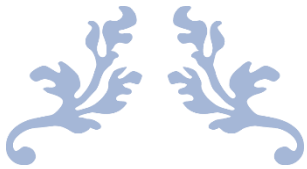

PAPIRUS E-COMMERCE PLATFORM REPORT

26 MAYIS 2024

MERTCAN KORKMAZ
2019556045

Papirus E-Commerce Platform

1. Introduction

Papirus E-Commerce Platform is a comprehensive online shopping platform where users can review and purchase various products online. Designed to meet the needs of modern consumers, this platform has a user-friendly interface and a powerful backend system. The project was developed using a PHP-based backend and MySQL database. Offering a user-friendly interface, this platform is designed to meet the needs of both end users and site administrators.

1.1. Purpose of the Project

Papirus E-Commerce Platform aims to provide a space where users can purchase various products online. This platform is designed to ensure that users can easily access the products they need and purchase them securely. The project has been developed to improve the user experience, speed up the purchasing process and increase customer satisfaction.

1.2. Development of the Project

Papirus E-Commerce Platform is developed using modern web technologies. PHP is a server-side programming language used to create dynamic web pages and MySQL is a popular relational database management system used for database management. These technologies ensure that the project is reliable, scalable and manageable.

1.3. Features for Users and Administrators

Papirus E-Commerce Platform offers various features for users and administrators. Users can register to the site, search for products, add them to the basket and place orders. Administrators can manage products, track orders and analyse sales data. These features make the platform user-friendly and functional.

1.4. Importance of the Project

Today, when e-commerce is growing rapidly, users need fast, secure and user-friendly platforms while shopping online. Papirus E-Commerce Platform has been developed to meet this need. It is an important tool to improve the shopping experience of users and increase online sales of businesses.

2. Purpose and Scope

2.1. Purpose

The main purpose of Papirus E-Commerce Platform is to facilitate the online shopping experience of users and to enable site administrators to manage product and order management effectively. The platform is designed to enable users to access the products they need quickly and easily. At the same time, it aims to increase customer satisfaction with secure payment methods and user-friendly interface.

2.2. Scope

The scope of the project is quite wide and includes many functions starting from user registration and login processes, listing the products, adding them to the basket, placing orders and product and order management through the management panel.

2.2.1. User Operations

User Registration and Login: Users can register and log in to the site. Information such as username, e-mail and password is requested during the registration process. Login is done with e-mail and password information.

Product Search and Review: Users can search and review products in various categories. Products are listed with details such as description, price and picture.

Add to Basket: Users can add the products they like to the basket. Products added to the basket are stored for later ordering.

Ordering: Users can order the products in their basket. During the ordering process, the user is asked for details such as delivery address and payment information.

2.2.2. Administrator Operations

Product Management: Administrators can add new products, edit or delete existing products. Product management includes product name, description, price, stock status and category information.

Order Management: Administrators can view and manage orders placed by users. The status of orders (e.g., processing, shipped, delivered) can be updated.

Graphical Reporting: Administrators can generate graphical reports on sales and orders. These reports are used to analyse sales trends and performance.

2.2.3. Technical Details

Database Design: The project uses a MySQL database with separate tables for users, products, orders and order items.

Security: Various measures have been taken for the security of user data. Passwords are hashed and stored using the bcrypt algorithm.

Responsive Design: The platform has a responsive design using Bootstrap. In this way, users can access seamlessly from different devices (computer, tablet, phone).

2.3. Project Goals

User Experience: It is aimed that users can easily access the site and perform their transactions easily.

Reliability and Security: User data will be securely stored and processed.

Manageability: A user-friendly management panel will be provided for administrators to easily manage products and orders.

Scalability: The platform will be scalable in accordance with the increasing number of users and products.

3. Technologies Used

3.1. Backend

PHP: A server-side programming language used to develop dynamic web applications.

MySQL: It is a relational database management system used for database management.

3.2. Frontend

HTML: Markup language used to create the structure of web pages.

CSS: A style language used to format the appearance of web pages.

JavaScript: It is a programming language used to add interactivity to web pages.

3.3. Libraries and Tools

Bootstrap: An open source CSS framework used to create modern and responsive web pages.

jQuery: A JavaScript library that simplifies JavaScript coding.

Chart.js: A JavaScript library for creating charts and tables.

Tempus Dominus: A JavaScript library used for date and time selection.

Owl Carousel: A responsive content carousel with touch support.

4. Database Design

The database contains main tables for users, products, orders and order items.

4.1. Users Table

Stores user information.

```sql
CREATE TABLE users (
    id INT AUTO_INCREMENT PRIMARY KEY,
    username VARCHAR(50) NOT NULL,
    email VARCHAR(100) NOT NULL,
    password VARCHAR(255) NOT NULL,
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);
```

4.2. Products Table

Stores product information.

```sql
CREATE TABLE products (
    id INT AUTO_INCREMENT PRIMARY KEY,
    name VARCHAR(100) NOT NULL,
    description TEXT NOT NULL,
    price DECIMAL(10, 2) NOT NULL,
    stock INT NOT NULL,
    category VARCHAR(50),
    image_url VARCHAR(255),
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);
```

4.3. Orders table

Stores order information.

```sql
CREATE TABLE orders (
    id INT AUTO_INCREMENT PRIMARY KEY,
    user_id INT,
    total DECIMAL(10, 2) NOT NULL,
    status VARCHAR(50) DEFAULT 'pending',
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    FOREIGN KEY (user_id) REFERENCES users(id)
);
```

4.4. Order_items Table

It stores the product information in the order.

```sql
CREATE TABLE order_items (
    id INT AUTO_INCREMENT PRIMARY KEY,
    order_id INT,
    product_id INT,
    quantity INT NOT NULL,
    price DECIMAL(10, 2) NOT NULL,
    FOREIGN KEY (order_id) REFERENCES orders(id),
    FOREIGN KEY (product_id) REFERENCES products(id)
);
```

5. Features

5.1. User Registration and Login

Users can register and log in to the site. This process enables users to create accounts and log in to the site with their existing accounts.

5.1.1. User Registration (register.php)

```php
<?php
if ($_SERVER['REQUEST_METHOD'] == 'POST') {
    include 'config.php';

    $username = $_POST['username'];
    $email = $_POST['email'];
    $password = password_hash($_POST['password'], PASSWORD_BCRYPT);

    $sql = "INSERT INTO users (username, email, password) VALUES ('$username', '$email', '$password')";
    if (mysqli_query($conn, $sql)) {
        echo "Registration successful";
    } else {
        echo "Error: " . $sql . "<br>" . mysqli_error($conn);
    }
}
?>
```

5.1.2. User Login (login.php)

```php
<?php
session_start();
if ($_SERVER['REQUEST_METHOD'] == 'POST') {
    include 'config.php';

    $email = $_POST['email'];
    $password = $_POST['password'];

    $sql = "SELECT * FROM users WHERE email='$email'";
    $result = mysqli_query($conn, $sql);
    $user = mysqli_fetch_assoc($result);

    if ($user && password_verify($password, $user['password'])) {
        $_SESSION['user_id'] = $user['id'];
        echo "Login successful";
    } else {
        echo "Invalid credentials";
    }
}
?>
```

5.2. Product Listing

Users can review products in different categories. This process is performed by pulling the products from the database and displaying them to the users.

5.2.1. Product Listing (products.php)

```php
<?php
include 'config.php';

$sql = "SELECT * FROM products";
$result = mysqli_query($conn, $sql);

while ($row = mysqli_fetch_assoc($result)) {
    echo "<div class='product'>";
    echo "<h2>" . $row['name'] . "</h2>";
    echo "<p>" . $row['description'] . "</p>";
    echo "<p>$" . $row['price'] . "</p>";
    echo "<img src='" . $row['image_url'] . "' alt='" . $row['name'] . "'>";
    echo "<button>Add to Cart</button>";
    echo "</div>";
}
?>
```

5.3. Add to basket

Users can add products to the basket and convert this basket into an order later.

5.3.1. Add to Cart (add_to_cart.php)

```php
<?php
session_start();
if ($_SERVER['REQUEST_METHOD'] == 'POST') {
    $product_id = $_POST['product_id'];
    $quantity = $_POST['quantity'];

    if (!isset($_SESSION['cart'])) {
        $_SESSION['cart'] = [];
    }

    if (isset($_SESSION['cart'][$product_id])) {
        $_SESSION['cart'][$product_id] += $quantity;
    } else {
        $_SESSION['cart'][$product_id] = $quantity;
    }

    echo "Product added to cart";
}
?>
```

5.4. Order Placing

Users can order the products in their basket. This process saves users' orders to the database and stores the order details.

5.4.1. Ordering (checkout.php)

```php
<?php
session_start();
if ($_SERVER['REQUEST_METHOD'] == 'POST') {
    include 'config.php';

    $user_id = $_SESSION['user_id'];
    $total = $_POST['total'];

    $sql = "INSERT INTO orders (user_id, total) VALUES ('$user_id', '$total')";
    if (mysqli_query($conn, $sql)) {
        $order_id = mysqli_insert_id($conn);

        foreach ($_SESSION['cart'] as $product_id => $quantity) {
            $price_query = "SELECT price FROM products WHERE id='$product_id'";
            $price_result = mysqli_query($conn, $price_query);
            $price_row = mysqli_fetch_assoc($price_result);
            $price = $price_row['price'];

            $order_item_sql = "INSERT INTO order_items (order_id, product_id, quantity, price) VALUES ('$order_id', '$product_id', '$quantity', '$price')";
            mysqli_query($conn, $order_item_sql);
        }

        unset($_SESSION['cart']);
        echo "Order placed successfully";
    } else {
        echo "Error: " . $sql . "<br>" . mysqli_error($conn);
    }
}
?>
```

5.5. Administration Panel

The administration panel allows site administrators to manage products and orders. It also offers the possibility to create graphical reports.

5.5.1. Product Management (admin/products.php)

```php
<?php
include 'config.php';

if ($_SERVER['REQUEST_METHOD'] == 'POST') {
    $name = $_POST['name'];
    $description = $_POST['description'];
    $price = $_POST['price'];
    $stock = $_POST['stock'];
    $category = $_POST['category'];
    $image_url = $_POST['image_url'];

    $sql = "INSERT INTO products (name, description, price, stock, category, image_url) VALUES ('$name', '$description', '$price', '$stock', '$category', '$image_url')";
    if (mysqli_query($conn, $sql)) {
        echo "Product added successfully";
    } else {
        echo "Error: " . $sql . "<br>" . mysqli_error($conn);
    }
}

$sql = "SELECT * FROM products";
$result = mysqli_query($conn, $sql);

while ($row = mysqli_fetch_assoc($result)) {
    echo "<div class='product'>";
    echo "<h2>" . $row['name'] . "</h2>";
    echo "<p>" . $row['description'] . "</p>";
    echo "<p>$" . $row['price'] . "</p>";
    echo "<p>Stock: " . $row['stock'] . "</p>";
    echo "<button>Edit</button>";
    echo "<button>Delete</button>";
    echo "</div>";
}
?>
```

5.5.2. Order Management (admin/orders.php)

```php
<?php
include 'config.php';

$sql = "SELECT orders.id, users.username, orders.total, orders.status, orders.created_at FROM orders JOIN users ON orders.user_id = users.id";
$result = mysqli_query($conn, $sql);

while ($row = mysqli_fetch_assoc($result)) {
    echo "<div class='order'>";
    echo "<h2>Order ID: " . $row['id'] . "</h2>";
    echo "<p>Customer: " . $row['username'] . "</p>";
    echo "<p>Total: $" . $row['total'] . "</p>";
    echo "<p>Status: " . $row['status'] . "</p>";
    echo "<p>Date: " . $row['created_at'] . "</p>";
    echo "<button>Update Status</button>";
    echo "</div>";
}
?>
```

5.5.3. Graphical Reporting (admin/reports.php)

```php
<?php
include 'config.php';
$sql = "SELECT DATE(created_at) as date, COUNT(*) as orders, SUM(total) as revenue FROM orders GROUP BY DATE(created_at)";
$result = mysqli_query($conn, $sql);
$dates = [];
$orders = [];
$revenue = [];
while ($row = mysqli_fetch_assoc($result)) {
    $dates[] = $row['date'];
    $orders[] = $row['orders'];
    $revenue[] = $row['revenue'];
}
?>
<!DOCTYPE html>
<html>
<head>
    <title>Reports</title>
    <script src="https://cdn.jsdelivr.net/npm/chart.js"></script>
</head>
<body>
    <h1>Sales and Orders Report</h1>
    <canvas id="salesChart"></canvas>
    <script>
        var ctx = document.getElementById('salesChart').getContext('2d');
        var salesChart = new Chart(ctx, {
            type: 'line',
            data: {
                labels: <?php echo json_encode($dates); ?>,
                datasets: [{
                    label: 'Orders',
                    data: <?php echo json_encode($orders); ?>,
                    borderColor: 'rgba(75, 192, 192, 1)',
                    borderWidth: 1,
                    fill: false
                },
                {
                    label: 'Revenue',
                    data: <?php echo json_encode($revenue); ?>,
                    borderColor: 'rgba(153, 102, 255, 1)',
                    borderWidth: 1,
                    fill: false
                }]
            },
            options: {
                scales: {
                    x: {
                        beginAtZero: true
                    },
                    y: {
                        beginAtZero: true
                    }
                }
            }
        });
    </script>
</body>
</html>
```

6. Conclusion

Papirus E-Commerce Platform is a comprehensive and user-friendly platform developed to meet the modern requirements of online shopping. This project offers various features to meet the needs of both end users and site administrators.

6.1. User Experience

Papirus E-Commerce Platform offers an environment where users can easily search, review and purchase products. Users can quickly register on the site and log in securely. Products are listed with detailed descriptions, images and price information. Users can add their favourite products to the basket and place their orders easily.

In this process, the user experience is optimised to provide a fast, secure and seamless shopping experience. Responsive design increases user satisfaction by providing the same quality experience on different devices (computer, tablet, phone).

6.2. Ease of Management

The management panel offered for site administrators provides effective management of products and orders. Administrators can add new products, update or delete existing products. The order management module allows users to track and manage orders placed by users. The status of orders can be updated and detailed order information can be viewed.

Graphical reporting tools allow managers to analyse sales and order data. These reports help managers evaluate sales trends and performance. In this way, business strategies can be developed and business processes can be optimised.

6.3. Security and Reliability

In Papirus E-Commerce Platform, the security of user data is kept at the highest level. Passwords are hashed using the bcrypt algorithm and stored in the database in this way. This prevents user passwords from being stored in plain text and increases security. In addition, prepared statements are used to protect against SQL injection attacks.

Database access controls ensure that only authorised users have access to the database. This increases database security and ensures that user data is protected. These security measures ensure that users and administrators have confidence in the platform.

6.4. Performance and Scalability

Papirus E-Commerce Platform is optimised to provide high performance and scalability. Developed using reliable and performant technologies such as PHP and MySQL, the platform is designed to meet high traffic and data processing requirements.

The database design is structured to increase performance while maintaining data integrity. The relational database model ensures that data is stored and processed accurately and efficiently. Furthermore, database indexing and query optimisation increases data access speed.

6.5. Future Development Opportunities

Papirus E-Commerce Platform has a flexible infrastructure for future developments and additions. The modular structure of the platform allows new features to be easily integrated. For example, additional features such as customer loyalty programs, personalised product recommendations and more advanced search functions can be added to the platform in the future.

Furthermore, if the platform expands to international markets, features such as multilingual support and the ability to trade in different currencies may also be added. This would enable the platform to serve a global customer base.

6.6. Business and Customer Satisfaction

Papirus E-Commerce Platform helps businesses increase their online sales and manage their operations more efficiently. With its user-friendly interface and powerful management panel, businesses can better serve their customers and increase customer satisfaction.

While users enjoy a safe and hassle-free shopping experience, businesses have the necessary tools to meet customer needs and stay ahead of the market competition. This supports the sustainable growth and success of businesses in the long term.

6.7. Conclusion

Papirus E-Commerce Platform is a project meticulously designed and developed to meet modern e-commerce requirements. It exhibits superior performance in important areas such as user experience, ease of management, security, performance and scalability. While the platform offers comprehensive features that meet the needs of users and administrators, it is also open to future developments.

This project aims to improve the online shopping experience of users and contribute to the digital transformation processes of businesses by providing an innovative and reliable solution in the field of e-commerce. Papirus E-Commerce Platform will be further strengthened with future additions and improvements and will consolidate its place in the sector.

SITE INTRODUCTION VIDEO

Youtube Tanıtım Videosu