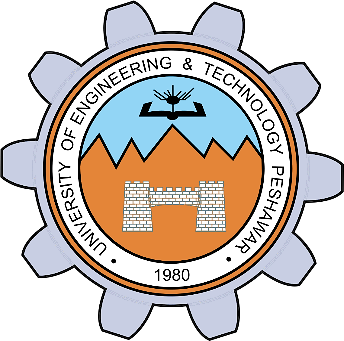
***Project Report***



|  |  |
| --- | --- |
| *Submitted By* | *Registration No.* |
| *Maira Zafar* | *22PWCSE2106* |
| *Saira Zafar* | *22PWCSE2128* |
|  |  |

**Subject:** *Database Management System Lab*

**Submitted To:** *Engr. Sumayyea Salahuddin*

**Project Title:** *Smart Food Delivery Box Online Mart*

**Department of Computer Systems Engineering**

**Abstract:**

**To adapt the VisionCare telemedicine platform structure and methodology to your e-commerce food delivery box project, we will follow the same systematic approach — defining:**

* **Introduction**
* **Objectives**
* **Entity Description Table**
* **Business Rules**

**✅ Introduction**

**In the modern digital marketplace, seamless and efficient food delivery systems have become essential. This project focuses on the development of SmartBox, a Laravel-based e-commerce platform for food delivery boxes. The goal is to provide a user-friendly, customizable, and secure system that allows customers to browse food boxes, place orders, track deliveries, and manage their accounts efficiently. SmartBox serves as a complete solution for vendors and customers, enabling a smooth and interactive food shopping experience.**

**✅ Objectives**

**✅ Objectives (Revised)**

**1. Develop User Management and Access Control  
Implement a role-based access system for admins and customers. Include secure registration and login with email verification and password hashing.**

**2. Customer & Order Management  
Enable customers to create profiles, browse food boxes, and place orders. Admins can view and manage all customer orders, shipping statuses, and order histories. Customers can only view their own order history.**

**3. Product & Inventory Management  
Only admins can add, update, or remove food delivery boxes (products) with details such as description, images, stock levels, pricing, and discounts. Customers can view and purchase available products.**

**4. Delivery & Payment Integration  
Integrate delivery tracking and multiple payment gateways (Cash on Delivery, Card, etc.). Each order will include delivery address, contact details, and payment method. Only admins can update delivery status.**

**5. Reporting and Analytics  
Provide order analytics dashboards for admins only, including best-selling boxes, total revenue, customer activity insights, and inventory performance.**

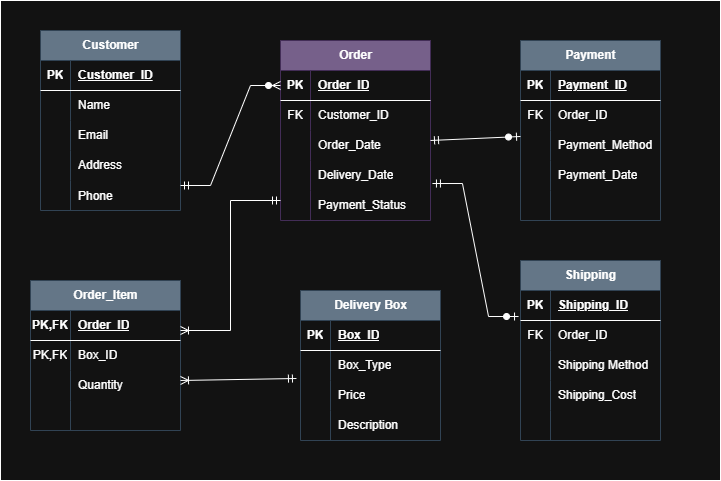
**✅ Entity Description Table**

| **Entity** | **Description** |
| --- | --- |
| Users | Contains user details including full name, email, password, role (admin, vendor, customer). |
| Roles | User role information with role name and timestamps. |
| Products | Food delivery box details (title, description, image, price, discount, quantity). |
| Orders | Customer order details including user ID, product ID, total, delivery status. |
| Order\_Items | Items associated with each order (order ID, product ID, quantity, subtotal). |
| Customers | Customer-specific info like name, email, phone, and address. |
| Payments | Payment information (order ID, method, amount, status, timestamp). |
| Shippings | Delivery details (order ID, shipping address, status, date). |
| Vendors | Vendor information including name, email, phone, and business profile. |
| Contacts | Customer inquiries (name, email, phone, subject, message). |
| Settings | App-wide configuration values (key, value, timestamps). |
| Subscribers | Newsletter subscriber emails and timestamps. |
| Reviews | Product reviews (product ID, user ID, rating, comment, created date). |

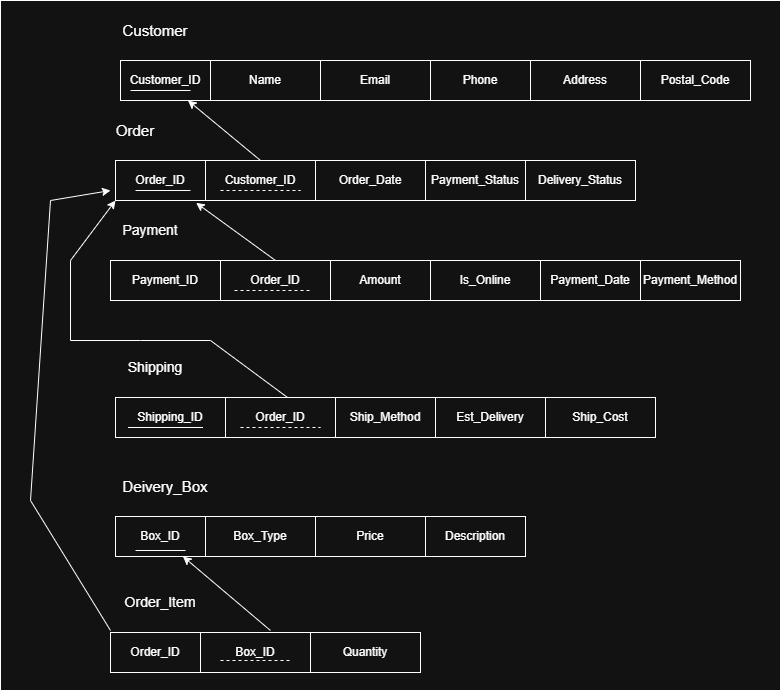
**✅ Business Rules**

1. **Each user must have a unique email and role (admin, vendor, customer).**
2. **Only vendors or admins can add/edit/delete products.**
3. **Products must have a title, image, price, and quantity defined.**
4. **Customers can browse products without login but must log in to order.**
5. **Each order must include at least one item and be associated with a customer ID.**
6. **Payments must be linked to an order and must store the method, amount, and status.**
7. **Shipping records must include the shipping address, status, and order ID.**
8. **Vendors can view analytics related to their products (orders, reviews, revenue).**
9. **Customers can leave reviews only after order delivery is confirmed.**
10. **All contact messages must include name, email, and subject for support.**
11. **Newsletter subscribers must provide a valid email.**
12. **Admins can manage global settings through a settings table.**
13. **Multiple items can be part of one order via the Order\_Items entity.**
14. **Discounts and promotions can be applied to products, visible to all users.**

**Conceptual Schema:**



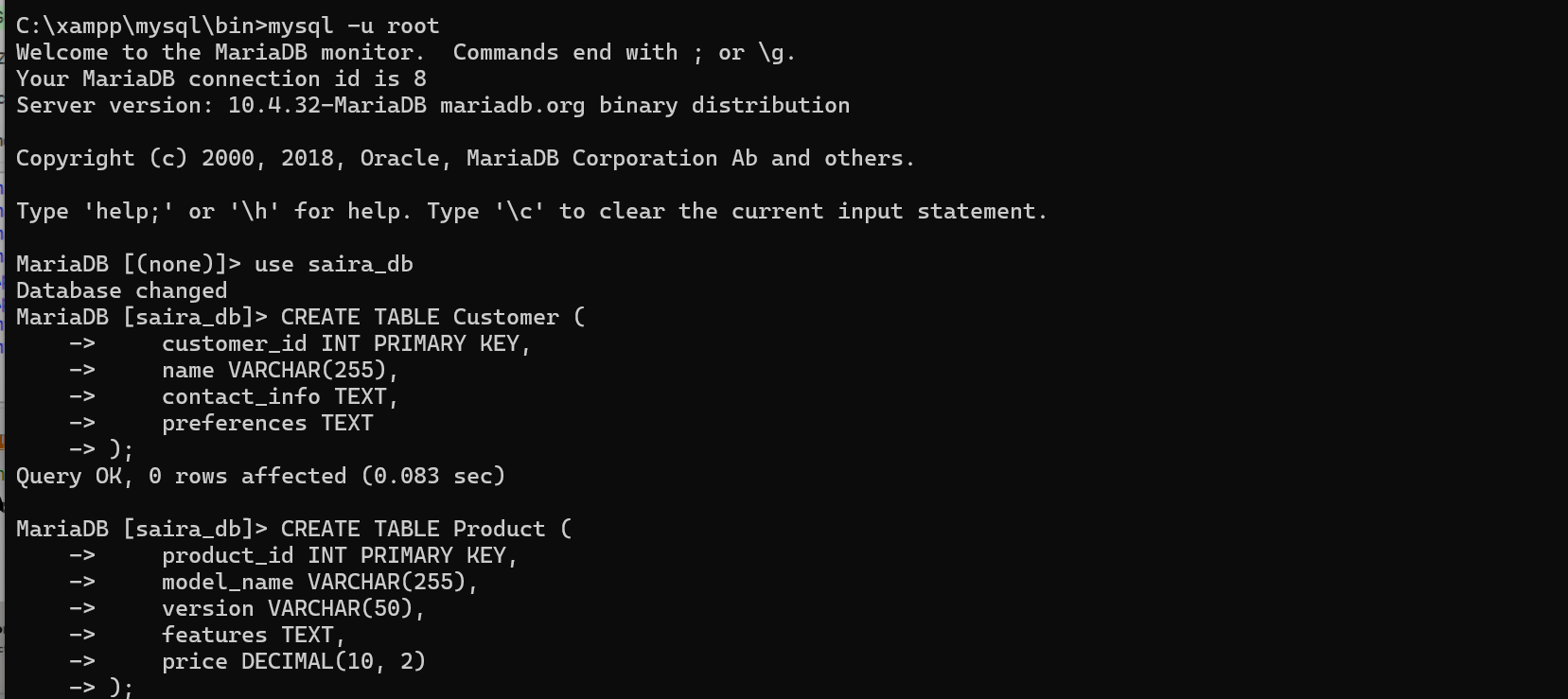
**Relational schema:**



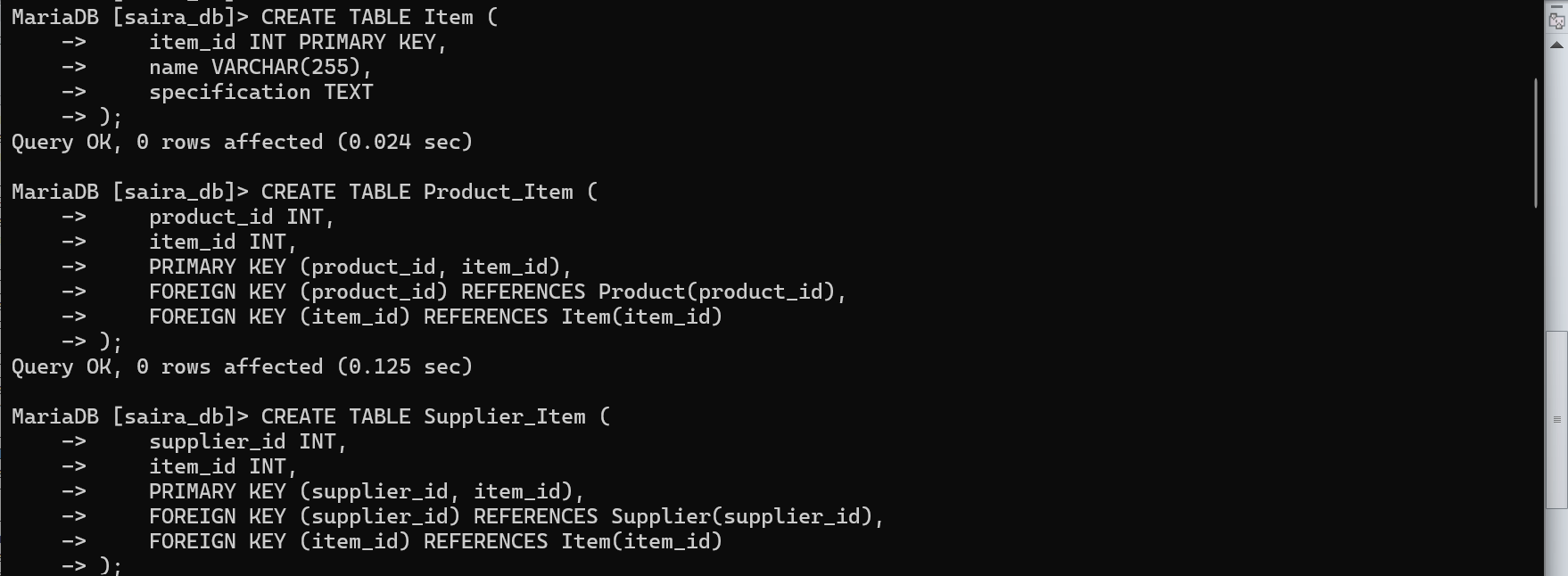
**CRM: SQL Databases and queries**

****

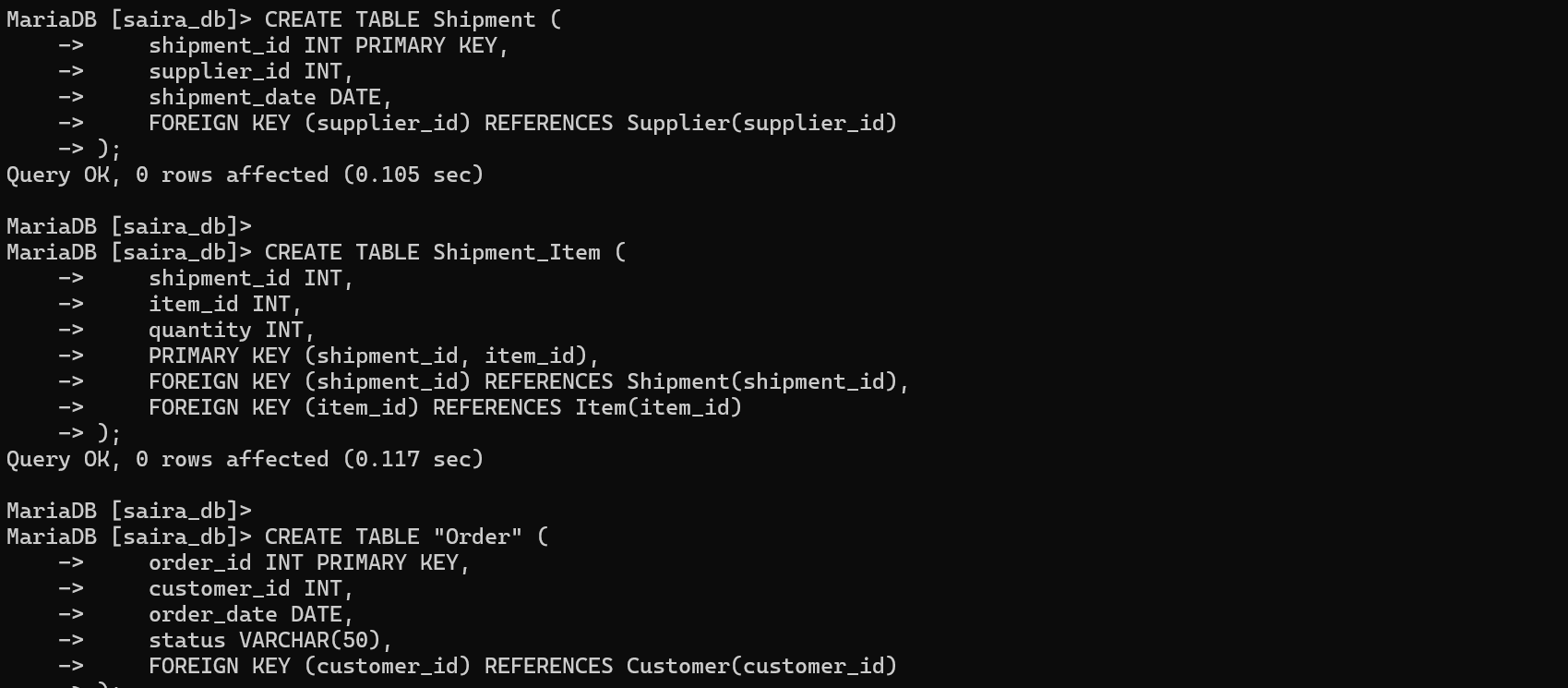
**Database Of Product , Customer:**

****

**Database Of Item , Product \_Item , Supplier\_Item**

****

**Database Of Shipment ,Shipment\_Item ,Order**

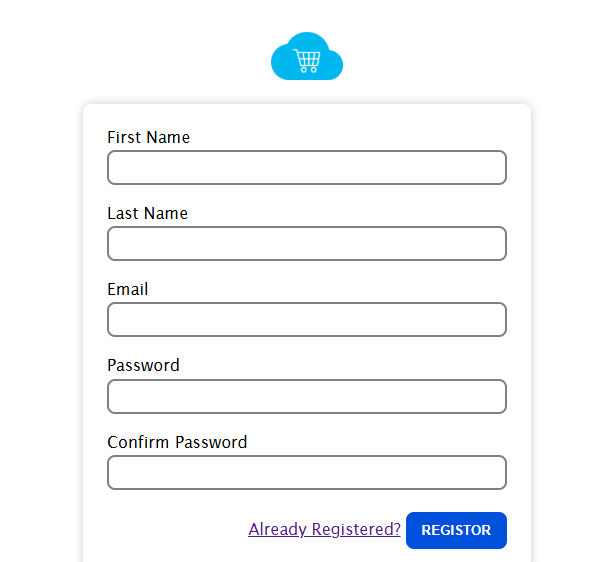
****

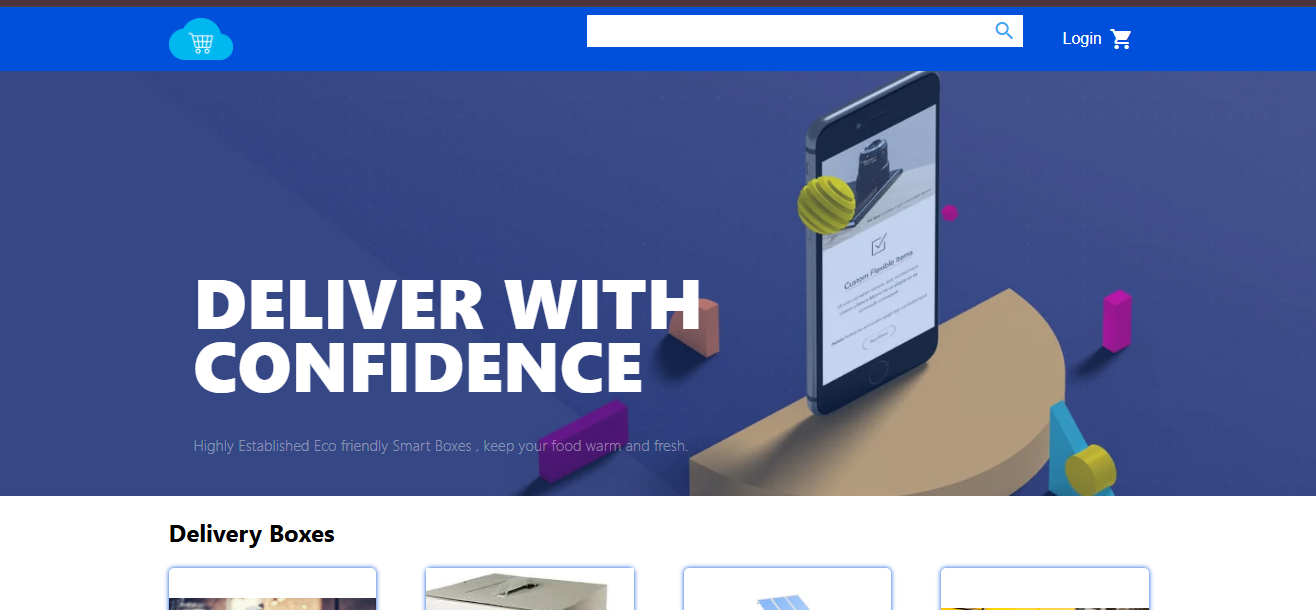
**Total Tables In Databases:**

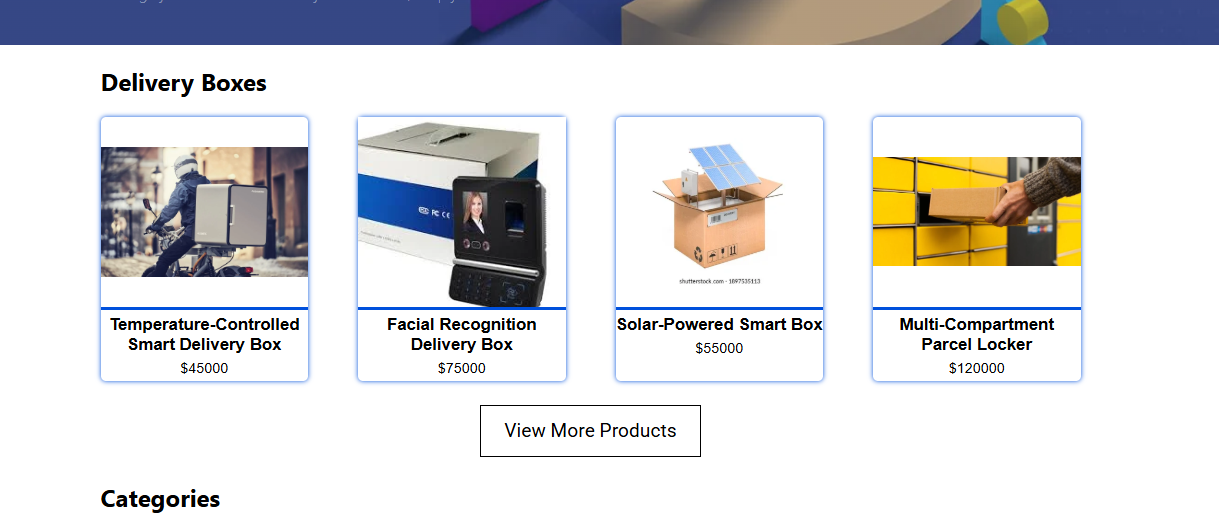
****

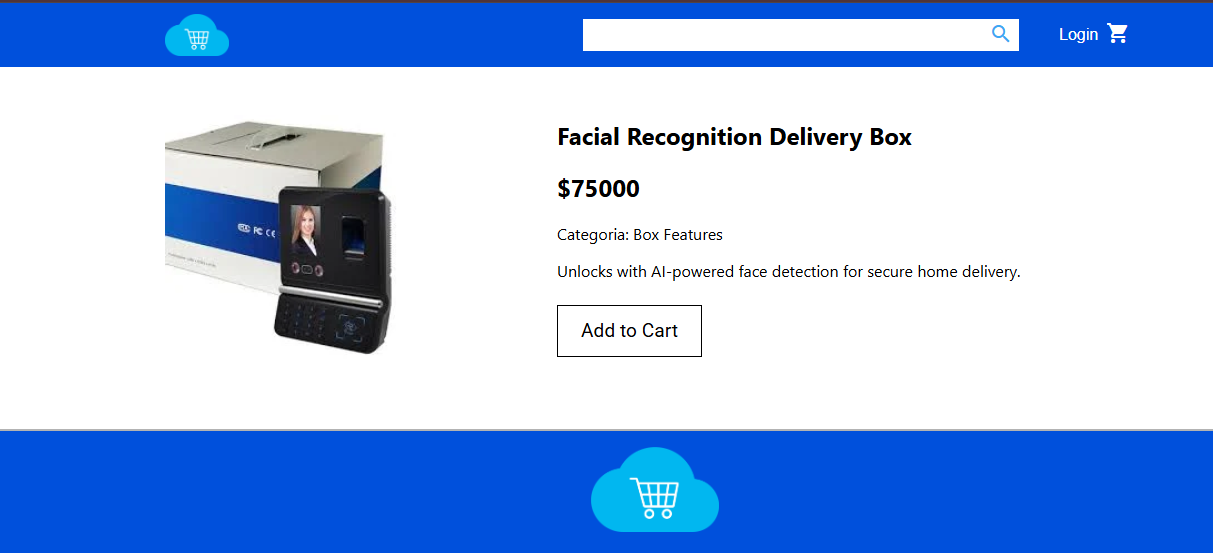
* **Comprehensive implementation details (Laravel):**

**Output:**









Website Link:

https://fyp.primehowlproject.com/