



"ECOMMERCE WEBSITE MANAGEMENT SYSTEM"

SHRESHTH SUDHIR

ABSTRACT

This project is a web-based shop management system which provide dual functionality such as providing user interface for shopping and admin control of products and user at the same time. The project objective is to deliver the online shopping application for web platform. This project is an attempt to provide the advantages of online shopping to customers of a real shop and also providing the ease of technology to the owner/seller/admin by helping him/her to manage products through his/ her portal. It helps buying the products in the shop anywhere through internet by merely accessing to the internet. Thus, the customer will get the service of online shopping and home delivery from his favorite shop and owner willl have to deal with less hassle in managing his/her account book. This system can be implemented to any shop in the locality or to multinational branded shops having retail outlet chains the possibilities are unlimited.

If shops are providing an online portal where their customers can enjoy easy shopping from anywhere, the shops won't be losing any more customers to the trending online shops such as flipcart or ebay. Since the application is a web based it is easily accessible and always available. Our application is both user and administration friendly att the same time.

TABLE OF CONTENTS

Abstract	1
Table Of Contents	2
List of Figures	4
Introduction	5
Objectives	5
Problem Statement	
Scope	6
System Requirements	6
Hardware Requirements	6
Software Requirements	7
Design	7
Requirements and Constraints	7
Entities and Attributes	8
ER Diagram	12
Implementation	13
Technologies/Frameworks used in building the project	13
Code Snippets	14
Create table commands	14
Altering Tables	16
Auto Increment Alteration	17
Constraints	18
Dumping Essential Data	19
SQL Queries in PHP Files	20

Some Test Cases	23
Snapshots	24
Limitaions	28
Conclusion	28
References	29

List of Figures

Fig No.	Description	Page No.
i.	Home Page	23
ii.	Login Page	23
iii.	Cart Page	24
iv.	Checkout	24
v.	Admin Login	25
vi.	Product List	25
vii.	Order & User Manager	26
viii.	Add Products & Users	26

INTRODUCTION

Computer plays an important role in our daily life. Anything we want we can get only in one mouse click. Speed, reliability and accuracy of the computer make it a powerful tool for different purposes. A very important and basic need of today's modern business world is the quick availability and processing of information using computer. One can easily get the type of required information within a fraction of a second. The project that I have taken is also in this category which is used in our daily life whenever we want to purchase some items, we can easily get them at our home.

OBJECTIVE

The objective of project on Online Shopping Portal is to developing a GUI based automated system, which will cover all the information Related to the all products which is used in our daily life. For example – Mobiles Phones, Laptops, Clothes, Books, Electronic Items and many more. So by this GUI based automated system a user want to purchase something then it only a mouse click away to purchase these products. And also provide an admin interface to manage users, products and order information of the users.

Basic Features of our project will include –

- Dashboard Transaction in the Admin Panel
- Product Management
- Cart adding and quantity adjustment
- Can shop even without logging into the site but can't checkout
- Virtual Credit Card checkout
- Admin Login
- Manage Users
- Keep track of Order information
- User CRUD
- Product CRUD

Need of the online shop management system

The “ONLINE SHOPPING PORTAL” is developed according the current need in different Fields. This is online shopping Website which provides facility for purchasing Mobiles, Laptops, Camera and many more items.

So by using this Online Shopping Portal users which want to purchase some products will first Register an account on this portal then Login through their Username and Password, and then Select items which they want to purchase and add them to cart and finally checkout by giving payment details. So by using this portal users can easily purchase products from their home.

PROBLEM STATEMENT

One must know what the problem is before it can be solved. The basis for the online shopping portal is to buy products online and save the timing. An Online shopping portal, who want to buy any product of their need, has to contact different Shoppers, before deciding upon a particular Product that best suit his needs, requirements and satisfaction. Moreover, most of the work involved in this development process has to be done manually which is very time consuming and cumbersome and also, it reduces the efficiency, accuracy. Even with online shopping system it is very difficult for the admin to manually keeep record of every user and products purchased. So, we also provided the solution for this problem by proiding an ADMIN interface in our system.

To know the facts and understanding of the problem in detail, *System Analysis* is carried out. It is the process of studying the business processes and procedures, generally referred to as business systems, to see how they can operate and whether improvement is needed.

SCOPE

The scope of this project is to ensure the best use of computer-based management system to bring the best available products to the fingertips of all users. Products from all over the world can be sold on a single platform thereby encouraging competition and ensuring the product is of the best quality. The project also ensures that any individual with a product to sell can do so, thus providing an open marketplace.

SYSTEM REQUIREMENTS

Hardware Requirements

- i. Processor: Pentium 4 or above
- ii. Ram: 2GB or more
- iii. Hard Disk: 500MB or more

Software Requirements

Technologies used:

- iv. Front end: HTML, CSS, JAVASCRIPT, PHP
- v. Back-end/Database: MySQL

Software:

- i. Text Editor: VS Code
- ii. Server: Apache (on XAMPP)
- iii. Operating System: Windows 10
- iv. Database Support: MySQL 5.7

DESIGN

In order to design a web site, the relational database must be designed first. Conceptual design can be divided into two parts: The data model and the process model. The data model focuses on what data should be stored in the database while the process model deals with how the data is processed. To put this in the context of the relational database, the data model is used to design the relational tables. The process model is used to design the queries that will access and perform operations on those tables.

4.1 Requirements and Constraints

- i. Dashboard Transaction in the Admin Panel
- ii. Product Management
- iii. Cart adding and quantity adjustment
- iv. Can shop even without logging into the site but can't checkout
- v. Virtual Credit Card checkout
- vi. Admin Login
- vii. Manage Users
- viii. Keep track of Order information
- ix. User CRUD
- x. Product CRUD

ENTITIES AND ATTRIBUTES

S.no.	NAME	DESCRIPTION
1.	ADMIN	It contains all the info of the ADMIN
2.	USER	It contains USERS info
3.	CART	It contains info related to the CART
4.	CATEGORIES	It contains the CATEGORY to which product belong
5.	ORDER_INFO	It contains all the info related to ORDER placed
6.	ORDER_PRODUCTS	It records the order info at the time order is placed
7.	PRODUCTS	It contains all the info related to the PRODUCT
8.	BRAND	It represents the BRAND of the product

Table 4.1 Table-names and their description

1. ADMIN

Name	Description	Type
Admin_id	Contains the admin id	int(10) NOT NULL
Admin_name	Contains the admin name	varchar(100) NOT NULL
Admin_email	Contains the admin email	varchar(300) NOT NULL
Admin_password	Contains the admin password	varchar(300) NOT NULL

Table 4.2 Admin table

2. USER

Name	Description	Type
User_id	Contains the user id	int(10) NOT NULL,
First_name	Contains the first name	varchar(100) NOT NULL
Last_name	Contains the last name	varchar(100) NOT NULL
Email	Contains the email	varchar(300) NOT NULL,
Password	Contains the password	varchar(300) NOT NULL
Mobile	Contains the mobile	varchar(10) NOT NULL,
Address1	Contains the address1	varchar(300) NOT NULL
Address2	Contains the address2	varchar(11) NOT NULL

Table 4.3 User table

3. CART

Name	Description	Type
Id	Contains the cart id	int(10) NOT NULL
P_id	Contains the product id	int(10) NOT NULL
Ip_add	Contains the ip address	varchar(250) NOT NULL
User_id	Contains the user id	int(10) DEFAULT NULL
Qty	Contains the quantity	int(10) NOT NULL

Table 4.4 Cart table

4. PRODUCTS

Name	Description	Type
Product_id	Contains the product id	int(100) NOT NULL
Product_cat	Contains the product category	int(100) NOT NULL
Product_brand	Contains the product brand	int(100) NOT NULL
Product_title	Contains the product title	varchar(255) NOT NULL
Product_price	Contains the product price	int(100) NOT NULL
Product_desc	Contains the product description	text NOT NULL
Product_image	Contains the product image	text NOT NULL
Product_keywords	Contains the product keywords	text NOT NULL

Table 4.5 Products table

5. CATEGORIES

Name	Description	Type
Cat_id	Contains the category id	int(100) NOT NULL
Cat_title	Contains the category title	text NOT NULL

Table 4.6 Categories table

6. BRAND

Name	Description	Type
Brand_id	Contains the brand id	int(100) NOT NULL,
Brand_title	Contains the brand title	text NOT NULL

Table 4.6 Brand table

7. ORDER_PRODUCTS

Name	Description	Type
Order_pro_id`	Contains the oredered product id	int(10) NOT NULL
Order_id	Contains the order id	int(11) NOT NULL
Product_id	Contains the product id	int(11) NOT NULL
Qty	Contains the quantity	DEFAULT NULL,
Amt	Contains the amount	DEFAULT NULL,

Table 4.7 Order_Products table

8. ORDER_INFO

Name	Description	Type
Order_id	Contains the order id	int(10) NOT NULL,
User_id	Contains the user id	int(11) NOT NULL,
F_name`	Contains the full name	varchar(255) NOT NULL,
Email	Contains the email	varchar(255) NOT NULL,
Address	Contains the address	varchar(255) NOT NULL,
City	Contains the city	varchar(255) NOT NULL,
State	Contains the state	varchar(255) NOT NULL,
Zip	Contains the zip	int(10) NOT NULL,
Cardname	Contains the cardname	varchar(255) NOT NULL,
Cardnumber	Contains the cardnumber	varchar(20) NOT NULL,
Expdate	Contains the expiry date	varchar(255) NOT NULL,
Prod_count	Contains the products count	int(15) DEFAULT NULL
Total_amt	Contains the total amount	int(15) DEFAULT NULL
Cvv	Contains the cvv	int(5) NOT NULL,

Table 4.7 Order_Info table

ER DIAGRAM

An **entity relationship diagram (ERD)** shows the relationships of entity sets stored in a database. An entity in this context is an object, a component of data. An entity set is a collection of similar entities. These entities can have attributes that define its properties. The fig. 4.1 shows the ERD for the database.

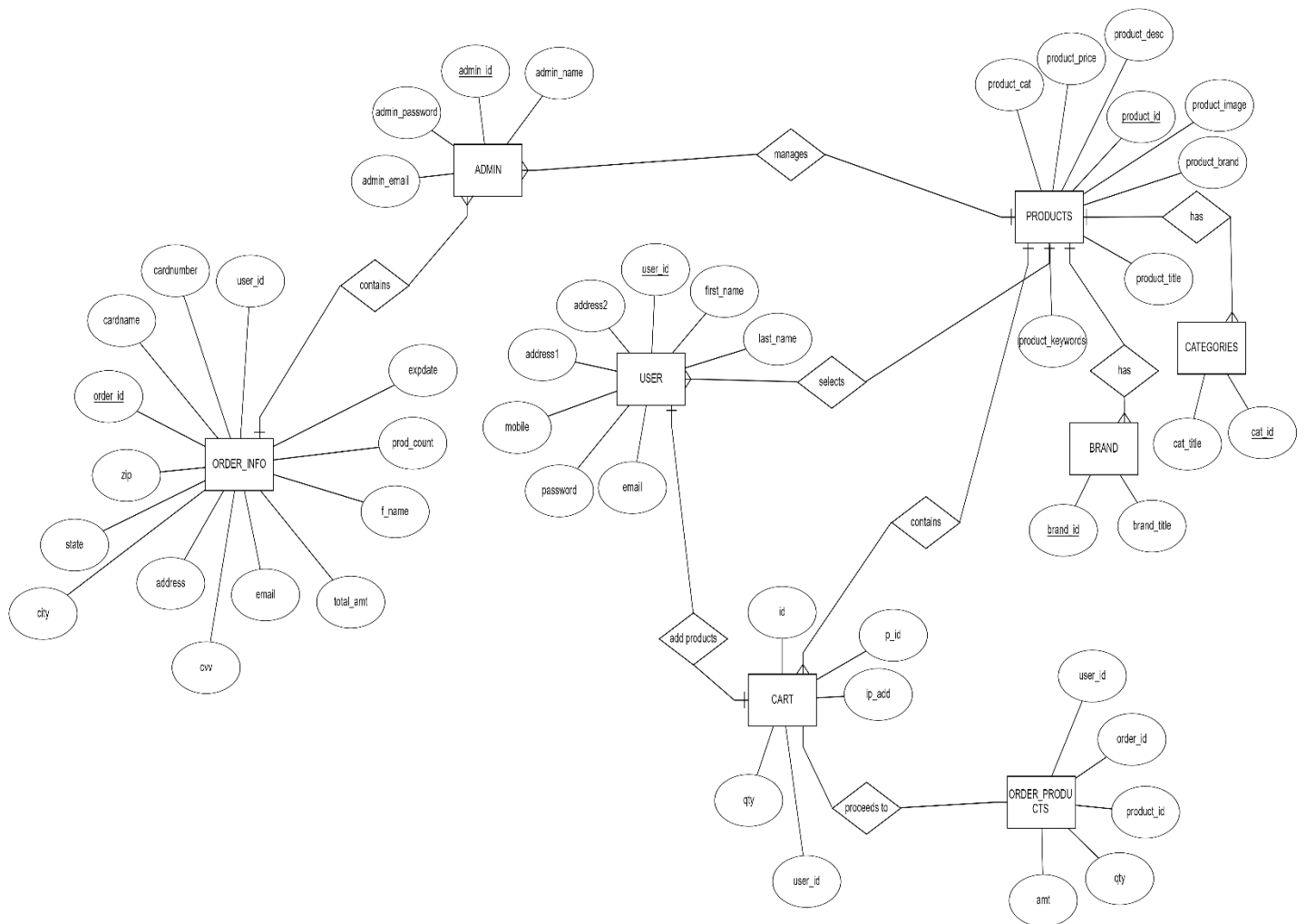


Fig 5.1 ER Diagram

IMPLEMENTATION

Technologies used in building the project

i. HTML5

HTML 5 is a mark-up language used for structuring and presenting content on the World Wide Web. It is the fifth and current major version of the HTML standard. HTML5 includes detailed processing models to encourage more interoperable implementations; it extends, improves and rationalises the mark-up available for documents and introduces mark-up and application programming interfaces (APIs) for Complex Web applications.

ii. PHP

The term PHP is an acronym for PHP: Hypertext Preprocessor. PHP is a server-side scripting language designed specifically for web development. It is open-source which means it is free to download and use. It is very simple to learn and use. The files have the extension “.php”.

iii. JAVASCRIPT

JavaScript is a cross-platform, object-oriented scripting language used to make webpages interactive (e.g., having complex animations, clickable buttons, popup menus, etc.). JavaScript contains a standard library of objects, such as Array, Date, and Math, and a core set of language elements such as operators, control structures, and statements. Core JavaScript can be extended for a variety of purposes by supplementing it with additional objects.

iv. SQL (Structured Query Language)

MySQL Database Management System is used to store the user and product information which is stored in the secondary storage device and can be altered anytime. Normalized and efficient schema is used to avoid redundancy and inconsistency. The data is updated in real time.

v. CSS

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a mark-up language like HTML5.

CODE SNIPPETS

Create table commands –

i. ADMIN

```
CREATE TABLE `admin_info` (  
  `admin_id` int(10) NOT NULL,  
  `admin_name` varchar(100) NOT NULL,  
  `admin_email` varchar(300) NOT NULL,  
  `admin_password` varchar(300) NOT NULL  
  ) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

ii. USER

```
CREATE TABLE `user_info` (  
  `user_id` int(10) NOT NULL,  
  `first_name` varchar(100) NOT NULL,  
  `last_name` varchar(100) NOT NULL,  
  `email` varchar(300) NOT NULL,  
  `password` varchar(300) NOT NULL,  
  `mobile` varchar(10) NOT NULL,  
  `address1` varchar(300) NOT NULL,  
  `address2` varchar(11) NOT NULL  
  ) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

iii. CART

```
CREATE TABLE `cart` (  
  `id` int(10) NOT NULL,  
  `p_id` int(10) NOT NULL,  
  `ip_add` varchar(250) NOT NULL,  
  `user_id` int(10) DEFAULT NULL,  
  `qty` int(10) NOT NULL  
  ) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

iv. CATEGORIES

```
CREATE TABLE `categories` (  
  `cat_id` int(100) NOT NULL,  
  `cat_title` text NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

v. ORDER_INFO

```
CREATE TABLE `orders_info` (  
  `order_id` int(10) NOT NULL,  
  `user_id` int(11) NOT NULL,  
  `f_name` varchar(255) NOT NULL,  
  `email` varchar(255) NOT NULL,  
  `address` varchar(255) NOT NULL,  
  `city` varchar(255) NOT NULL,  
  `state` varchar(255) NOT NULL,  
  `zip` int(10) NOT NULL,  
  `cardname` varchar(255) NOT NULL,  
  `cardnumber` varchar(20) NOT NULL,  
  `expdate` varchar(255) NOT NULL,  
  `prod_count` int(15) DEFAULT NULL,  
  `total_amt` int(15) DEFAULT NULL,  
  `cvv` int(5) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

vi. ORDER_PRODUCTS

```
CREATE TABLE `order_products` (  
  `order_pro_id` int(10) NOT NULL,  
  `order_id` int(11) NOT NULL,  
  `product_id` int(11) NOT NULL,  
  `qty` int(15) DEFAULT NULL,  
  `amt` int(15) DEFAULT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```


vii. PRODUCTS

```
CREATE TABLE `products` (  
  `product_id` int(100) NOT NULL,  
  `product_cat` int(100) NOT NULL,  
  `product_brand` int(100) NOT NULL,  
  `product_title` varchar(255) NOT NULL,  
  `product_price` int(100) NOT NULL,  
  `product_desc` text NOT NULL,  
  `product_image` text NOT NULL,  
  `product_keywords` text NOT NULL  
  ) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

viii. BRAND

```
CREATE TABLE `brands` (  
  `brand_id` int(100) NOT NULL,  
  `brand_title` text NOT NULL  
  ) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

ALTERING TABLES

1. Indexes for table `admin_info`

- a. ALTER TABLE `admin_info`
- b. ADD PRIMARY KEY (`admin_id`);

2. Indexes for table `brands`

- a. ALTER TABLE `brands`
- b. ADD PRIMARY KEY (`brand_id`);

3. Indexes for table `cart`

- a. ALTER TABLE `cart`
- b. ADD PRIMARY KEY (`id`);

4. Indexes for table `categories`

- a. ALTER TABLE `categories`
- b. ADD PRIMARY KEY (`cat_id`);

5. Indexes for table **`orders_info`**

- a. ALTER TABLE `orders_info`
- b. ADD PRIMARY KEY (`order_id`),
- c. ADD KEY `user_id` (`user_id`);

6. Indexes for table **`order_products`**

- a. ALTER TABLE `order_products`
- b. ADD PRIMARY KEY (`order_pro_id`),
- c. ADD KEY `order_products` (`order_id`),
- d. ADD KEY `product_id` (`product_id`);

7. Indexes for table **`products`**

- a. ALTER TABLE `products`
- b. ADD PRIMARY KEY (`product_id`);

8. Indexes for table **`user_info`**

- a. ALTER TABLE `user_info`
- b. ADD PRIMARY KEY (`user_id`);

AUTO_INCREMENT ALTERATION

1. AUTO_INCREMENT for table **`admin_info`**

- a. ALTER TABLE `admin_info`
- b. MODIFY `admin_id` int(10) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=2;

2. AUTO_INCREMENT for table **`brands`**

- a. ALTER TABLE `brands`
- b. MODIFY `brand_id` int(100) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=7;

3. AUTO_INCREMENT for table **`cart`**

- a. ALTER TABLE `cart`
- b. MODIFY `id` int(10) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=178;

4. AUTO_INCREMENT for table **`categories`**

- a. ALTER TABLE `categories`
- b. MODIFY `cat_id` int(100) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=8;

5. AUTO_INCREMENT for table **`orders_info`**

- a. ALTER TABLE `orders_info`
- b. MODIFY `order_id` int(10) NOT NULL AUTO_INCREMENT,
AUTO_INCREMENT=16;

6. AUTO_INCREMENT for table **`order_products`**

- a. ALTER TABLE `order_products`
- b. MODIFY `order_pro_id` int(10) NOT NULL AUTO_INCREMENT,
AUTO_INCREMENT=111;

7. AUTO_INCREMENT for table **`products`**

- a. ALTER TABLE `products`
- b. MODIFY `product_id` int(100) NOT NULL AUTO_INCREMENT,
AUTO_INCREMENT=86;

8. AUTO_INCREMENT for table **`user_info`**

- a. ALTER TABLE `user_info`
- b. MODIFY `user_id` int(10) NOT NULL AUTO_INCREMENT,
AUTO_INCREMENT=30;

CONSTRAINTS

a) Constraints for table **`orders_info`**

```
ALTER TABLE `orders_info`  
ADD CONSTRAINT `user_id` FOREIGN KEY (`user_id`) REFERENCES `user_info`  
(`user_id`);
```

b) Constraints for table **`order_products`**

```
ALTER TABLE `order_products`  
ADD CONSTRAINT `order_products` FOREIGN KEY (`order_id`) REFERENCES  
`orders_info` (`order_id`) ON DELETE NO ACTION ON UPDATE CASCADE,  
ADD CONSTRAINT `product_id` FOREIGN KEY (`product_id`) REFERENCES  
`products` (`product_id`);  
COMMIT;
```

DUMPING ESSENTIAL DATA IN TABLES

Dumping data for table `brands`

```
INSERT INTO `brands` (`brand_id`, `brand_title`) VALUES
(1, 'HP'),
(2, 'Samsung'),
(3, 'Apple'),
(4, 'motorolla'),
(5, 'LG'),
(6, 'Cloth Brand');
```

Dumping data for table `categories`

```
INSERT INTO `categories` (`cat_id`, `cat_title`) VALUES
(1, 'Electronics'),
(2, 'Ladies Wears'),
(3, 'Mens Wear'),
(4, 'Kids Wear'),
(5, 'Furnitures'),
(6, 'Home Appliances'),
(7, 'Electronics Gadgets');
```

Dumping data for table `user_info`

```
INSERT INTO `user_info` (`user_id`, `first_name`, `last_name`, `email`, `password`,
`mobile`,
`address1`, `address2`) VALUES
(12, 'Manish', 'Toshwal', 'manishtoshwal@gmail.com', '123', '9448121558', '123456789', 'abc,
def'),
(16, 'Karan', 'Gupta', 'karangupta@gmail.com', '123', '9877654334', 'snhdgvajfehyfygv', 'asr'),
(26, 'demo', 'demo', 'demo@gmail.com', '123', '1234567890', 'delhi', 'delhi'),
(28, 'demo1', 'demo1', 'demo1@gmail.com', '123', '1234567890', 'delhi', 'delhi'),
(29, 'DTU', 'DTU', 'dtu@gmail.com', 'dtu@gmail.com', '1234567890', 'Rohini', 'Delhi');
```

Dumping data for table `admin_info`

```
INSERT INTO `admin_info` (`admin_id`, `admin_name`, `admin_email`, `admin_password`)
VALUES
(1, 'admin', 'admin@gmail.com', '25f9e794323b453885f5181f1b624d0b');
```

SQL QUERIES USED IN PHP FILES –

ACTION.PHP

- a) `$sql = "SELECT COUNT(*) AS count_items FROM products WHERE product_cat=$i";`
- b) `$sql = "SELECT COUNT(*) AS count_items FROM products WHERE product_brand=$i";`
- c) `$sql = "SELECT * FROM products";`
- d) `$sql = "SELECT * FROM products, categories WHERE product_cat = '$id' AND product_cat=cat_id";`
- e) `sql = "SELECT * FROM products, categories WHERE product_brand = '$id' AND product_cat=cat_id";`
- f) `$sql = "SELECT * FROM products, categories WHERE product_cat=cat_id AND product_keywords LIKE '%$keyword%'";`
- g) `$sql = "SELECT * FROM cart WHERE p_id = '$p_id' AND user_id = '$user_id'";`
- h) `$sql = "INSERT INTO `cart``
- i) `$sql = "SELECT id FROM cart WHERE ip_add = '$ip_add' AND p_id = '$p_id' AND user_id = -1";`
- j) `$sql = "INSERT INTO `cart``
- k) `$sql = "SELECT COUNT(*) AS count_item FROM cart WHERE user_id = $_SESSION[uid]";`
- l) `$sql = "SELECT COUNT(*) AS count_item FROM cart WHERE ip_add = '$ip_add' AND user_id < 0";`
- m) `$sql = "SELECT a.product_id,a.product_title,a.product_price,a.product_image,b.id,b.qty FROM products a, cart b WHERE a.product_id=b.p_id AND b.user_id='$_SESSION[uid]'";`
- n) `$sql = "SELECT a.product_id,a.product_title,a.product_price,a.product_image,b.id,b.qty FROM products a, cart b WHERE a.product_id=b.p_id AND b.ip_add='$ip_add' AND b.user_id < 0";`
- o) `$sql = "SELECT a.product_id,a.product_title,a.product_price,a.product_image,b.id,b.qty FROM products a, cart b WHERE a.product_id=b.p_id AND b.user_id='$_SESSION[uid]'";`
- p) `$sql = "DELETE FROM cart WHERE p_id = '$remove_id' AND user_id = '$_SESSION[uid]'";`
- q) `$sql = "DELETE FROM cart WHERE p_id = '$remove_id' AND ip_add = '$ip_add'";`
- r) `$sql = "UPDATE cart SET qty='$qty' WHERE p_id = '$update_id' AND user_id = '$_SESSION[uid]'";`
- s) `$sql = "UPDATE cart SET qty='$qty' WHERE p_id = '$update_id' AND ip_add = '$ip_add'";`

CHECKOUT.PHP

- a) `$sql = "INSERT INTO `orders_info` ('order_id', `user_id`, `f_name`, `email`, `address`, `city`, `state`, `zip`, `cardname`, `cardnumber`, `expdate`, `prod_count`, `total_amt`, `cvv`) VALUES ($order_id, '$user_id', '$f_name', '$email', '$address', '$city', '$state', '$zip', '$cardname', '$cardnumberstr', '$expdate', '$total_count', '$prod_total', '$cvv)";`
- b) `$sql = "SELECT * FROM user_info WHERE user_id='$_SESSION[uid]'";`
- c) `$sql = "SELECT product_id FROM products WHERE product_title='$item_name_'";`

HOME ACTION.PHP

- a) `$sql = "SELECT COUNT(*) AS count_items FROM products,categories WHERE product_cat=cat_id";`
- b) `$sql = "SELECT * FROM products";`
- c) `$sql = "SELECT * FROM products,categories WHERE product_cat = '$id' AND product_cat=cat_id";`
- d) `$sql = "SELECT * FROM products,categories WHERE product_cat=cat_id AND product_keywords LIKE '%$keyword%'";`

LOGIN.PHP

- a) `$sql = "SELECT * FROM user_info WHERE email = '$email' AND password = '$password'";`
- b) `$sql = "SELECT * FROM admin_info WHERE admin_email = '$email' AND admin_password = '$password'";`

PAYMENT_SUCCESS.PHP

- a) `$sql = "SELECT p_id,qty FROM cart WHERE user_id = '$cm_user_id'";`
- b) `$sql = "INSERT INTO orders (user_id,product_id,qty,trx_id,p_status) VALUES ('$cm_user_id','.$product_id[$i].','.$qty[$i].','$trx_id','$p_st')";`
- c) `$sql = "DELETE FROM cart WHERE user_id = '$cm_user_id'";`

PRODUCT.PHP

- a) `$sql = " SELECT * FROM products ";`
- b) `$sql = " SELECT * FROM products WHERE product_id = $product_id";`

ADDPRODUCT.PHP

- a) `mysqli_query($con,"insert into products (product_cat, product_brand,product_title,product_price, product_desc, product_image,product_keywords) values ('$product_type','$brand','$product_name','$price','$details','$pic_name','$tags')") or die ("query incorrect");`

ADDUSER.PHP

- a) `mysqli_query($con,"insert into user_info(first_name, last_name,email,password,mobile,address1,address2) values ('$first_name','$last_name','$email','$user_password','$mobile','$address1','$address2')") or die ("Query 1 is inncorrect.....");`

EDITUSER.PHP

- a) `mysqli_query($con,"select user_id,first_name,last_name, email, password from user_info where user_id='$user_id')or die ("query 1 incorrect.....");`
- b) `mysqli_query($con,"update user_info set first_name='$first_name', last_name='$last_name', email='$email', password='$user_password' where user_id='$user_id')or die("Query 2 is inncorrect.....");`

INDEX.PHP

- a) `$result=mysqli_query($con,"select * from user_info")or die ("query 1 incorrect.....");`
- b) `mysqli_query($con,"select * from categories")or die ("query 1 incorrect.....");`
- c) `$sql = "SELECT COUNT(*) AS count_items FROM products WHERE product_cat=$i";`
- d) `mysqli_query($con,"select * from brands")or die ("query 1 incorrect.....");`
- e) `$sql = "SELECT COUNT(*) AS count_items FROM products WHERE product_brand=$i";`
- f) `mysqli_query($con,"select * from email_info")or die ("query 1 incorrect.....");`

MANAGEUSER.PHP

- a) `mysqli_query($con,"delete from user_info where user_id='$user_id')or die("query is incorrect...");`
- b) `mysqli_query($con,"select user_id, email, password from user_info")or die ("query 2 incorrect.....");`

ORDERS.PHP

- a) `mysqli_query($con,"delete from orders where order_id='$order_id')or die("delete query is incorrect...");`
- b) `mysqli_query($con,"select f_name, address, f_name, email, address, total_amt from orders_info Limit $page1,10")or die ("query 1 incorrect.....");`

PRODUCTLIST.PHP

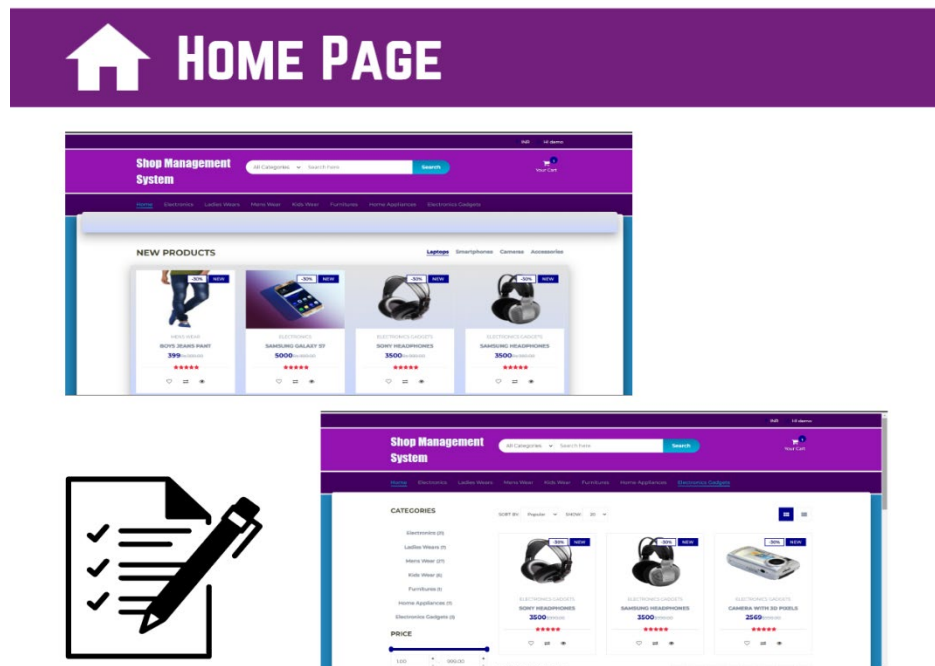
- a) `mysqli_query($con,"select product_image from products where product_id='$product_id')or die("query is incorrect...");`
- b) `mysqli_query($con,"delete from products where product_id='$product_id')or die("query is incorrect...");`
- c) `mysqli_query($con,"select product_id,product_image, product_title,product_price from products where product_cat=2 or product_cat=3 or product_cat=4 Limit $page1,12")or die ("query 1 incorrect.....");`
- d) `mysqli_query($con,"select product_id,product_image, product_title,product_price from products");`

Some Test Cases

S NO	TEST RESULTS	EXPECTED RESULTS	OBSERVED RESULTS	REMARKS
1	Insert a record	New tuple should be inserted	Query OK 1 row affected or inserted	PASS
2	Search a record	Display the Record	Return the requested record	TRUE
3	Update a record	Update the particular record in the table	Update successful	PASS
4	Delete a record	Tuple removed from database	Query OK record deleted	PASS

SNAPSHOTS

i. Home Page

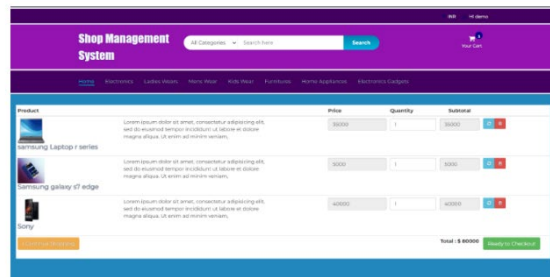
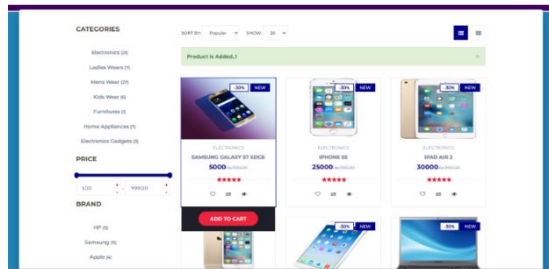


ii. Login Page



iii. Cart

CART (ADDING PRODUCTS TO CART)



iv. Checkout

CHECKOUT

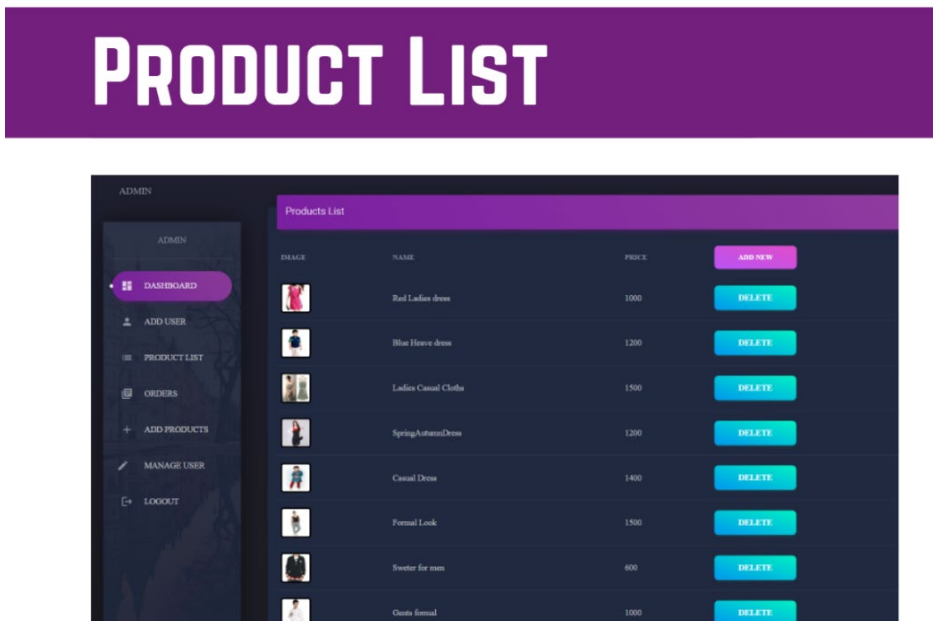


Billing Address Full Name demo demo Email demo@gmail.com Address delhi City delhi State Delhi Zip 110001 <input checked="" type="checkbox"/> Shipping address same as billing Continue to checkout	Payment Accepted Cards Name on Card Demo Card Number 1234567890101010 Exp Date 10/24 CVV 653	Cart no product title qty amount 1 samsung Laptop r series 2 35000 2 Samsung galaxy s7 edge 1 5000 3 Sony 1 40000 total Rs.80000
---	--	--

v. Admin Login

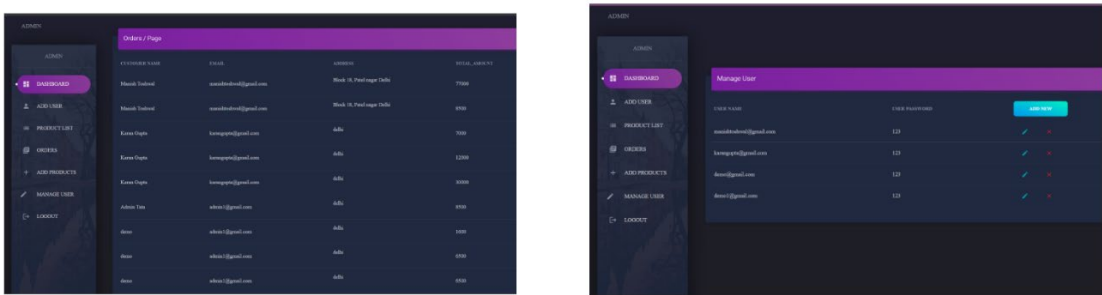


vi. Product List



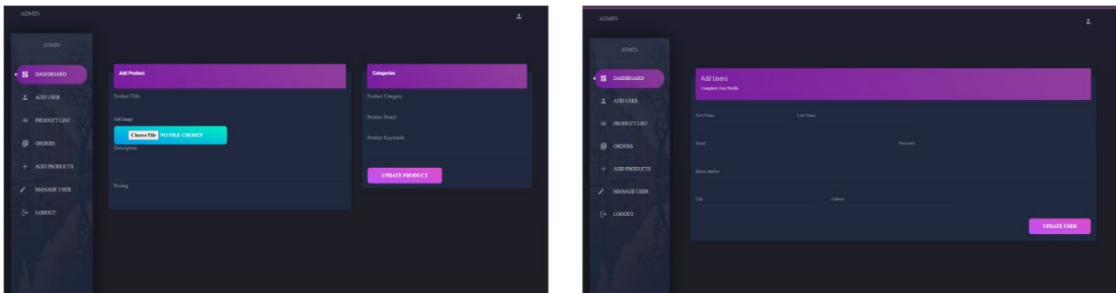
vii. Orders & User Manager

ORDERS & USER MANAGER



viii. Add Products and Users

ADD PRODUCTS & USERS



LIMITATIONS

Although I have tried to do the best and try to do all the things that are possible in an Online System, but still the system contains some of the limitations. The reason of these limitations is the time constraints. Time is the major problem. We have to deliver the project in a particular time period. That's why we have to leave some of the topics that actually we want to cover, we are still working on this software and my next goal is to remove these limitations and develop a more efficient and elegant system.

Limitations of the System:

- This project does not give the information about the stock (quantity) present within the shop.
- This project does not create monthly, yearly Reports.

After removing these and other minor limitations we hope this project will very efficient and effective.

CONCLUSION

The E-commerce Database Management System is created by taking into account all of the requirements of today's market. Providing ordinary citizens with the ability to buy and sell the best of products from anywhere in the globe in the ever-changing environment of the traditional marketplace, while also providing admin with all the essential controls for the website's smooth operation. The website's design is well-thought-out and gives the best of speed and appearance, as well as quick access to all of the platform's functions. The MySQL database, together with Javascript and PHP, provides unrivalled speed while keeping the project simple and easy to understand. This application has been thoroughly tested using appropriate test cases. It is user-friendly and includes options that are appropriate for users. This project's front-end was built with a combination of HTML, CSS, and JS. The database is stored and managed using MySQL.

References

- <https://www.javatpoint.com/dbms-tutorial>
- <https://medium.com/linkit-intecs/fundamentals-of-database-981c133c02b1>
- [Fundamentals of database systems \(Ramez Elmsari,Shamkant B.Navathe\)](#)
- [Database System Concepts \(Avi Silberschatz · Henry F.Korth · S. Sudarshan\)](#)
- <https://www.zend.com/training/php-fundamentals-1>
- <https://www.w3schools.com/php/>
- <https://www.guru99.com/php-tutorials.html>
- <https://javascript.info/first-steps>
- <https://code.tutsplus.com/courses/javascript-fundamentals>
- <https://www.dataquest.io/blog/sql-fundamentals/>
- https://www.w3schools.com/sql/sql_intro.asp