EFFICIENT BULK MATERIAL LOGISTICS

A MINI PROJECT REPORT

Submitted by

MANIKANDAN S- 2116220701159

in partial fulfillment of the award of the degree

of

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING

RAJALAKSHMI ENGINEERING COLLEGE

AUTONOMOUS, CHENNAI

NOV/DEC, 2024

BONAFIDE CERTIFICATE

Certified that this project report "**EFFICIENT BULK MATERIAL LOGISTICS**" is the bonafide work of **"MANIKANDAN S (220701159)"** 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 Practical Examination held on _____

INTERNAL EXAMINER **EXTERNAL EXAMINER**

ACKNOWLEDGEMENT

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

I am greatly indebted to my respected and honorable 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 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 **Bulk Transporters** project is a web-based platform designed to streamline the process of bulk material ordering and transportation. Aimed primarily at businesses and individuals seeking large-scale materials, such as construction aggregates, minerals, or other bulk goods, the platform provides a seamless interaction between producers (suppliers of bulk materials) and consumers (businesses or individual buyers) in need of materials.

The primary functionalities of **Bulk Transporters** include product catalog management, real-time stock updates, an order processing system, and a secure payment integration. **Consumers** can browse a diverse range of bulk materials, view product details, check stock levels, and place orders by specifying the required quantity and preferred delivery location. To ensure product availability, the system checks real-time stock levels before allowing order placement. Upon successful order creation, consumers receive a confirmation along with an estimated delivery timeline.

For **Admins**, the platform includes a comprehensive management dashboard where they can add, update, and manage the product catalog, adjust stock levels, and oversee orders in progress. The **Admin** role is critical for ensuring the accuracy of product information and the timely replenishment of stock levels to meet consumer demand. Additionally, the platform supports session management, allowing secure access control for both consumer and admin users.

The **Rithi Bulk Transporters** project employs a scalable architecture, combining **Flask** as the backend framework, **MySQL** for database management, and **HTML, CSS, and JavaScript** for the user interface. The system also incorporates **Bootstrap** for responsive design, ensuring accessibility across devices, and it maintains a centralized stock management system to update material quantities as orders are processed.

TABLE OF CONTENTS

CHAPTER NO	TITLE	PAGE
	ABSTRACT	4
1	INTRODUCTION	6
1.1	INTRODUCTION	6
1.2	SCOPE OF THE WORK	6
1.3	AIM AND OBJECTIVES OF THE PROJECT	7
2.	SYSTEM SPECIFICATIONS	8
2.1	HARDWARE SPECIFICATIONS	8
2.2	SOFTWARE SPECIFICATIONS	8
3	ARCHITECTURE DIAGRAM	9
4	MODULE DESCRIPTION	10
5	SYSTEM DESIGN	12
5.1	USE CASE DIAGRAM	12
5.2	ER DIAGRAM	13
5.3	DATA FLOW DIAGRAM	14
5.4	ACTIVITY DIAGRAM	15
6	SAMPLE CODING	16
7	SCREENSHOTS	20
8	CONCLUSION	30
	REFERENCES	25

CHAPTER 1
INTRODUCTION

1. INTRODUCTION

In this application, users can perform essential operations related to bulk material transport. The system allows users to create and manage orders for various building materials, including specifying the quantity, location for delivery, and preferred delivery time. Each product has a unique identification number that helps in tracking and managing stock levels effectively. Users can place orders based on available stock, while the admin has the capability to update stock quantities as needed. The system ensures that orders are fulfilled efficiently by decreasing the stock level whenever a new order is placed. Additionally, users can view their order history and receive notifications regarding order status, enhancing the overall user experience.

1.2 SCOPE OF THE WORK

The Bulk Material Transport Management System encompasses a wide range of functionalities aimed at streamlining the process of transporting building materials. This includes the development of a user-friendly interface for consumers and administrators, facilitating efficient order placement and management. The system incorporates real-time stock level monitoring to ensure that users can only place orders based on available quantities. Additionally, it allows administrators to update stock levels seamlessly, thereby improving inventory management. The implementation of a secure login system ensures that user data is protected, while the order tracking feature enhances transparency in the delivery process. Moreover, the project aims to integrate data analytics for better forecasting of demand, thereby optimizing resource allocation. Public awareness campaigns will be included to educate users about the benefits of using the system, ensuring broader adoption. Overall, the project takes a comprehensive approach to improve the efficiency and effectiveness of bulk material transport operations.

1.3 AIM AND OBJECTIVES OF THE PROJECT

The aim of the Bulk Material Transport Management System is to create a comprehensive and efficient platform that optimizes the transportation and management of building materials. The system seeks to enhance user experience by providing streamlined order processing, real-time inventory management, and secure transactions, thereby improving operational efficiency and reducing delays in the delivery of materials.

Objectives:

1. **Streamlined Order Processing**: Develop an intuitive interface for users to easily place orders for building materials, ensuring a smooth and efficient ordering experience
2. **Real-Time Inventory Management:** Implement a robust inventory management system that allows administrators to monitor stock levels in real time and make necessary adjustments to quantities, ensuring availability and reducing the risk of stockouts.
3. **Secure User Authentication:** Establish a secure login system that protects user data and ensures only authorized personnel can access sensitive information, enhancing overall system security.

CHAPTER 2

SYSTEM SPECIFICATIONS

1. HARDWARE SPECIFICATIONS

Processor: Intel Core i3 or Higher

Memory Size: 16 GB (Minimum)

HDD: 256 GB SSD (Minimum)

Network: High-Speed Internet Connection (Broadband recommended)

Display: Minimum resolution of 1366 x 768 pixels

2. SOFTWARE SPECIFICATIONS

Operating System: Windows 10 or Higher

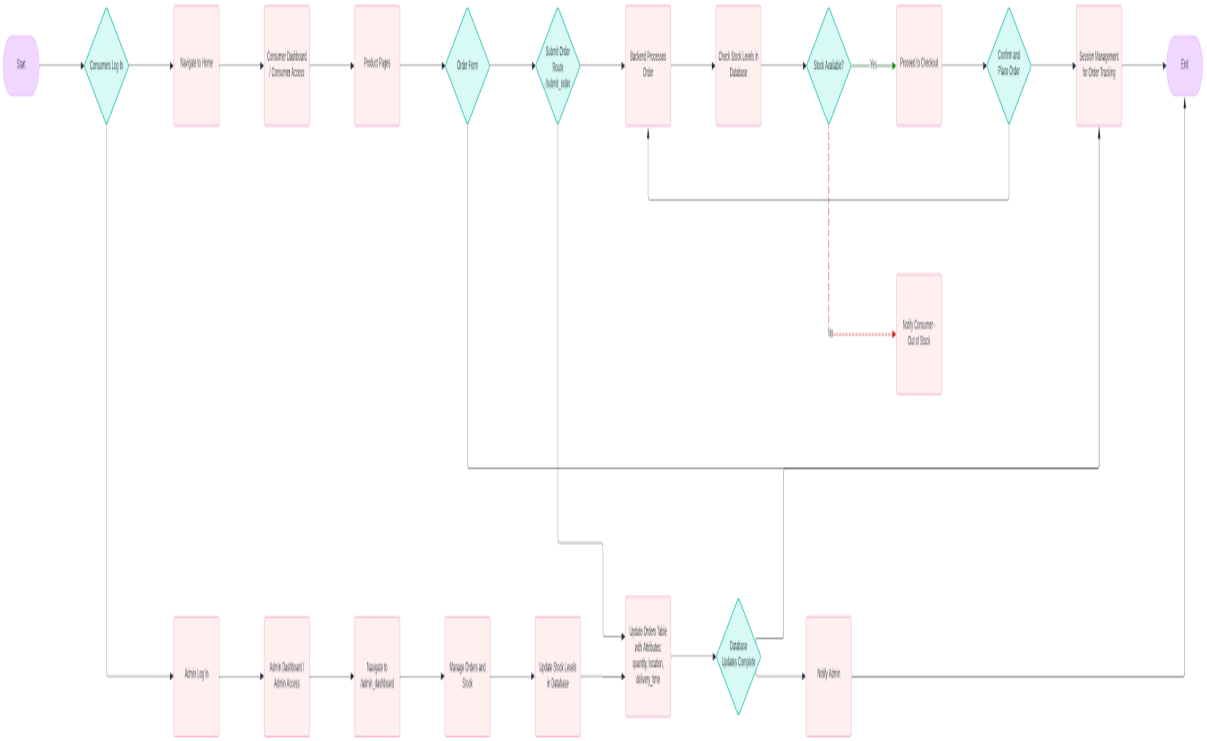
Front-End: HTML, CSS, JavaScript

Back-End: Python (Flask), MySQL

Development Environment: Visual Studio Code or any preferred IDE

CHAPTER 3

ARCHITECTURE DIAGRAM



CHAPTER 4

MODULE DESCRIPTION

4.1. User Registration and Login Module

The User Registration and Login Module for the Bulk Material Transport Management System is designed to provide secure and efficient access to authorized users, including administrators, producers, and consumers. Users can create profiles by providing essential information, undergo verification processes, and receive role-based access. This ensures accountability, data integrity, and the protection of sensitive information within the system. The module enables seamless user management, facilitating easy login and secure access to various functionalities, such as order placement and stock management.

4.2. Admin Module

The Admin Module within the Bulk Material Transport Management System serves as the command center, equipping administrators with essential tools for overseeing and managing all operational activities. It enables efficient user management, allowing the addition, modification, and deactivation of accounts with diverse roles and permissions, including producers and consumers. The module provides a comprehensive dashboard with analytics and reporting capabilities, facilitating data-driven decision-making and regulatory compliance. Administrators can configure system settings, manage product inventory, and communicate critical updates to stakeholders. With robust security features, incident response tools, and collaboration functionalities, the Admin Module ensures a coordinated, secure, and adaptive response to inventory management and order fulfillment, maintaining the integrity and efficiency of the transport management system.

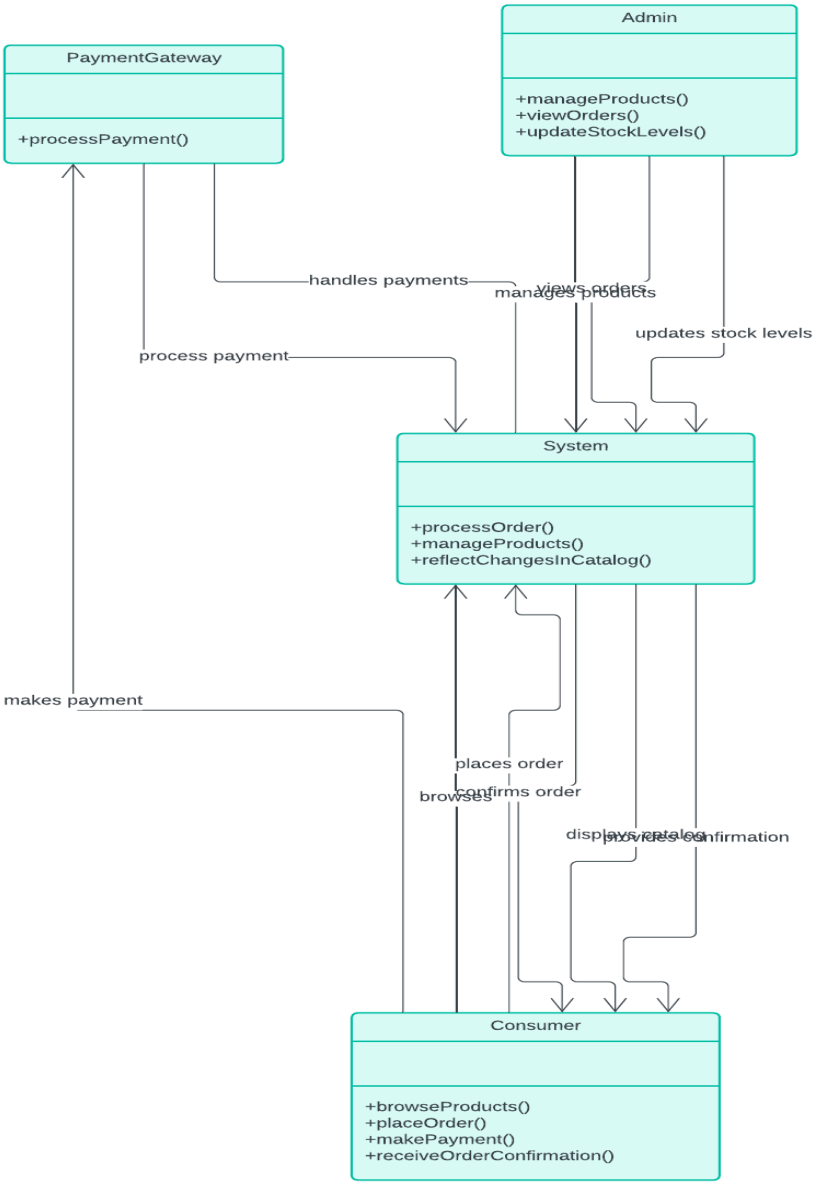
4.3 Inventory Management Module

The Inventory Management Module in the Bulk Material Transport Management System is designed to efficiently manage stock levels for various building materials. It allows users to add, modify, and delete inventory items as needed. Additionally, the module provides alerts for low stock levels, ensuring that users are promptly informed when inventory needs replenishing. It also tracks changes to inventory, enabling accurate oversight of stock levels and facilitating better decision-making for inventory management.

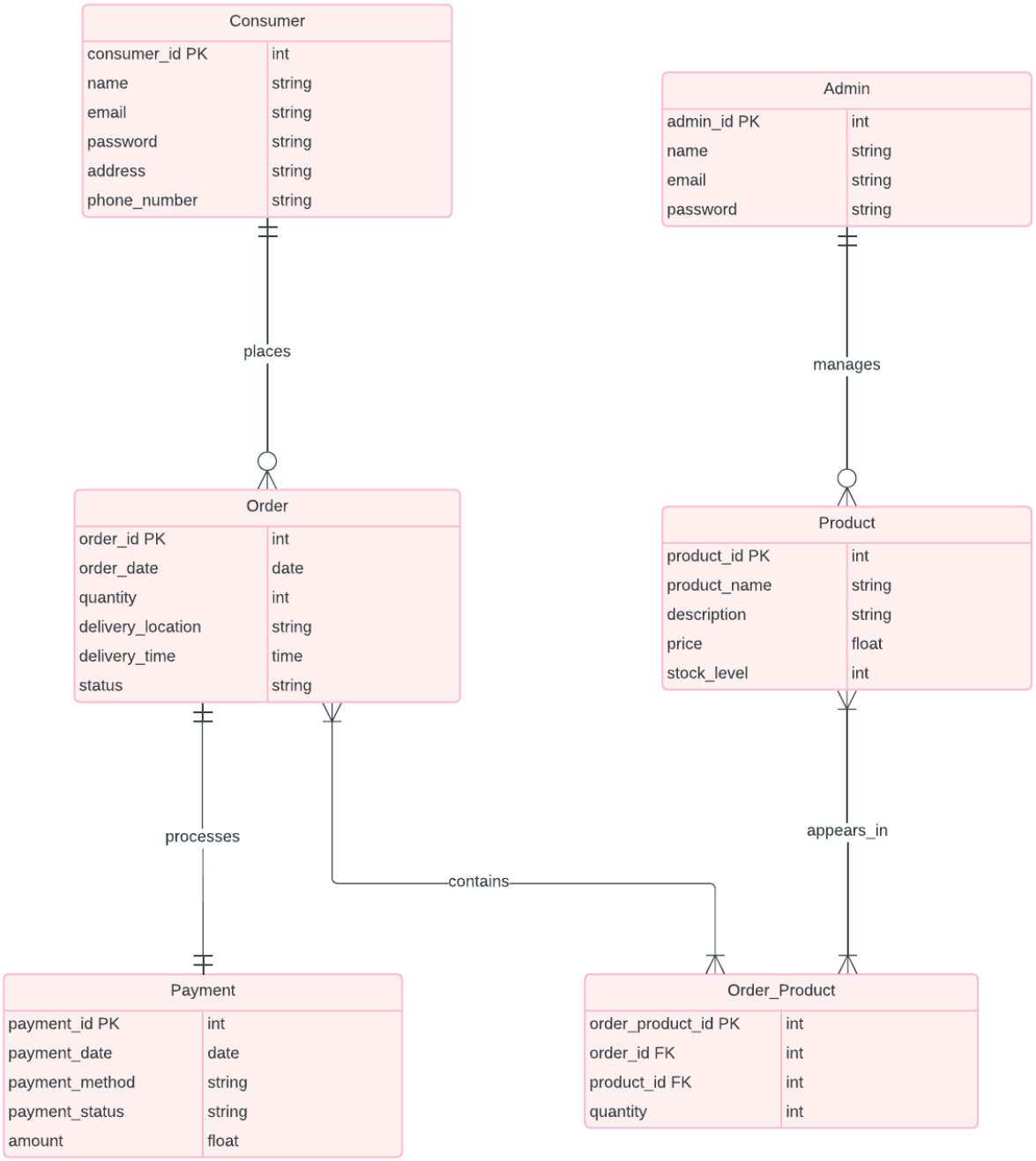
CHAPTER 5

SYSTEM DESIGN

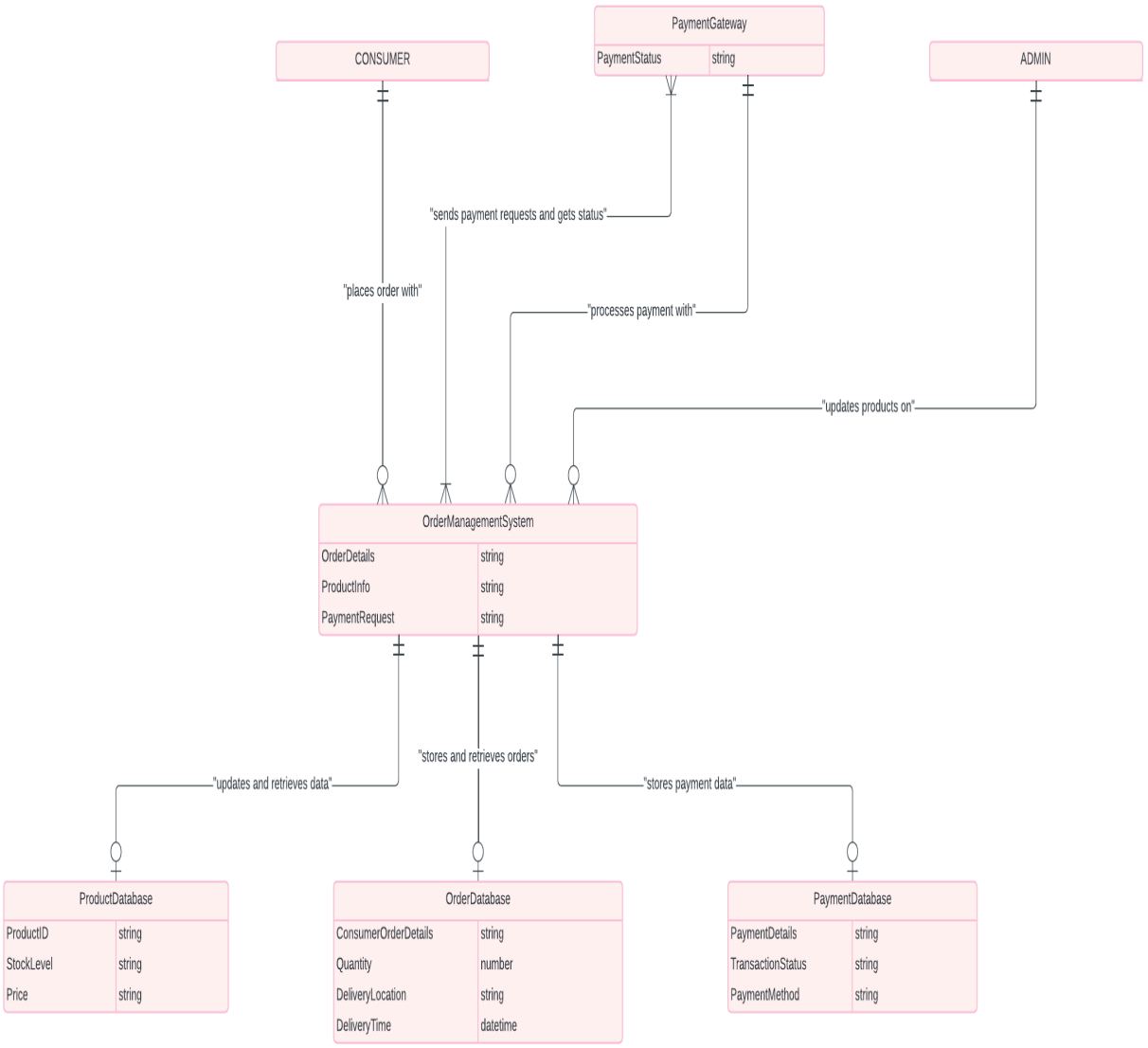
5.1 USE CASE DIAGRAM



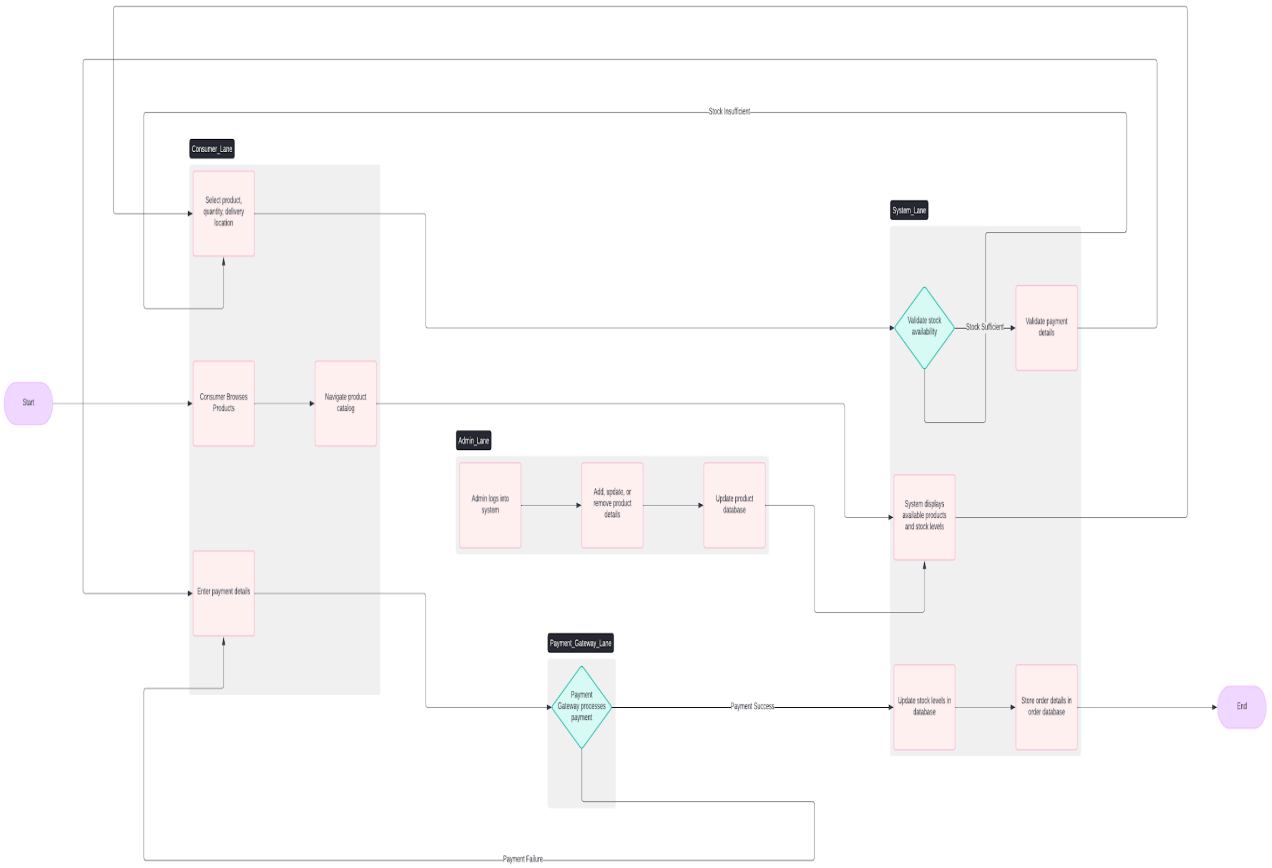
5.2 ER DIAGRAM



5.3 DFD DIAGRAM



5.4 ACTIVITY DIAGRAM



CHAPTER 6

SAMPLE CODING

HOME.HTML

```
<!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
   <title>Raichu</title>
   <link rel="stylesheet" href="static/customer_home.css">
   <link rel="preconnect" href="https://fonts.googleapis.com">
   <link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>
   <link
href="https://fonts.googleapis.com/css2?family=Josefin+Sans:ital,wght@0,100..70
0;1,100..700&family=Nunito:ital,wght@0,200..1000;1,200..1000&display=swap"
rel="stylesheet">
</head>
<body>
   <header>
      <div class="logo">R Bulk Transporters</div>
      <nav>
         <a href="{{ url_for('admin_dashboard') }}">Admin Dashboard</a>
         {% if not session.get('consumer_logged_in') %}
            <a href="{{ url_for('consumer_login') }}">Login</a>
         {% endif %}
         {% if session.get('consumer_logged_in') %}
            <a href="{{ url_for('consumer_logout') }}">Logout</a>
         {% endif %}
      </nav>
       </header>
   <main>
      <section class="hero">
         <div class="carousel">
 <div class="carousel-inner">
               <img src="/static/images/c1.jpg" alt="Promo 1" class="carousel-item
active">
               <img src="/static/images/c2.jpg" alt="Promo 2"
class="carousel-item">
               <img src="/static/images/c3.jpg" alt="Promo 3"
```

```
class="carousel-item">
            </div>
            <button class="carousel-control prev"
onclick="moveSlide(-1)">&#10094;</button>
            <button class="carousel-control next"
onclick="moveSlide(1)">&#10095;</button>
        </div>
    </section>
    <section class="product-listings">
        <h2>Product Types</h2>
        <div class="product-grid">
            <a href="{{ url_for('product_2') }}">
                <div class="product-item1">
                    <p class="materials">Agriculture Materials</p>
                </div>
            </a>
            <a href="{{ url_for('building_products') }}">
                <div class="product-item2">
                    <p class="materials" >Builiding material</p>
                </div>
            </a>
            <a href="{{ url_for('product_3') }}">
                <div class="product-item3">
                    <p class="materials">Chemical and industrial material</p>
                </div>
            </a>
        </div>
    </section>
     </main>
<footer>
    <div class="footer-links">
        <a href="{{ url_for('about') }}">About</a>
        <a href="{{ url_for('careers') }}">Careers</a>
        <a href="{{ url_for('privacy_policy') }}">Privacy Policy</a>
        <a href="{{ url_for('terms_of_service') }}">Terms of Service</a>
    </div>
    <div class="social-media">
        <a href="#"><img src="{{ url_for('static',
filename='images/facebook_icon.webp') }}" alt="Facebook"></a>
        <a
href="https://www.instagram.com/manickvishal14/profilecard/?igsh=ZDJucGJ1N2
ZoaDJt"><img src="{{ url_for('static', filename='images/instagram_icon.png') }}"
alt="Instagram"></a>
    </div>
```

```
    </footer>
    <script src="/static/home_js.js"></script>
</body>
</html>
```

app.py

```
from flask import Flask, render_template, request, redirect, url_for, session, flash
import mysql.connector
app = Flask(__name__)
app.secret_key = 'fghfjknrglsnvldnswer3432efbdaw3%##$r23d'
# Database connection setup
def create_connection():
    connection = mysql.connector.connect(
        host='localhost',
        user='host',  # Your MySQL username
        password='root',  # Your MySQL password
        database='rithi'  # Your database name)
    return connection
@app.route('/')
def home():
    return render_template('customer_login.html')
@app.route('/admin-login', methods=['GET', 'POST'])
def admin_login():
    if request.method == 'POST':
        username = request.form['username']
        password = request.form['password']
        # Admin login credentials hardcoded
        if username == 'admin' and password == 'adminpass':
            session['admin_logged_in'] = True
            return redirect(url_for('admin_dashboard'))
        else:
            return 'Invalid Admin credentials, please try again.'
    return render_template('admin_login.html')
@app.route('/consumer-login', methods=['GET', 'POST'])
def consumer_login():
    if request.method == 'POST':
        username = request.form['username']
        password = request.form['password']
        # Connect to the database and validate credentials
        conn = create_connection()
        cursor = conn.cursor(dictionary=True
        cursor.execute("SELECT password FROM users WHERE username = %s",
(username,))
        consumer = cursor.fetchone()  # Fetch the user data from the database
```

```
        if consumer and consumer['password'] == password:
            session['consumer_logged_in'] = True
            session['username'] = username
            print(f"Logged in as {username}")  # Debug print for testing
            return redirect(url_for('consumer_dashboard'))
        else:
            return 'Invalid Consumer credentials, please try again.'
 return render_template('customer_login.html')
if __name__ == '__main__':
    app.run(debug=True)
```

CUSTOMER_LOGIN.HTML

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Login</title>
    <link rel="stylesheet" href="/static/customer.css">
</head>
<body>
    <div class="container">
        {% with messages = get_flashed_messages(with_categories=True) %}
          {% if messages %}
            <ul class="flashes">
              {% for category, message in messages %}
                <li class="{{ category }}">{{ message }}</li>
              {% endfor %}
            </ul>
          {% endif %}
        {% endwith %}
        <form id="loginForm" class="login-form" action="{{ url_for('consumer_login')
}}" method="POST">
```

ADMIN_LOGIN.HTML

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Admin Login</title>
```

```
  <link rel="stylesheet" href="static/admin.css">

</head>
<body>
  <div class="login-container">
    <h2>Admin Login</h2>
    <form method="POST" action="/admin-login">
      <input type="text" name="username" placeholder="Username" required />
      <input type="password" name="password" placeholder="Password" required />
      <button type="submit">Login</button>
    </form>
```

CUSTOMER_DASHBOARD.HTML

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Raichu</title>
    <link rel="stylesheet" href="static/customer_home.css">
    <link rel="preconnect" href="https://fonts.googleapis.com">
    <link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>
<link
href="https://fonts.googleapis.com/css2?family=Josefin+Sans:ital,wght@0,100..700;1,10
0..700&family=Nunito:ital,wght@0,200..1000;1,200..1000&display=swap"
rel="stylesheet">
</head>
<body>
    <header>
        <div class="logo">R Bulk Transporters</div>
        <nav>
            <a href="{{ url_for('admin_login') }}">Admin Dashboard</a>
            <!-- Show the login button only if the user is NOT logged in -->
            {% if not session.get('consumer_logged_in') %}
                <a href="{{ url_for('consumer_login') }}">Login</a>
            {% endif %}
            <!-- Show the logout button only if the user is logged in -->
            {% if session.get('consumer_logged_in') %}
                <a href="{{ url_for('consumer_logout') }}">Logout</a>
            {% endif %}
        </nav>
    </header>
```

ADMIN_DASHBOARD.HTML

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Admin Dashboard - R Bulk Transporters</title>
    <link rel="stylesheet" href="static/customer_home.css">
    <link rel="stylesheet" href="static/admin_dashboard.css">
</head>
<body>
    <header>
        <div class="logo">R Bulk Transporters - Admin Dashboard</div>
        <nav>
            <a href="{{ url_for('home') }}">Home</a>
            <a href="{{ url_for('admin_logout') }}">Logout</a>
        </nav>
    </header>

    <main>
        <section class="orders-section">
            <h2>Customer Orders</h2>
            <table class="orders-table">
                <thead>
                    <tr>
                        <th>Order ID</th>
                        <th>Quantity</th>
                        <th>Location</th>
                        <th>Delivery Time</th>
                        <th>Action</th>
                    </tr>
                </thead>
                <tbody>
                    {% for order in orders %}
                    <tr>
                        <td>{{ order.id }}</td>
                        <td>{{ order.quantity }} tons</td>
                        <td>{{ order.location }}</td>
                        <td>{{ order.delivery_time }}</td>
                        <td>
                            <!-- Status Update Form -->
```

```
                    <form action="{{ url_for('update_status', order_id=order.id) }}"
method="POST">
                    <select name="status">
                    <option value="pending" {% if order.status == 'pending'
%}selected{% endif %}>Pending</option>
                    <option value="processing" {% if order.status == 'processing'
%}selected{% endif %}>Processing</option>
                    <option value="shipped" {% if order.status == 'shipped' %}selected{%
endif %}>Shipped</option>
                    <option value="delivered" {% if order.status == 'delivered'
%}selected{% endif %}>Delivered</option>
                    </select>
                    <button type="submit">Update Status</button>
                    </form>
```

CHAPTER 7

SCREEN SHOTS

Fig. 7.1 Home page

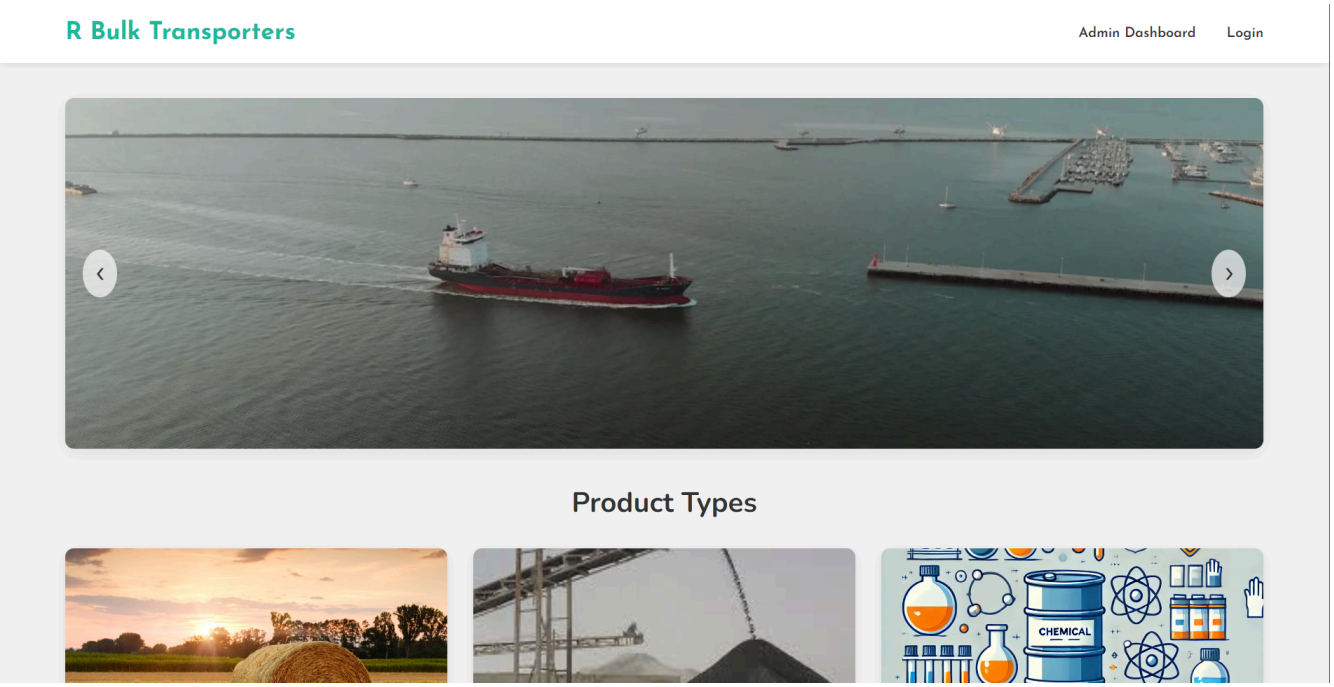


Fig. 7.2 Customer login page

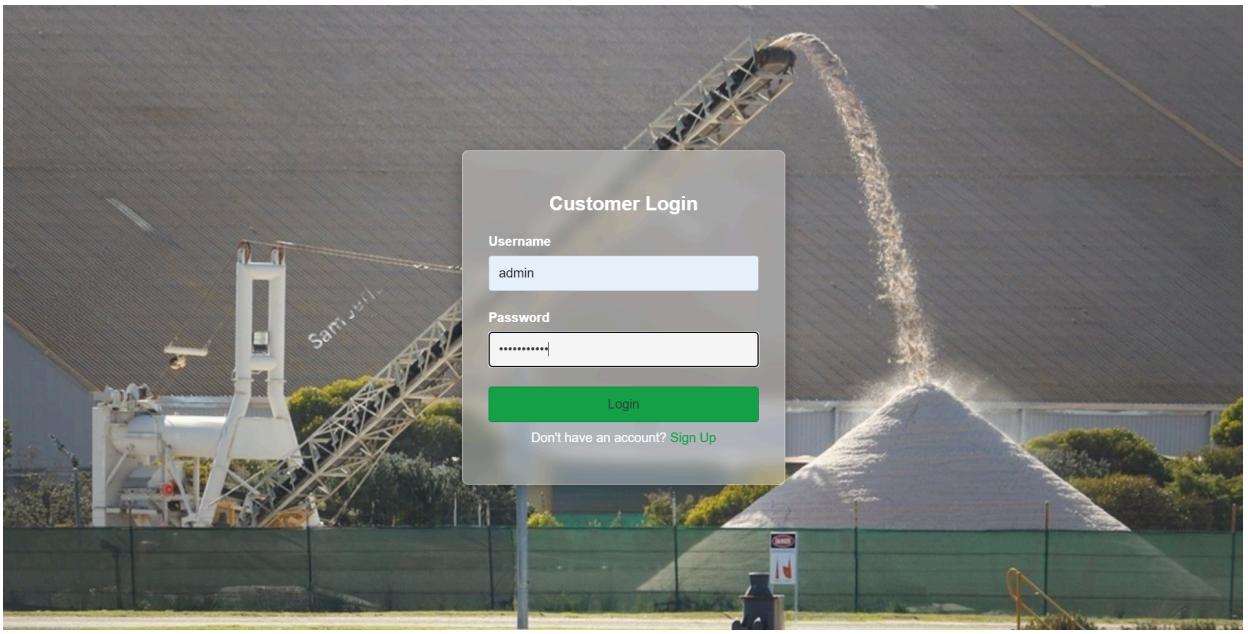


Fig. 7.3 Product page

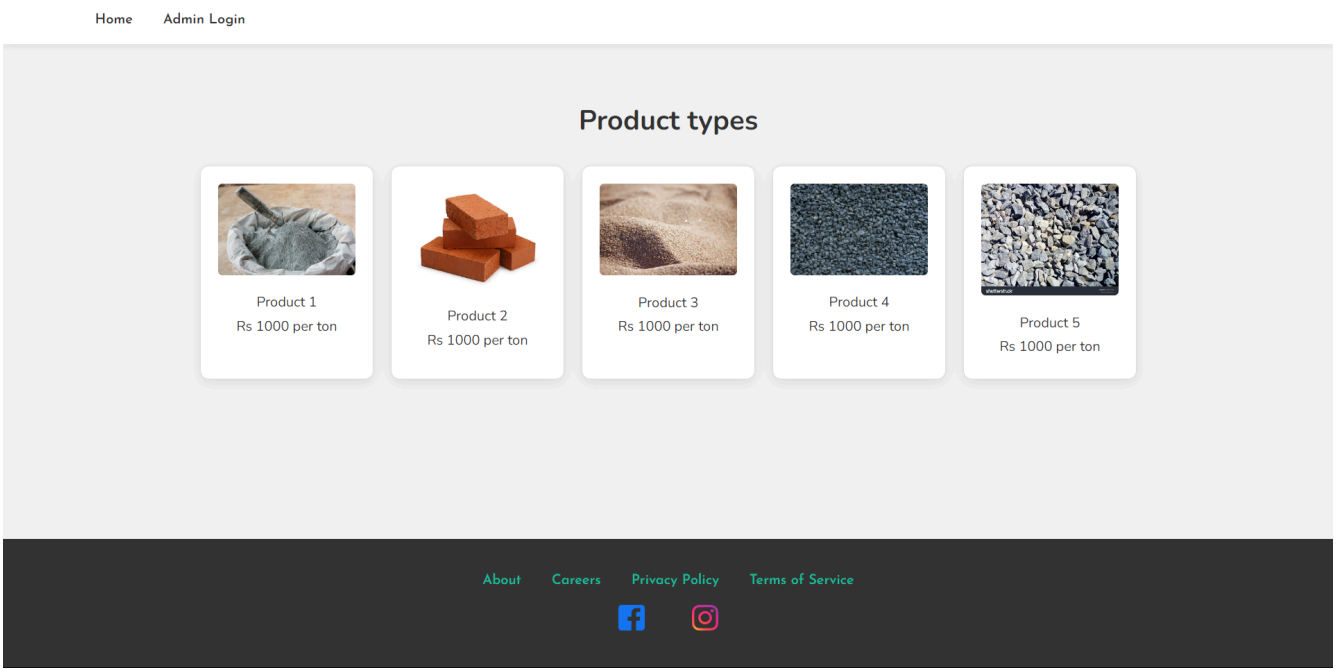
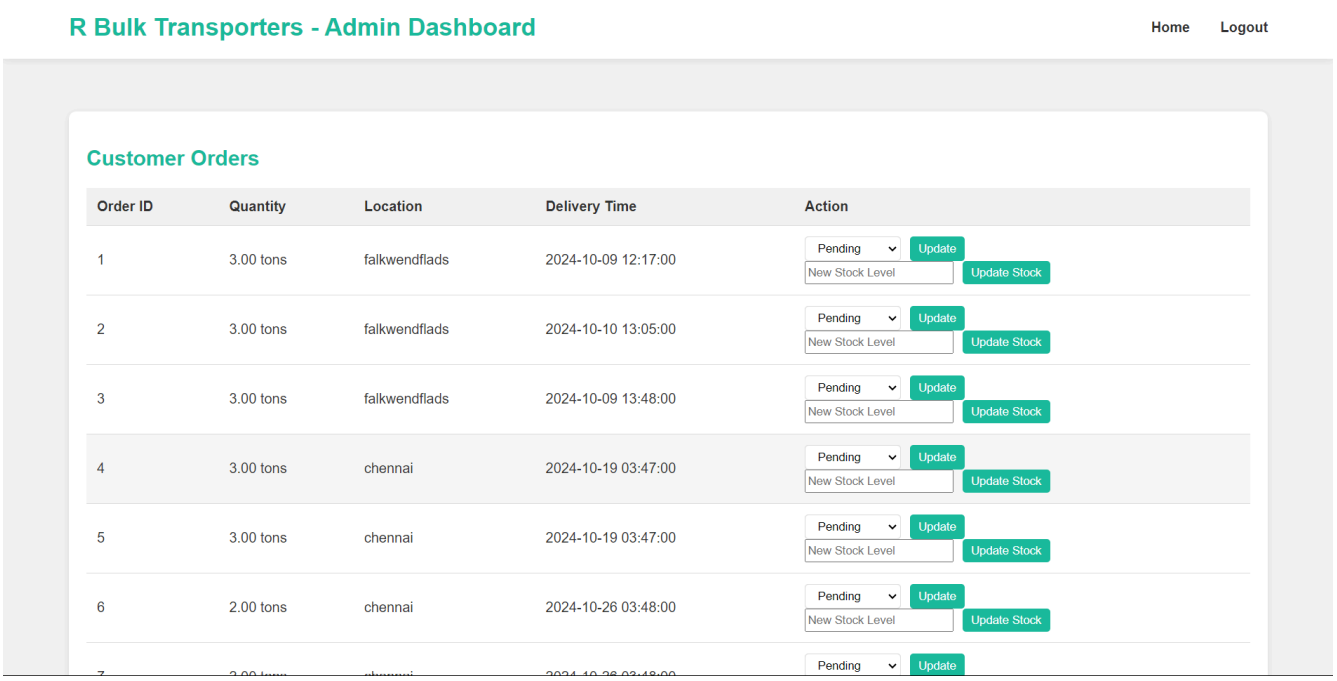


Fig. 7.4 Admin Dashboard



CHAPTER 8

CONCLUSION

In conclusion, the development and implementation of a building material transporter management system is essential for enhancing the efficiency and effectiveness of material logistics within the construction industry. This system provides a comprehensive framework for managing the procurement, distribution, and tracking of various building materials, ensuring timely deliveries and optimal resource utilization. By integrating advanced inventory management techniques, streamlined processes, and data-driven decision-making, the system significantly improves the accuracy of order fulfillment and inventory control.

The establishment of such a robust management system empowers contractors, suppliers, and project managers to mitigate delays and reduce costs associated with material shortages. Furthermore, the system facilitates real-time monitoring and reporting, allowing stakeholders to make informed decisions and maintain transparency throughout the supply chain.

Ongoing assessment, regular updates to system functionalities, and collaboration among industry professionals are critical to the system's success. By leveraging emerging technologies, such as artificial intelligence and data analytics, the management system can further enhance its capabilities, enabling predictive analysis of material demands and optimization of transportation routes.

Ultimately, a well-structured building material transporter management system is a cornerstone for improving operational efficiencies in the construction sector. It not only addresses the immediate logistical challenges faced by industry stakeholders but also sets a precedent for future innovations in material management, contributing to a more efficient and sustainable construction landscape.

REFERENCES

1. HTML and CSS : Design and Build Websites by Jon Duckett
2. JavaScript and JQuery : Interactive Front-End Web Development by Jon Duckett
3. Font Awesome Icons– www.fontawesome.com
4. HTML,CSS,JS : https://www.w3school.com
5. PHP, MYSQL : www.youtube.com
6. Eloquent JavaScript by Marijn Haverbeke