
**A PROJECT REPORT**

On

# SHOP MANAGEMENT SYSTEM

Submitted By

MOHAMMED ARSH KHAN 228R1A1294

Under the Guidance of

**Mrs. T.SWATHI**

Assistant Professor, Department of IT

**DEPARTMENT OF INFORMATION TECHNOLOGY**

**CMR ENGINEERING COLLEGE**

**UGC AUTONOMOUS**

(Approved by AICTE-New Delhi & J.N.T.U, Hyderabad)
Kandlakoya(v),Medchal Road,Hyderabad-501 401,Telangana State, India

# TABLE OF CONTENTS

# ABSTRACT

The Shop Management System (SMS) is a comprehensive software application designed to streamline and enhance the administrative and operational functions of retail shops. It aims to improve the efficiency and effectiveness of retail operations by integrating various essential processes within a shop environment. This system encompasses the management of product information, sales transactions, inventory, billing, and reporting. The SMS provides a centralized platform where product data is securely stored and easily accessible, enabling shop owners to manage their business effectively. By automating sales transactions, the system reduces the administrative burden on staff and minimizes the risk of errors. Inventory management ensures that products are adequately stocked, preventing shortages and overstock situations. Additionally, the SMS includes robust reporting features that generate various reports such as daily sales summaries, financial statements, and inventory reports.

# INTRODUCTION

The Shop Management System (SMS) is a critical technological advancement designed to streamline the administrative and operational functions within a retail shop. As retail environments become increasingly complex, the need for an integrated system to manage product data, sales transactions, inventory, billing, and reporting has never been more apparent. The SMS addresses these challenges by providing a centralized platform that enhances efficiency, reduces errors, and improves the overall quality of shop operations. Traditionally, these tasks have been managed through manual processes, which are prone to errors, inefficiencies, and delays. The advent of digital technologies has paved the way for automated systems that can handle these tasks more effectively. The SMS is one such solution designed to automate and integrate the myriad processes involved in shop management. These reports provide valuable insights for shop management, aiding in strategic decision-making and operational planning. Overall, the Shop Management System enhances the quality of retail services by improving data accuracy, reducing administrative workload, and ensuring efficient resource management. Its user-friendly interface and comprehensive functionalities make it an indispensable tool for modern retail operations, contributing to improved business performance.

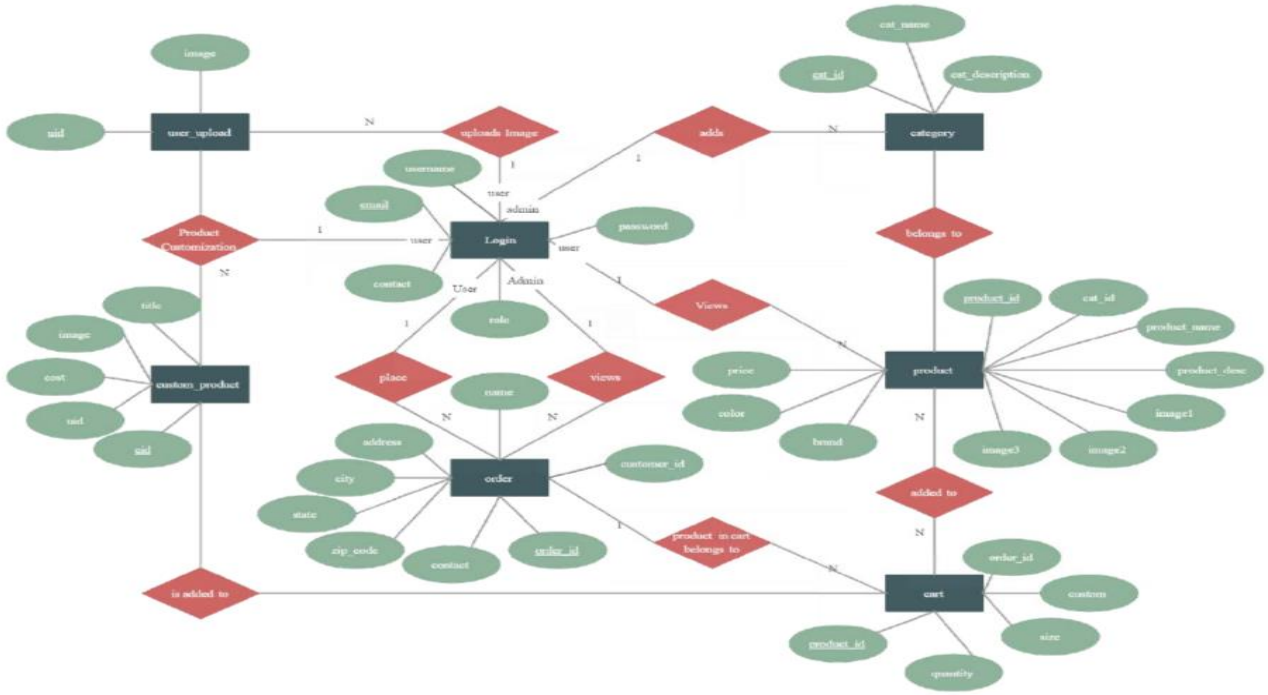
# SOFTWARE REQUIREMENS

This software package is developed using HTML and Bootstrap for the front end, and PHP and MySQL Server for the backend. For backend management, we are using XAMPP server.

**Operating System:** Windows 7, 8, 9, 10
**Language:** HTML, CSS, JavaScript, PHP, SQL
**Database:** MS SQL Server (backend)

# ER DIAGRAM

ER Diagram is known as Entity-Relationship Diagram; it is used to analyze the structure of the Database. It shows relationships between entities and their attributes. An ER Model provides a means of communication.




Entities:

Product:

Attributes: productID (Primary Key), name, category, price, quantity

Customer:

Attributes: customerID (Primary Key), name, contactDetails

Sales:

Attributes: salesID (Primary Key), productID (Foreign Key), customerID (Foreign Key), salesDate, quantity, totalPrice

Relationships:

Product – Sales (One-to-Many): One product can have multiple sales.

Customer – Sales (One-to-Many): One customer can have multiple sales.

## TABLES

```
CREATE DATABASE shop_management;
USE shop_management;
CREATE TABLE products (
  id INT PRIMARY KEY AUTO_INCREMENT,
  name VARCHAR(255) NOT NULL,
  category VARCHAR(100) NOT NULL,
  price DECIMAL(10, 2) NOT NULL,
  quantity INT NOT NULL
);
CREATE TABLE customers (
  id INT PRIMARY KEY AUTO_INCREMENT,
  name VARCHAR(255) NOT NULL,
  contactDetails TEXT
);
CREATE TABLE sales (
  id INT PRIMARY KEY AUTO_INCREMENT,
  productID INT,
  customerID INT,
  salesDate DATE NOT NULL,
  quantity INT NOT NULL,
  totalPrice DECIMAL(10, 2) NOT NULL,
  FOREIGN KEY (productID) REFERENCES products(id),
  FOREIGN KEY (customerID) REFERENCES customers(id)
);
INSERT INTO products (name, category, price, quantity) VALUES
('Product A', 'Category 1', 100.00, 50),
('Product B', 'Category 2', 150.00, 30);
INSERT INTO customers (name, contactDetails) VALUES
('Customer A', '123 Main St, Cityville'),
('Customer B', '456 Oak St, Townsville');
```

Here are the tables in image format:

## Products Table

| Field | Type | Null | Key | Default | Extra |
|---|---|---|---|---|---|
| productID | int(11) | NO | PRI | NULL | auto_increment |
| name | varchar(255) | NO | | NULL | |
| category | varchar(255) | NO | | NULL | |
| price | decimal(10,2) | NO | | NULL | |
| quantity | int(11) | NO | | NULL | |

## Customers Table

| Field | Type | Null | Key | Default | Extra |
|---|---|---|---|---|---|
| customerID | int(11) | NO | PRI | NULL | auto_increment |
| name | varchar(255) | NO | | NULL | |
| contactDetails | text | NO | | NULL | |

## Sales Table

| Field | Type | Null | Key | Default | Extra |
|---|---|---|---|---|---|
| salesID | int(11) | NO | PRI | NULL | auto_increment |
| productID | int(11) | NO | MUL | NULL | |
| customerID | int(11) | NO | MUL | NULL | |
| salesDate | date | NO | | NULL | |
| quantity | int(11) | NO | | NULL | |
| totalPrice | decimal(10,2) | NO | | NULL | |

## SOURCE CODE

**add_product.php:**

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Shop Management System</title>
  <style>
    body {
      background-color: #f0f0f0;
      font-family: Arial, sans-serif;
      margin: 0;
      padding: 0;
      text-align: center;
    }
    h1 {
      background-color: #4CAF50;
      color: white;
      padding: 20px;
    }
    form {
      margin: 20px auto;
      padding: 20px;
      background-color: white;
      border-radius: 10px;
      width: 300px;
      box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
    }
    input, select, textarea {
      width: 100%;
      padding: 10px;
      margin: 5px 0;
      box-sizing: border-box;
    }
    input[type="submit"] {
      background-color: #4CAF50;
```

```
    color: white;
    border: none;
    cursor: pointer;
   }
   input[type="submit"]:hover {
    background-color: #45a049;
   }
  </style>
</head>
<body>
  <h1>Shop Management System</h1>
  <form action="add_product.php" method="post">
   <label for="name">Product Name:</label>
   <input type="text" id="name" name="name" required>
   <label for="category">Category:</label>
   <input type="text" id="category" name="category" required>
   <label for="price">Price:</label>
   <input type="number" id="price" name="price" required>
   <label for="quantity">Quantity:</label>
   <input type="number" id="quantity" name="quantity" required>
   <input type="submit" value="Add Product">
  </form>
</body>
</html>
```

**index.php:**

```
<?php
if ($_SERVER['REQUEST_METHOD'] === 'POST') {
  $name = $_POST['name'];
  $category = $_POST['category'];
  $price = $_POST['price'];
  $quantity = $_POST['quantity'];

  $conn = new mysqli('localhost', 'root', '', 'shop_management');
  if ($conn->connect_error) {
   die("Connection failed: " . $conn->connect_error);
```

```
  }

  $sql = "INSERT INTO products (name, category, price, quantity) VALUES ('$name', '$category', $price, $quantity)";
  if ($conn->query($sql) === TRUE) {
    echo "Product added successfully!";
  } else {
    echo "Error: " . $sql . "<br>" . $conn->error;
  }

  $conn->close();
}
?>

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Add Product</title>
</head>
<body>
  <h2>Add Product</h2>
  <form method="post" action="">
    <label for="name">Product Name:</label>
    <input type="text" id="name" name="name" required>
    <label for="category">Category:</label>
    <input type="text" id="category" name="category" required>
    <label for="price">Price:</label>
    <input type="number" id="price" name="price" required>
    <label for="quantity">Quantity:</label>
    <input type="number" id="quantity" name="quantity" required>
    <input type="submit" value="Add Product">
  </form>
</body>
```

```
</html>
```

**view_products.php**:

```
<?php
$conn = new mysqli('localhost', 'root', '', 'shop_management');
if ($conn->connect_error) {
  die("Connection failed: " . $conn->connect_error);
}

$sql = "SELECT * FROM products";
$result = $conn->query($sql);
?>
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>View Products</title>
  <style>
    body {
      font-family: Arial, sans-serif;
      background-color: #f4f4f4;
      margin: 0;
      padding: 0;
      text-align: center;
    }
    h2 {
      background-color: #4CAF50;
      color: white;
      padding: 20px;
      margin: 0;
    }
    table {
      margin: 20px auto;
      border-collapse: collapse;
      width: 80%;
```

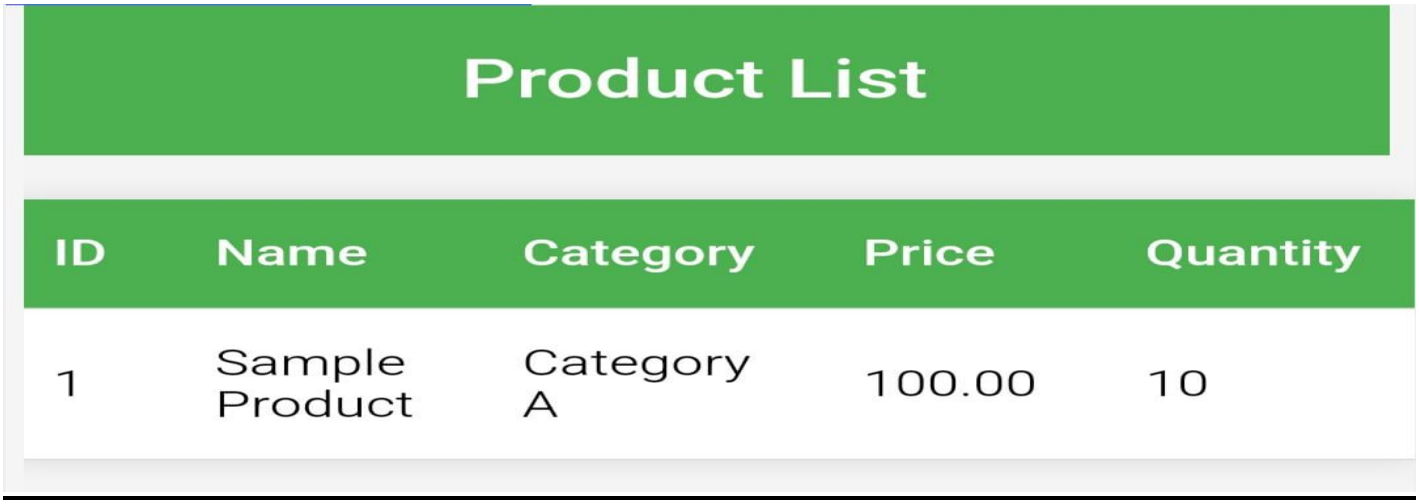
```
    background-color: white;
    box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
   }
   th, td {
    padding: 15px;
    text-align: left;
    border-bottom: 1px solid #ddd;
   }
   th {
    background-color: #4CAF50;
    color: white;
   }
  </style>
</head>
<body>
  <h2>Product List</h2>
  <table>
   <tr>
    <th>ID</th>
    <th>Name</th>
    <th>Category</th>
    <th>Price</th>
    <th>Quantity</th>
   </tr>
   <?php
   if ($result->num_rows > 0) {
    while($row = $result->fetch_assoc()) {
     echo "<tr>
          <td>" . $row["id"]. "</td>
          <td>" . $row["name"]. "</td>
          <td>" . $row["category"]. "</td>
          <td>" . $row["price"]. "</td>
          <td>" . $row["quantity"]. "</td>
         </tr>";
    }
```

```
    } else {
      echo "<tr><td colspan='5'>No products available</td></tr>";
    }
    $conn->close();
    ?>
  </table>
</body>
</html>
```

## OUTPUT

Add Product Page: Interface for adding a new product with fields for name, category, price, and quantity.

View Products Page: Interface displaying a list of products in a tabular format with columns for ID, name, category, price, and quantity.


## CONCLUSION

The Shop Management System is an essential tool for modern retail businesses, providing streamlined operations and enhanced data management capabilities. By integrating product management, sales tracking, inventory control, and reporting into a single platform, this system improves efficiency, accuracy, and decision-making for shop owners. The automated processes reduce manual errors, save

time, and allow for better resource allocation. The use of HTML, PHP, and MySQL ensures that the system is robust, scalable, and easy to maintain. Overall, the Shop Management System significantly contributes to the effective management and growth of retail businesses.

## REFERENCES

HTML Documentation: W3Schools HTML

PHP Manual: PHP.net

MySQL Document