

A PROJECT REPORT
ON
ECOMMERCE-WEBSITE

By
Jaivin Domadiya



DEPARTMENT OF B.C.A

COLLEGE : S.S.S.D.I.I.T. COLLEGE – JUNAGADH

UNIVERSITY : B.K.N.M. UNIVERSITY JUNAGADH -362640



શાસ્ત્રી સ્વામી શ્રી ધર્મજીવનદાસજી

સાયન્સ & IT ગુરુકુલ કોલેજ





Shastri Swami Shree Dharmajivandasji Institute of I.T.

(Affiliated to Bhakta Kavi Narsinh Mehta University, Junagadh)

C/o Shree Swaminarayan Gurukul, College Road, Junagadh

Web: www.junagadhgurukul.org

Email : sssdiit_jnd@yahoo.com

PROJECT COMPLETION CERTIFICATE

THIS CERTIFICATE IS AWARDED TO

Member Name : Domadiya Jaivin Vallabhbhai

Stream : BCA

Semester : 5th

IN COMPLETION OF PROJECT WORK

Platform : Python Django Begin Date : 01/07/2021

Database : MySQL End Date: 13/10/2021

DIRECTOR



Shastri Swami Shree Dharmajivandasji Institute of I.T.

(Affiliated to Bhakta Kavi Narsinh Mehta University, Junagadh)

C/o Shree Swaminarayan Gurukul, College Road, Junagadh

Web: www.junagadhgurukul.org

Email : sssdiit_jnd@yahoo.com

PROJECT EVALUATION CERTIFICATE

THIS CERTIFICATE IS AWARDED TO

Member Name : Domadiya Jaivin Vallabhbhai

Stream : BCA

Semester : 5th

IN COMPLETION OF PROJECT WORK

Platform : Python Django Begin Date : 01/07/2021

Database : MySQL End Date : 13/10/2021

**PROJECT
EVALUATOR**

**PROJECT
GUIDE**

Table Of Content

	Topic	Page No.
1.	Abstract	5
2.	Introduction	6
2.1	Project Details: Brief Introduction	6
2.2	Technology And Tools Used	6
3	Software Requirements Specifications	7
3.1	Scope	7
3.2	System Functional Requirements	7
3.3	Non-Functional Requirements	10
4	Design	11
4.1	Use Case Diagram	11
4.2	Class Diagram	12
4.3	Data Flow Diagram	13
4.4	Sequence Diagram	15
4.5	Activity Diagram	16
4.6	Data Disctionary	17
5	Implementation Details	19
5.1	Modules	19
5.2	Implementation of module	20
6	Testing	24
7	Snapshot of System	25
8	Conclusion	31
9	Limitations And Future Extensions	32
10	Bibliography	33

1. Abstract

- **E-Commerce** is the online buying and selling process that is extremely important in **our** daily life now. The foremost **reason** behind the growth of Internet users besides social media in **e-commerce**. **E-Commerce** is at the heart of the Internet and **e-commerce** is as important as a heart is for a body.
- E-Commerce Website is an online shopping Website. The platform supports seller as well as the buyer. The Seller uploads the products to sell along with necessary information related to the display product . Buyer then goes through the available stock for the products and buys the products of his/her choice by adding the products into the cart where the summary of the products can be seen along with the total amount payable.
- In Order to place order, the buyers need to register himself/herself onto the website and then only can proceed to place order further. However, the buyer can browse in order to just see through the website. They can browse through various categories as per the filter available.

2. Introduction

2.1 Brief Introduction

- Nowadays, the people across the globe are so busy with their work as well as the responsibilities that they don't have time to spend few hours just for the sake of shopping, Which is an essential part of our day-to-day life habits. So, the lack of time due to the hactic as well as the non-Availability of certain products at nearby places leads people to waste their time just to find the right place to get their task done at one time.

So, inorder to save people's time so that they can focus on other important life events, has lead to the development of the this E-Commerce Website, where buyers can buy anything of their choice from the available list of products at the convenience of their comfort.

2.2 Tools/Technologies Used :

❖ **Technologies :**

- Django
- Python
- MySQL
- HTML
- CSS

❖ **Tools :**

- Visual Studio Code

❖ **Platform :**

- Local Development Server

3. Software Requirement Specifications

3.1 Product Scope :

This Site is created to enable the buyers to buy various products be it mobile phones, laptops, etc.

From the comfort of their home and at their convenience i.e 24 X 7.

3.2 System Functional Requirements:

1. **MANAGE ACCOUNT :**

1.1 **SIGN-UP:**

- The customers who are visiting the website for the first time, need to sign-up with their number and other details, in order to view the products and shop, buy anything from the website.

INPUT : User will input his/her username and new password.

OUTPUT : Account Created.

1.2 **LOGINPAGE :**

- Here, the customers who already have an account and are frequent users of the website , have to login in order to view, shop as well as place order from the website.

INPUT : Username and password are entered in fields.

OUTPUT : Login Successful if username and password are correct and can continue shopping.

2. ORDER DETAILS :

2.1 ADD/REMOVE CARD:

- Here, the customer can add or remove card details.

INPUT : ask user if he wants to remove or add card details

OUTPUT: card details added/removed successfully

2.2: ADD ADDRESS DETAILS:

- Here, the customer add his/her Address details where he/she want to deliver his/her order.

INPUT : Add Address and Mobile number and checkout the order

OUTPUT: ordered successfully

2.3: Check Your Order Status:

- Here, The customer can track the status of his/her order.

INPUT : Information of Order Which is to be Tracked.

OUTPUT : Order Status.

3.PRODUCT SELECTION :

3.1 CART :

- Customers can temporarily store item they wish to buy while browsing for other items , which they want to buy alongside, also they can add various payment method, select delivery address etc.

INPUT : Add to Cart button is selected for a product by the customer.

OUTPUT: Item is added to Cart Successfully

3.2 PRODUCT PREVIEW :

- Here, customers can see the product listed on the site in detail. In the preview, they can see all the information related to the particular product that they have selected for preview.

INPUT : Particular product selected.

OUTPUT : All the details related to the product selected are displayed on screen.

4: SIDE BAR /NAVIGATION BAR:

4.1: CATEGORIES :

- It is a option through which the buyer can see the products as per the selected category of the option.
- **INPUT**: select price range, specification, types of products user wants
- **OUTPUT**: displays product of selected options

4.2: STORE BUTTON:

- **INPUT**: click on Store button
- **OUTPUT**: Return to home page

4. Manage Payment :

5.1 Make payment

1. Select Option for payment

- **Input** : User can select any type of payment method like online (GPay, PhonePe, PayTm) or offline method (COD).

- **Output:** If Chosen option is COD, then display payment successful Or else ask for amount details and do transaction.

3.3 Other Nonfunctional Requirements

1. Performance

- The system must be interactive and must not involve long delays. Though in case of opening the app components or loading the page the system shows the delays less than 2 seconds.

2. Safety

- The users' data is highly personal. The system has authorization to avoid any un- authorized access to user's private data.

3. Reliability

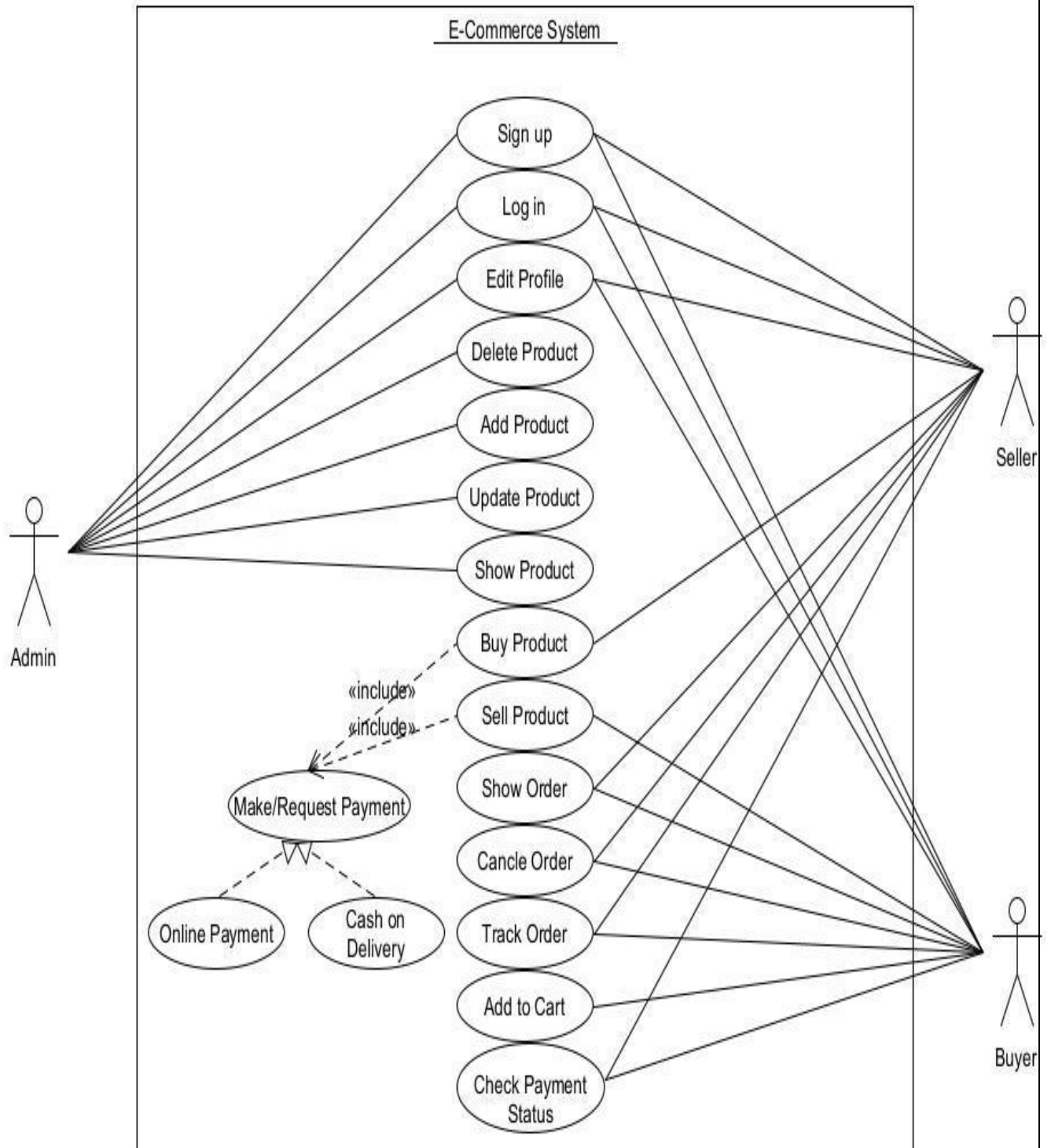
- As the system has personal data, its reliability is the major factor for consideration.

4. Database

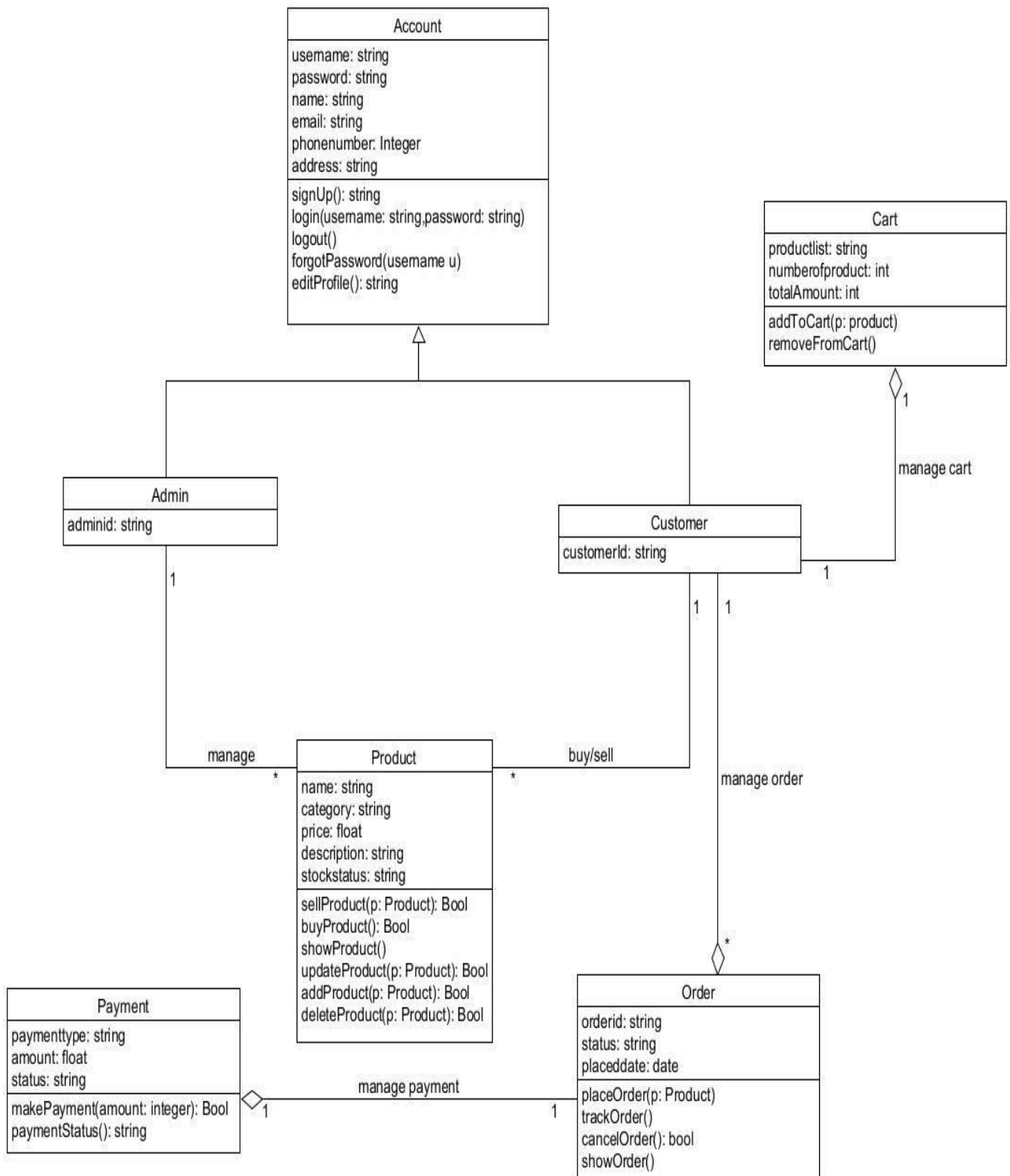
- System requires to access user submissions, versions, reviews and profile data fast to maintain the performance.

4. DESIGN

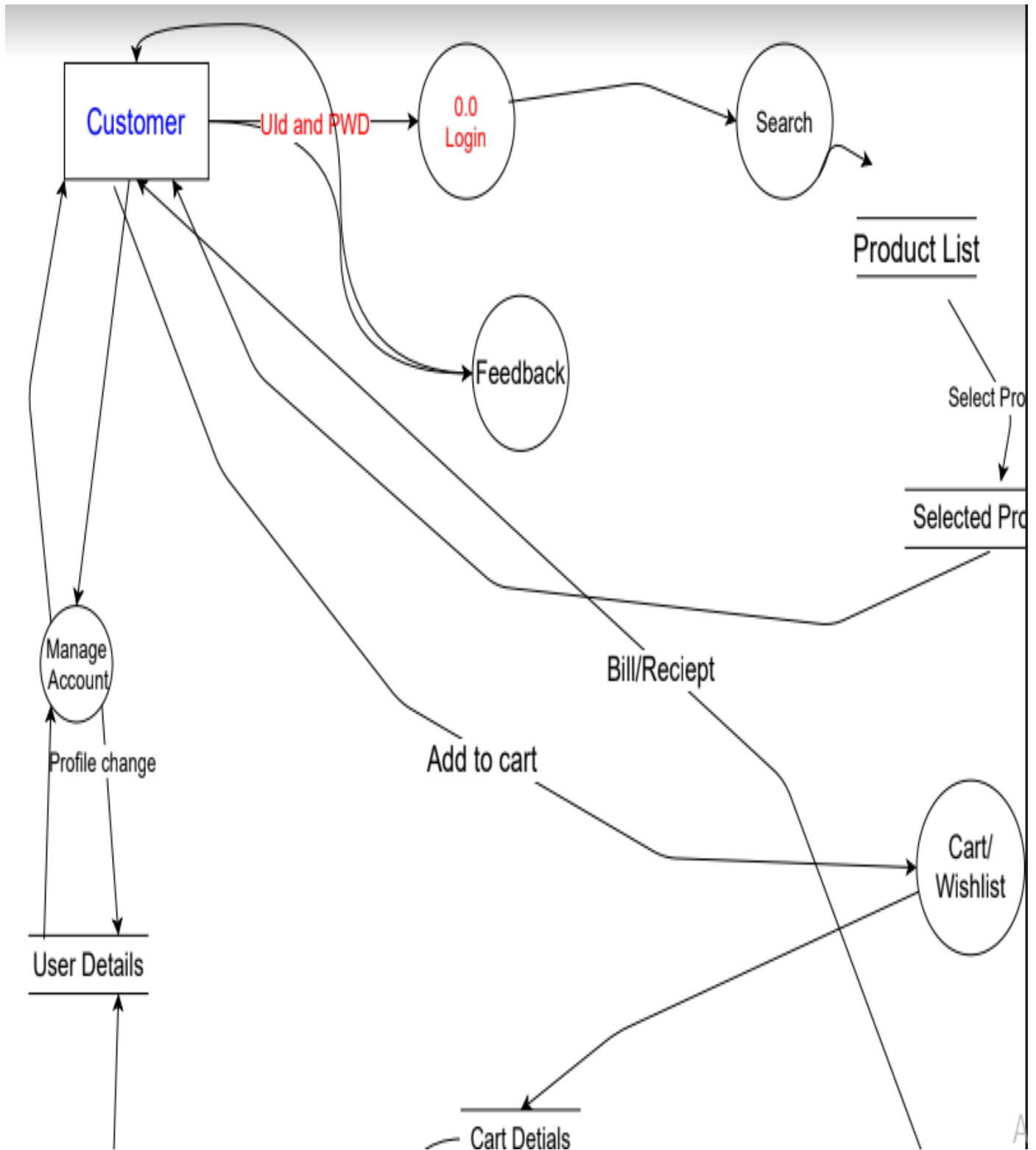
4.1 Use Case Diagram

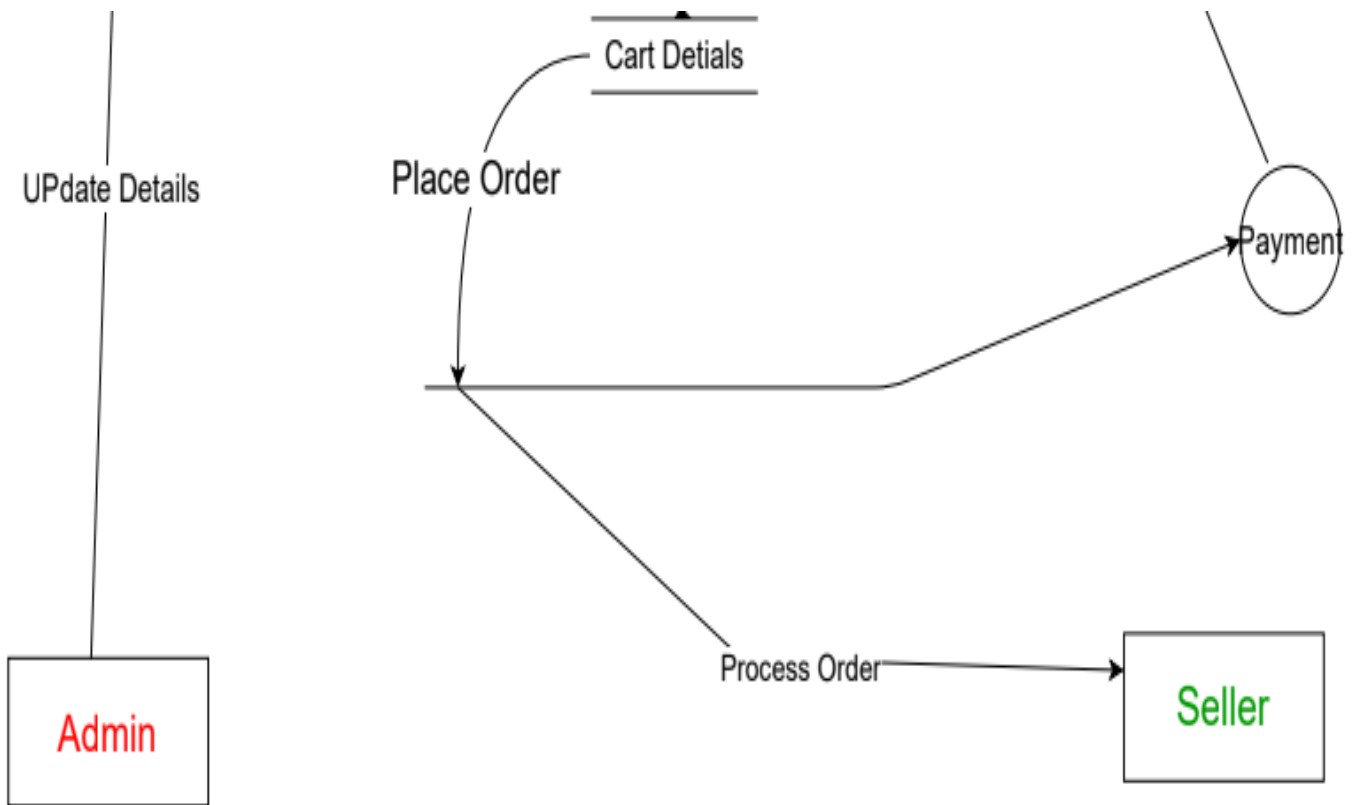


4.2 Class Diagram

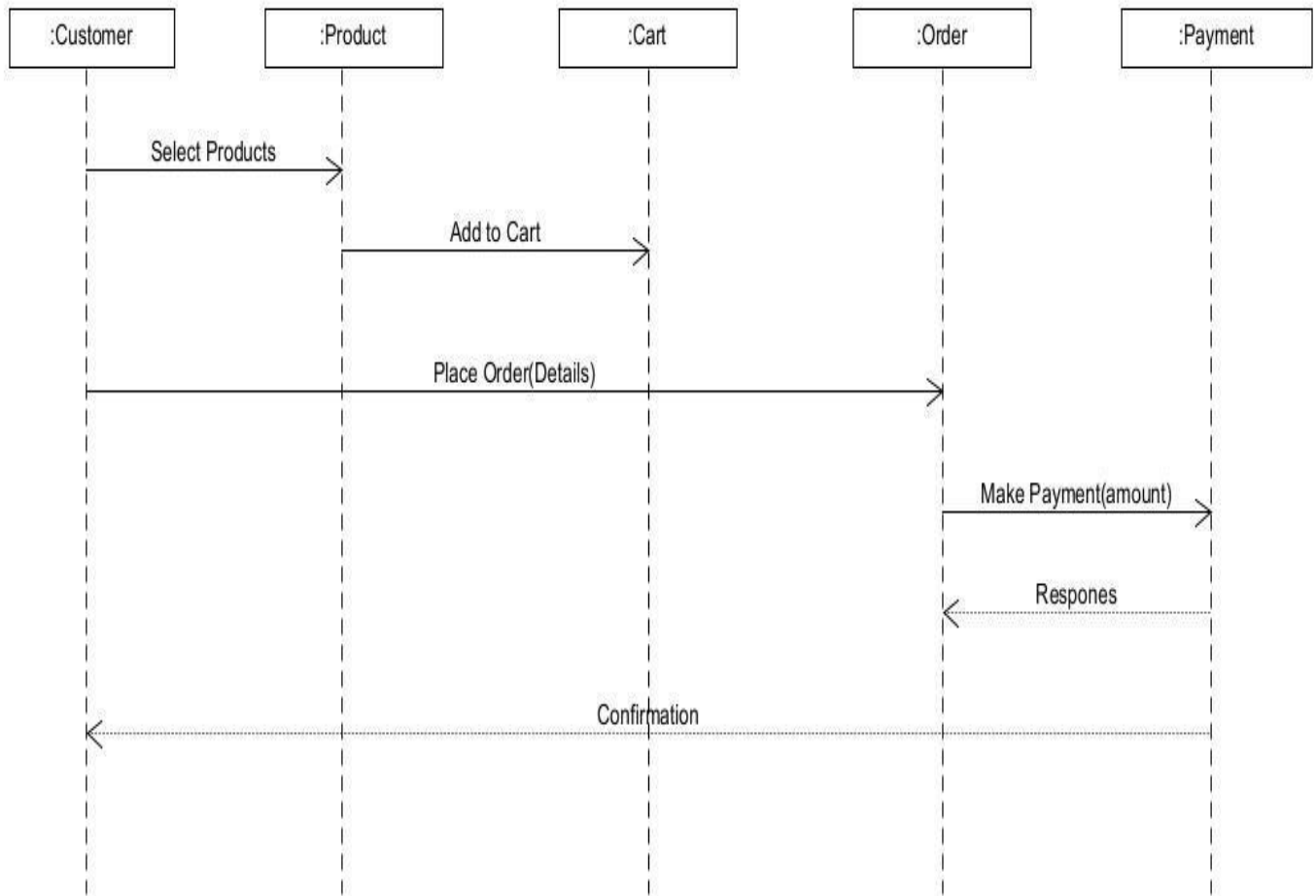


4.3 Data Flow Diagram



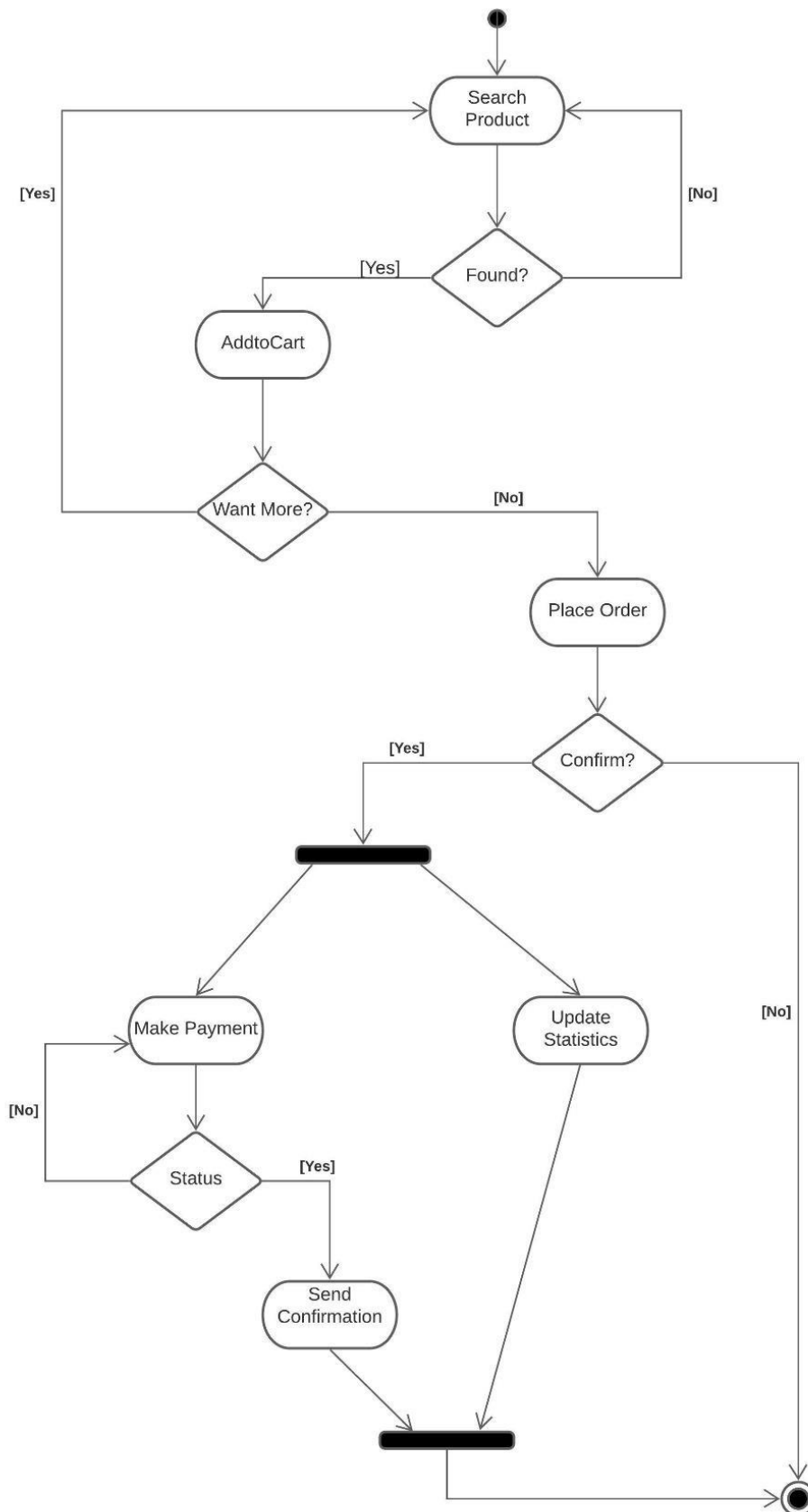


4.4 Sequence Diagram



4.5 Activity Diagram

Activity Diagram for Buy Product



4.6 Data Dictionary :

1)Category Table :

Name of Feild	Type	Length	Attributes
Id	Int	-	AI, Primary Key
Name	Varchar	100	-

2)Customer Table:

Name of Feild	Type	Length	Attributes
Id	Int	-	AI, Primary Key
First Name	Varchar	100	-
Last Name	Varchar	100	-
Email	Varchar	254	-
Password	Varchar	500	-

3)Product :

Name of Feild	Type	Length	Attributes
Id	Int	-	AI, Primary Key
Name	Varchar	100	-
Price	Int	100	-
Description	Varchar	200	-
Image	-	-	-

5. Implementation Details :

5.1 Modules

The System consist of 4 various modules namely

- 1) Manage Account
- 2) Manage Product
- 3) Manage Order
- 4) Manage Payment

These modules are implements several methods to perform major functionalities. Implementation is done using Django and MySQL.

5.2 Implementation Of modules

Manage Account

This module contain all methods related to account. It contains Signup, Authorization, Authentication, Login.

➤ **Implementation of Sign Up functionality** (Signup.py) :

```
class Signup(View):
    def get(self,request):
        return render(request,'signup.html')
    def post(self,request):
        regdata=request.POST
        fname=regdata.get('first_name')
        lname=regdata.get('last_name')
        mail=regdata.get('email')
        passw=regdata.get('password')
        value={
            'first_name':fname,
            'last_name':lname,
            'email':mail,
            'password':passw,
        }
        customer=Customer(first_name=fname,last_name=lname,email=mail,password=passw)
        error_mess=self.validate(customer)
        if not error_mess:
            customer.password=make_password(customer.password)
            customer.save()
            messages.success(request,('Registration Successfull!!'))
            return redirect('/')
        else:
            data={
                'error':error_mess,
                'values':value,
            }
            return render(request,'signup.html',data)
```

➤ **Implementation of Authentication and Login** (Login.py) :

```
from django.shortcuts import render,redirect
from django.http import HttpResponseRedirect
from django.contrib.auth.hashers import check_password
from ecommerce.models.Customer import Customer
from django.views import View

class Login(View):
    def get(self,request):
        return render(request,'Login.html')
    def post(self,request):
        email=request.POST.get('email')
        password=request.POST.get('password')
        customer=Customer.get_customer_by_email(email)
        error_message=None
        if customer:
            flag=check_password(password,customer.password)
            if flag:
                request.session['customer_id']=customer.id
                return redirect('/')
            else:
                error_message="Email or Password is Invalid!"
        else:
            error_message="Email or Password is Invalid!"
        return render(request,'Login.html',{'error':error_message})
    def logout(request):
        request.session.clear()
        return redirect('login')
```

➤ **Implementation of Logout** : (Login.py)

```
def logout(request):
    request.session.clear()
    return redirect('login')
```

➤ **After Login setting up Cart (Home.py) :**

```
from django.shortcuts import render, redirect
from ecommerce.models.Product import Product
from ecommerce.models.categories import Category
from django.views import View
# Create your views here.
class index(View):
    def get(self, request):
        cart=request.session.get('cart')
        if not cart:
            request.session.cart={}
        product=None
        category=Category.get_all_categories()
        category_id=request.GET.get('category')
        if category_id:
            product= Product.get_all_Products_byid(category_id)
        else:
            product= Product.get_all_Products()
        data={}
        data['products']=product
        data['categories']=category
        return render(request, 'index.html', data)
    def post(self, request):
        product=request.POST.get('product')
        remove=request.POST.get('remove')
        cart=request.session.get('cart')
        if cart:
            quantity=cart.get(product)
            if quantity:
                if remove:
                    if quantity<=1:
                        cart.pop(product)
                else:
                    cart[product]=quantity-1
            else:
                cart[product]=quantity+1
        else:
            cart[product]=1
        request.session['cart']=cart
        print('cart', request.session['cart'])
        return redirect('homepage')
```

➤ **Show Cart** (cart.py) :

```
ite > views > cart.py > ...
from django.shortcuts import render,redirect
from django.http import HttpResponse
from django.contrib.auth.hashers import check_password
from ecommerce.models.Product import Product
from django.views import View

class Cart(View):
    def get(self,request):
        if request.session.get('cart'):
            ids=list(request.session.get('cart').keys())
            products=Product.get_product_by_id(ids)
            return render(request,'cart.html',{'products':products})
        return render(request,'cart.html',{'products':0})
```

6. Testing :

Testing Method : Manual

Sr no.	Test Details	Expected Result	Actual Result	Status
1.	Login with invalid credential	User shouldn't be allowed to Login	User is redirected to login page with message of invalid login details	Success
2.	Login with Valid Details	User is shown homepage	User is shown Homepage	Success
3.	Checkout of cart without login	User is redirected to login page	User is given message of login first and redirected to login page	Success
4.	Product Add to Cart	Product should be added to cart	Quantity of product increases	Success
5.	Accessing direct Homepage	User should be redirected to homepage	User is redirected to homepage	Success
6.	Category Listing	Selected Category Products will be shown	Selected category products are shown	Success
7.	Home button	Redirect to home page	Redirected to homepage	Success
8.	Signup with less than 6 chars Password	Validation corresponding to Password	User is redirected to Signup Page with valid message	Success
9.	Signup with Already Registered Email ID	Validation Corresponding to Email	User is redirected to Signup Page with Valid message	Success
10	Cart while Logged in	Cart is shown with product and price details	Cart is shown with product and price	Success
11.	Payment	Payment Successful page should be shown	Payment Successful page is shown	Success
12.	Logout	User is Logged Out	User is Logged Out	Success

7. Snapshot of System :

Home Page Of Website :

Ecommerce-Store

Cart Login Signup

Select Category

Mobile Phones


Laptops

Tablets

Audio


Television

Cameras




Realme X7 Max (128GB ROM, 8GB RAM, RMX3031, Asteroid Black)
₹25999

Add To Cart




Apple iPhone 11 (128GB ROM, 4GB RAM, MHDJ3HN/A, White)
₹50999

Add To Cart




OnePlus Nord CE 5G (256GB ROM, 12GB RAM, Blue Void)
₹27999

Add To Cart




Vivo Y12s (32GB ROM, 3GB RAM, Phantom Black)
₹12999

Add To Cart




Dell Inspiron 3501 (D560385WIN9S) Core i5 11th Gen Windows 10 Home Notebook (8GB RAM, 1TB HDD + 256G
₹64999

Add To Cart




Apple MacBook Air (MGND3HN/A) M1 Chip macOS Big Sur Laptop (8GB RAM, 256GB SSD, Apple M1 GPU, 33.78c
₹82990

Add To Cart




Apple MacBook Pro (MYD82HN/A) M1 Chip macOS Big Sur Laptop (8GB RAM, 256GB SSD, Apple M1 GPU, 33.78c
₹109000

Add To Cart




HP 15s-du3060TX (360L6PA) Core i5 11th Gen Windows 10 Home Laptop (8GB RAM, 1TB HDD, NVIDIA GeForce
₹56990

Add To Cart




Apple iPad mini 6 WiFi + 5G iPadOS Tablet (iPadOS 15 Apple A15 Bionic Chip 21.08cm (8.3 Inches) 256G
₹74990

Add To Cart




Apple iPad mini 6 WiFi + 5G iPadOS Tablet (iPadOS 15, Apple, A15 Bionic Chip, 21.08cm (8.3 Inches),
₹60990

Add To Cart



Apple iPad mini 6 WiFi + 5G iPadOS Tablet (iPadOS 15, Apple, A15 Bionic Chip, 21.08cm (8.3 Inches),
₹60990

Add To Cart



Apple iPad mini 6 WiFi iPadOS Tablet (iPadOS 15, Apple A15 Bionic chip, 21.08cm (8.3 inches), 64GB R
₹46900

Add To Cart

Select Category :

Select Category
Mobile Phones
Laptops
Tablets
Audio
Television
Cameras

Signup :

Ecommerce-Store

[Cart](#) [Login](#) [Signup](#)

Register

Create your account. It's free and only takes a minute.

First Name

Last Name

Email

Password

Register Now

Login :

Login

Email

Password

Login

After Login Add to Cart :

Ecommerce-Store

Welcome sir/madam!!! Cart Logout

Select Category

Mobile Phones


Laptops

Tablets

Audio


Television

Cameras




Realme X7 Max (128GB ROM, 8GB RAM, RMX3031, Asteroid Black)
₹25999

Add To Cart




Apple iPhone 11 (128GB ROM, 4GB RAM, MHDJ3HN/A, White)
₹50999

Add To Cart



OnePlus Nord CE 5G (256GB ROM, 12GB RAM, Blue Void)
₹27999

Add To Cart



Vivo Y12s (32GB ROM, 3GB RAM, Phantom Black)
₹12999

Add To Cart

Developed by Jaivin Domadiya

Page 27

Select Category

[Mobile Phones](#)[Laptops](#)[Tablets](#)[Audio](#)[Television](#)[Cameras](#)

Realme X7 Max (128GB ROM, 8GB RAM, RMX3031, Asteroid Black)

₹25999

[Add To Cart](#)

Apple iPhone 11 (128GB ROM, 4GB RAM, MHDJ3HN/A, White)

₹50999

[-](#) Qty : 1 [+](#)

OnePlus Nord CE 5G (256GB ROM, 12GB RAM, Blue Void)

₹27999




[Add To Cart](#)

Vivo Y12s (32GB ROM, 3GB RAM, Phantom Black)

₹12999

[Add To Cart](#)[Show Cart :](#)

Cart

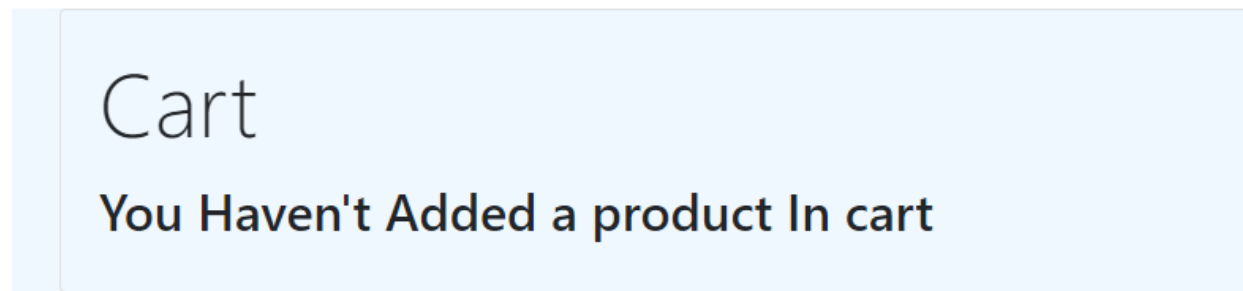
Image	Product	Price	Quantity	Total
	Apple iPhone 11 (128GB ROM, 4GB RAM, MHDJ3HN/A, White)	₹50999	1	₹50999
	Dell Inspiron 3501 (D560385WIN9S) Core i5 11th Gen Windows 10 Home Notebook (8GB RAM, 1TB HDD + 256G	₹64999	1	₹64999
	pTron Bassbuds Evo In-Ear Passive Noise Cancellation Truly Wireless Earbuds with Mic (Bluetooth 5.0	₹999	1	₹999
Total:				₹116997

[Fill shipping Details](#)

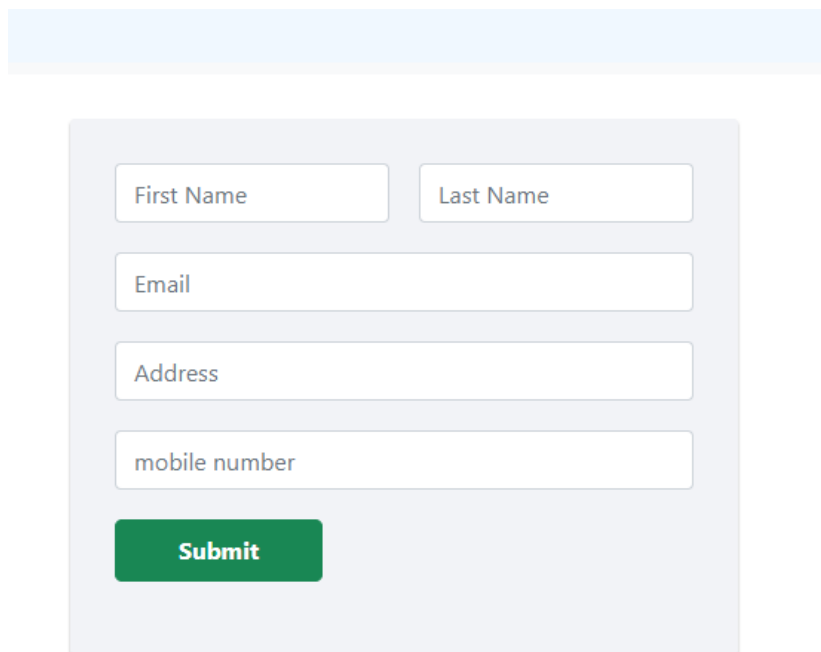
Activate Windows

Go to Settings to activate Windows.

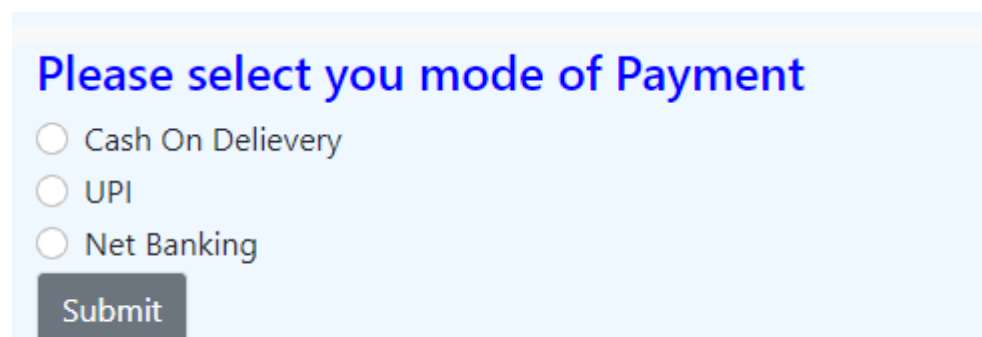
When Cart is Empty :



 **Fill Shipping Details Form :**

A light blue rectangular box with a white border. Inside, there is a form with four input fields: "First Name", "Last Name", "Email", and "Address". Below these fields is a "mobile number" field. At the bottom of the form is a green "Submit" button.

 **Choose Payment Options :**

A light blue rectangular box with a white border. Inside, the text "Please select you mode of Payment" is written in a blue font. Below this text are three radio button options: "Cash On Delievery", "UPI", and "Net Banking". At the bottom of the form is a grey "Submit" button.

Select Payment Options :

Please select you mode of Payment

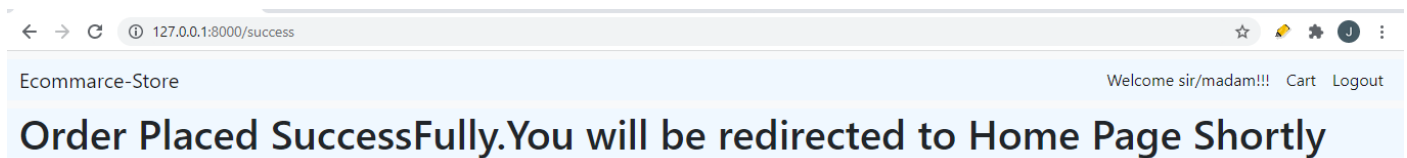
☒ Cash On Delievery

☐ UPI

☐ Net Banking

Submit

Order Placed :



8. Conclusion :

- In this project we have successfully implemented the Admin-side & Customer-side functionality, Admin can add the products or change the product information. Admin can show all the products with their necessary details, admin can also delete the particular product from website.

Here two types of roles are there so ..

1)Admin

- Admin can sell the products with product's details, after based on admin response the customer either will get confirmation or rejection.

2)Customer

- Buyer can also use the cart for buying more product and make payment for all once.

- Of all the Functionalities we have listed in SRS we have implemented as many as we can in the given time . Following were executed Successfully:

1)Login

2)Signup

3)Search by Category

4)Cart

5)Logout

6)Login Validation

7)Signup Validation

9. Limitations And Future Extensions :

9.1 Limitations :

=> Some Limitations of the application are:

- 1) Non Editable Personal Details.
- 2) Non Integrated Payment
- 3) No Payment so no order management.

=> We can make incremental model by developing this functionalities one by one and publishing it.

9.1 Future Extensions :

To take over the limitations we are planning this future extension in our system.

- 1) Good and better User Interface.
- 2) Implements Payment Gateway.
- 3) Implements Method to display statistics data.
- 4) Customer will be able to cancel order in future.
- 5) By modifying this system, system can be used broader scale.
- 6) Customer can filter the product according to price, condition.

10. Bibliography :

- docs.djangoproject.com
- [Stackoverflow.com](https://stackoverflow.com)
- youtube.com
- javatpoint.com