Table of Contents

Table of Figures

Introduction

E-commerce is fast gaining ground as an accepted and used business paradigm. More and more business houses are implementing web sites providing functionality for performing commercial transactions over the web. It is reasonable to say that the process of shopping on the web is becoming commonplace. The objective of this project is to develop a general purpose e-commerce store where product like clothes can be bought from the comfort of home through the Internet.

An online store is a virtual store on the Internet where customers can browse the catalog and select products of interest. The selected items may be collected in a shopping cart. At checkout time, the items in the shopping cart will be presented as an order. At that time, more information will be needed to complete the transaction. Usually, the customer will be asked to fill or select a billing address, a shipping address, a shipping option, and payment information such as credit card number.

This project is very user friendly developed using laravel framework with MySQL database. Almost all the function are dynamic except the logo. It is very user friendly and secure as larvae framework itself is the most secure framework. The most advance and important feature of this project is the PayPal integration. User can pay by payment gateway using PayPal however it is not the only one option, customer can also can choose cash on delivery.

Objective

1. More than reach, it also important for increasing digital transactions of the country.
2. Customer satisfaction will be improved more.
3. More offers and discounts will be provided for buyers.
4. Easy to spread business across countries
5. Helps to increase our national income.

Prerequisite

- Xampp server
- Window 7 or plus
- laravel setup
- internet connection

ER diagram

The relational diagram of the ecommerce system is given below.

Relation:

i.   Users and orders

There exit one to many relationship between the users and orders table, it means that one user can place multiple of orders.

ii.   Users and carts

One user can add multiple product in cart i.e they have one to many relationship

iii.   Payment and cart

It is one to one relation as only one payment exit for one cart. That is to say for one payment for one order

iv.   Orders and cart

It is also one to one relation which means only one order exit in for one cart. One the product in cart is order it will be removed from the carts table.

v.   Carts and products

There is one to many relation between cart and product that multiple product exit in cart.
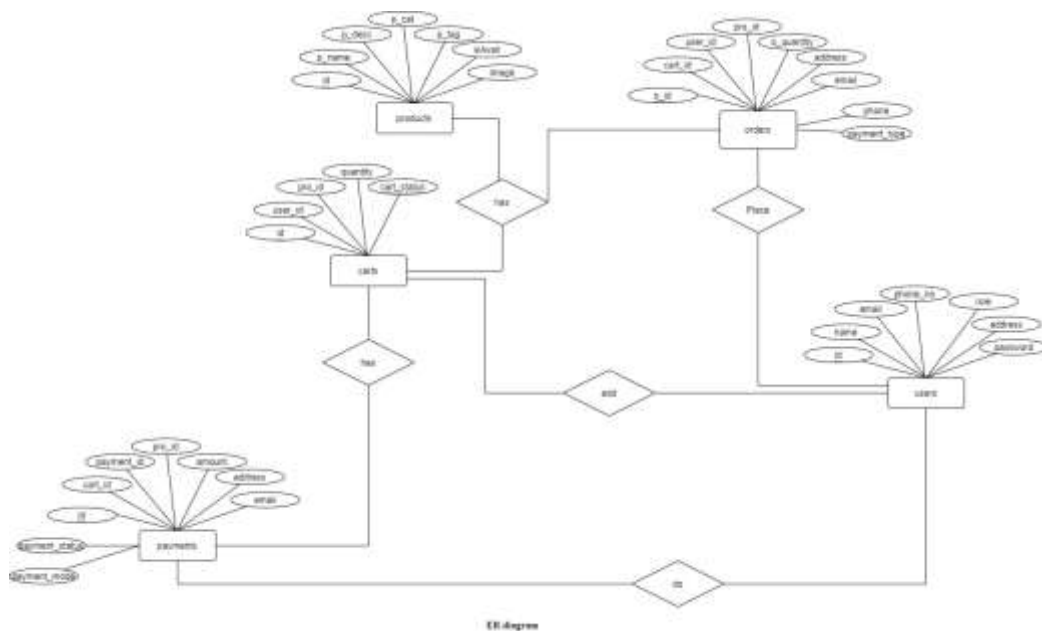

Figure 1: ER Daigram

Use case

There are main actor admin and customer. Admin can add, update and delete products, user, analyze data whereas customer can view all the products, login and logout, add product etc. the main use cases of admin and customer are listed below.

Admin

1. View order, payment, user data
2. Add/ edit product
3. Update/ delete user

Customer

1. Browse catalog
2. Register
3. Login/logout
4. Edit profile
5. Add to cart
6. Order
7. Payment
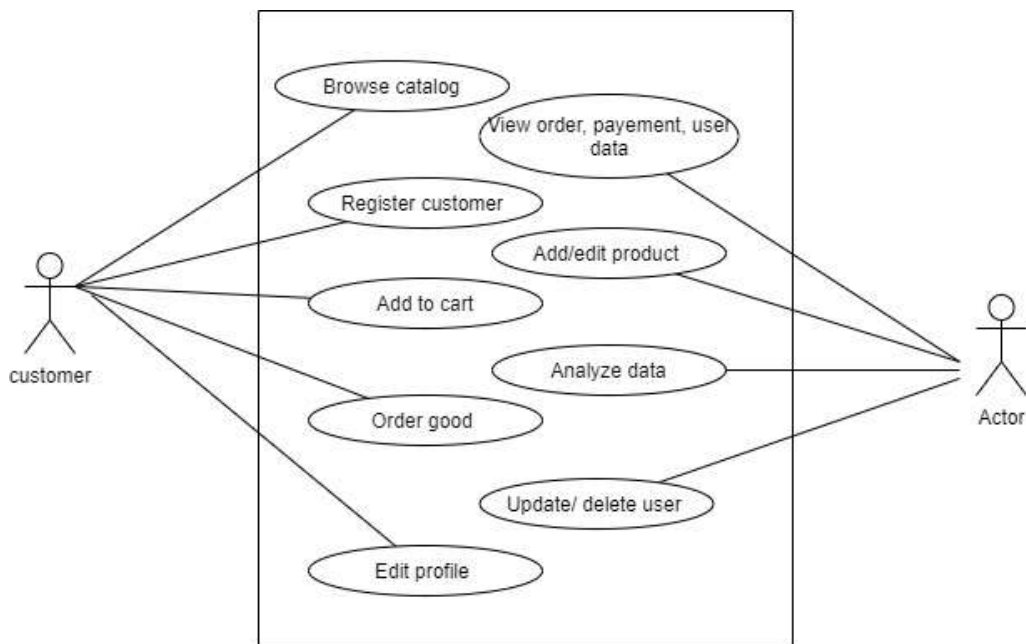
use case diagram

Figure 2: use case diagram

Relational diagram

The relational diagram of my website 'the shopping site' is given below. It has five table which are users, products, orders, carts and payments.

1. Users

   This table hold the user information. It has ten field including primary key 'id'.

2. Products
   This table hold the details of products. it has ten fields including primary key 'id'.

3. Carts
   This hold the detail of product that user has added. It has seven fields including two foreign key user_id which point to id of user table and pro_id which point to product table and primary key id.

4. Orders

   The users orders are stored in the orders table which has thirteen fields. It had three foreign key i.e user_id, pro_id and cart_id which point to user table, product table and cart respectively. o_id is the primary key.

5. Payments
   The payment details are stored in payments table which has nine field. it has one primary key id and one foreign key cart_id.
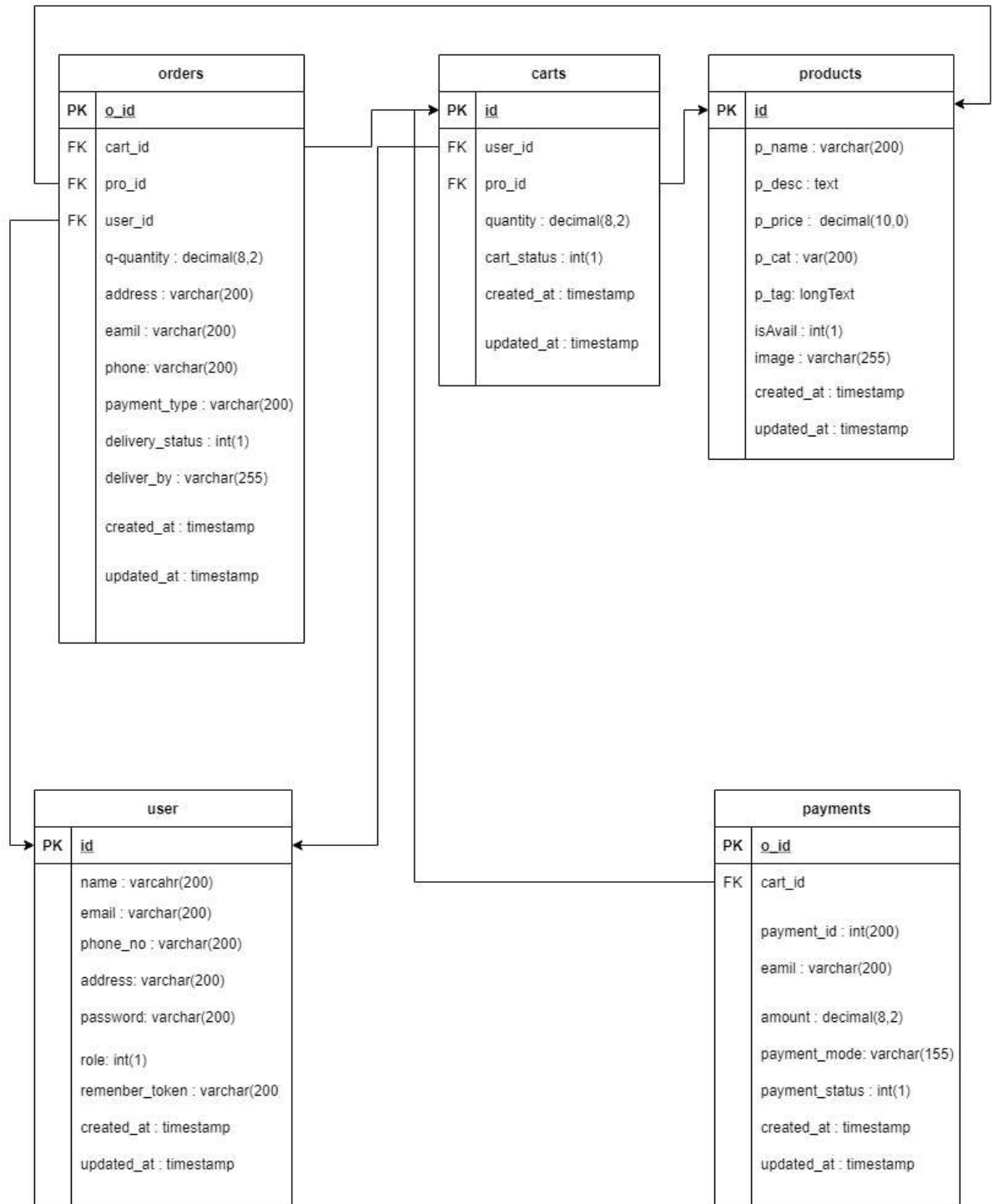
orders

PK	o_id
FK	cart_id
FK	pro_id
FK	user_id
	q-quantity : decimal(8,2)
	address : varchar(200)
	eamil : varchar(200)
	phone: varchar(200)
	payment_type : varchar(200)
	delivery_status : int(1)
	deliver_by : varchar(255)
	created_at : timestamp
	updated_at : timestamp

carts

PK	id
FK	user_id
FK	pro_id
	quantity : decimal(8,2)
	cart_status : int(1)
	created_at : timestamp
	updated_at : timestamp

products

PK	id
	p_name : varchar(200)
	p_desc : text
	p_price : decimal(10,0)
	p_cat : var(200)
	p_tag: longText
	isAvail : int(1)
	image : varchar(255)
	created_at : timestamp
	updated_at : timestamp

user

PK	id
	name : varcahr(200)
	email : varchar(200)
	phone_no : varchar(200)
	address: varchar(200)
	password: varchar(200)
	role: int(1)
	remenber_token : varchar(200
	created_at : timestamp
	updated_at : timestamp

payments

PK	o_id
FK	cart_id
	payment_id : int(200)
	eamil : varchar(200)
	amount : decimal(8,2)
	payment_mode: varchar(155)
	payment_status : int(1)
	created_at : timestamp
	updated_at : timestamp

Figure 3: Relational Class Diagram

Scope

- run small scale business
- Transport and logistics.
- Automatic trading of clothes, devices, books and so on.
- Accounting and financial management.
- Legal advice
- Marketing, sales and sales promotion.
- Selling can be centered around the Global client

Conclusion

A simple ecommerce site is built with Laravel framework. The main feature of this project is that it follow MVC pattern and it has PayPal integration.
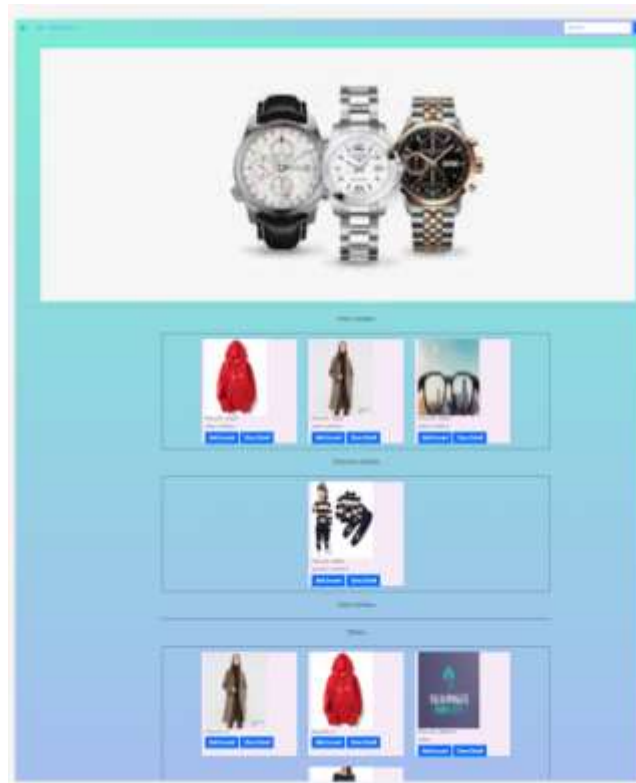
Appendix

Fig: index page

Fig: Add to cart page

Fig: Place Order

Fig: PayPal integration
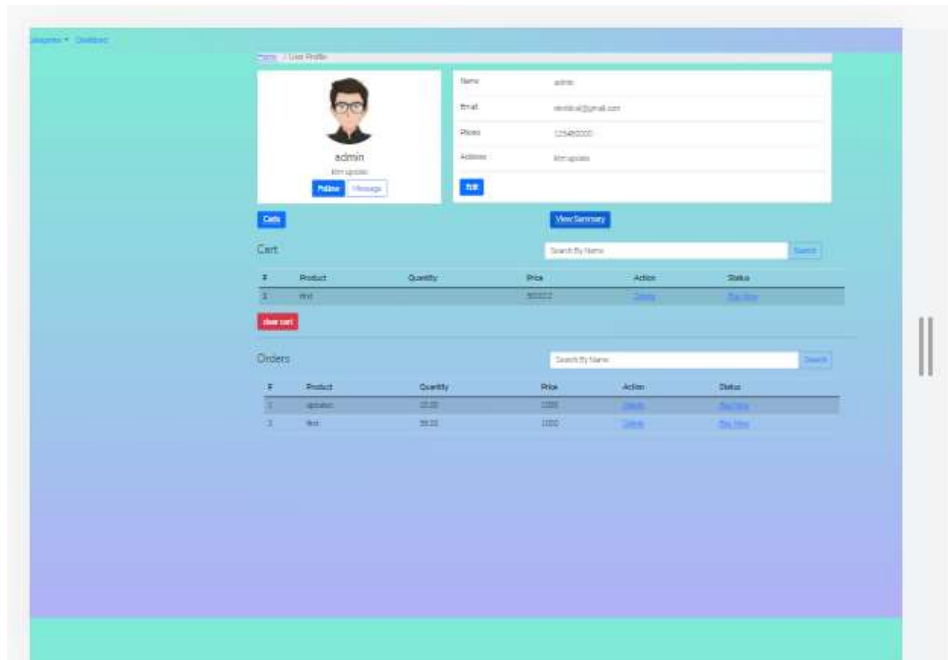
Fig: User Profile

Fig: Admin Dashboard

Fig: edit user profile