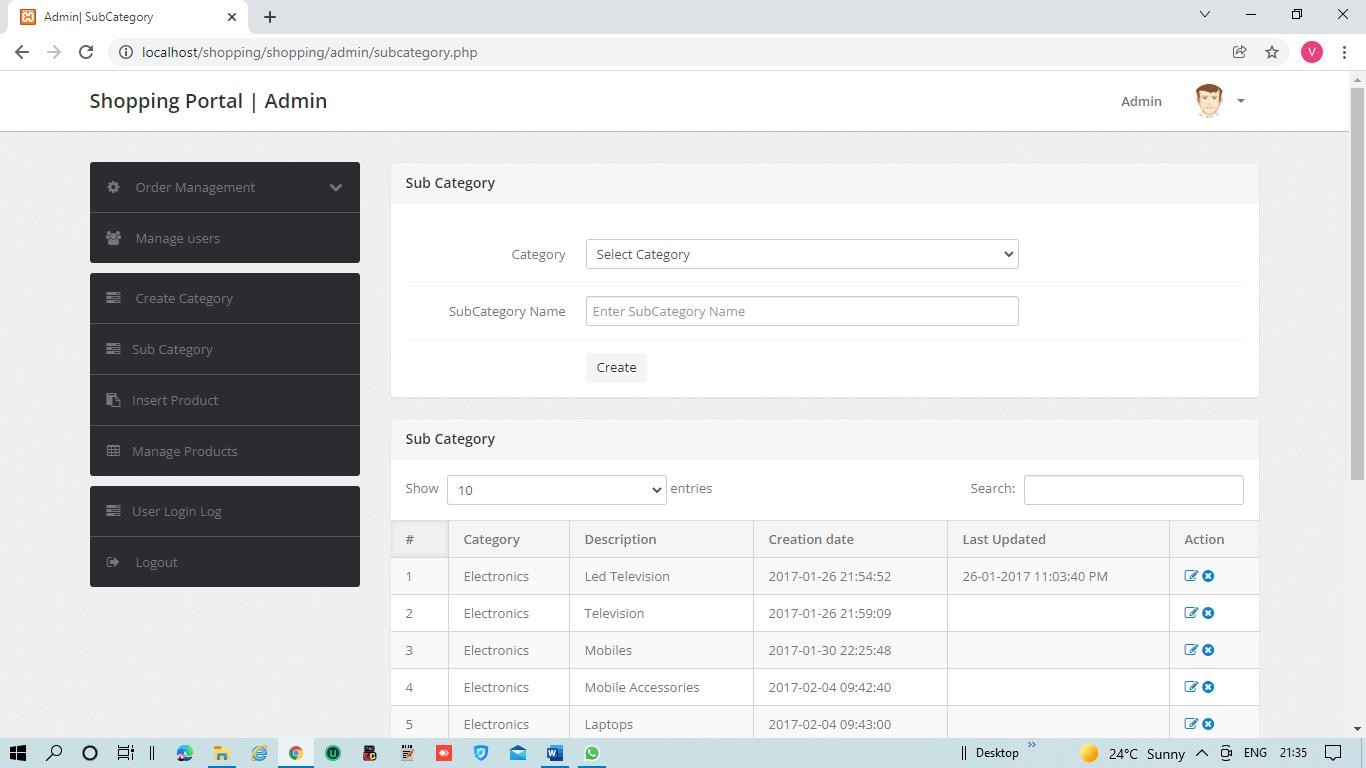
**Fig 8.2 Admin Password Change**



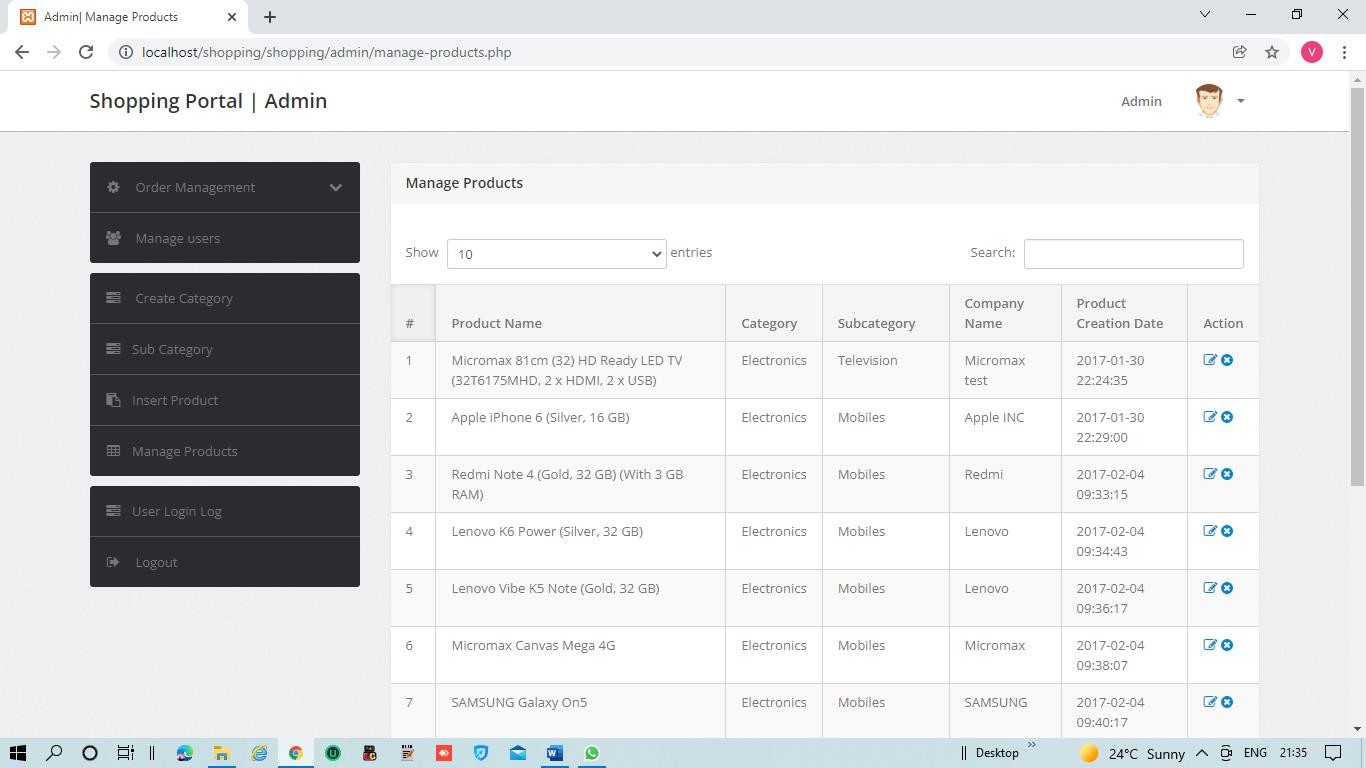
**Fig 8.3 Admin Login Form**



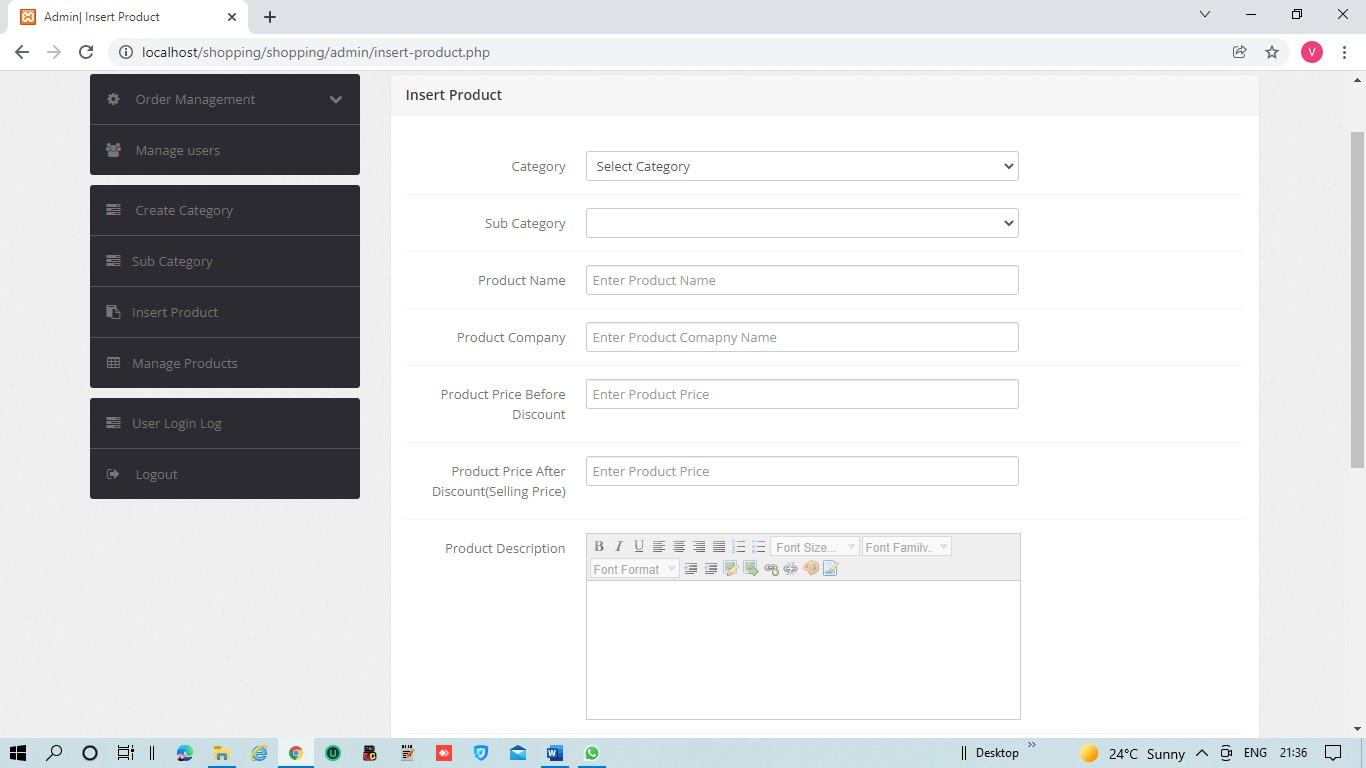
**Fig 8.4 Create & Manage Category Form**



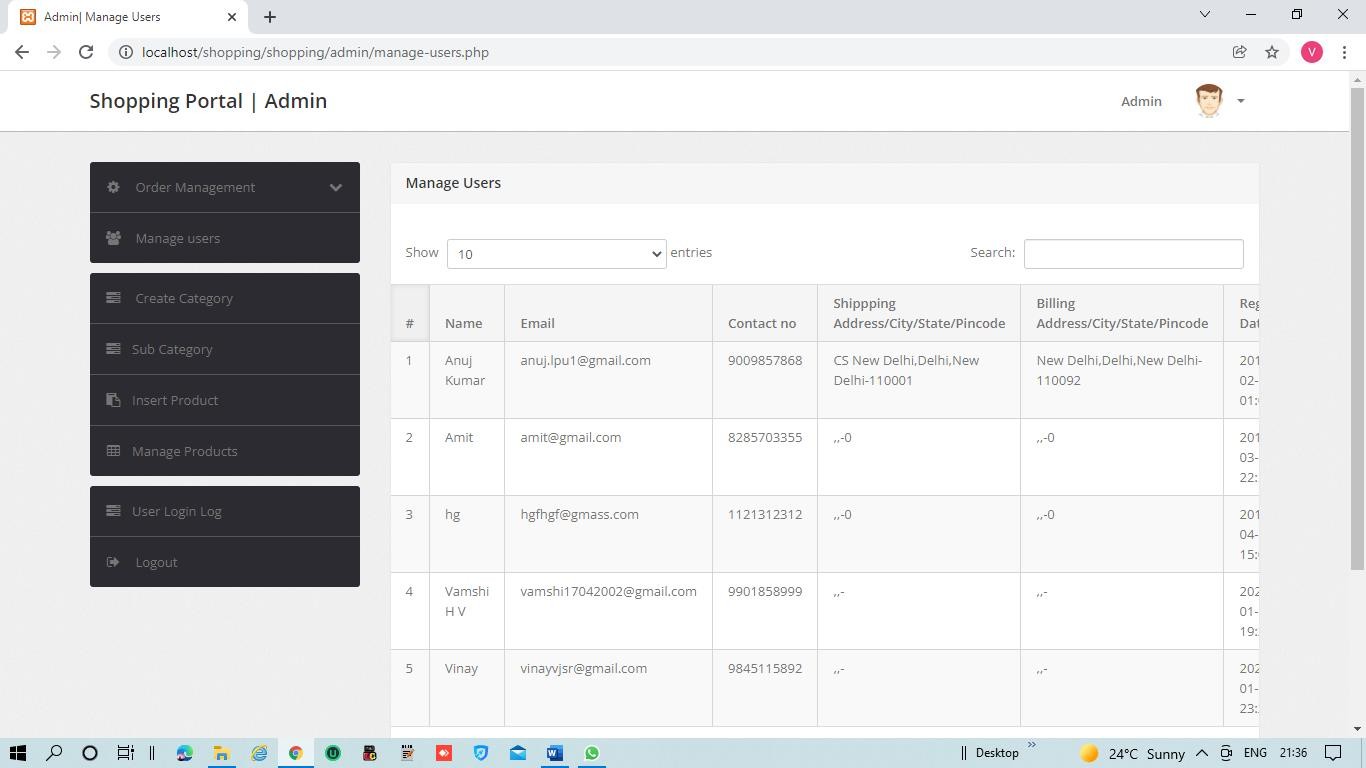
**Fig 8.5 Create & Manage Sub Category Form**



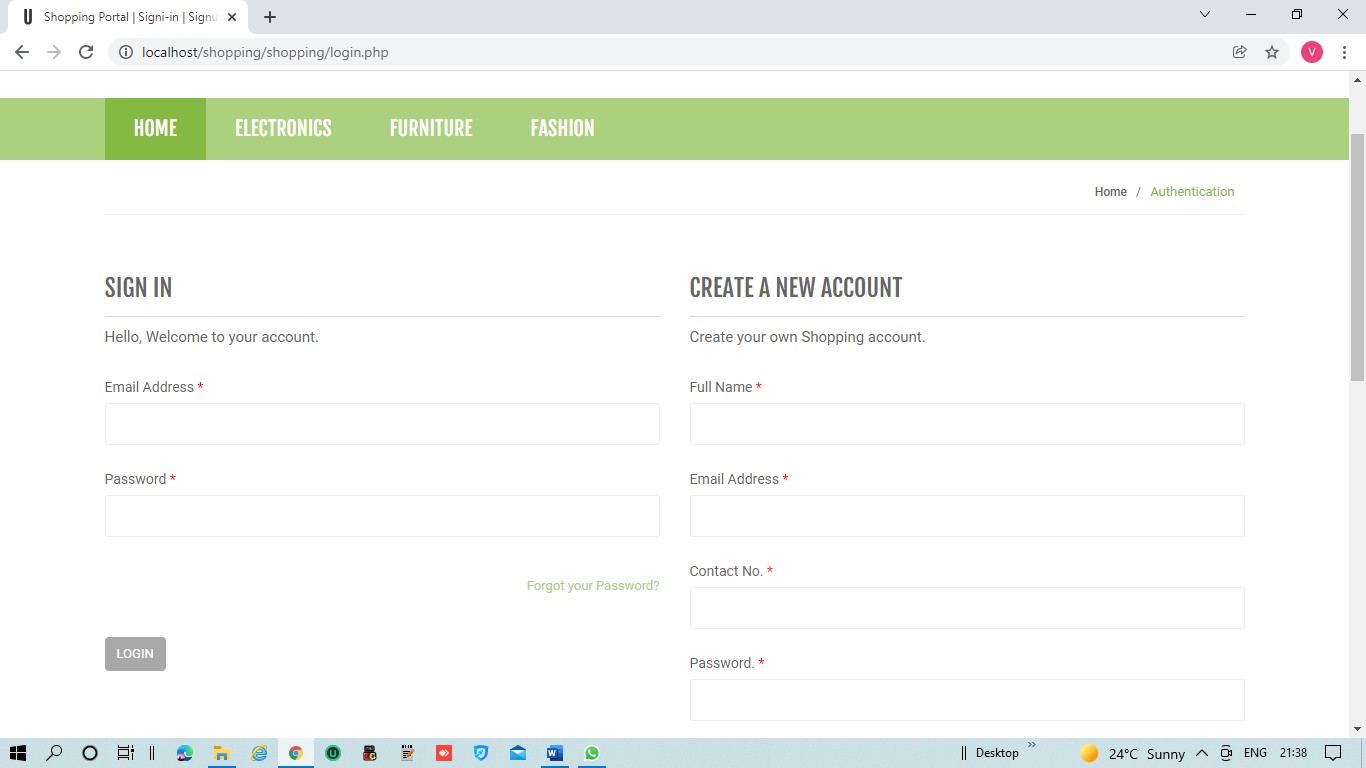
**Fig 8.6 Manage Products Form**



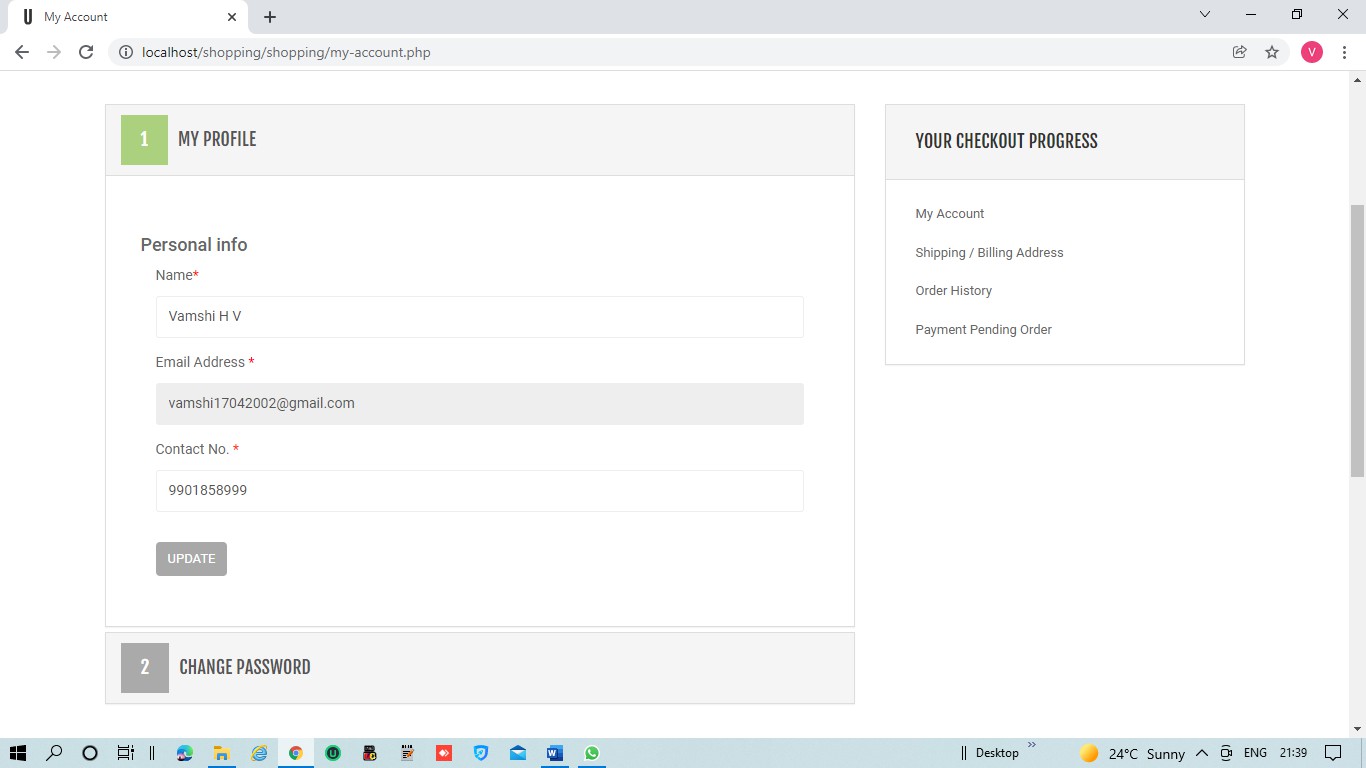
**Fig 8.7 Insert Products Form**



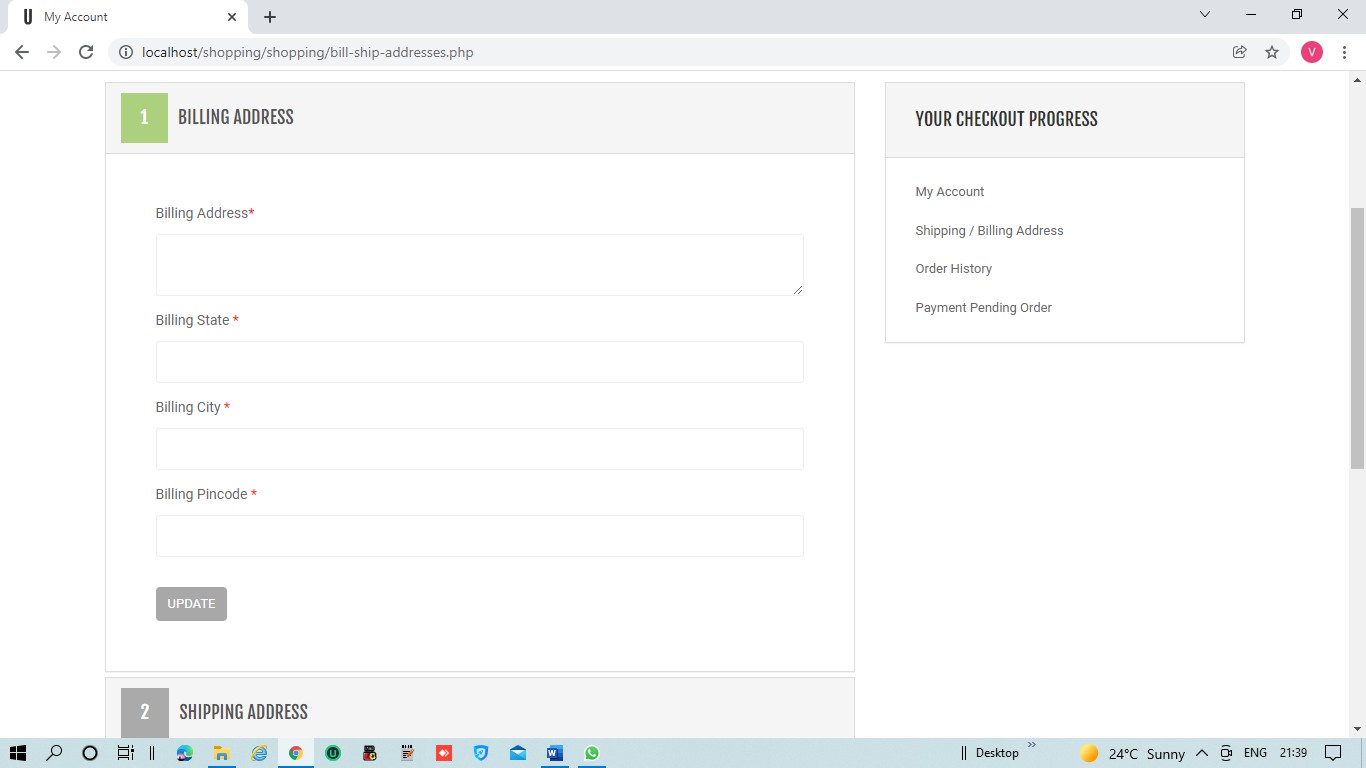
**Fig 8.8 Manage Users Form**



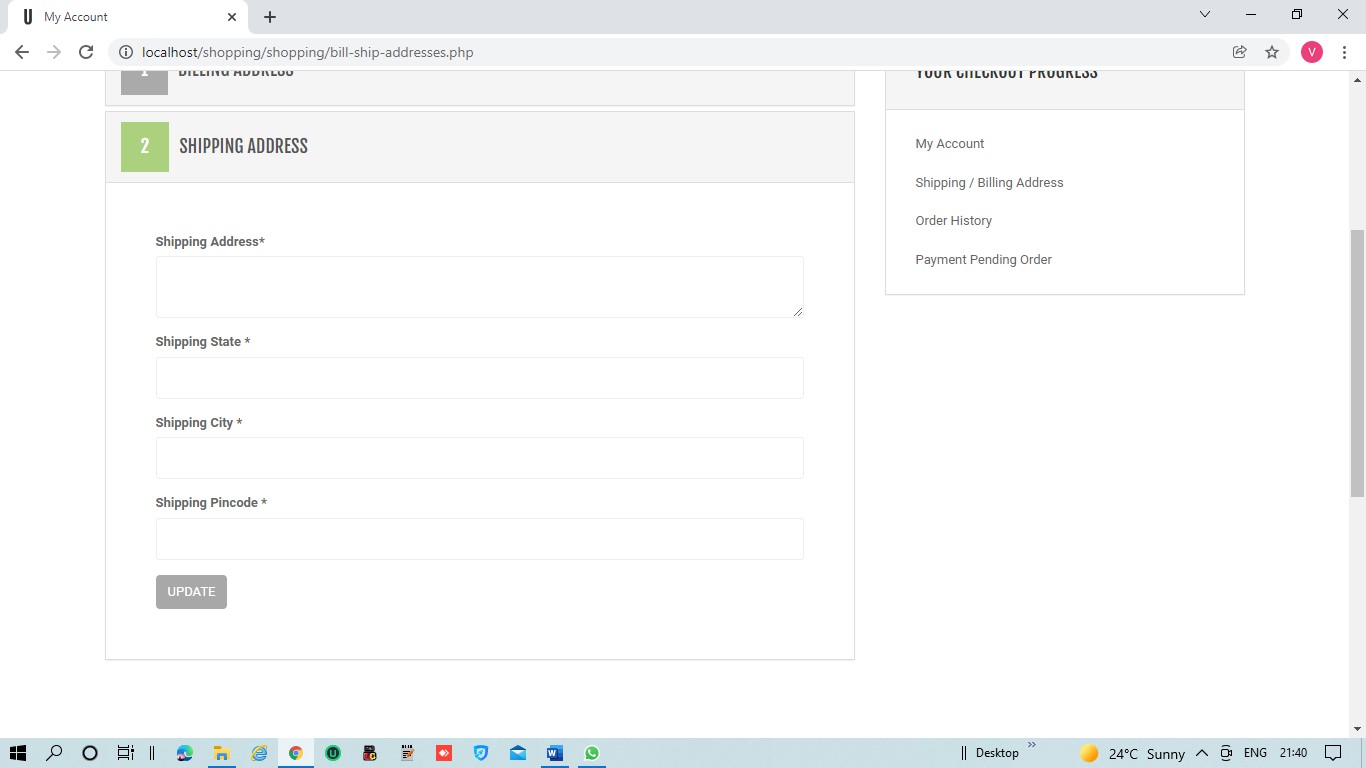
**Fig 8.9 User Login & Register Form**



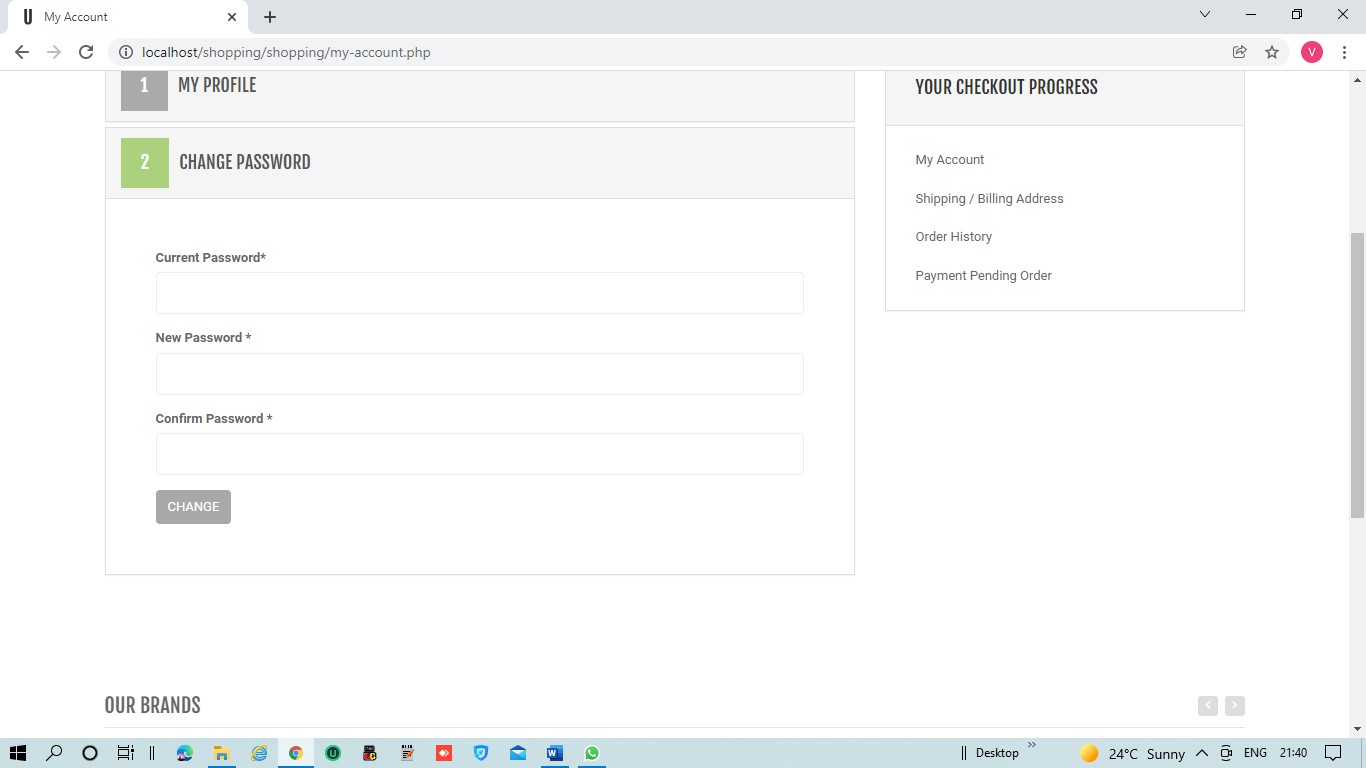
**Fig 8.10 My Profile Form**



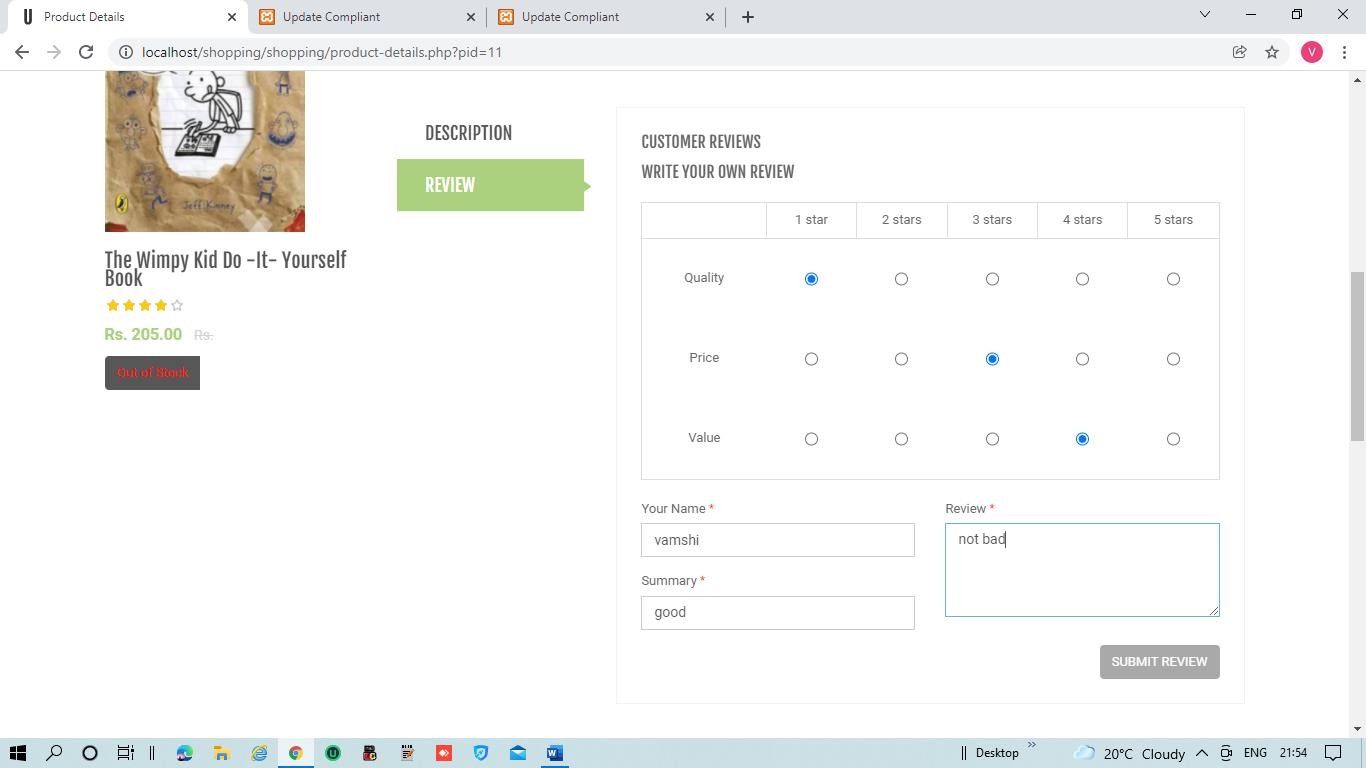
**Fig 8.11 Billing Address Form**



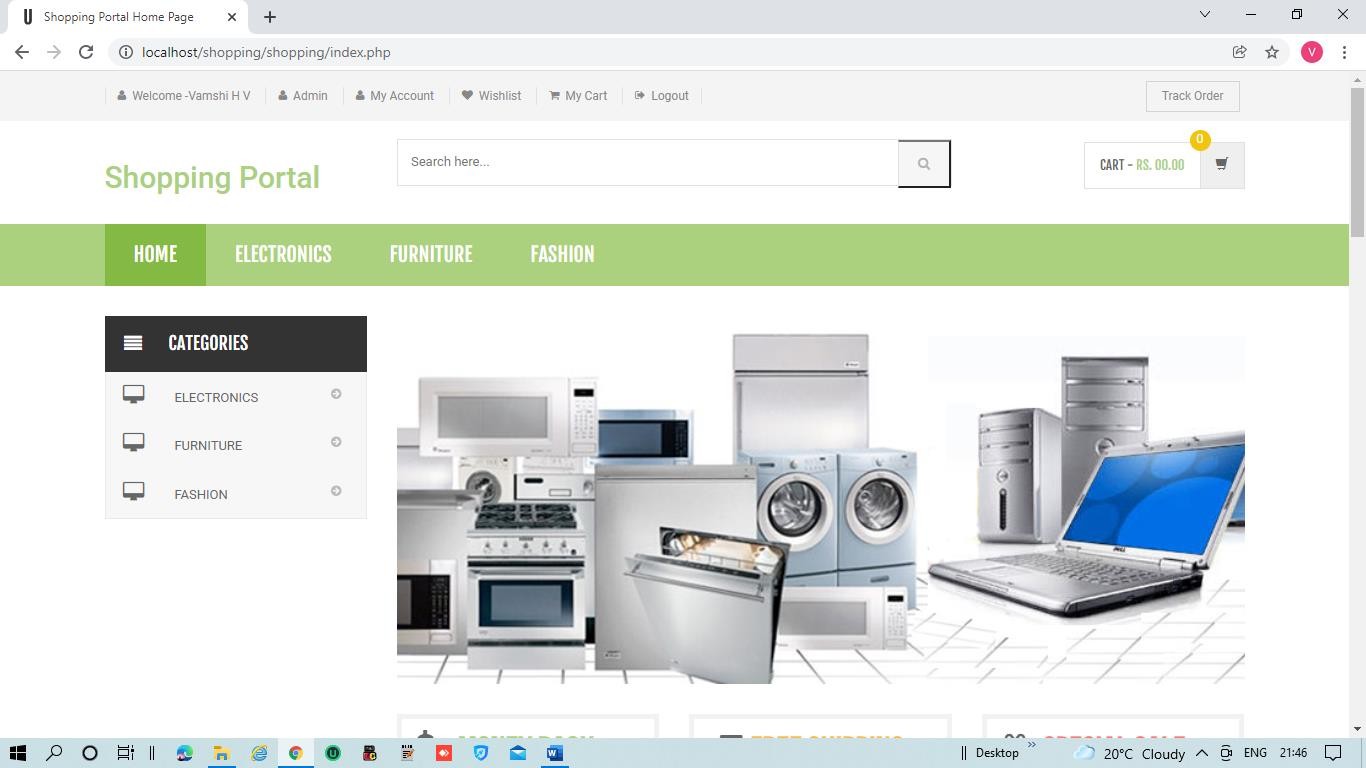
**Fig 8.12 Shipping Address Form**



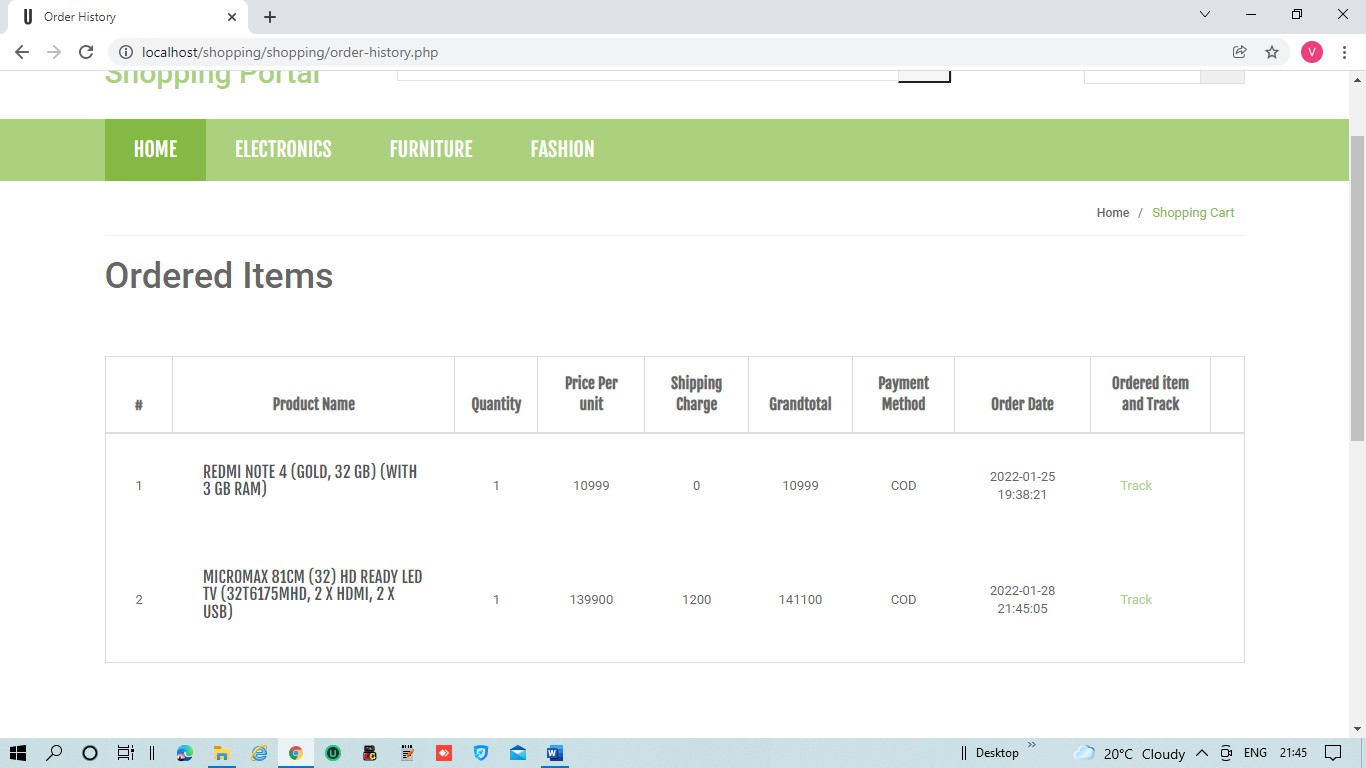
**Fig 8.13 User Change Password Form**



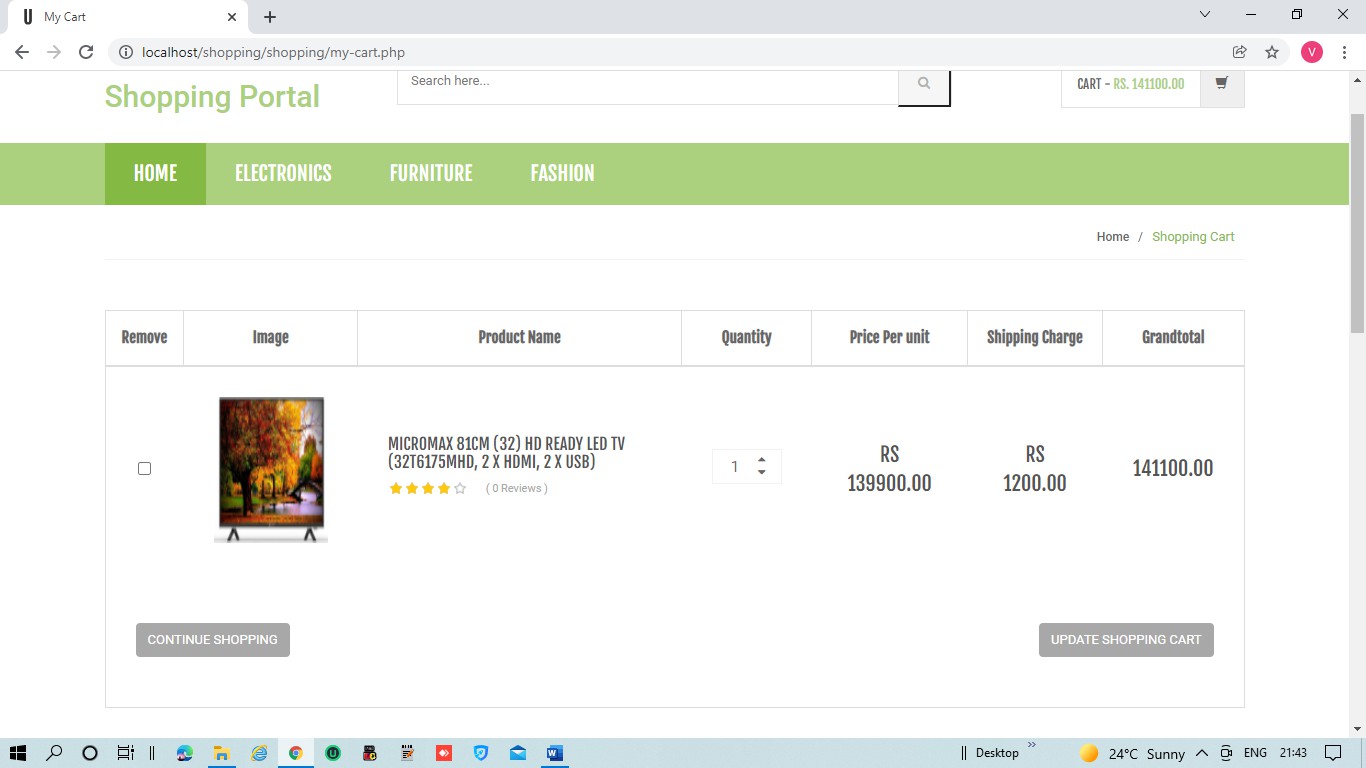
**Fig 8.14 Customers Review Form**



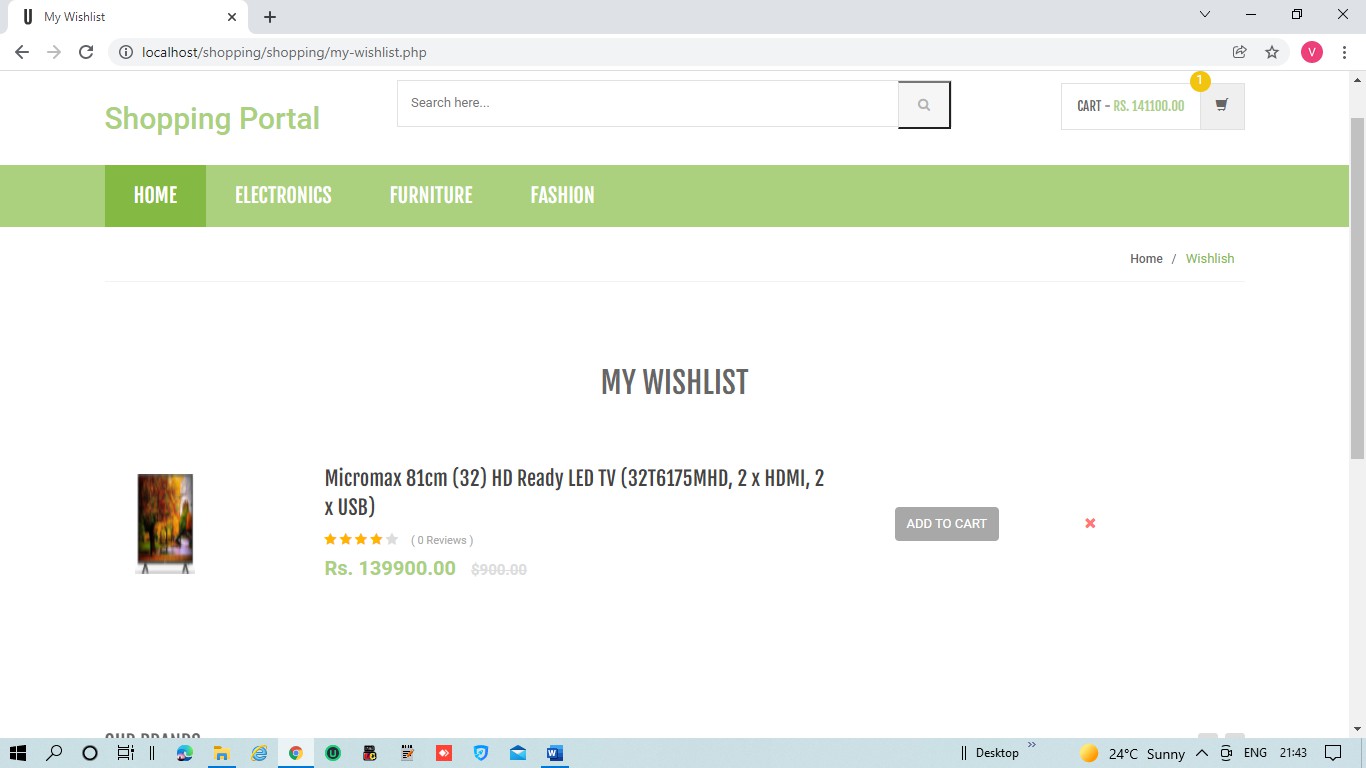
**Fig 8.15 Category & Sub Category Form**



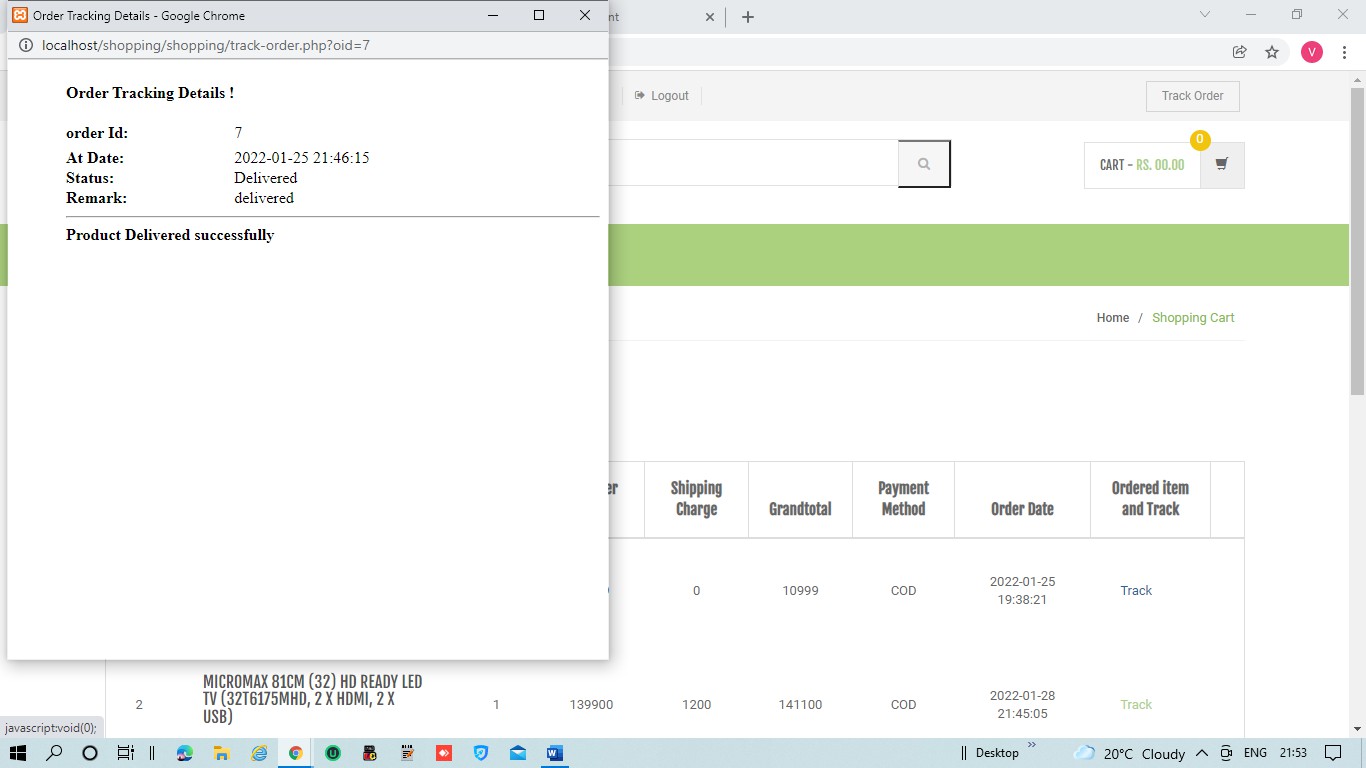
**Fig 8.16 Ordered Items Form**



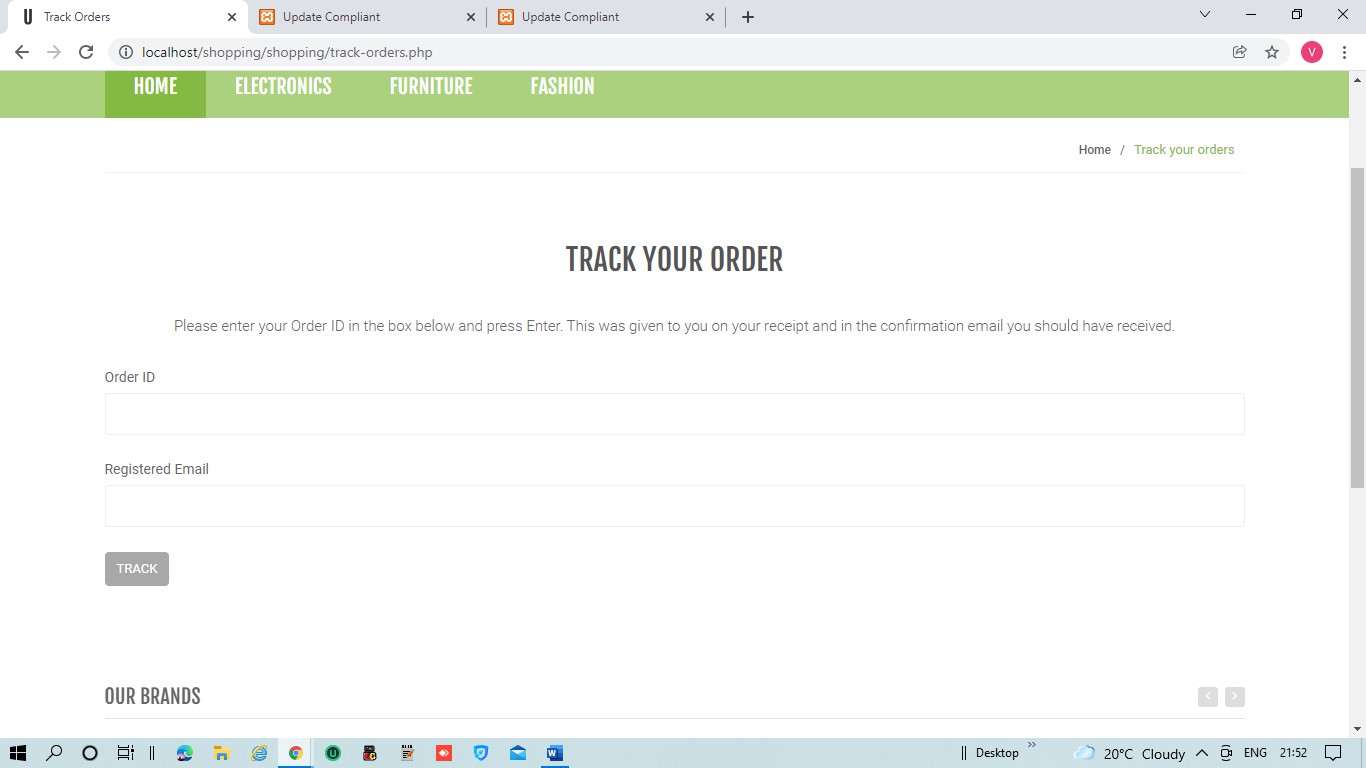
**Fig 8.17 Shopping Cart Form**



**Fig 8.18 Wish list Form**



**Fig 8.19 Track Order Details**



**Fig 8.20 Track Order Form**

**CONCLUSION**

The given project “Online Shopping Portal” was successfully completed and the required reports are generated. The project has been designed to fulfill the needs of the customer and company. The system is extremely user friendly. The system is tested with real data. The results or the live data are compacted with the actual system verified for its accuracy. The system is flexible so that there is a lot of scope to update the system. As the system is flexible the system can be changed if any changes come in future. The developed system is portable. The developed system has been completed which is customized for the satisfaction of the user. The system has been analysed, designed and developed with meticulous care and can be executed without any faults or errors.

**FUTURE ENHANCEMENT**

E-Marketer estimates that there are 220 million online shoppers in China today (defined as having bought at least one thing online), estimated to grow to 420 million by 2016. That’s compared to roughly 150 million in the US. The comparable number in India is 10 million right now. So the Chinese market is 22 times bigger than in India, and consequently has several e- Commerce companies either already IPO’ed and acquired or in line to do so. Only a handful has gone public in India so far. When it comes to India, though, you can’t just go by numbers and comparisons. The thing we repeatedly hear is that India is a unique market with unique challenges, which also means it creates unique opportunities. Given all this, who are going to be these successful e-Commerce players of tomorrow? I think the successful companies of tomorrow are going to look different from the ones that are on the top now. There’s going to be 100-200 million new e-Commerce customers that are going to be up for grabs in the next few years. Another way to look at it is that in 80% of the e-Commerce shoppers of 2016 still are available to be nabbed by e-Commerce sites. And they’re going to come from predominantly two categories. First, from Tier-2 and ISSN2320-5407 International Journal of Advanced Research (2016), Volume 4, Issue 5, 1528-1544 1531 Tier-3 cities as the logistics and connectivity there improve and second, young people that get jobs/pocket-money and start shopping online.

# BIBLIOGRAPHY

**Books referred:**

* + 1. Software Engineering
       - Iansommervile
    2. HTML5 Black Book
       - Paperback

#### WEBSITES:

1. https://[www.w3schools.com](http://www.w3schools.com/)
2. https://[www.w3schools.com/js/default.asp](http://www.w3schools.com/js/default.asp)
3. https://[www.w3schools.com/php/default.asp](http://www.w3schools.com/php/default.asp)