CEMENT ORDERING MANAGEMENT SYSTEM

A MINI-PROJECT REPORT

Submitted by

LALIT PRASANNA G-2116220701142

in partial fulfilment of the award of the degree

of

BACHELOR OF ENGINEERING
IN
COMPUTER SCIENCE AND ENGINEERING

HARDWORK DISCIPLINE REC

RAJALAKSHMI
ENGINEERING COLLEGE
An AUTONOMOUS Institution
Affiliated to ANNA UNIVERSITY, Chennai

RAJALAKSHMI ENGINEERING COLLEGE

AUTONOMOUS, CHENNAI

NOV-DEC 2024

BONAFIDE CERTIFICATE

Certified that this mini project "**Cement ordering management system**" is the bonafide work of "**LALIT PRASANNA G (2116220701142)**" who carried out the project work under my supervision.

SIGNATURE
Mrs. JANANEE V,
Assistant Professor,
Computer Science & Engineering
Rajalakshmi Engineering College
Thandalam, Chennai -602105.

Submitted for the End semester practical examination to be held on ____

INTERNAL EXAMINER **EXTERNAL EXAMINER**

ACKNOWLEDGEMENT

I express my sincere thanks to my beloved and honourable chairman **MR.S.MEGANATHAN** and the chairperson **DR.M.THANGAM MEGANATHAN** for their timely support and encourage men.

I am greatly indebted to my respected and honourable principal **Dr. S.N. MURUGESAN** for his able support and guidance.

No words of gratitude will suffice for the unquestioning support extended to us by my head of the department **Dr. P. KUMAR,** and my Academic Head **Dr. R. SABITHA,** for being ever supporting force during my project work.

I also extend my sincere and hearty thanks to my internal guide **Mrs. JANANEE V** for her valuable guidance and motivation during the completion of this project.

My sincere thanks to my family members, friends and other staff members of Computer Science and Engineering.

ABSTRACT

The Cement Ordering Management System is a comprehensive platform designed to optimize and automate the process of ordering cement, aimed at improving efficiency for both customers and administrators. This system addresses the challenges of manual cement ordering processes, such as delays, errors, and a lack of transparency, by providing a streamlined, intuitive interface that simplifies interactions.

The user interface is designed to be responsive and accessible across various devices, including desktops, tablets, and smartphones. Customers can easily navigate the platform to browse cement types, check availability, and place orders with minimal effort. A key feature of the system is real-time order tracking, which allows customers to monitor their orders from placement to delivery. This ensures transparency and builds trust between the company and its clients.

On the administrative side, the system offers robust tools for inventory management, order processing, and logistics oversight. Administrators can track stock levels, process orders efficiently, and manage delivery schedules, reducing errors and ensuring timely deliveries. The platform also supports automated notifications, keeping both customers and administrators updated on order status, stock availability, and delivery timelines.

Incorporating these features into a single platform leads to improved customer satisfaction, better order accuracy, and a reduction in processing times. The Cement Ordering Management System is a powerful tool for enhancing operational efficiency, promoting seamless customer interaction, and maintaining a competitive edge in the construction industry.

TABLE OF CONTENTS

CHAPTER 1
INTRODUCTION

1.1 INTRODUCTION

1.1 Introduction: Cement Ordering Management System

The Cement Ordering Management System is designed to simplify and automate the cement procurement process for construction companies, contractors, and other users in the construction industry. This platform provides a centralized system for managing cement orders, offering services such as product selection, order placement, real-time tracking, and delivery scheduling. By catering to both customers and suppliers, the system streamlines communication and ensures efficient order fulfillment. It offers users an easy-to-navigate interface that enables them to quickly browse available products, place orders, track shipments, and manage payments, all in one place.

For suppliers, the Cement Ordering Management System offers tools to efficiently handle inventory, order processing, and logistics, ensuring timely deliveries and reducing the risk of stock shortages or order delays. This platform aims to improve the overall operational efficiency of the cement supply chain, minimizing errors associated with manual ordering processes. By integrating essential features such as automated notifications, stock monitoring, and delivery updates, the system provides an efficient and transparent ordering experience, improving customer satisfaction. The Cement Ordering Management System is designed to enhance the procurement experience for all users, making cement ordering more efficient, accessible, and convenient.

1.2 Scope of the Work

The scope of the Cement Ordering Management System includes several key functionalities aimed at improving both the customer and supplier experience. Users will be able to create and manage profiles, allowing for personalized order tracking and purchase history. The system will serve as a unified platform for browsing different cement grades, placing bulk or individual orders, and tracking order status in real-time. Secure payment integration ensures seamless financial transactions, while the system's real-time inventory management feature keeps stock levels and product availability updated to prevent overbooking.

The platform will offer additional features like automated email and SMS notifications for order confirmations, shipment tracking, and delivery updates. It will also provide special offers and discounts to regular customers, further enhancing the value of the system for buyers. On the administrative side, suppliers will have tools to manage incoming orders, track inventory, coordinate deliveries, and respond to customer inquiries. Analytical tools will be included to track performance metrics such as order volume, sales trends, and customer satisfaction. These features collectively improve order management, reduce operational inefficiencies, and help maintain a competitive edge in the construction industry.

1.3 Aim and Objectives of the Project

The aim of the Cement Ordering Management System is to create an efficient and user-friendly platform that streamlines the process of ordering and delivering cement. The system seeks to improve the overall experience for both customers and suppliers by integrating essential functionalities like order tracking, real-time inventory updates, and secure payment options. It aims to enhance customer satisfaction while providing robust tools for suppliers to manage their operations effectively.

Objectives:

1. User-Centric Design: Develop an intuitive and aesthetically appealing interface that simplifies the cement ordering process for users across different sectors.

2. Comprehensive Product Offering: Provide a centralized platform for customers to browse a wide range of cement products and place bulk orders.

3. Real-Time Updates: Ensure accurate, real-time updates on product availability, order status, and delivery schedules to minimize delays and errors.

4. Customer Support: Establish efficient customer support channels, such as live chat or a helpdesk, to assist users with order inquiries, issues, and general queries.

CHAPTER 2

SYSTEM SPECIFICATIONS

2.1 SOFTWARE SPECIFICATIONS

➢ **Operating System**

- **Server**:
  - Linux (Ubuntu Server, CentOS, or Debian) for stability and performance.
  - Alternatively, Windows Server (2016 or later) if required by specific applications.
- **Client**:
  - Windows 10 or later, macOS (latest version), or a recent Linux distribution.

➢ **Web Server**

- **Apache HTTP Server** or **Nginx**: For handling HTTP requests and serving web pages.

➢ **Database Management System**

- **MySQL** or **PostgreSQL**: For relational database management to store user data, bookings, and transaction records.
- **MongoDB**: Optionally, for handling non-relational data if required by specific features.

➢ **Programming Languages**

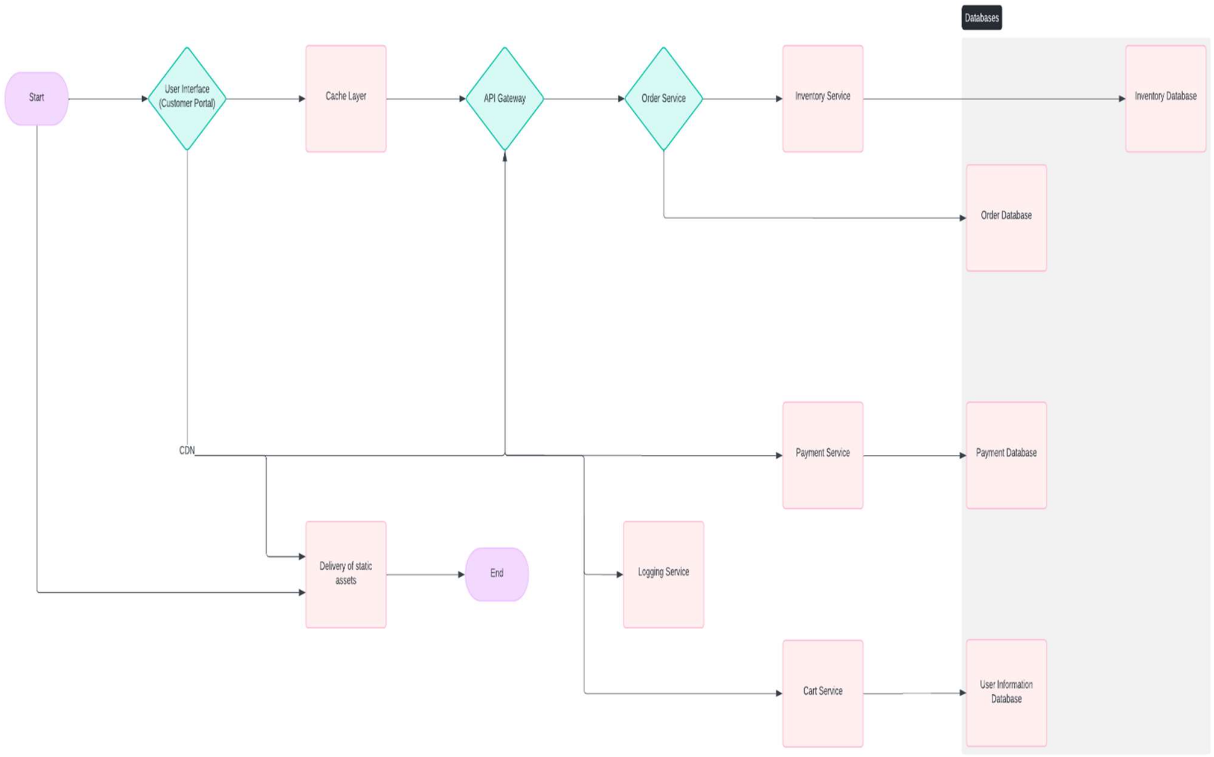
- **Frontend**:
  - **HTML5**, **CSS3**, and **JavaScript** for building responsive web interfaces.
  - **Frameworks**:
    - **React**, **Angular**, or **Vue.js** for dynamic user interfaces.
- **Backend**:
  - **Node.js** (JavaScript) or **Python** (Flask/Django) for server-side development.

  - **PHP** or **Java** (Spring) can also be used based on team expertise.

- **Frameworks and Libraries**
- **Bootstrap** or **Tailwind CSS**: For responsive and visually appealing design.
- **Express.js**: If using Node.js for a structured web application.
- **RESTful API**: For communication between the frontend and backend.

CHAPTER 3

ARCHITECTURE DIAGRAM



CHAPTER 4

MODULE DESCRIPTION

- **User Management Module**

This module allows users to register, log in, and manage their profiles. It includes functionalities for user authentication, password recovery, and the ability to update personal information. The module supports various user roles, such as travellers and administrators, each with different access levels and capabilities.

- **Search and Booking Module**

The Search and Booking Module is central to the system, enabling users to search for and book various travel services, including flights, hotels, vehicle rentals, and tour packages. Users can filter search results based on preferences such as price, location, and availability. The module facilitates the booking process, allowing users to view details, modify, or cancel their reservations.

- **Special Offers and Discounts Module**

This module allows administrators to create and manage special offers, discounts, and promotional campaigns. Users can view available promotions, apply discount codes at checkout, and receive targeted offers based on their preferences and past bookings.

- **Customer Support Module**

The Customer Support Module provides users with various support options, including live chat, email, and ticketing systems for inquiries and issues. It may also include an FAQ section to address common questions and facilitate quicker resolutions.

- **Analytics and Reporting Module**

This module gathers data on user behaviour, booking trends, and customer satisfaction. Administrators can generate reports to analyse system performance, identify growth opportunities, and enhance user experience through data-driven decisions.

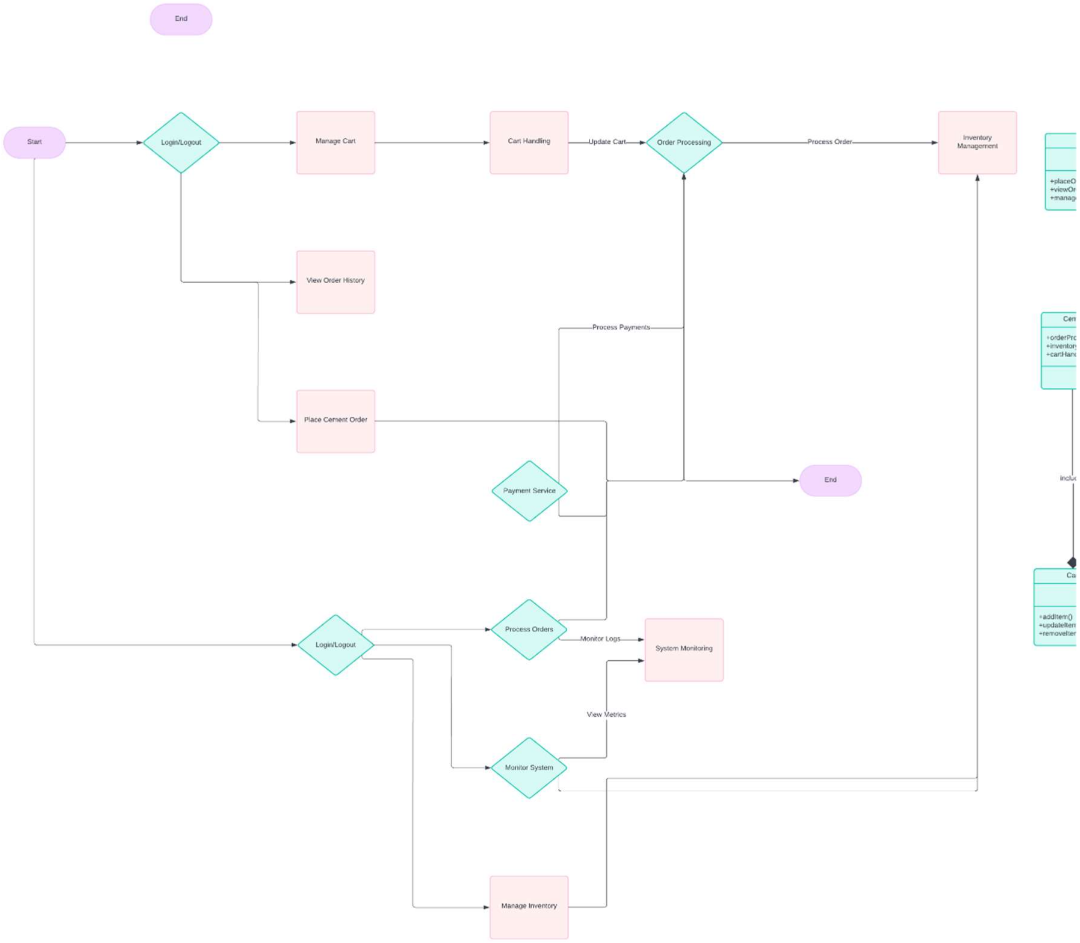
- **Feedback and Review Module**

This module allows users to submit feedback and reviews for services they have used. It helps improve service quality by enabling administrators to monitor user satisfaction and make necessary adjustments based on feedback

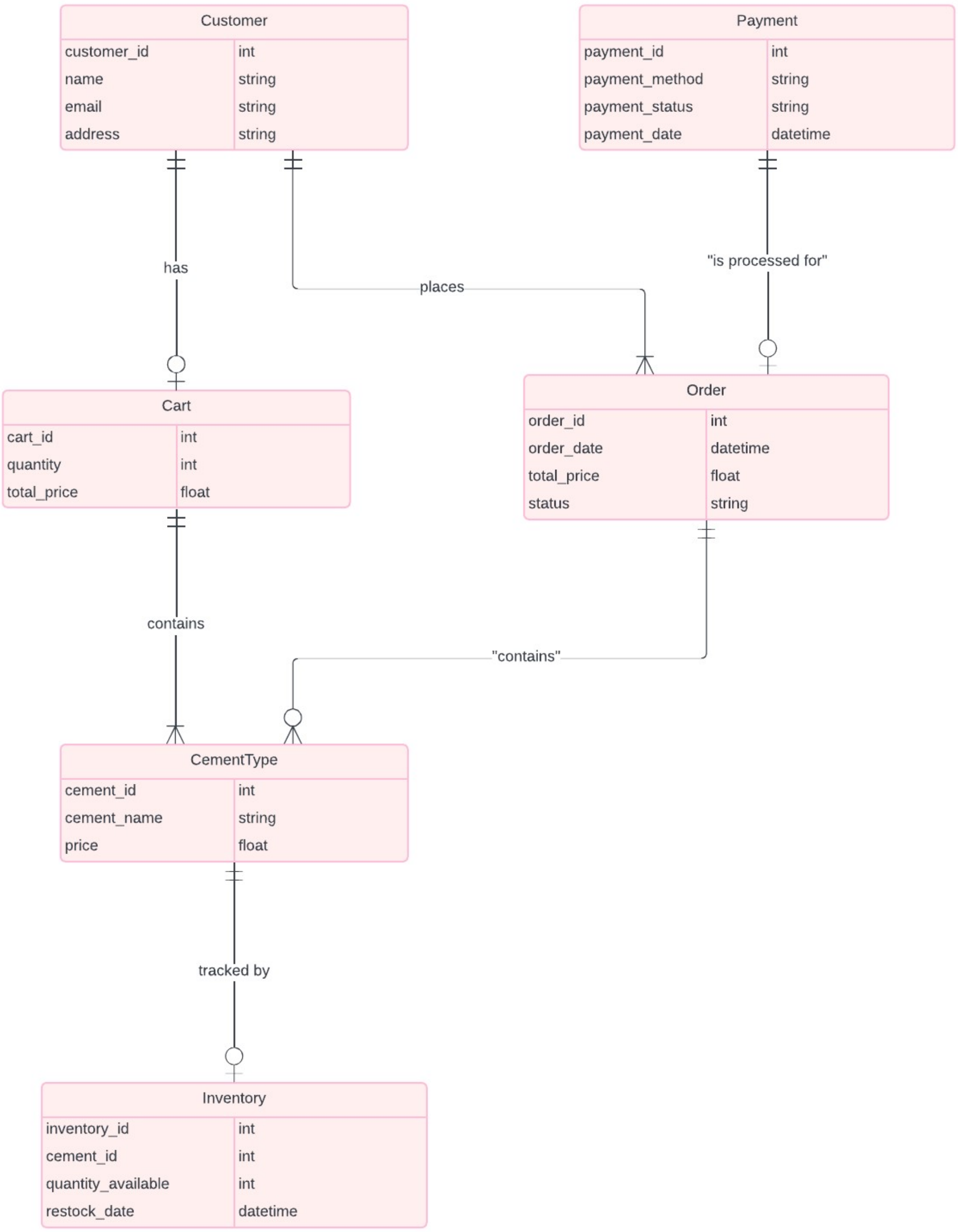
CHAPTER 5

SYSTEM DESIGN

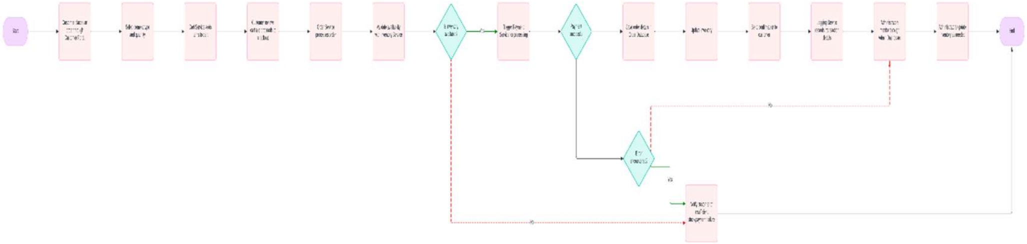
5.1 USE CASE DIAGRAM



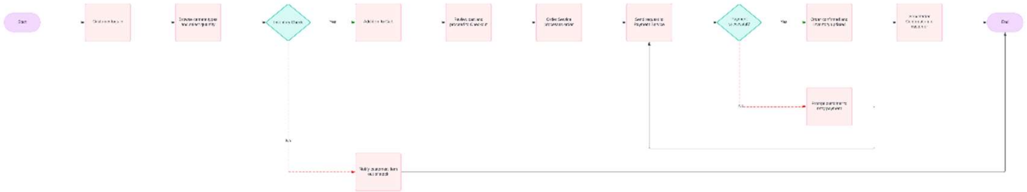
5.2. ER DIAGRAM



5.3 DATA FLOW DIAGRAM



5.4.ACTIVITY DIAGRAM


Start
Yes
Add item to Cart
Send request to Payment Service
Yes
Order confirmed and inventory updated
End
No
Notify customer: Item out of stock

CHAPTER 6

SAMPLECODING

WELCOME.PHP

```
<?php
session_start();

// Initialize cart session if not already done
if (!isset($_SESSION['cart'])) {
   $_SESSION['cart'] = [];
}

if (!isset($_SESSION['username'])) {
   header("Location: index.php"); // Redirect to login page if not logged in
   exit();
}

include 'db_connect.php'; // Database connection file

// Fetch user orders from the database
$username = $_SESSION['username'];
$sql = "SELECT * FROM orders WHERE username = ?";
$stmt = $conn->prepare($sql);
$stmt->bind_param("s", $username);
$stmt->execute();
$result = $stmt->get_result();

// Check if a cart item is being added
```

```
if ($_SERVER['REQUEST_METHOD'] == 'POST' &&
isset($_POST['add_to_cart'])) {
    $cart_item = [
        'cement_type' => $_POST['cement_type'],
        'quantity' => (int)$_POST['quantity'],
        'total_price' => $_POST['total_price']
    ];
    // Add to session cart
    $_SESSION['cart'][] = $cart_item;
}

// Place Order
if (isset($_POST['place_order'])) {
    // Loop through the cart and process each item
    foreach ($_SESSION['cart'] as $item) {
        // Insert order into the database
        $order_sql = "INSERT INTO orders (username, cement_type, quantity,
total_price, order_date) VALUES (?, ?, ?, ?, ?)";
        $stmt = $conn->prepare($order_sql);
        $order_date = date('Y-m-d');
        $stmt->bind_param("ssids", $username, $item['cement_type'],
$item['quantity'], $item['total_price'], $order_date);
        $stmt->execute();
    }
    // Clear the cart after placing the order
    $_SESSION['cart'] = [];
    header("Location: index.php?order_success=1"); // Redirect to a success page
    exit();
}
?>
```

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Welcome</title>
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-
alpha1/dist/css/bootstrap.min.css" rel="stylesheet">
    <style>
        body {
            background-color: #f4f7f6;
            font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
            color: #2c3e50;
            overflow-x: hidden;
        }
        header {
            background-color: #2980b9;
            padding: 20px;
            text-align: center;
            color: white;
        }
        .container {
            max-width: 1200px;
            margin: 30px auto;
        }
        .card {
            border: none;
            border-radius: 12px;
            box-shadow: 0 4px 15px rgba(0, 0, 0, 0.05);
            margin-bottom: 30px;
            background-color: white;
        }
        .table thead th {
            background-color: #2980b9;
            color: white;
            text-align: center;
            padding: 15px;
        }
        .table tbody td {
            text-align: center;
            padding: 15px;
        }
    </style>
</head>
<body>
    <header>
        <h1>Welcome, <?php echo $_SESSION['username']; ?>!</h1>
    </header>

    <div class="container">
        <!-- Orders Section -->
        <div class="card p-4">
            <h2>Your Orders</h2>
            <table class="table table-striped">
                <thead>
                    <tr>
                        <th>Order ID</th>
                        <th>Product</th>
                        <th>Quantity</th>
                        <th>Total Price</th>
                        <th>Date</th>
                        <th>Actions</th>
                    </tr>
                </thead>
                <tbody>
                    <?php
                    if ($result->num_rows > 0) {
                        while($row = $result->fetch_assoc()) {
                            echo "<tr>
                                    <td>" . $row['order_id'] . "</td>
                                    <td>" . $row['cement_type'] . "</td>
                                    <td>" . $row['quantity'] . "</td>
                                    <td>₹" . $row['total_price'] . "</td>
                                    <td>" . $row['order_date'] . "</td>
                                    <td><form method='post'
action='remove_order.php'><input type='hidden' name='order_id' value='" .
$row['order_id'] . "'><button class='btn btn-
danger'>Cancel</button></form></td>
                                  </tr>";
                        }
                    } else {
                        echo "<tr><td colspan='6'>No orders found.</td></tr>";
                    }
                    ?>
                </tbody>
            </table>
        </div>
        <!-- Order Form -->
        <div class="card p-4 mt-4">
            <h2>Order Cement</h2>
            <form action="" method="post">
                <div class="row g-3">
                    <div class="col-md-6">
                        <label for="cementType" class="form-label">Select Cement Type</label>
                        <select id="cementType" name="cement_type" class="form-select" required>
                            <option value="portland_cement">Portland Cement (₹100)</option>
                            <option value="ready_mix">Ready Mix Cement (₹120)</option>
                            <option value="white_cement">White Cement (₹150)</option>
                            <option value="hydraulic_cement">Hydraulic Cement (₹130)</option>
                            <option value="fly_ash_cement">Fly Ash Cement (₹110)</option>
                            <option value="colored_cement">Colored Cement (₹140)</option>
                        </select>
                    </div>
                    <div class="col-md-6">
                        <label for="quantity" class="form-label">Quantity (in bags)</label>
                        <input type="number" id="quantity" name="quantity" class="form-control" min="1" value="1" required>
                    </div>
                    <input type="hidden" name="total_price" id="totalPrice">
                </div>
                <button type="submit" class="btn btn-primary mt-3"
name="add_to_cart">Add to Cart</button>
            </form>
        </div>

        <!-- Cart Section -->
        <div class="card p-4 mt-4">
            <h2>Your Cart</h2>
            <table class="table table-striped">
                <thead>
                    <tr>
                        <th>Cement Type</th>
                        <th>Quantity</th>
                        <th>Total Price</th>
                    </tr>
                </thead>
                <tbody>
                    <?php if (!empty($_SESSION['cart'])): ?>
                        <?php foreach ($_SESSION['cart'] as $item): ?>
                            <tr>
                                <td><?php echo $item['cement_type']; ?></td>
                                <td><?php echo $item['quantity']; ?></td>
                                <td>₹<?php echo $item['total_price']; ?></td>
                            </tr>
                        <?php endforeach; ?>
                    <?php else: ?>
                        <tr>
                            <td colspan="3">No items in cart.</td>
                        </tr>
                    <?php endif; ?>
                </tbody>
            </table>
            <form method="post">
                <button type="submit" name="place_order" class="btn btn-success">Place Order</button>
            </form>
        </div>
    </div>

    <script>
        const cementPrices = {
            portland_cement: 100,
            ready_mix: 120,
            white_cement: 150,
            hydraulic_cement: 130,
            fly_ash_cement: 110,
            colored_cement: 140
        };

        function calculateTotalPrice() {
            const cementType = document.getElementById('cementType').value;
            const quantity = parseInt(document.getElementById('quantity').value);
            const totalPrice = quantity * cementPrices[cementType];
            document.getElementById('totalPrice').value = totalPrice.toFixed(2);
        }
        document.getElementById('cementType').addEventListener('change', calculateTotalPrice);
        document.getElementById('quantity').addEventListener('input', calculateTotalPrice);
        calculateTotalPrice();
    </script>
</body>
</html>
```

ADMIN_DASHBOARD

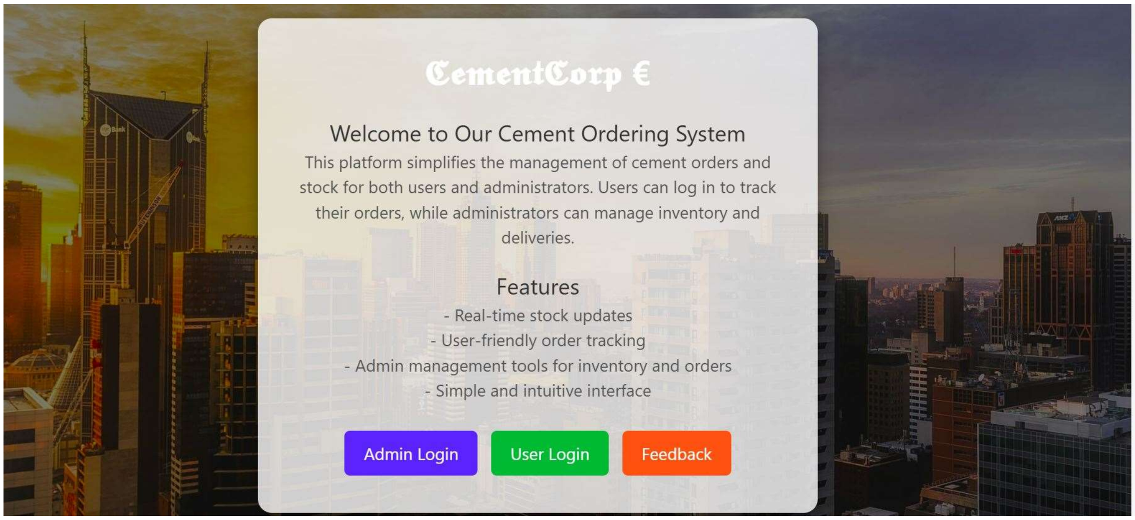
```
<?php
session_start();

if (!isset($_SESSION['username'])) {
    header("Location: index.php"); // Redirect to login page if not logged in
    exit();
}

include 'db_connect.php'; // Database connection file

// Fetch user orders from the database
$username = $_SESSION['username'];
$sql = "SELECT * FROM orders WHERE username = ?";
$stmt = $conn->prepare($sql);
$stmt->bind_param("s", $username);
$stmt->execute();
$result = $stmt->get_result();

// Fetch product data for stock updates
```

```
$products_query = "SELECT * FROM products";
$products_result = mysqli_query($conn, $products_query);

// Set a maximum stock level for percentage calculations
$max_stock = 500;

?>

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Admin Dashboard</title>
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-
alpha1/dist/css/bootstrap.min.css" rel="stylesheet">
    <style>
        body {
            background-color: #f4f7f6;
            font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
            color: #2c3e50;
            overflow-x: hidden;
        }
        header {
            background-color: #2980b9;
            padding: 20px;
            text-align: center;
            color: white;
        }
        .container {
            max-width: 1200px;
            margin: 30px auto;
        }
        .card {
            border: none;
            border-radius: 12px;
            box-shadow: 0 4px 15px rgba(0, 0, 0, 0.05);
            margin-bottom: 30px;
            background-color: white;
        }
        .table thead th {
            background-color: #2980b9;
            color: white;
            text-align: center;
            padding: 15px;
        }
        .table tbody td {
            text-align: center;
            padding: 15px;
        }
        .stock-percentage {
            color: green;
            font-weight: bold;
        }
    </style>
</head>
<body>
    <header>
        <h1>Welcome, <?php echo $_SESSION['username']; ?>!</h1>
    </header>

    <div class="container">
        <!-- Orders Section -->
        <div class="card p-4">
            <h2>Your Orders</h2>
            <table class="table table-striped">
                <thead>
                    <tr>
                        <th>Order ID</th>
                        <th>Product</th>
                        <th>Quantity</th>
                        <th>Total Price</th>
                        <th>Date</th>
                        <th>Actions</th>
                    </tr>
                </thead>
                <tbody>
                    <?php
                    if ($result->num_rows > 0) {
                        while($row = $result->fetch_assoc()) {
                            echo "<tr>
                                    <td>" . $row['order_id'] . "</td>
                                    <td>" . $row['cement_type'] . "</td>
                                    <td>" . $row['quantity'] . "</td>
                                    <td>₹" . $row['total_price'] . "</td>
                                    <td>" . $row['order_date'] . "</td>
```

```
                                    <td><form method='post' action='remove_order.php'><input type='hidden' name='order_id' value='" . $row['order_id'] . "'><button class='btn btn-danger'>Cancel</button></form></td>
                                  </tr>";
                        }
                    } else {
                        echo "<tr><td colspan='6'>No orders found.</td></tr>";
                    }
                    ?>
                </tbody>
            </table>
        </div>

        <!-- Stock Update Section -->
        <div class="card p-4 mt-4">
    <h2>Update Stock Levels</h2>
    <form method="POST" action="update_stock.php">
        <div class="row g-3">
            <div class="col-md-6">
                <label for="product_id" class="form-label">Select Product</label>
                <select id="product_id" name="product_id" class="form-select" required>
                    <option value="" disabled selected>Select a product</option>
                    <option value="1">Portland Cement (Price: ₹100)</option>
                    <option value="2">Ready Mix (Price: ₹120)</option>
                    <option value="3">White Cement (Price: ₹150)</option>
                    <option value="4">Hydraulic Cement (Price: ₹130)</option>
                    <option value="5">Fly Ash Cement (Price: ₹110)</option>
                    <option value="6">Colored Cement (Price: ₹140)</option>
                </select>
            </div>
            <div class="col-md-6">
                <label for="new_stock" class="form-label">New Stock Level</label>
                <input type="number" id="new_stock" name="new_stock"
class="form-control" required>
            </div>
        </div>
        <button type="submit" class="btn btn-primary mt-3">Update
Stock</button>
    </form>
</div>

</script>



        <!-- Cement Type Selection for Total Amount Calculation -->
        <div class="card p-4 mt-4">
            <h2>Select Cement Type and Quantity</h2>
            <form method="POST" action="calculate_total.php">
                <div class="row g-3">
                    <div class="col-md-6">
                        <label for="cementType" class="form-label">Select Cement
Type</label>
                        <select id="cementType" name="cement_type" class="form-
select">
                            <option value="portland_cement">Portland Cement
(₹100)</option>
                            <option value="ready_mix">Ready Mix Cement
(₹120)</option>
```

```
                            <option value="white_cement">White Cement (₹150)</option>
                            <option value="hydraulic_cement">Hydraulic Cement (₹130)</option>
                            <option value="fly_ash_cement">Fly Ash Cement (₹110)</option>
                            <option value="colored_cement">Colored Cement (₹140)</option>
                        </select>
                    </div>
                    <div class="col-md-6">
                        <label for="quantity" class="form-label">Quantity (in bags)</label>
                        <input type="number" id="quantity" name="quantity" class="form-control" min="1" value="1">
                    </div>
                </div>
                <button type="submit" class="btn btn-success mt-3">Calculate Total</button>
            </form>
        </div>
    </div>

    <!-- Stock and Cement Price Calculation JavaScript -->
    <script>
        const cementPrices = {
            portland_cement: 100,
            ready_mix: 120,
            white_cement: 150,
            hydraulic_cement: 130,
            fly_ash_cement: 110,
            colored_cement: 140
        };

        function calculateTotalPrice() {
            const cementType = document.getElementById('cementType').value;
            const quantity = parseInt(document.getElementById('quantity').value);
            const totalPrice = quantity * cementPrices[cementType];
            document.getElementById('totalPrice').value = totalPrice.toFixed(2);
        }

        document.getElementById('cementType').addEventListener('change', calculateTotalPrice);
        document.getElementById('quantity').addEventListener('input', calculateTotalPrice);
        calculateTotalPrice();
    </script>

</body>
</html>
```

CHAPTER 7
SCREENSHOTS



CementCorp €

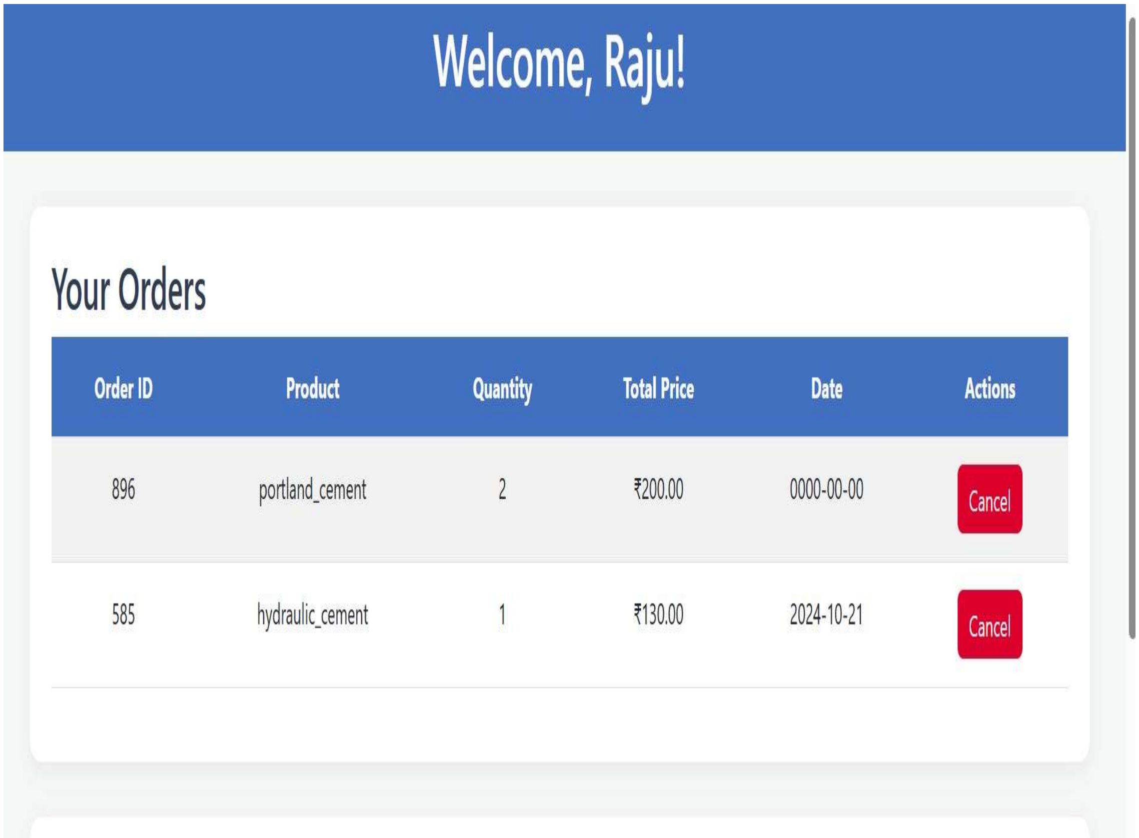
Welcome to Our Cement Ordering System

This platform simplifies the management of cement orders and stock for both users and administrators. Users can log in to track their orders, while administrators can manage inventory and deliveries.

Features

- Real-time stock updates
- User-friendly order tracking
- Admin management tools for inventory and orders
- Simple and intuitive interface

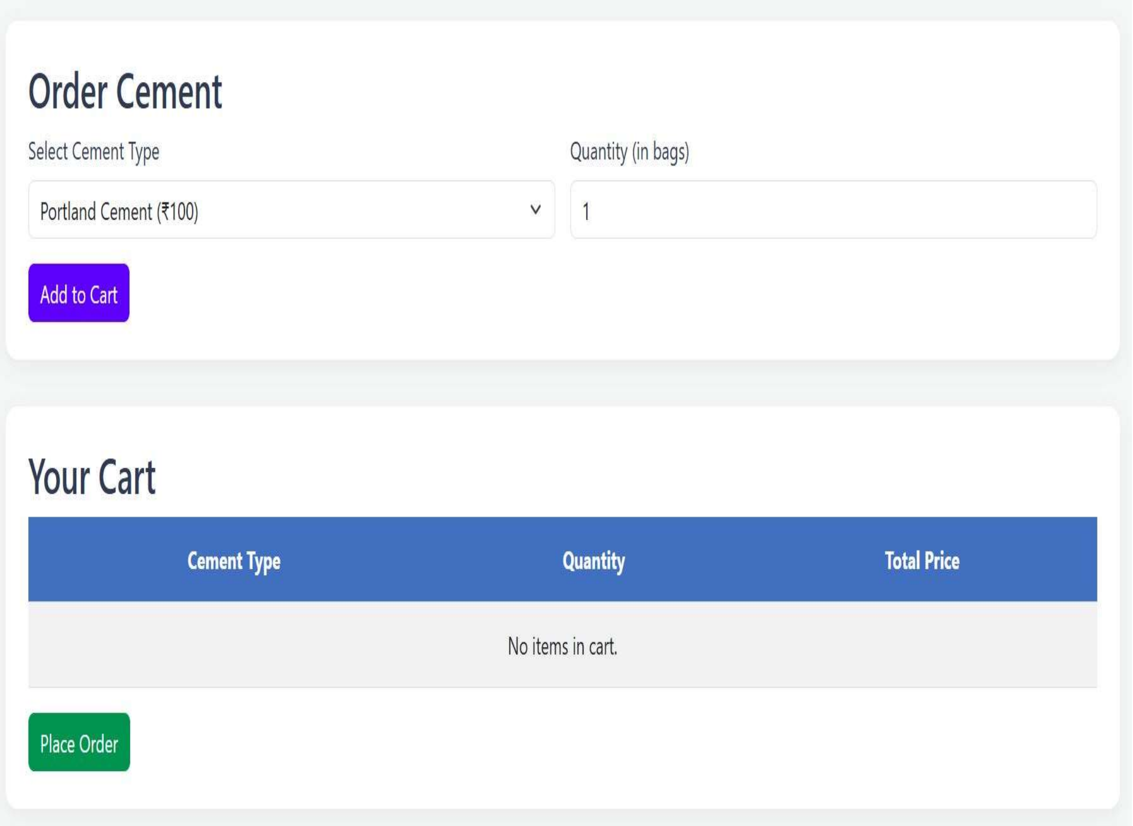
Fig.7.1 USER PAGE



Welcome, Raju!

Your Orders

Order ID	Product	Quantity	Total Price	Date	Actions
896	portland_cement	2	₹200.00	0000-00-00	Cancel
585	hydraulic_cement	1	₹130.00	2024-10-21	Cancel



Order Cement

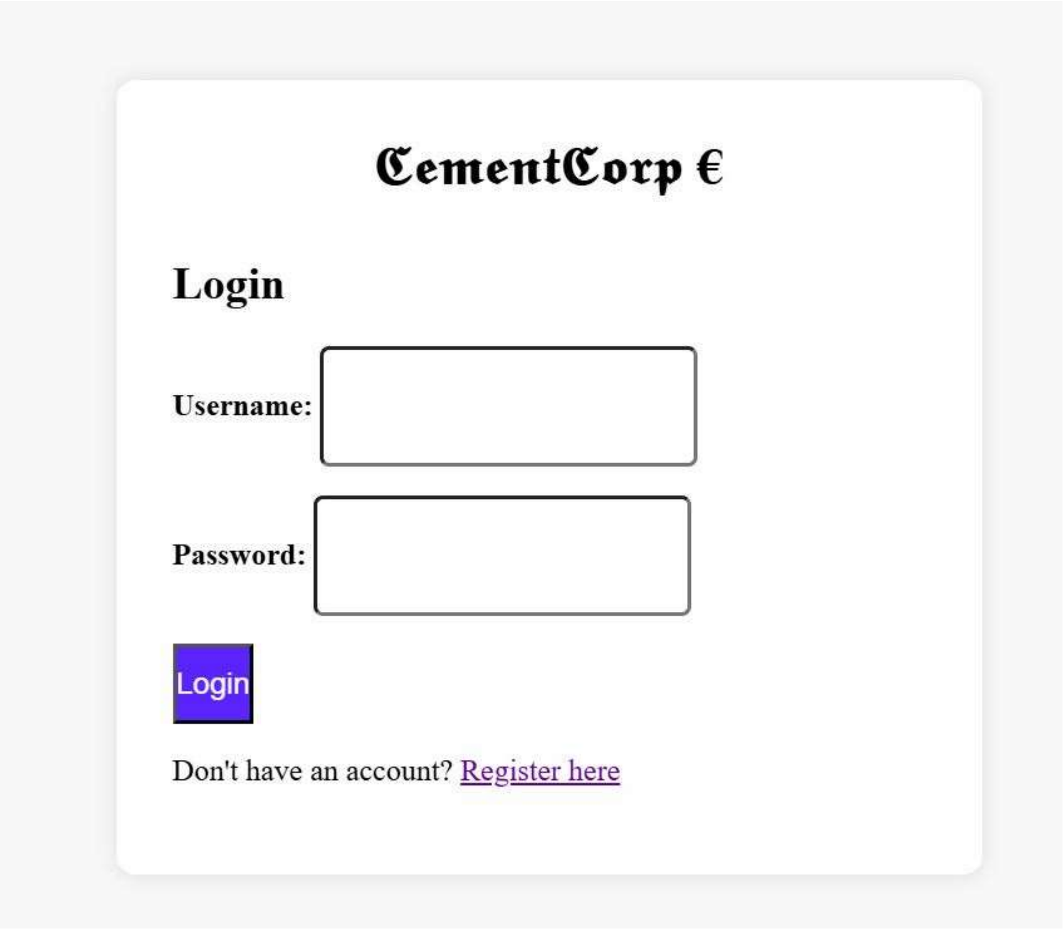
Select Cement Type: Portland Cement (₹100)

Quantity (in bags): 1

Your Cart

Cement Type	Quantity	Total Price
No items in cart.		

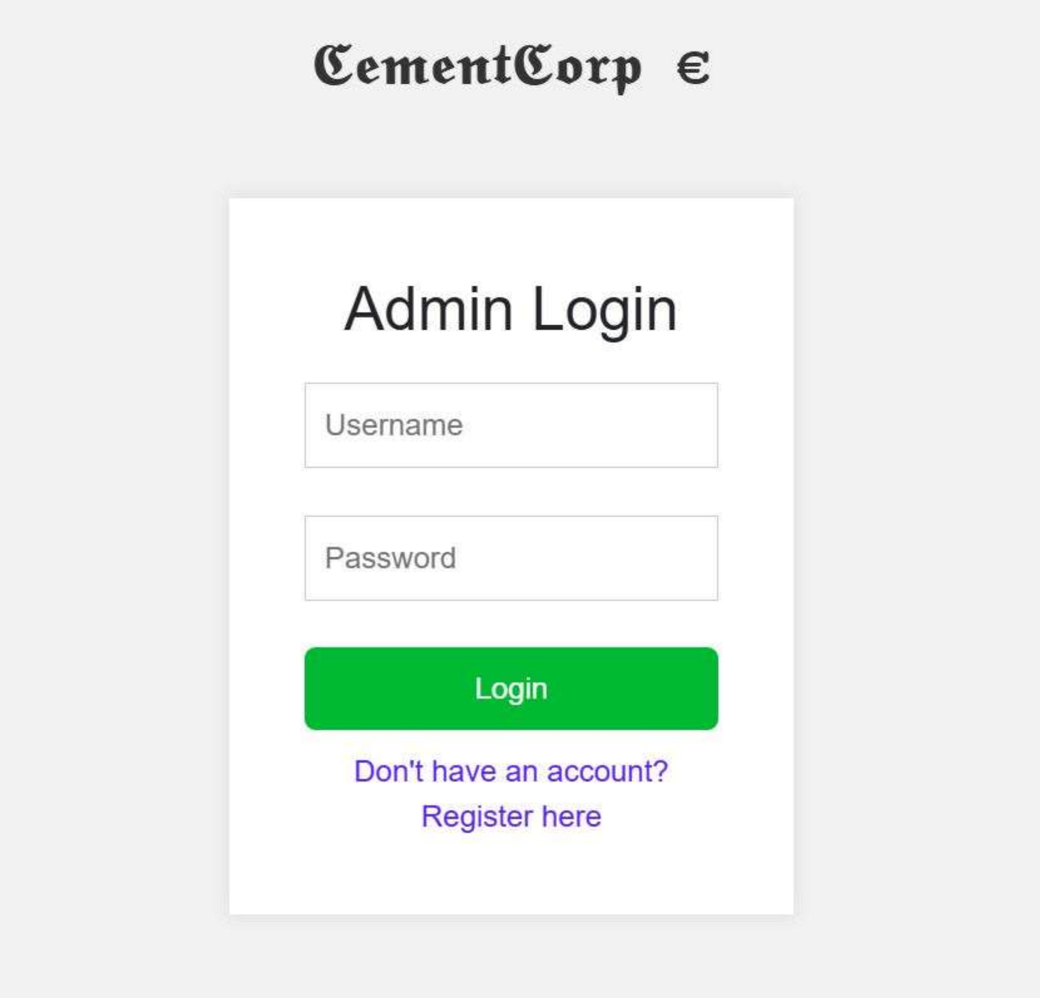
Fig. 7.2 ADMIN AND USER LOGIN



CementCorp €

Login

Don't have an account? Register here



CementCorp €

Admin Login

Don't have an account? Register here

CHAPTER 8
CONCLUSION

A cement ordering management system simplifies the entire process of managing cement-related services, enabling users to browse, compare, and place orders for various types of cement and quantities with ease. This system is essential for construction companies, suppliers, and individual contractors, as it improves operational efficiency by automating tasks like order placement, inventory management, and customer inquiries. For customers, the system provides a user-friendly interface where they can explore different types of cement, filter options based on their specific requirements (such as type, price, or quantity), and finalize their orders seamlessly.

A key advantage of such a system is its ability to manage real-time updates regarding stock availability, ensuring that customers receive up-to-date information on product availability, reducing the chance of issues like ordering out-of-stock products. The system can also provide recommendations based on previous orders, popular trends, or project requirements, helping customers make informed decisions and improving overall satisfaction.

However, like in the travel system, this cement ordering management system does not include a payment gateway. Payments must be processed through separate channels, which can be an advantage for businesses that prefer to integrate third-party services for secure transactions. This separation of the financial aspect allows businesses to focus on streamlining the ordering and fulfillment process while ensuring security through external payment solutions.

In summary, this type of cement ordering system greatly enhances the efficiency of the cement procurement process, benefiting both suppliers and customers by automating manual tasks and providing real-time data, without directly handling payment processing.

REFERENCES

1. HTML,CSS,JS : https://www.w3school.com
2. PHP, MYSQL : www.youtube.com
3. Eloquent JavaScript by Marijn Haverbeke