

CAR ACCESSORIES SHOP MANAGEMENT SYSTEM

*A project report submitted to
Bishop Heber College (Autonomous), Tiruchirappalli
affiliated to Bharathidasan University, Tiruchirappalli – 620024
in partial fulfillment of the requirements for the award of the degree of*

Bachelor of Vocation in Information Technology

By
S. MOHANRAJ
(REGISTER NO: 225915135)

Under the guidance of
Mrs. P. USHA, MCA, M.Phil., SET., NET.,



Department of Information Technology

Bishop Heber College(Autonomous)

(Nationally Re-accredited with 'A++' Grade by NAAC with a CGPA of 3.69 out of 4)

(Recognized by UGC as "College of Excellence")

Tiruchirappalli 620 017

NOVEMBER 2024



Department of Information Technology
Bishop Heber College(Autonomous)
Tiruchirappalli-620 017,Tamilnadu,India
Phone No:0431-277 0136

CERTIFICATE

This Viva-Voce examination for the candidate **S.MOHANRAJ**
(Reg No:225915135) was held on_____

Signature of HOD

Examiners:

- 1.
- 2.

Mrs. P.USHA, MCA .,M. Phil.,SET.,NET..

Assistant Professor,

Department of Information Technology,

Bishop Heber College (Autonomous),

Tiruchirapalli-620 017



Date:

CERTIFICATE

This is to certify that the project work entitled “**CAR ACCESSORIES SHOP MANAGEMENT SYSTEM**” is a bonafide work done under my supervision by **S.MOHANRAJ (Reg No:225915135)** and submitted to Bishop Heber College (Autonomous), Tiruchirappalli – 620 017 in partial fulfillment of the requirements for the award of the degree of Bachelor of Vocation in Information Technology during the odd semester of the academic year (2024-2025).

Signature of the Guide

DECLARATION

I hereby declare that the work presented in this project work report is the original work done by me under the guidance of **Mrs. P.USHA, MCA., M.Phil., SET., NET.**, Assistant Professor of Information Technology, Bishop Heber College (Autonomous), Tiruchirapalli-620 017 and has not been included in any other project work submitted for any other degree.

Name of the Candidate :S.MOHANRAJ

Register Number :225915135

Semester :FIFTH

Academic Year :2024 – 2025

Course Code :U21ITPJ5

Signature of the Candidate

ACKNOWLEDGEMENTS

First of foremost, I thank **THE ALMIGHTY GOD** for granting abundant grace. Good health and knowledge to do this Project.

I express my sincere gratitude to **Dr.J.PRINCY MERLIN, M.Sc., SET., B.Ed., M.Phil., Ph.D., PGDCL.**, Principal., Bishop Heber College (Autonomous), Tiruchirappalli for providing the opportunity to pursue this programme.

I am highly indebted to thanks **Dr. J. JOHN RAYBIN JOSE, M.sc.,MCA, M.Phill, PGDCA ,Ph.D, SET.** Associate professor and Head , Department of Information Technology , Bishop Heber College (Autonomous), Trichy for providing all the support and facilities to do this project work.

I wish to place on record my gratitude to **Mrs. P.USHA, MCA., M.Phil., SET., NET.,** Assistant Professor in Department of Information Technology, Bishop Heber College (Autonomous), Tiruchirappalli for grant me permission to pursue carry on with this project and guiding me to complete the project successfully.

I thank all the Staff members of the Department of Information Technology for their contribution to complete this project work successfully.

I record my deep sense of gratitude to my beloved parents and my friends for their encouragement and moral support extended during the period of my project.

S.MOHANRAJ

ABSTRACT

The **Car Accessories Shop Management System** is a PHP-based web application with SQL as the database, designed to simplify the management of a car accessories store. It allows the admin to manage products, track inventory, and view customer orders, while providing customers with a smooth shopping experience. The admin can add, update, and delete products, monitor stock levels, and generate sales reports for better business insights. Customers can register, log in, browse car accessories, add items to their cart, and place orders. The system maintains a detailed order history, allowing customers to track past and ongoing purchases. The database includes tables for users, products, orders, and order items, ensuring efficient data organization and retrieval. The system is scalable and easy to modify, making it adaptable to the evolving needs of the store.

CONTENTS

S. NO.	TITLE	PAGE NO.
1	INTRODUCTION	1
2	SYSTEM STUDY	2
	2.1. Project Description	2
	2.1.1. Existing System	2
	2.1.2. Proposed System	3
	2.1.3 Modules and Description	4
	2.2 Requirement Specification	6
	2.2.1 Hardware Requirements	6
	2.2.2 Software Requirements	6
3.	SYSTEM DESIGN	8
	3.1. Logical Design	8
	3.2 Database Design	9
4.	SYSTEM DEVELOPMENT	11
	4.1. Program Code	11
5.	SYSTEM TESTING	22
	5.1. Unit testing	22
	5.2.Integration testing	23
	5.3 Validation Testing	24
6.	SYSTEM IMPLEMENTATION	25
7.	CONCLUSION	33
	BIBLIOGRAPHY	34

1.INTRODUCTION

The Car Accessories Shop Management System is an application designed for maintaining and managing various aspects of a car accessories store. This project covers the basic functionality of managing inventory, sales, and customer orders within a car accessories shop environment. Car accessory shops play a critical role in the automotive industry by providing a wide range of parts and accessories to enhance the performance, comfort, and aesthetics of vehicles. These shops also offer services such as product installations, maintenance, and customization for vehicles.

The goal of this project is to develop a system that solves the operational needs of a car accessories store, ensuring efficient management of products and customer demands. The system should provide various ways to perform tasks such as inventory tracking, order management, sales processing, and customer service. Additionally, it should enhance the store's workspace with extra functionalities that go beyond the scope of a traditional management system, such as customer feedback tracking and product recommendations.

There are different types of car accessory stores, including those specializing in performance upgrades, aesthetic modifications, and essential maintenance parts. Some stores may focus on performance accessories, offering parts to improve speed, handling, and overall vehicle dynamics. Others may focus on aesthetic enhancements such as custom wheels, lighting, or body kits. There are also general car accessory shops that provide a broad range of products, including basic items like seat covers, air fresheners, and car care products. This system will cater to the specific needs of these different types of car accessory shops, ensuring smooth operations and improved customer satisfaction.

2.SYSTEM STUDY

System analysis is a process of gathering the facts concerning the system, breaking them into elements, and understanding the relationships between these elements. It provides a framework for visualizing the organizational and environmental factors that operate on a system. The quality of work performed by a machine is usually uniform, neat, and more reliable when compared to performing the same operations manually.

2.1.Project Description

The scope of the project is to store and access the database consisting of customer details, product inventories, and transaction records. This data can be shared with the concerned departments, such as sales, inventory management, and customer service, to streamline operations. The Car Accessories Shop Management System is an application that manages information on customer orders, available products, and transactions, typically for small to medium-sized car accessory shops.

2.1.1.Existing System

The existing system operates manually. It involves a lot of complexity and requires significant human effort, including paperwork and manual tracking of sales, inventory, and customer data. All data must be recorded on physical ledgers, making it a tedious and risky process. As the number of transactions increases, so does the volume of data, making data management more challenging. To retrieve any information, an extensive manual search through papers is required, making it time-consuming and inefficient.

2.1.2.Proposed System.

The proposed system is designed to provide the best services to both the car accessory shop and its customers by being user-friendly and reducing the time it takes to complete tasks compared to the current manual system. The new system is highly computerized, ensuring that data related to customer orders, product inventories, and sales are securely stored and managed with high accuracy. This reduces errors caused by human mistakes or machine malfunctions. The system also validates data as it is entered, ensuring data integrity. When necessary, appropriate messages are displayed to prevent user confusion. The data entry screen offers features for viewing records and modifying various types of data as needed.

Advantages:

1. It satisfies the user's requirements.
2. The system generates various types of reports and information for different purposes (sales, inventory, etc.).
3. It is easy for both users and operators to understand.
4. It is simple to operate.
5. The system is expandable, allowing for future upgrades or the addition of new features.

2.1.3 Modules and Description

Car Accessories Shop Management System is the add record and delete record to all details.

❖ **Register Account :**

This module is use to create an account with personal information for easier access to the shop's services.

❖ **View Transaction Details :**

This module is use to view all transaction details.

❖ **Manage Inventory :**

Enables the administrator to add, update, or remove products from the inventory, manage stock levels, and set prices.

❖ **Manage Orders :**

Administrator to view, approve, or cancel customer orders, and track order statuses.

❖ **View customer Details :**

This module is use to view all customer data.

❖ **View Sales Report :**

This module is use to displays sales reports, helping administrators analyze sales trends and business performance.

❖ **Manage Service Bookings :**

This module is use to view, schedule, and manage customer service bookings, including confirmations and cancellations.

❖ **View Profile :**

This module is use to customers to view and edit their account details, such as name, email, and contact information.

❖ **Add Product to Cart :**

This module is use to add selected products to their shopping cart for future purchase.

❖ **Checkout :**

This module is use to process for customers to review their cart, select payment options, and place orders.

❖ **View Delivery Status :**

customers to track the delivery status of their purchased products, including shipment and estimated delivery time.

❖ **Book Service :**

This module is use to schedule and book services, such as car maintenance or installation of accessories.

2.2. Requirement Specification

Requirements specification involves frequent communication with system users to determine specific feature expectations, resolve conflicts or ambiguities, and ensure that the system meets the needs of its users. The goal is to ensure the system or product conforms to the client's needs rather than forcing users to adapt to predefined requirements. Requirements analysis is a team effort that demands a combination of hardware, software, and human factors engineering, along with strong communication and interpersonal skills.

2.2.1. Hardware Requirement

The hardware specification of the computer system required for developing the **Car Accessories Shop Management System** is as follows:

- **Processor:** Intel Core i3
- **Hard Disk:** 500 GB
- **RAM:** 8 GB
- **Keyboard:** Standard Keyboard
- **Mouse:** Standard Mouse Pad

2.2.2. Software Requirement

A **Software Requirement Specification** is a detailed description of the system's behavior. It includes the use cases that describe all user interactions with the system. The software tools used for the **Car Accessories Shop Management System** are as follows:

- **Operating System:** Windows 10
- **Software Applications:** WAMP
- **Front End:** PHP, HTML, CSS, JavaScript
- **Back End:** MySQL Database

Operating System

The **Operating System** manages the communication between hardware and software, allowing the system to function. This project will use **Windows 10** , which is widely supported and ideal for running the required software stack.

Development Tools

- **PHP:** PHP (Hypertext Preprocessor) is a widely-used, open-source scripting language that is especially suited for web development and can be embedded in HTML. PHP is server-side and allows the creation of dynamic content and interaction with the database.
- **HTML/CSS:** HTML (HyperText Markup Language) and CSS (Cascading Style Sheets) are used for structuring and designing the web pages of the system. HTML creates the structure, while CSS is responsible for the layout, design, and responsiveness.
- **JavaScript:** JavaScript is a client-side scripting language used to add dynamic elements, interactivity, and enhanced functionality to the web pages. It enables features such as form validation, dynamic content updates, and user interaction with the interface without reloading the page.

Database:

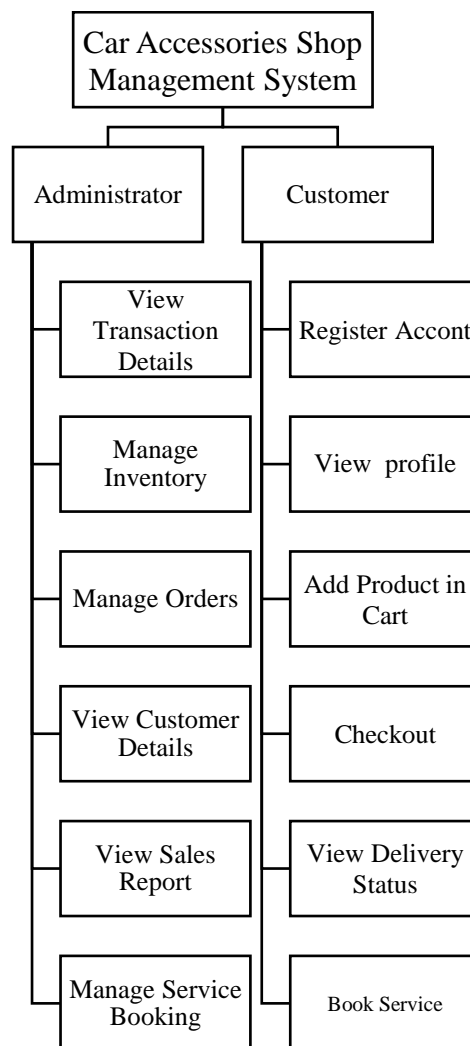
- **MySQL:** MySQL is an open-source relational database management system that will be used to manage all data related to customers, products, orders, and transactions. MySQL is known for its reliability, ease of use, and support for high transaction loads.

3.SYSTEM DESIGN

System design is the process of defining the architecture, components, modules, interfaces, and data for a system to satisfy specified requirements. System design could be seen as the application of system theory to product development.

3.1. Logical Design

Logical design is an abstract concept in computer programming by which programmers arrange data in a series of logical relationships known as attributes or entities. An entity refers to a chunk of information, whereas an attribute defines the unique properties of an entity.



• Fig.3.1 Car Accessories Shop Management System

3.2 Database Design

Database design is the process of producing a detailed data model of the database. This data model contains all the needed logical and physical design choices and physical storage parameters needed to generate a design in a data definition language, which can then be used to create a database. A fully attributed data model contains detailed attributes for each entity. A good database design is important in ensuring consistent data, elimination of data redundancy, efficient execution of queries and high-performance application.

Users Table :

Field	Data Type	Null
Id	Int	No
Username	Varchar(255)	No
Email	Varchar(255)	No
Password	Varchar(255)	No
Image	Longblob	No
Address	Varchar(255)	No
Phone No	Int	No
Date of Birth	Date	No
Created_at	Timestamp	No

Products Table :

Field	Data Type	Null
Id	Int	No
Name	Varchar(255)	No
Category	Varchar(255)	No
Price	Decimal(10,2)	No
Image	Longblob	No
Original_Price	Decimal(10,2)	No
Discount_Percentage	Int	No
Description	Varchar(255)	No
Sold_By	Varchar(255)	No
Total_Products	Int	No
Warranty	Varchar(255)	No

Cart Table :

Field	Data Type	Null
Id	Int	No
User_Id	Int	No
Product_Id	Int	No
Quantity	Int	No

Orders Table :

Field	Data Type	Null
Order_Id	Int	No
User_Id	Int	No
Order_Date	Datetime	No
Delivery_Status	Varchar(255)	No
Total_Amount	Decimal(10,2)	No
Payment_Method	Varchar(255)	No
Payment_Status	Varchar(255)	No
Shipping_Address	Varchar(255)	No
Billing_Address	Varchar(255)	No
Created_At	Timestamp	No
Updated_At	Timestamp	No
Customer_Name	Varchar(255)	No

Service_Booking Table :

Field	Data Type	Null
Id	Int	No
User_Id	Int	No
Service	Varchar(255)	No
Booking_date	Date	No

Payment Table :

Field	Data Type	Null
Id	Int	No
User_Id	Int	No
Product_Id	Int	No
Amount	Decimal(10,2)	No
Payment_Status	Varchar(255)	No

4.SYSTEM DEVELOPMENT

The Software Development Life Cycle(SDLC), also referred to as the application development life-cycle, is a term used in system engineering, information system and software engineering to describe a process for planning, creating testing and deploying an information system. The system developments life-cycle concept applies to a range of hardware and software configurations, as a system can be composed of hardware only, software only, or a combination of both.

4.1 Program Code

Here I displayed the coding of my program:

db.php:

```
<?php
$servername = "localhost";
$username = "root";
$password = "";
$dbname = "car_accessories";

$conn = new mysqli($servername, $username, $password, $dbname);
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);}?>
```

Login.php:

```
<?php
session_start();
require './db.php';
if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $email = $_POST['email'];
    $password = $_POST['password'];
```

```

$sql = "SELECT * FROM users WHERE email = ? LIMIT 1";
$stmt = $conn->prepare($sql);
$stmt->bind_param('s', $email);
$stmt->execute();
$result = $stmt->get_result();
if ($result->num_rows > 0) {
    $user = $result->fetch_assoc();
    if (password_verify($password, $user['password'])) {
        $_SESSION['email'] = $user['email'];
        $_SESSION['user_id'] = $user['id'];
        $_SESSION['loggedin'] = true;
        if ($email === 'admin@gmail.com') {
            header("Location: ../Admin/Dashboard.php");
        } else {
            header("Location: ../index.html"); }exit;
        } else {
            echo "Invalid password.";}
    } else {
        echo "No user found with that email address."; }
    $stmt->close();}
$conn->close();?>

```

Add_Product.php:

```

<?php
require '../db.php';
if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $name = isset($_POST['name']) ? mysqli_real_escape_string($conn,
$_POST['name']) : "";
    $category = isset($_POST['category']) ? mysqli_real_escape_string($conn,
$_POST['category']) : "";
    $price = isset($_POST['price'])?mysqli_real_escape_string($conn, $_POST['price']) : "";

```

```

$rating=isset($_POST['rating'])?mysqli_real_escape_string($conn,$_POST['rating']);

$original_price=isset($_POST['original_price'])?mysqli_real_escape_string($conn,$_POST['original_price']) : NULL;

$discount_percentage = isset($_POST['discount_percentage']) ?
mysqli_real_escape_string($conn, $_POST['discount_percentage']) : NULL;

$description = isset($_POST['description']) ? mysqli_real_escape_string($conn,
$_POST['description']) : NULL;

$material = isset($_POST['material']) ? mysqli_real_escape_string($conn,
$_POST['material']) : NULL;

$care = isset($_POST['care']) ? mysqli_real_escape_string($conn, $_POST['care'])
: NULL;

$sold_by = isset($_POST['sold_by']) ? mysqli_real_escape_string($conn,
$_POST['sold_by']) : NULL;

$positive_feedback_percentage = isset($_POST['positive_feedback_percentage']) ?
mysqli_real_escape_string($conn, $_POST['positive_feedback_percentage']) : NULL;

$total_products = isset($_POST['total_products']) ?
mysqli_real_escape_string($conn, $_POST['total_products']) : NULL;

$warranty = isset($_POST['warranty']) ? mysqli_real_escape_string($conn,
$_POST['warranty']) : NULL;

if (isset($_FILES['image']['tmp_name'])) {
    $image = $_FILES['image']['tmp_name'];
    $imageData = addslashes(file_get_contents($image));} else {
    $imageData = NULL; }

$sql = "INSERT INTO products (name, category, price, rating, original_price,
discount_percentage, description, material, care, sold_by,
positive_feedback_percentage, total_products, warranty, image)

VALUES ('$name', '$category', '$price', '$rating', '$original_price',
'$discount_percentage', '$description', '$material', '$care', '$sold_by',
'$positive_feedback_percentage', '$total_products', '$warranty', '$imageData')";

if ($conn->query($sql) === TRUE) {
    echo "New product added successfully!";

    echo '<br><a href="add_product.php"><button>Add Another
Product</button></a>';

    echo '<br><a href="inventory.php"><button>View Inventory</button></a>';
} else {

```

```

        echo "Error: " . $sql . "<br>" . $conn->error;

        echo          '<br><a          href="inventory.php"><button>Go          TO
Inventory</button></a>';}}?>

```

Edit_Product.php:

```

<?php
require '../db.php';
if (isset($_GET['id'])) {
    $product_id = $_GET['id'];
    $sql = "SELECT * FROM products WHERE id = $product_id";
    $result = $conn->query($sql);
    if ($result->num_rows > 0) {
        $product = $result->fetch_assoc();} else {
            echo "No product found!";exit;}
    } else {
        echo "Invalid product ID!";exit;}
if ($_SERVER['REQUEST_METHOD'] == 'POST') {
    $name = $_POST['name'];
    $category = $_POST['category'];
    $price = $_POST['price'];
    $rating = $_POST['rating'];
    $original_price = $_POST['original_price'];
    $discount_percentage = $_POST['discount_percentage'];
    $description = $_POST['description'];
    $material = $_POST['material'];
    $care = $_POST['care'];
    $sold_by = $_POST['sold_by'];
    $positive_feedback_percentage = $_POST['positive_feedback_percentage'];
    $total_products = $_POST['total_products'];
    $warranty = $_POST['warranty'];
    if (!empty($_FILES['image']['name'])) {

```

```

$image = addslashes(file_get_contents($_FILES['image']['tmp_name']));
$image_sql = ", image='$image'";
} else {
    $image_sql = "";
$update_sql = "UPDATE products SET
    name='$name',
    category='$category',
    price='$price',
    rating='$rating',
    original_price='$original_price',
    discount_percentage='$discount_percentage',
    description='$description',
    material='$material',
    care='$care',
    sold_by='$sold_by',
    positive_feedback_percentage='$positive_feedback_percentage',
    total_products='$total_products',
    warranty='$warranty'
    $image_sql
    WHERE id=$product_id";
if ($conn->query($update_sql) === TRUE) {
    echo "Product updated successfully!";
    header('Location: inventory.php');
    exit;
} else {
    echo "Error updating product: " . $conn->error;}}?>

```

Add_to_Cart.php:

```

<?php
session_start();

```

```

require '../db.php';
if (!isset($_SESSION['user_id'])) {
    echo 'Please log in to add items to the cart.';
    exit;
}
$user_id = $_SESSION['user_id'];
$product_id = $_POST['product_id'];
$quantity = $_POST['quantity'];
$query = "SELECT * FROM cart_items WHERE user_id = ? AND product_id = ?";
$stmt = $conn->prepare($query);
$stmt->bind_param('ii', $user_id, $product_id);
$stmt->execute();
$result = $stmt->get_result();
if ($result->num_rows > 0) {
    $update_query = "UPDATE cart_items SET quantity = quantity + ? WHERE
user_id = ? AND product_id = ?";
    $stmt_update = $conn->prepare($update_query);
    $stmt_update->bind_param('iii', $quantity, $user_id, $product_id);
    $stmt_update->execute();
} else {
    $insert_query = "INSERT INTO cart_items (user_id, product_id, quantity)
VALUES (?, ?, ?)";
    $stmt_insert = $conn->prepare($insert_query);
    $stmt_insert->bind_param('iii', $user_id, $product_id, $quantity);
    $stmt_insert->execute();
}
header("Location: cart.php");
exit();?>

```

Checkout.php:

```
<?php

session_start();

require '../db.php';

require_once 'stripe-php-master/init.php';

\Stripe\Stripe::setApiKey('sk_test_51Q2Zi0Lzf1Nkv11Jkp5MvnrDv7FyVnDJSMMPr
s0GAPxYwGaLcbJRssSzO3ZtZXJUldwAmzOUso5QaGWfhusOeaKt00gSVTacvk')
;

$user_id = $_SESSION['user_id'];

$name = isset($_GET['name']) ? htmlspecialchars($_GET['name']) : 'N/A';

$address = isset($_GET['address']) ? htmlspecialchars($_GET['address']) : 'N/A';

$phone = isset($_GET['phone']) ? htmlspecialchars($_GET['phone']) : 'N/A';

$payment_method = isset($_GET['payment_method']) ?
htmlspecialchars($_GET['payment_method']) : 'N/A';

$sql = "SELECT products.id, products.name, products.price, cart_items.quantity
FROM cart_items
JOIN products ON cart_items.product_id = products.id
WHERE cart_items.user_id = ?";

$stmt = $conn->prepare($sql);

$stmt->bind_param("i", $user_id);

$stmt->execute();

$result = $stmt->get_result();

$total_price = 0;

$line_items = [];

while ($row = $result->fetch_assoc()) {

    $total_price += $row['price'] * $row['quantity'];

    $line_items[] = [

        'price_data' => [

            'currency' => 'inr',

            'product_data' => [

                'name' => $row['name'],
```



```

    ],
    'unit_amount' => $row['price'] * 100,
    ],
    'quantity' => $row['quantity'],];
}if ($_SERVER['REQUEST_METHOD'] === 'POST') {
    // Get shipping details from the session or POST request
    $name = $_SESSION['shipping_details']['name'];
    $address = $_SESSION['shipping_details']['address'];
    $phone = $_SESSION['shipping_details']['phone'];
    $payment_method = $_SESSION['shipping_details']['payment_method'];
    $session = \Stripe\Checkout\Session::create([
        'payment_method_types' => ['card'],
        'line_items' => $line_items,
        'mode' => 'payment',
        'success_url' => 'http://localhost/Car-
Accessories/User/checkout_success.php?session_id={CHECKOUT_SESSION_ID}' .
        '&name=' . urlencode($name) .
        '&address=' . urlencode($address) .
        '&phone=' . urlencode($phone) .
        '&payment_method=' . urlencode($payment_method), // Include the
shipping details in the success URL
        'cancel_url' => 'http://localhost/Car-Accessories/User/checkout_cancel.php', //
Change to your cancel URL]);
    header("Location: " . $session->url);
    exit();}?>

```

Checkout_Success.php:

```

<?php
session_start();
require './db.php';
require_once 'stripe-php-master/init.php';

```

```

\Stripe\Stripe::setApiKey('sk_test_51Q2Zi0Lzf1Nkv11Jkp5MvnrDv7FyVnDJSmmPr
s0GAPxYwGaLcbJRssSzO3ZtZXJUldwAmzOUso5QaGWfhusOeaKt00gSVTacvk')
;

if (!isset($_GET['session_id'])) {
    header("Location: checkout.php"); // Redirect if no session_id
    exit();}

$session_id = $_GET['session_id'];
$session = \Stripe\Checkout\Session::retrieve($session_id);
$user_id = $_SESSION['user_id'];
$name = isset($_GET['name']) ? htmlspecialchars($_GET['name']) : 'N/A';
$address = isset($_GET['address']) ? htmlspecialchars($_GET['address']) : 'N/A';
$phone = isset($_GET['phone']) ? htmlspecialchars($_GET['phone']) : 'N/A';
$payment_method = isset($_GET['payment_method']) ?
htmlspecialchars($_GET['payment_method']) : 'N/A';
$billing_address = $address;
$order_date = new DateTime();
$order_date_formatted = $order_date->format('Y-m-d H:i:s');
$delivery_status = "Processing";
$total_amount = $session->amount_total / 100;
$currency_code = $session->currency;
$txn_id = $session->id;
$payment_status = $session->payment_status;
$payment_response = json_encode($session);

$stmt = $conn->prepare("INSERT INTO tbl_payment (`user_id`, `amount`,
`currency_code`, `txn_id`, `payment_status`, `payment_response`) VALUES (?, ?, ?,
?, ?, ?)");

$stmt->bind_param("isssss", $user_id, $total_amount, $currency_code, $txn_id,
$payment_status, $payment_response);

if (!$stmt->execute()) {
    echo "Error inserting payment details: " . $stmt->error;}

$stmt1 = $conn->prepare("INSERT INTO orders(`user_id`, `customer_name`,
`order_date`, `delivery_status`, `total_amount`, `payment_method`, `payment_status`,
`shipping_address`, `billing_address`) VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?)");

```

```

$stmt1->bind_param("isssssss", $user_id, $name, $order_date_formatted,
$delivery_status, $total_amount, $payment_method, $payment_status,
$shipping_address, $billing_address);

if ($stmt1->execute()) {

    $order_id = $stmt1->insert_id;

    $cart_query = $conn->prepare("SELECT cart_items.*, products.price AS
product_price FROM cart_items

                                JOIN products ON cart_items.product_id = products.id

                                WHERE cart_items.user_id = ?");

    $cart_query->bind_param("i", $user_id);

    $cart_query->execute();

    $cart_result = $cart_query->get_result();

    $insert_item_stmt = $conn->prepare("INSERT INTO order_items (`order_id`,
`product_id`, `quantity`, `price`) VALUES (?, ?, ?, ?)");

    while ($row = $cart_result->fetch_assoc()) {

        $product_id = $row['product_id'];

        $quantity = $row['quantity'];

        $price = isset($row['price']) ? $row['price'] : $row['product_price'];

        $insert_item_stmt->bind_param("iiid", $order_id, $product_id, $quantity,
$price);

        if (!$insert_item_stmt->execute()) {

            echo "Error inserting order item: " . $insert_item_stmt->error;

        }

    }

    $insert_item_stmt->close();

    $delete_stmt = $conn->prepare("DELETE FROM cart_items WHERE user_id =
?");

    $delete_stmt->bind_param("i", $user_id);

    if (!$delete_stmt->execute()) {

        echo "Error removing items from cart: " . $delete_stmt->error; }

    $_SESSION['success_message'] = "Thank you for your purchase!";

} else {

```

```
echo "Error inserting order details: " . $stmt1->error;}?>
```

Remove_From_Cart.php:

```
<?php
session_start();
require'../db.php';
if (!isset($_SESSION['user_id'])) {
    $_SESSION['error_message'] = "You must be logged in to remove items from the
    cart.";
    header("Location: cart.php");
    exit;}
$user_id = $_SESSION['user_id'];
$product_id = isset($_POST['product_id']) ? intval($_POST['product_id']) : 0;
$sql = "DELETE FROM cart_items WHERE user_id = ? AND product_id = ?";
$stmt = $conn->prepare($sql);
$stmt->bind_param("ii", $user_id, $product_id);
if ($stmt->execute()) {
    $_SESSION['success_message'] = "Item removed from cart successfully.";
} else {
    $_SESSION['error_message'] = "Error removing item from cart. Please try again.";
}
$stmt->close();
$conn->close();
header("Location: cart.php");
exit;
?>
```

5.SYSTEM TESTING

System testing is the process of evaluation and software item to detect differences between given input and expected output. Testing assesses the quality of the product. Software testing is a process that should be done during the development process. In other words, software testing is a verification and validation process.

5.1. Unit Testing:

Testing of individual software components or modules. Typically done by the programmer and not by testers, as it requires detailed knowledge of the internal program design and code. may require developing test driver modules or test harnesses.

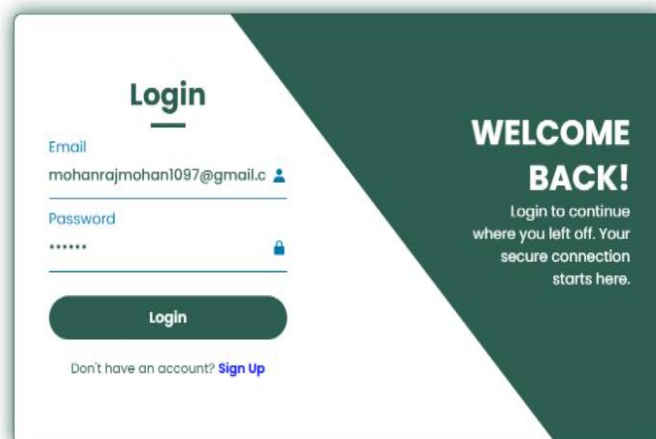
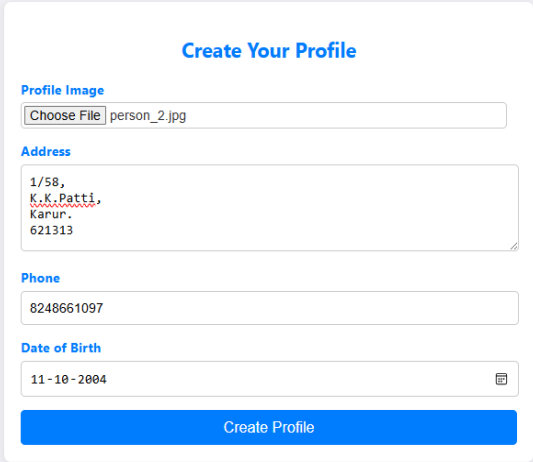


Fig:5.1 Login Page

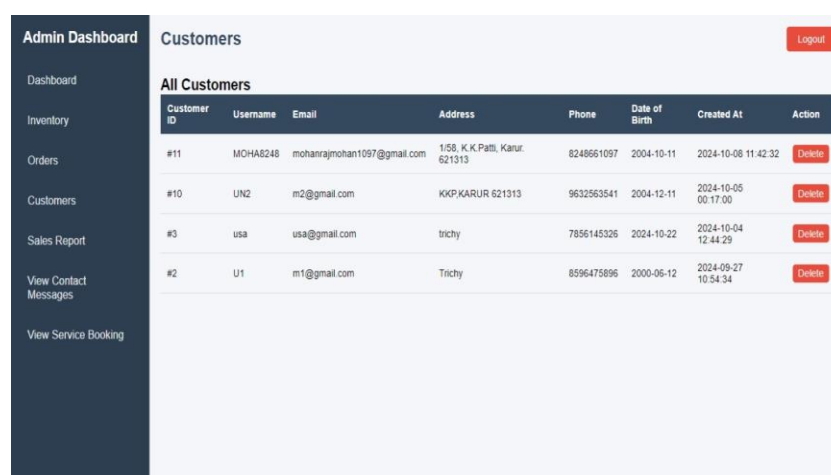
5.2. Integration Testing:

Testing of integrated modules to verify combined functionality after integration. Modules are typically code modules, individual applications, client and server applications on a network, etc. This type of testing is especially relevant to client/server and distributed systems.



The image shows a 'Create Your Profile' form. It has four input fields: 'Profile Image' with a 'Choose File' button and the text 'person_2.jpg'; 'Address' with the text '1/58, K.K.Patti, Karur, 621313'; 'Phone' with the text '8248661097'; and 'Date of Birth' with the text '11-10-2004' and a calendar icon. A blue 'Create Profile' button is at the bottom.

Fig:5.2 Register Account Page



The image shows an 'Admin Dashboard' with a sidebar menu and a 'Customers' section. The sidebar menu includes: Dashboard, Inventory, Orders, Customers, Sales Report, View Contact Messages, and View Service Booking. The 'Customers' section has a 'Logout' button and a table titled 'All Customers'.

Customer ID	Username	Email	Address	Phone	Date of Birth	Created At	Action
#11	MCHA6248	mohanrajmohan1097@gmail.com	1/58, K.K.Patti, Karur, 621313	8248661097	2004-10-11	2024-10-08 11:42:32	Delete
#10	UN2	m2@gmail.com	KKP:KARUR 621313	9632563541	2004-12-11	2024-10-05 00:17:00	Delete
#3	usa	usa@gmail.com	trichy	7856145326	2024-10-22	2024-10-04 12:44:29	Delete
#2	U1	m1@gmail.com	Trichy	8596475896	2000-06-12	2024-09-27 10:54:34	Delete

Fig:5.3 View Customer Details Page

5.3. Validation Testing:

The process of validating a software product, or determining if it meets high level criteria, involves determining whether it is up to part. It is the procedure used to verify that the product we are producing is the proper one. The actual and anticipated product are being validated.

Testing in motion is validation.

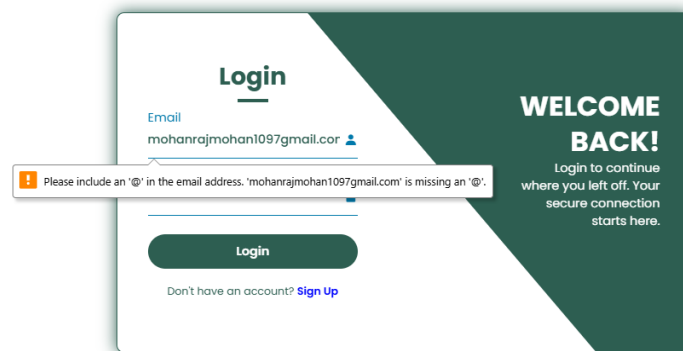


Fig:5.4 Validation Testing on Login

6.SYSTEM IMPLEMENTATION

Implementation is the stage of the project when the theoretical design is turned out into a working system. Thus, it can be considering the most critical stage in achieving a successful new system and in giving the user, confidence that the new system will work and be effective.

6.1 SOFTWARE DEMONSTRATION

6.1.1 Home Page

The home page of the Car Accessories Shop Management System offers a user-friendly interface for browsing products, accessing services, and discovering the latest deals.

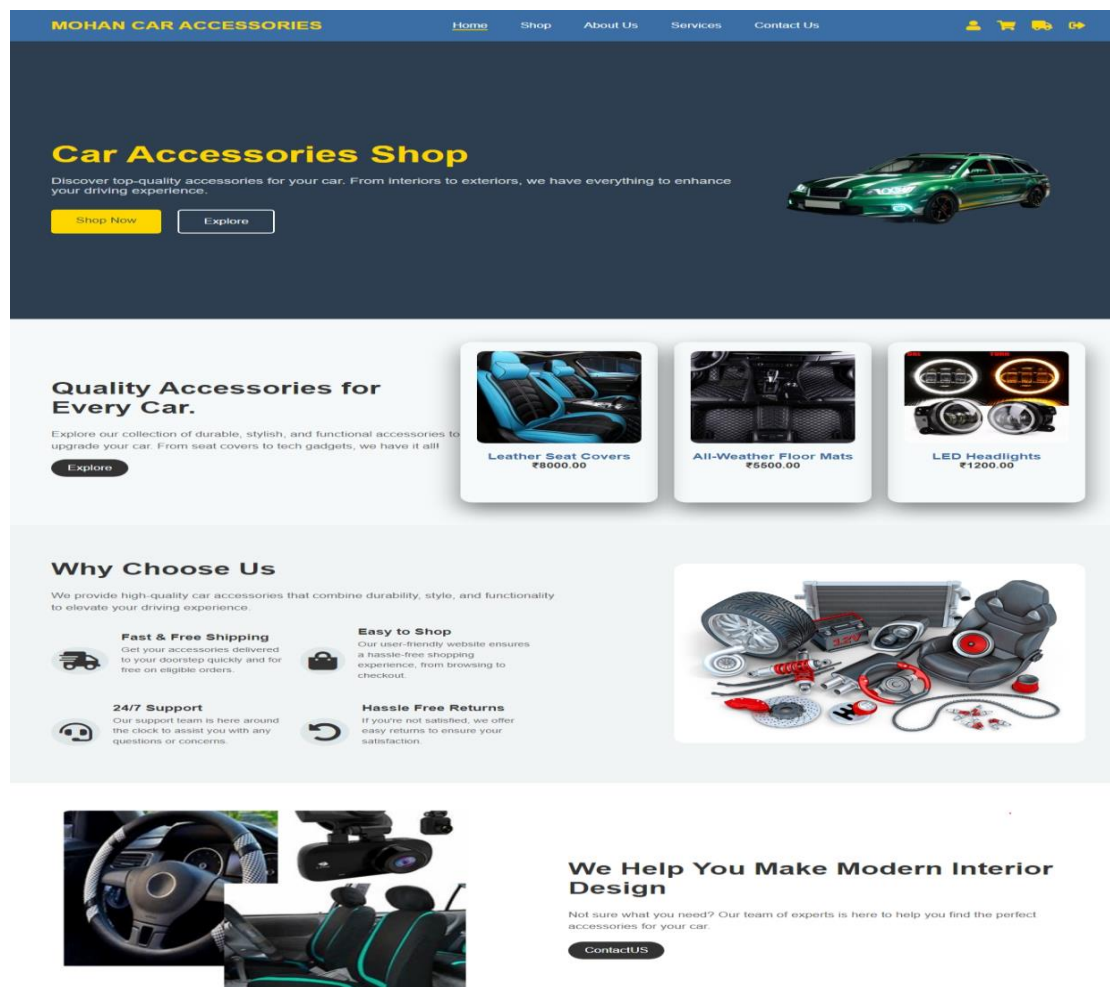
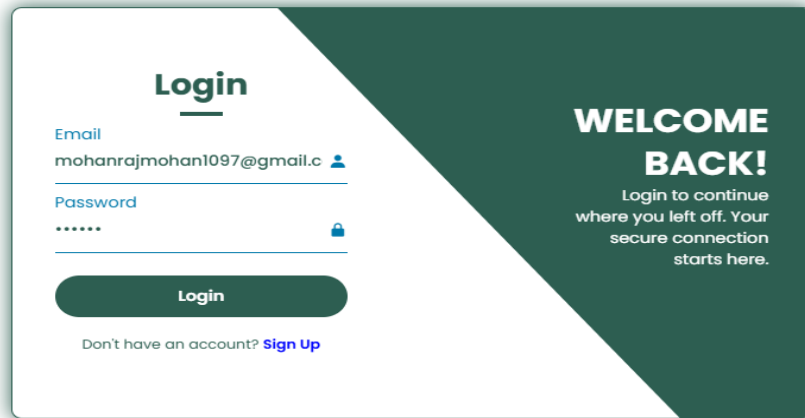


Fig:6.1 Home Page

6.1.2 Login Page

A user authentication interface allowing access to an website by entering credentials such as email and password.

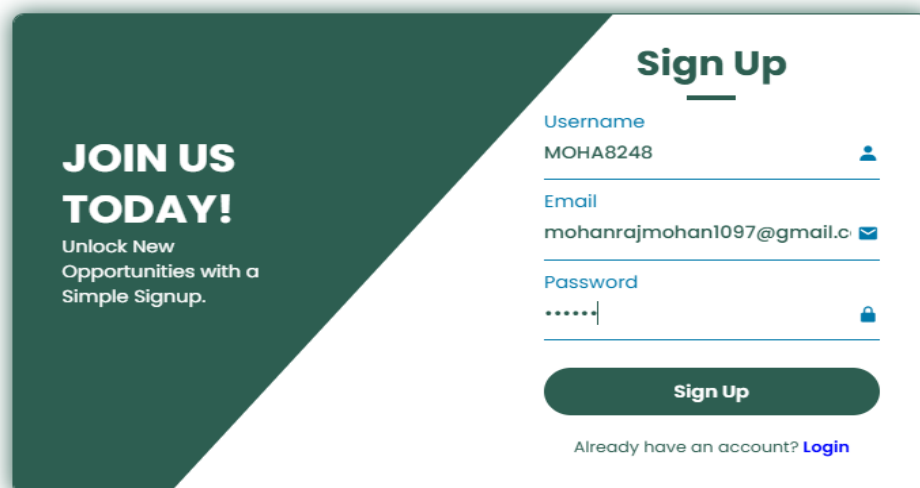


The login page features a dark green diagonal header on the right side. The main content area is white. The title 'Login' is centered at the top. Below it, there are two input fields: 'Email' with the value 'mohanrajmohan1097@gmail.c' and a user icon, and 'Password' with masked characters '.....' and a lock icon. A dark green 'Login' button is positioned below the fields. At the bottom, there is a link 'Don't have an account? Sign Up'. On the right side, the text 'WELCOME BACK!' is displayed in large, bold letters, followed by a smaller message: 'Login to continue where you left off. Your secure connection starts here.'

Fig:6.2 Login Page

6.1.3 Signup Page

An interface within the website enabling users to input email and password to signup.



The signup page features a dark green diagonal header on the left side. The main content area is white. The title 'Sign Up' is centered at the top. Below it, there are three input fields: 'Username' with the value 'MOHA8248' and a user icon, 'Email' with the value 'mohanrajmohan1097@gmail.c' and an envelope icon, and 'Password' with masked characters '.....' and a lock icon. A dark green 'Sign Up' button is positioned below the fields. At the bottom, there is a link 'Already have an account? Login'. On the left side, the text 'JOIN US TODAY!' is displayed in large, bold letters, followed by a smaller message: 'Unlock New Opportunities with a Simple Signup.'

Fig:6.3 SignUp Page

6.1.4 Shop

The shopping page displays a wide range of car accessories with detailed images, descriptions, and prices for easy browsing and purchasing.

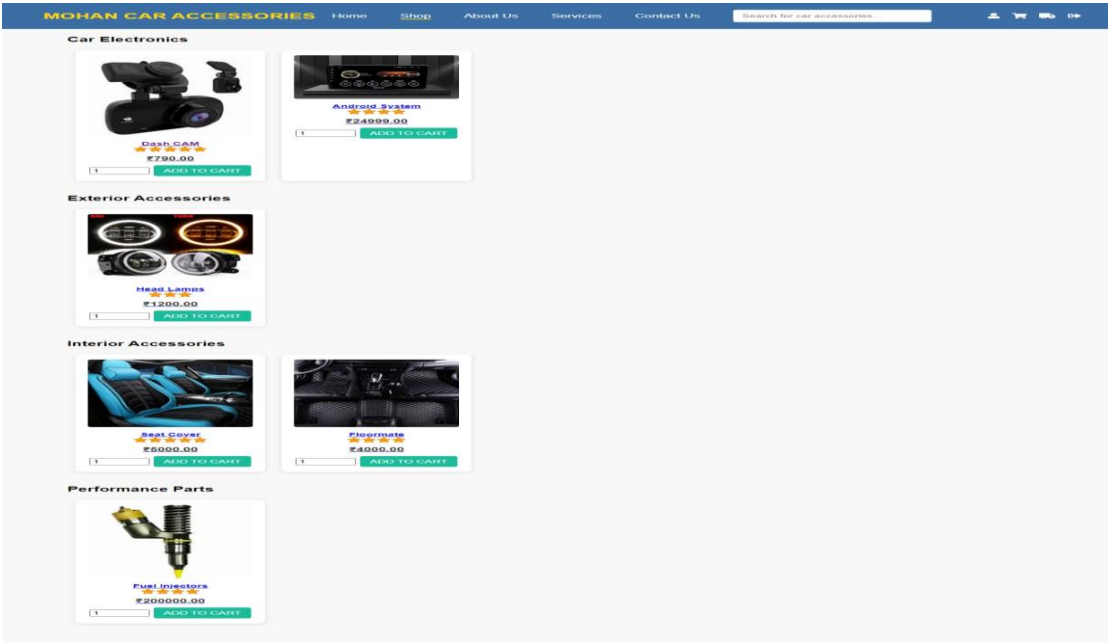


Fig:6.4 Shop

6.1.5 Book a Service

The "Book a Service" page allows customers to schedule car maintenance or installation services by selecting their preferred service type, date, and time.

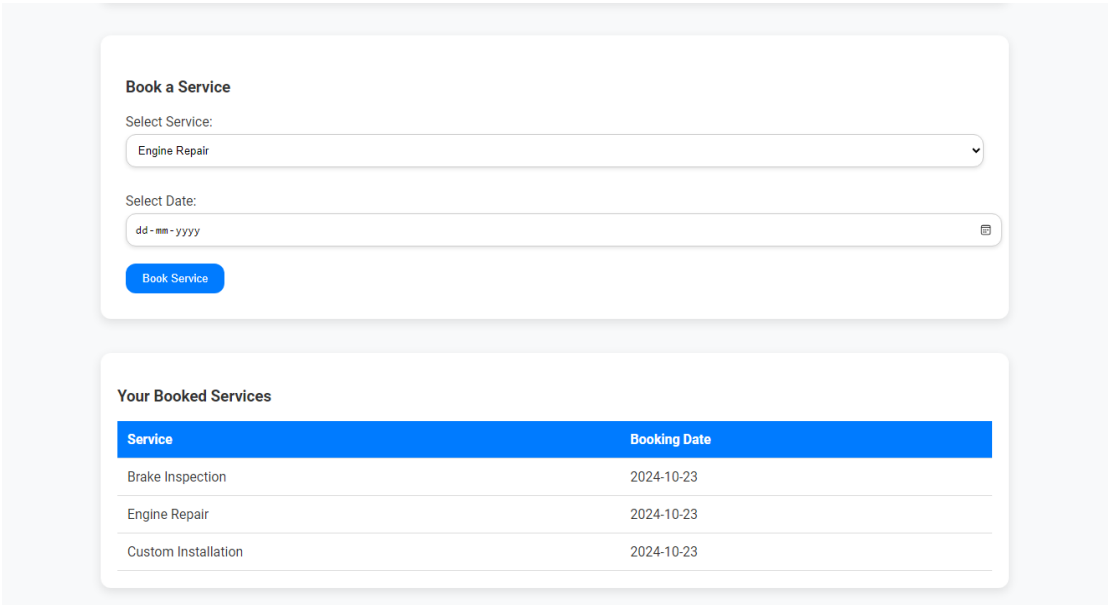



Fig:6.5 Book a Service

6.1.6 Profile

The profile page allows users to view and update their personal details, including contact information and account settings.

View Profile

Edit






Name	MOHA8248
Email	mohanrajmohan1097@gmail.com
Address	1/58, K.K.Patti, Karur. 621313
Phone	8248661097
Date of Birth	2004-10-11
Edit Profile	

Fig:6.6 Profile

6.1.7 Cart

The cart page displays selected products, quantities, and total price, allowing users to review items before proceeding to checkout.

Your Cart

Image	Product	Price	Quantity	Actions
	Dash CAM	₹790.00	1	Remove
	Floormate	₹4000.00	1	Remove
	Head Lamps	₹1200.00	1	Remove

Total Price: ₹5990

[Buy Now](#)

Fig:6.7 Cart

6.1.8 Checkout

The checkout page allows users to confirm their order, choose a payment method, and provide shipping details for a seamless purchase process.

All fields are required!

Checkout

Full Name
MOHANRAJ

Shipping Address
1/58, KKP, Karur, Tamilnadu -621313

Phone Number
8249661097

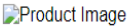
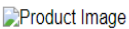
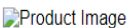
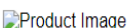
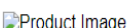
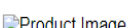
Payment Method
☒ Online Payment ☐ Cash on Delivery

Proceed to Payment

Fig:6.8 Checkout

6.1.9 Order Status

The Order Status page enables users to track the progress of their orders, from processing and shipping to delivery, providing real-time updates.

Your Order Status						
Order ID	Product Image	Product Name	Quantity	Price	Status	Paid
16		Dash CAM	1	790.00	PROCESSING	Paid
16		Floormate	1	4,000.00	PROCESSING	Paid
16		Head Lamps	1	1,200.00	PROCESSING	Paid
14		Dash CAM	1	790.00	DELIVERED	Paid
14		Fuel Injectors	1	200,000.00	DELIVERED	Paid
14		Head Lamps	1	1,200.00	DELIVERED	Paid

[Back to Shop](#)

Fig:6.9 Order Status

6.1.10 View Transactions Details

The View Transaction Details page displays a comprehensive history of user transactions, including dates, amounts, payment methods for easy tracking and reference.

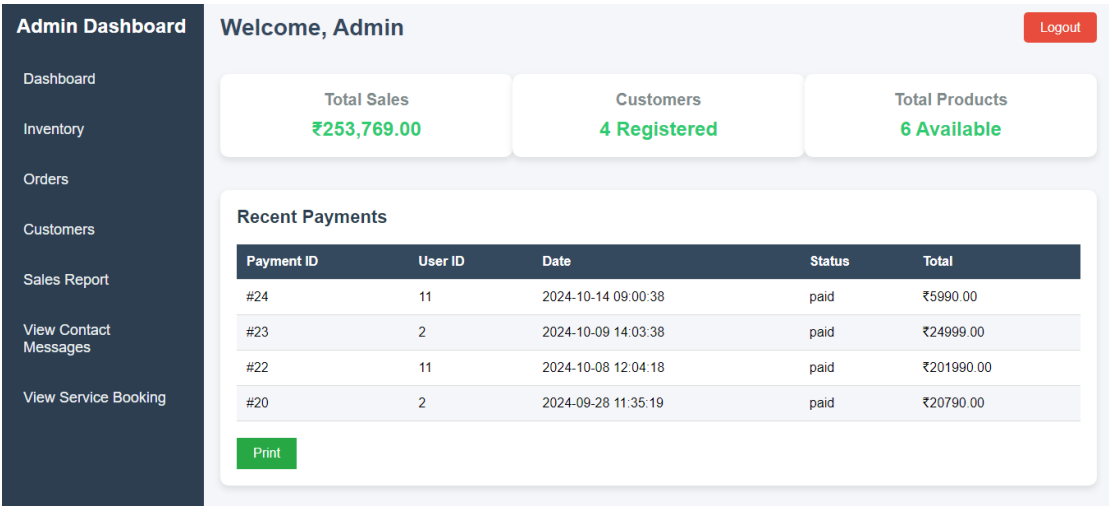


Fig:6.10 ViewTransactions Details

6.1.11 Inventory

The inventory page provides administrators with a complete overview of all available products, including stock levels, prices, and detailed descriptions, allowing for efficient management and updates.

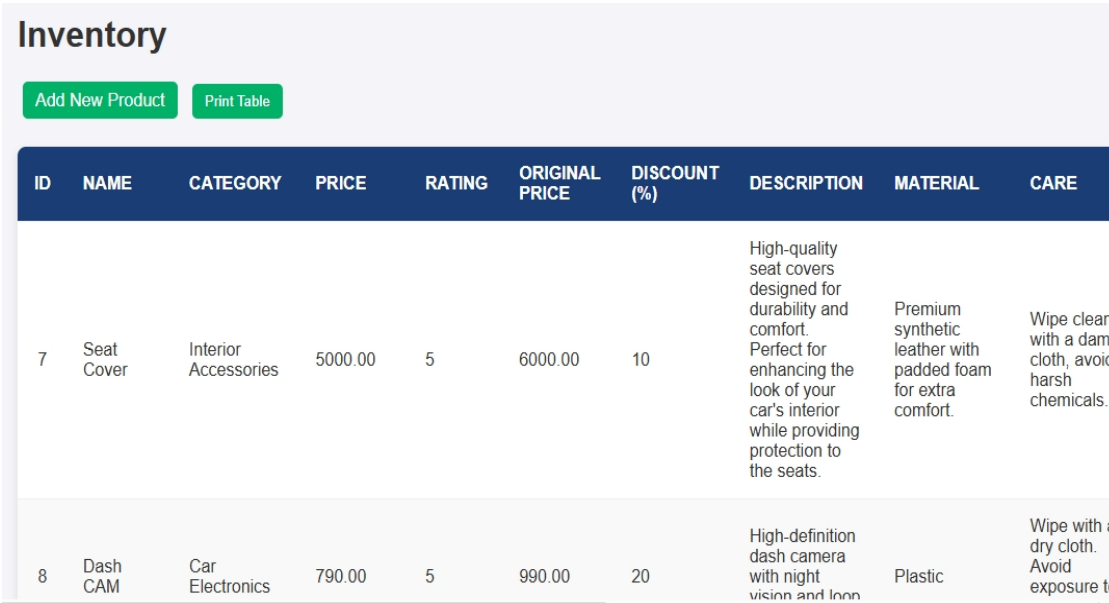


Fig:6.11 Inventory

6.1.12 Orders

The orders page allows administrators to view, manage, and process customer orders, including details on order status, payment, and shipment information.

OrdersLogout

Start Date: End Date: Select Product: Apply Filter

All Orders
Print Orders

Order ID	Customer Name	Order Date	Product Name	Quantity	Price	Status	Total Amount	Created At	Action
#16	Mohanraj	2024-10-14 03:30:38	Dash CAM	1	₹790.00	Processing	₹5990.00	2024-10-14 09:00:38	Edit Delete
#16	Mohanraj	2024-10-14 03:30:38	Floormate	1	₹4000.00	Processing	₹5990.00	2024-10-14 09:00:38	Edit Delete
#16	Mohanraj	2024-10-14 03:30:38	Head Lamps	1	₹1200.00	Processing	₹5990.00	2024-10-14 09:00:38	Edit Delete
#15	mohan	2024-10-09 08:33:38	Android System	1	₹24999.00	Processing	₹24999.00	2024-10-09 14:03:38	Edit Delete

Fig:6.12 Orders

6.1.13 View Customer Details

The View Customer Details page provides administrators with access to customer information, including contact details, order history, and account status, facilitating better customer support and management.

CustomersLogout

All Customers

Customer ID	Username	Email	Address	Phone	Date of Birth	Created At	Action
#11	MOHA8248	mohanrajmohan1097@gmail.com	1/58, K.K.Patti, Karur. 621313	8248661097	2004-10-11	2024-10-08 11:42:32	Delete
#10	Dhiva	m2@gmail.com	KKP,KARUR 621313	9632563541	2004-12-11	2024-10-05 00:17:00	Delete
#3	usha	usha@gmail.com	trichy	7856145326	2024-10-22	2024-10-04 12:44:29	Delete
#2	U1	m1@gmail.com	Trichy	8596475896	2000-06-12	2024-09-27 10:54:34	Delete

Fig:6.13 View Customer Details

6.1.14 Sales Report

The Sales Report page offers administrators insights into sales performance through detailed analytics, including total sales, revenue trends, and product performance over specific periods.

Sales Report

Logout

Start Date: End Date: Select Product: Apply Filter

Delivered Orders

Order ID	Customer Name	Order Date	Product Name	Total Amount	Delivery Status	Created At
#14	MOHANRAJ	2024-10-08 06:34:00	Dash CAM	₹201,990.00	Delivered	2024-10-08 12:04:18
#14	MOHANRAJ	2024-10-08 06:34:00	Fuel Injectors	₹201,990.00	Delivered	2024-10-08 12:04:18
#14	MOHANRAJ	2024-10-08 06:34:00	Head Lamps	₹201,990.00	Delivered	2024-10-08 12:04:18
#13	usha	2024-10-04 07:16:00	Dash CAM	₹790.00	Delivered	2024-10-04 12:46:55
#11	Vignesh	2024-09-28 06:05:00	Dash CAM	₹20,790.00	Delivered	2024-09-28 11:35:19
#11	Vignesh	2024-09-28 06:05:00	Seat Cover	₹20,790.00	Delivered	2024-09-28 11:35:19

Print Report

6.1.14 Sales Report

6.1.15 View Service Bookings

The View Service Bookings page allows administrators to manage and review all scheduled service appointments, displaying details such as customer information, service type, date, time, and booking status.

Service Booking Details

Logout

Start Date: End Date: Apply Filter

Booking ID	User ID	Username	Service	Booking Date
8	3	usha	Tire Rotation	2024-12-25
9	3	usha	Lighting Upgrades	2024-12-25
6	11	MOHA8248	Brake Inspection	2024-10-23
5	11	MOHA8248	Engine Repair	2024-10-23
7	11	MOHA8248	Custom Installation	2024-10-23
10	3	usha	Custom Installation	2024-10-23
4	10	Dhiva	Custom Installation	2024-10-11

Print Report

6.1.15 View Service Bookings

7. CONCLUSION

In the **Car Accessories Shop Management System**, the platform provides a user-friendly interface that ensures efficient management of shop services, offering convenience for both customers and administrators. The admin can oversee product inventory, manage customer orders, and track sales effectively. Customers can easily register, view product listings, add items to their cart, place orders, and view their purchase history. The platform also ensures secure and seamless transactions. Future improvements will focus on expanding functionality and further enhancing the overall user experience.

BIBLIOGRAPHY

BOOK REFERENCES:

1. Robin Nixon, “Learning PHP, MySQL & JavaScript: With jQuery, CSS & HTML5”, 2021.
2. Luke Welling, Laura Thomson, “PHP and MySQL Web Development”, 2016.

WEBSITE REFERENCES:

1. <https://www.php.net/manual/en/index.php>
2. <https://www.w3schools.com/php/>
3. <https://www.mysql.com/>