A

PROJECT REPORT

ON

E-COMMERCE WEBSITE

By

SHAH ANSH (CE-117)

PATEL LUV P. (CE-104)

B.Tech CE Semester-IV

Subject: SEPP

Department Of Computer Engineering

Faculty Of Technology,

Dharmsinh Desai University, Nadiad.

Contents:

1. **ABSTRACT .**

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 inorder to just seethrough the website.They can browse through various catagories as per the filter available.

2.INTRODUCTION .

2.1Brief 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:

- Django
- Python
- Database
- Bootstrap
- Html

2.3. Tools :

- Visual Studio Code
- Git
- Database

3. Software Requirement Specifications .

3.1 Product Scope:

This Site is created to enable the buyers to buy various products be it clothes, 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.REGISTRATION :

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. MANAGE ACCOUNT :

2.1: EDIT PROFILE DETAIL:

- Here ,the Customer if by mistake has entered any wrong info he/she can correct it.
  INPUT: Field where mistake was done.
  OUTPUT: Field changed Successfully .

2.2: VIEW PAST ORDER:

- Here, the customers who already have an account can give input as date range and can see past orders between those dates.
  INPUT: Start Date and end date in DD/MM/YYYY format
  OUTPUT: All orders that were placed in/between given dates.

2.3: 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

2.4: ADD WALLET DETAILS:

- Here, the customer if not made can make wallet and if already made deposit money and use them to buy products
  INPUT: registration page if wallet not there else money deposit page/during payment
  OUPUT: registered successfully/deposited xxx/- successfully/used xxx/-rupee to buy xyz product successfully.

2.5: Track Your Order:

- **Here,**The customer can track the status of his order.
  INPUT: Information of Order Which is to be Tracked.
  Output: Order Status.

2.6: Cancel Your Order:

- Here customer can cancel his/her order.

INPUT: Select Order which has to be cancelled.

OUTPUT: Order Cancelled Successfully. Refund will be added to your account in 2 to 3 working days.

2.7: Refund Again:

- In Case refund is not received in account, customer can use this option to get refund.
  INPUT: Order for which refund is not received.
  OUTPUT: Refund will be added in 2-3 working days.

3.PRODUCT SELECTION :

3.1 WISHLIST :

- In this, the customers can save/store the item they liked while browsing the website which they wish to buy in near future or want to add it in their cart if they wish to checkout.

INPUT: Save to wishlist button in the product page is selected/Tick Marked.

OUTPUT:Item in which customer is interested in, is added to wishlist successfully.

3.2 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, apply coupon code if they have which help them to save more on purchase and then can finally place order.

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

OUTPUT: Item is added to Cart Successfully

3.3 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 i.e from bulit quality, manufacturer detail,and all the other product related details.
  INPUT : Particular product selected.]
  OUTPUT : All the details related to the product selected are displayed on screen.

4..: SEARCH 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:HOME BUTTON:

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

5.: CUSTOMER SUPPORT :

- It serves as the backbone of the whole system in which if the buyer has any queries or has problem related to any of the of the experience in viewing or is related to his/her account or order.

INPUT : Any Queries related to the website or Shopping experience or any general support arised by customers.

OUTPUT : Information closest to solution is displayed or Support staff details are displayed.

6.: SELLER UPLOAD/MODIFY :

6.1: ADD PRODUCT:

- Here, the sellers can add new products.

INPUT: Product Details which has to be uploaded

OUTPUT: Product Added Successfully.

6.2: Update Product:

- Here, the seller can update the stock of products he want.
- **INPUT:** Product name and amount to increase it
- **OUPUT:** Product xyz incremented by amount**i.**

6.3: Remove Product:

- .Here, the seller can remove product he wants.

INPUT: Product name that is to be removed

OUTPUT: Product xyz removed successfully

7. SELLER ACCOUNT :

7.1.: ORDER SUMMARY:

- Here, Seller can see all the orders he has received that have been shipped,that has to be shipped that are just dispatched etc.
-

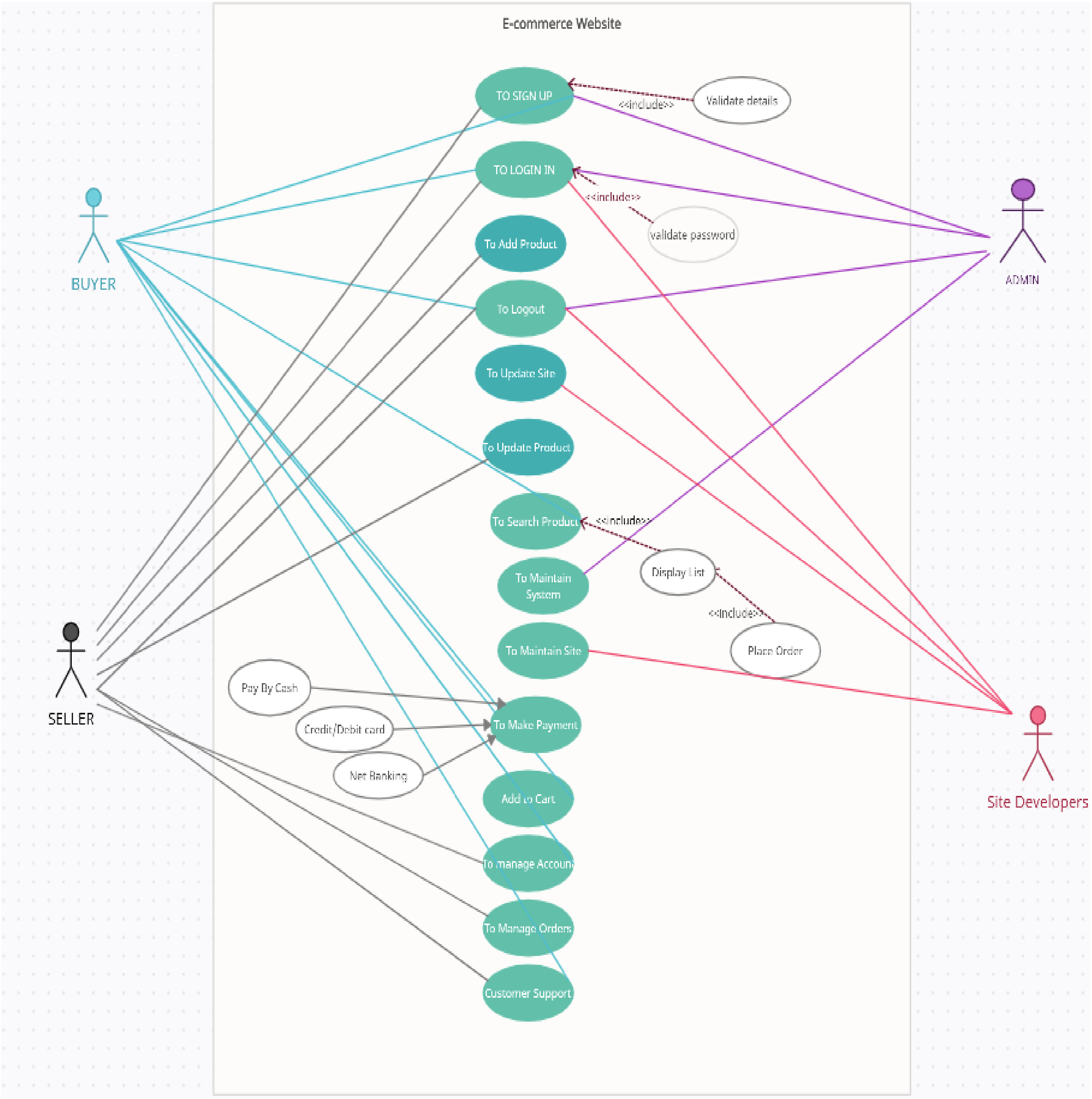
  INPUT: Product info (shipped, shipping, etc)
  OUTPUT: List of products satisfying criteria

7.2.: Product Money change:

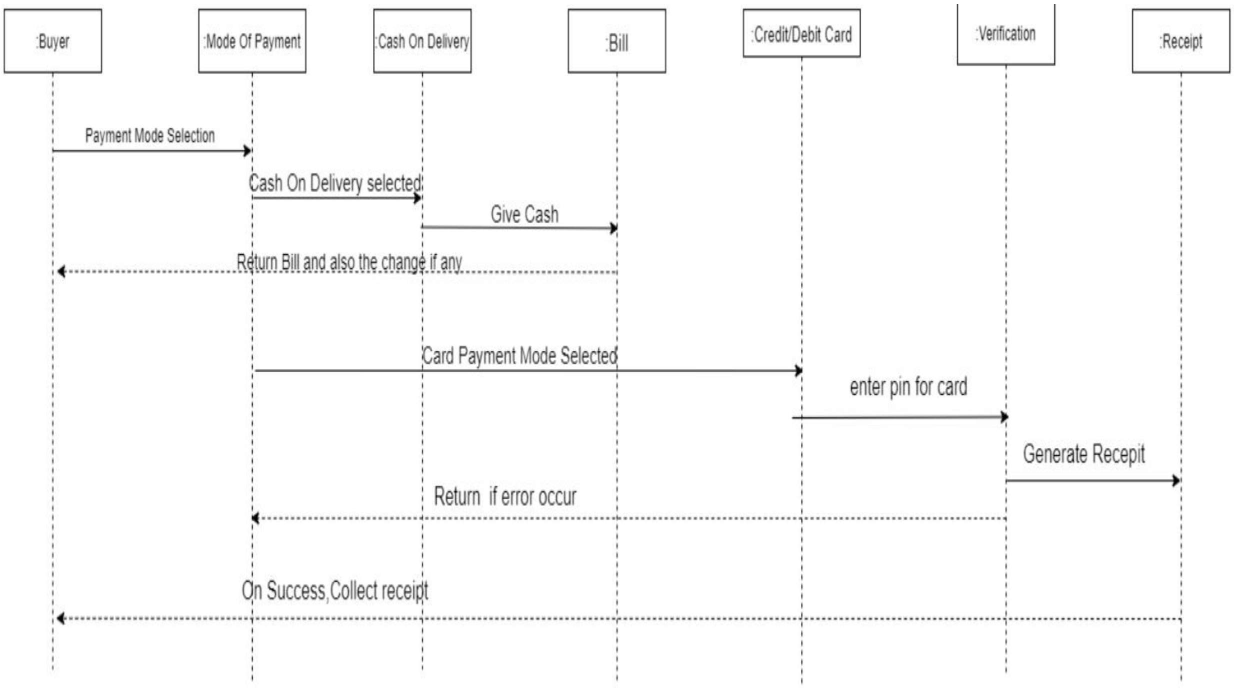
- Here, the seller can change the price of given product.
  INPUT: Product name and amount and(+for increment or – for decrement)
  OUTPUT: Price of Product xyz updated Successfully**.**

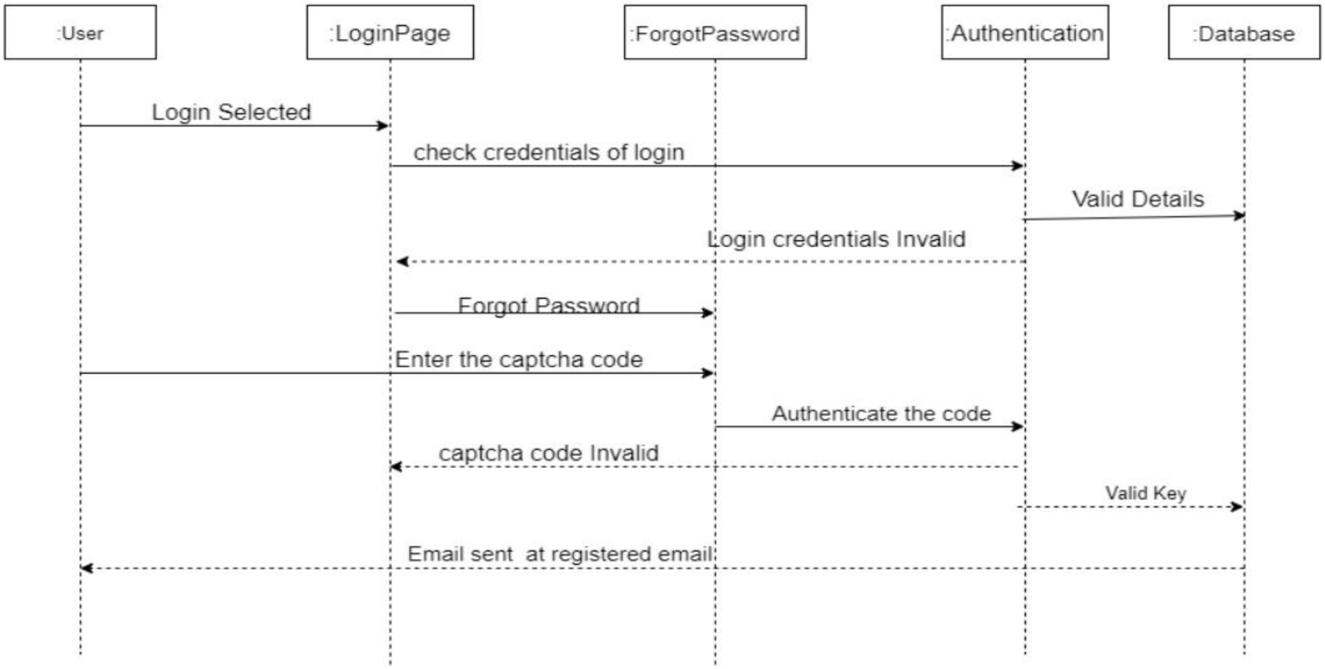
4.DESIGN .

4.1 USE CASE DIAGRAM:

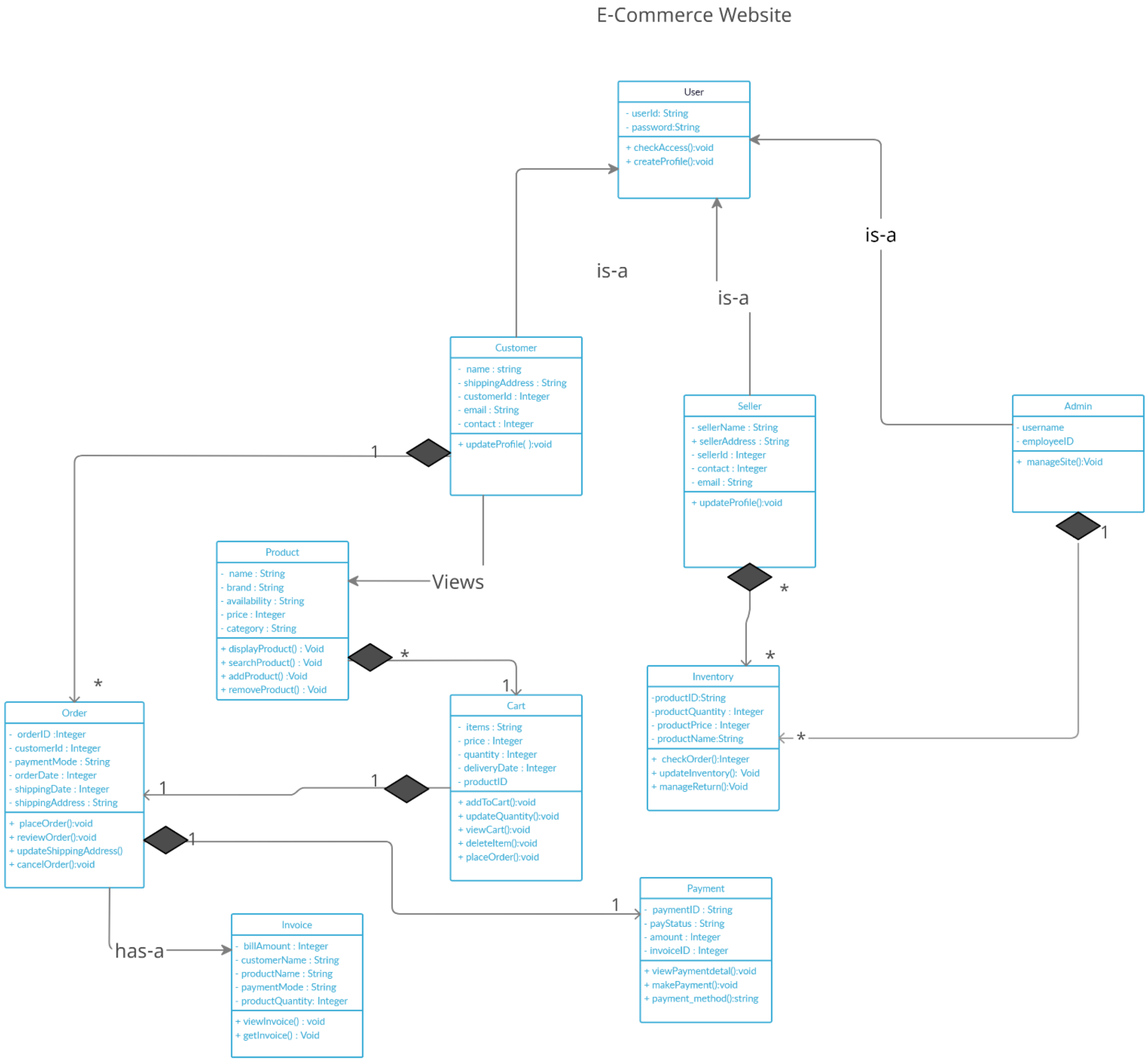


4.2SEQUENCE DIAGRAM :

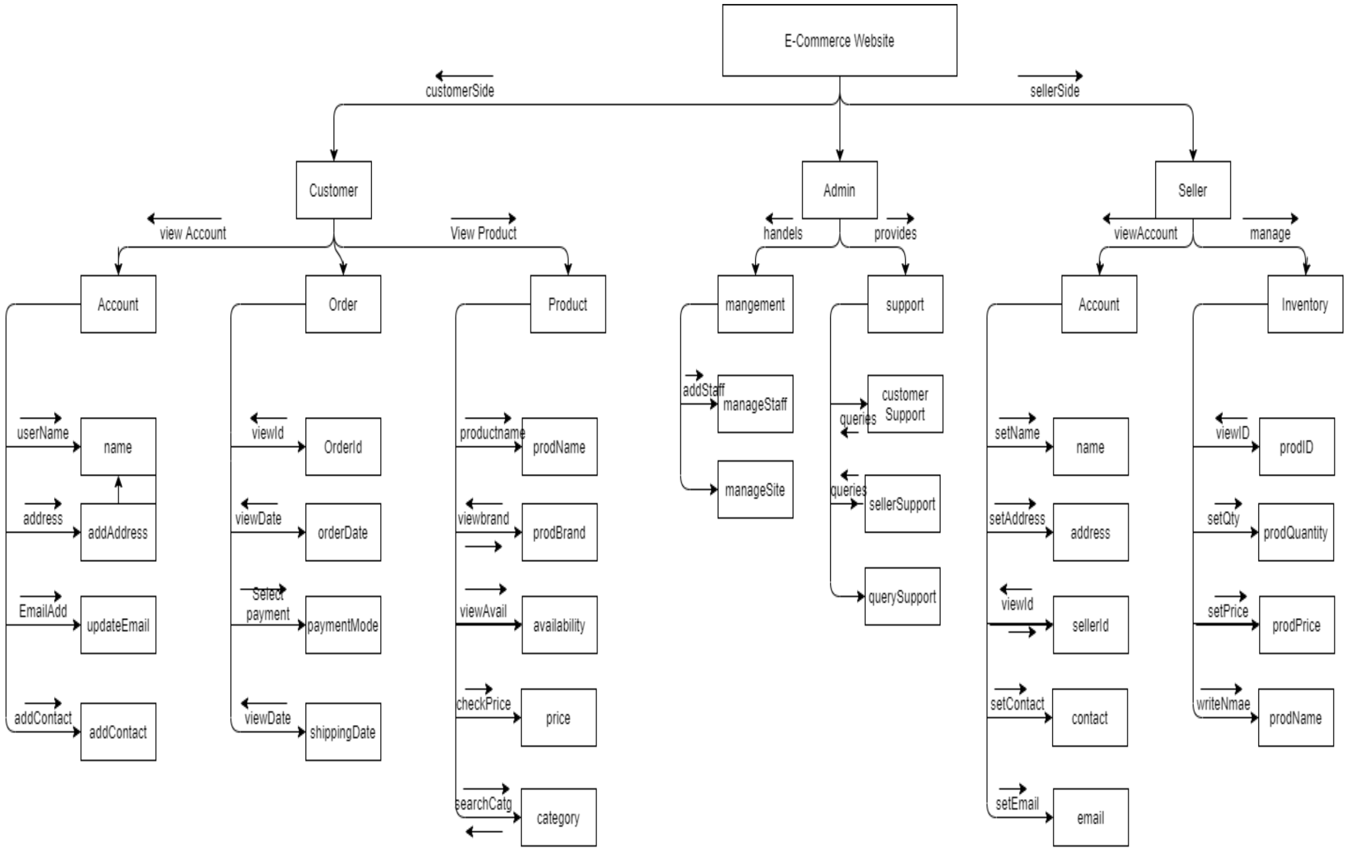




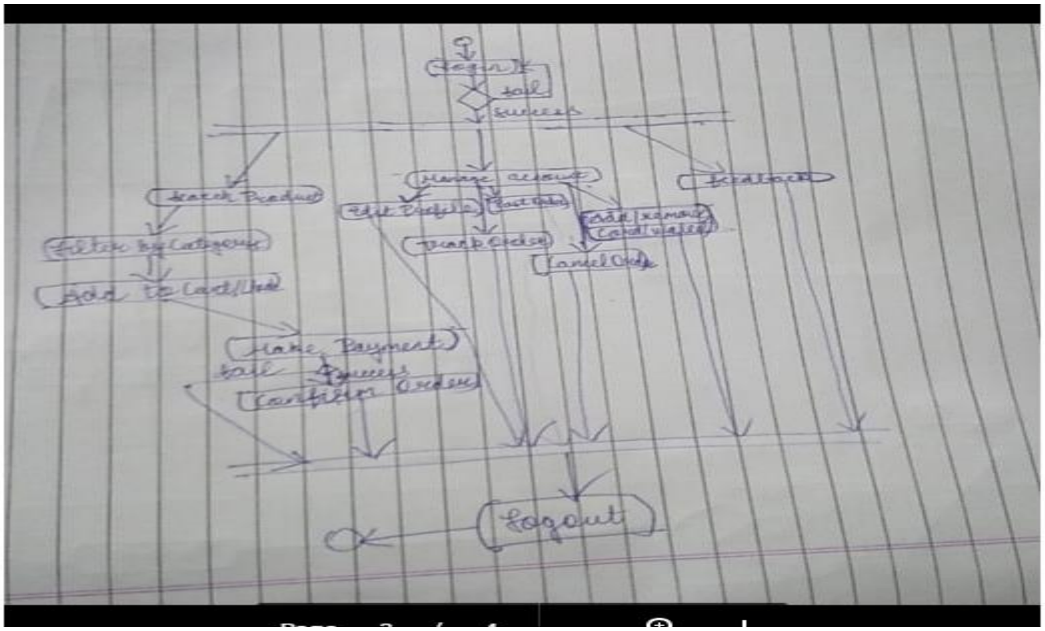
4.3 CLASS DIAGRAM :



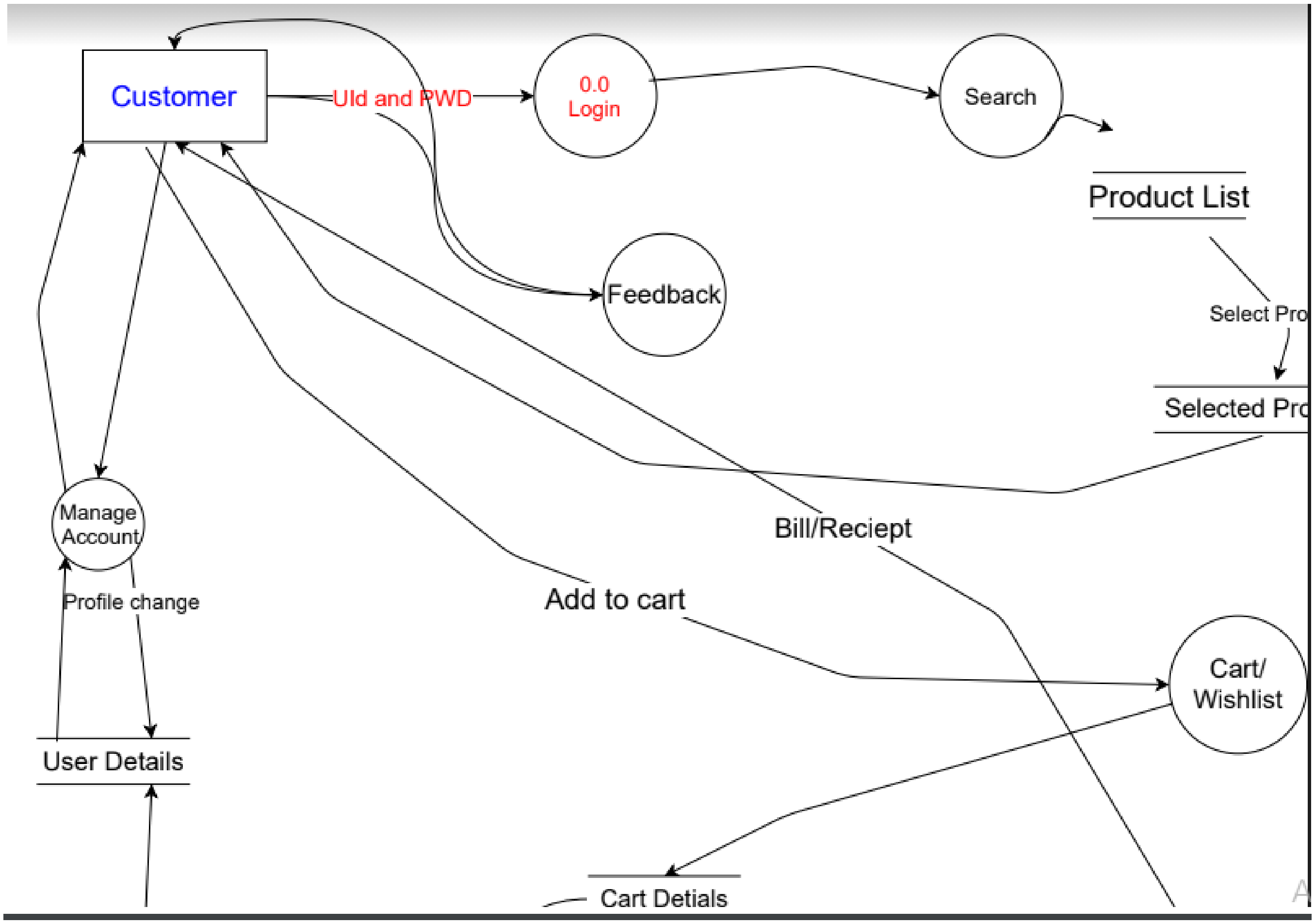
4.4 Structure Diagram :

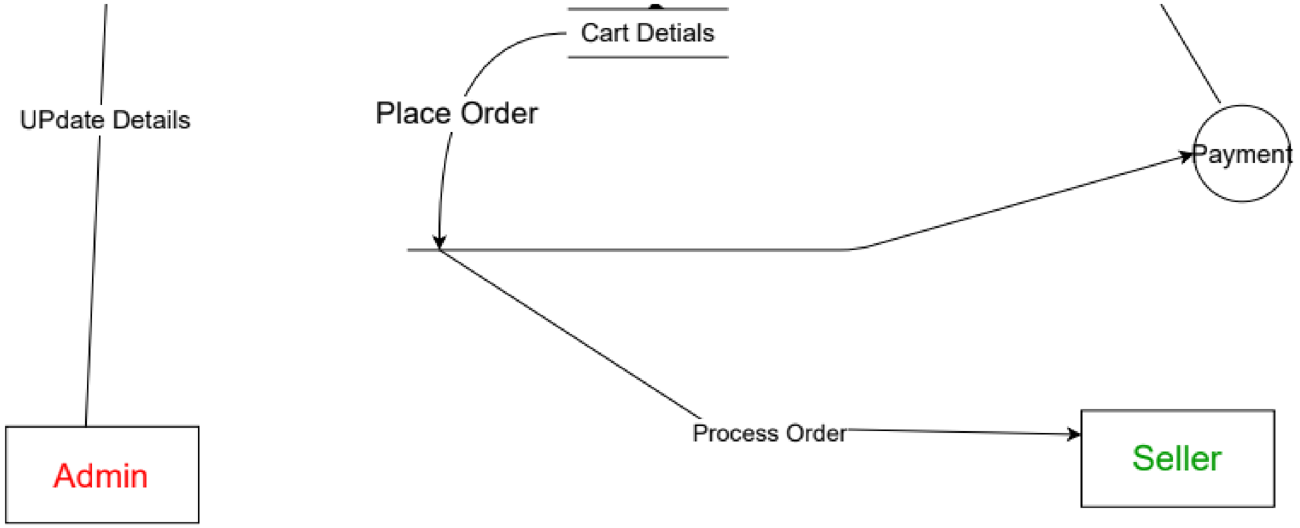


4.5:Activity Diagram:



4.6 Data Flow Diagram:





4.7 Data Dictionary:

1)Category Table:

Name of the Field	Type	Length	Attributes
ID	INT	-	AI,Primary Key
Name	Varchar	100	-

2)Customer Table:

Name of the Field	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 the Field	Type	Length	Attributes
ID	Int	-	AI,Primary Key
Name	Varchar	100	-
Price	Int	-	-
Description	Varchar	200	-
Image	Varchar	100	-
Fk_id	Int	-	-

5)Implementation Details:

The System mainly consists of following modules. Screenshots are added in separate section:

1)Registration:

=>Here, User can become user of the ecommerce website.

2)Manage Account:

=>Here, User can edit personal info and check order status.

3)Add/Remove Card:

=>Here, User can add or remove card details as a part of payment

4)Add Wallet Details:

=>Here, User can add wallet details to make payment easier.

5)Search Module:

=>Here, User can Filter out products according to category.

6)Product Selection:

=>Here, User can select some product and add it to cart.

7)Customer Support:

=>Here, User can put up query and get answers to it.

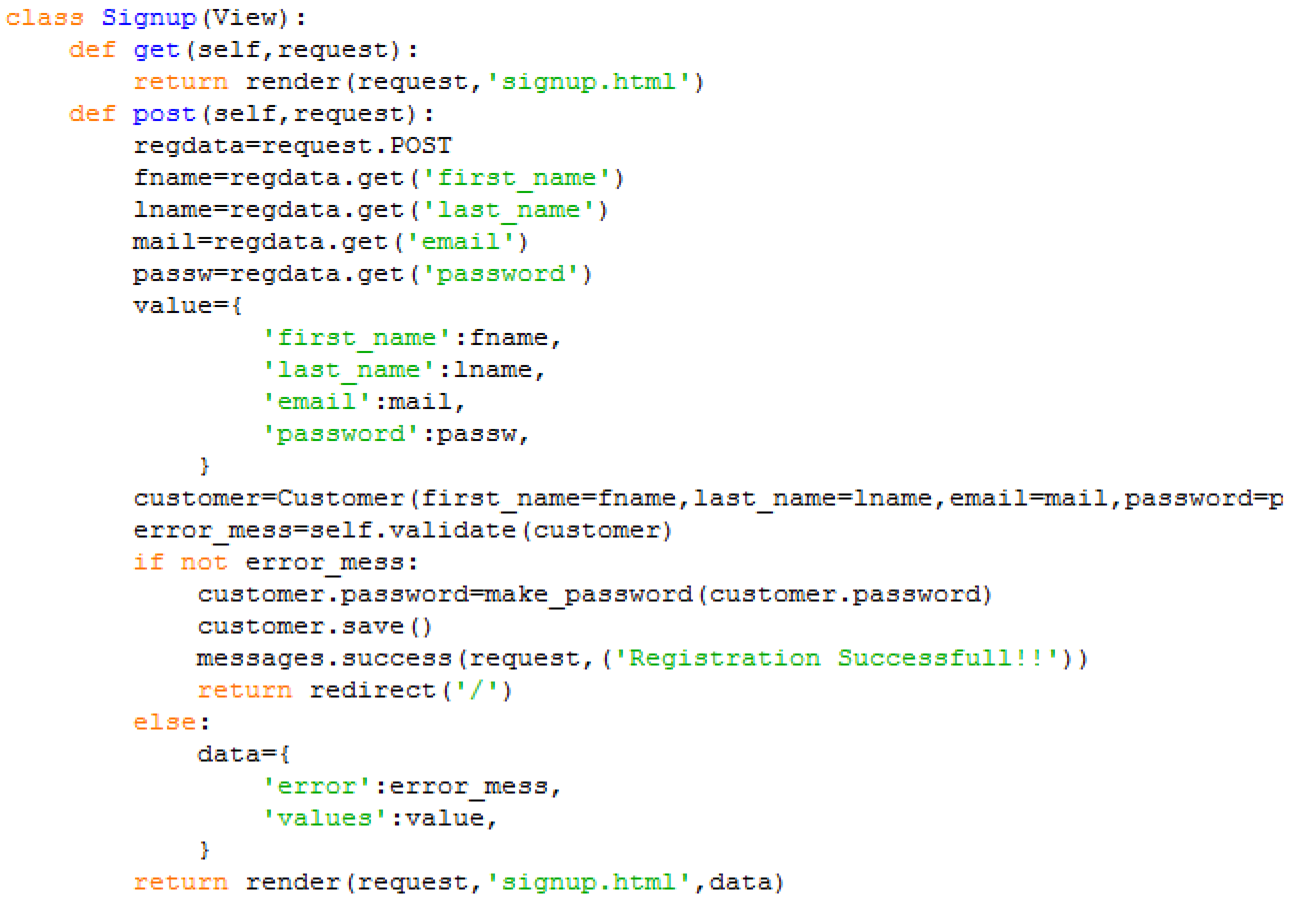
8)Seller Functionality:

=>This was made to deliver order related issues.

Major Functions:

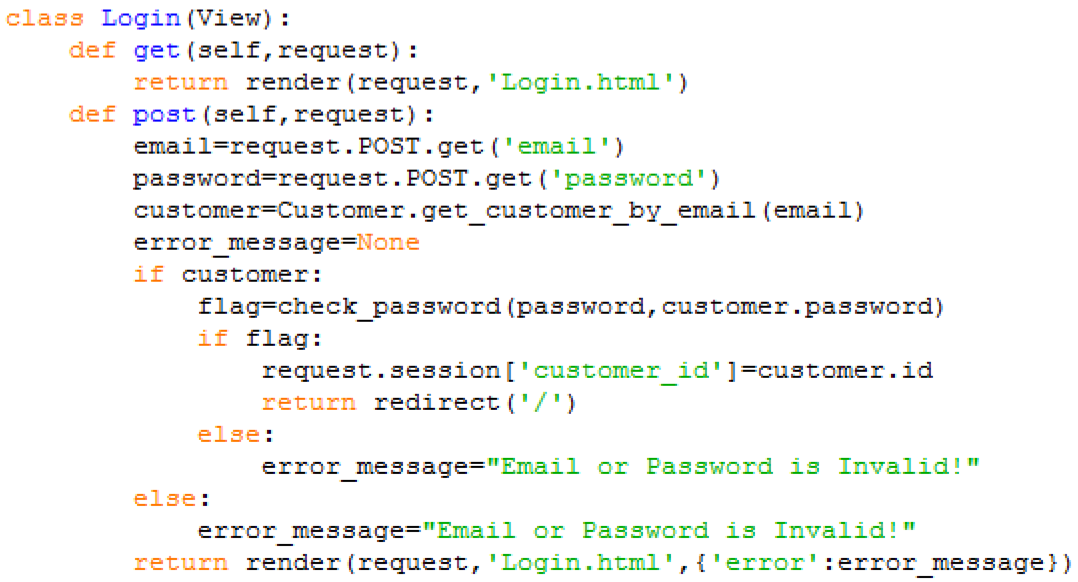
1)Signup Funcn:

=>After Customer Enter Details on Signup Page how data goes to database is shown:



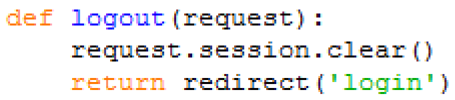
```
class Signup(View):
    def get(self,request):
        return render(request,'signup.html')
    def post(self,request):
        regdata=request.POST
        fname=regdata.get('first_name')
        lname=regdata.get('last_name')
        mail=regdata.get('email')
        passw=regdata.get('password')
        value={
                'first_name':fname,
                'last_name':lname,
                'email':mail,
                'password':passw,
            }
        customer=Customer(first_name=fname,last_name=lname,email=mail,password=p
        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)
```

2)After Customer Enter Details they are valid or not if valid then login are done in this function:



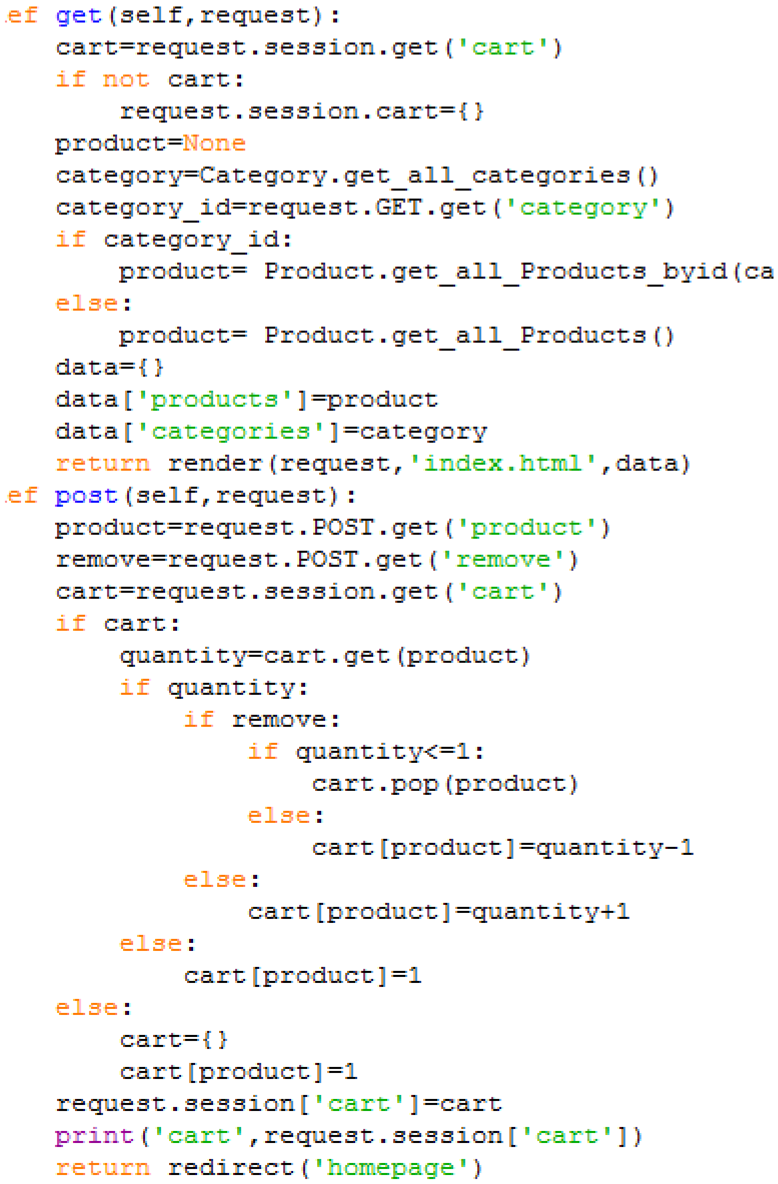
```
class Login(View):
    def get(self,request):
        return render(request,'Login.html')
    def post(self,request):
        email=request.POST.get('email')
        password=request.POST.get('password')
        customer=Customer.get_customer_by_email(email)
        error_message=None
        if 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})
```

3)Logout Function:



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

4)After Login setting up Cart using Session:



```
ef 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(ca
   else:
       product= Product.get_all_Products()
   data={}
   data['products']=product
   data['categories']=category
   return render(request,'index.html',data)
ef 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
   else:
       cart={}
       cart[product]=1
   request.session['cart']=cart
   print('cart',request.session['cart'])
   return redirect('homepage')
```

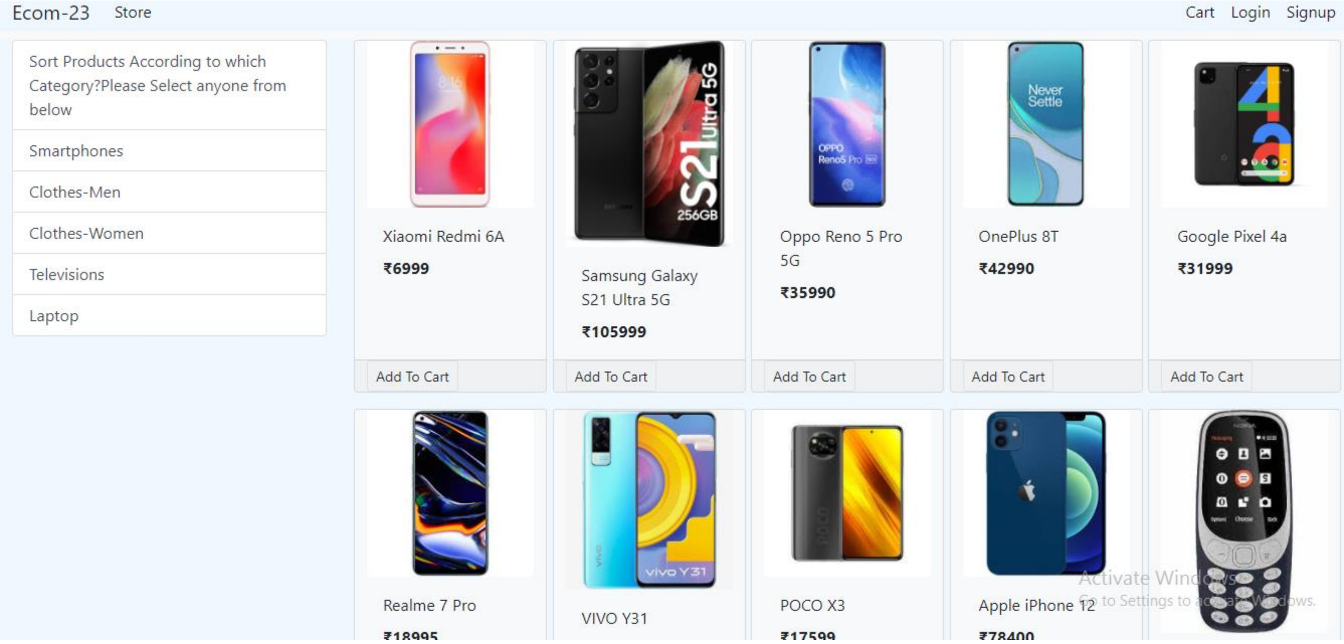
6)Testing:

=>We Had adopted Manual Testing Approach to check and fix Bugs. Here is a brief report of it:

Sr No.	Test Detail	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	Selected	Selected	Success

	Listing	**Category Products will be shown**	**category products are shown**	
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)Screenshots:



Sort Products According to which Category?Please Select anyone from below

- Smartphones
- Clothes-Men
- Clothes-Women
- Televisions
- Laptop

Xiaomi Redmi 6A
₹6999
Add To Cart

Samsung Galaxy S21 Ultra 5G
₹105999
Add To Cart

Oppo Reno 5 Pro 5G
₹35990
Add To Cart

OnePlus 8T
₹42990
Add To Cart

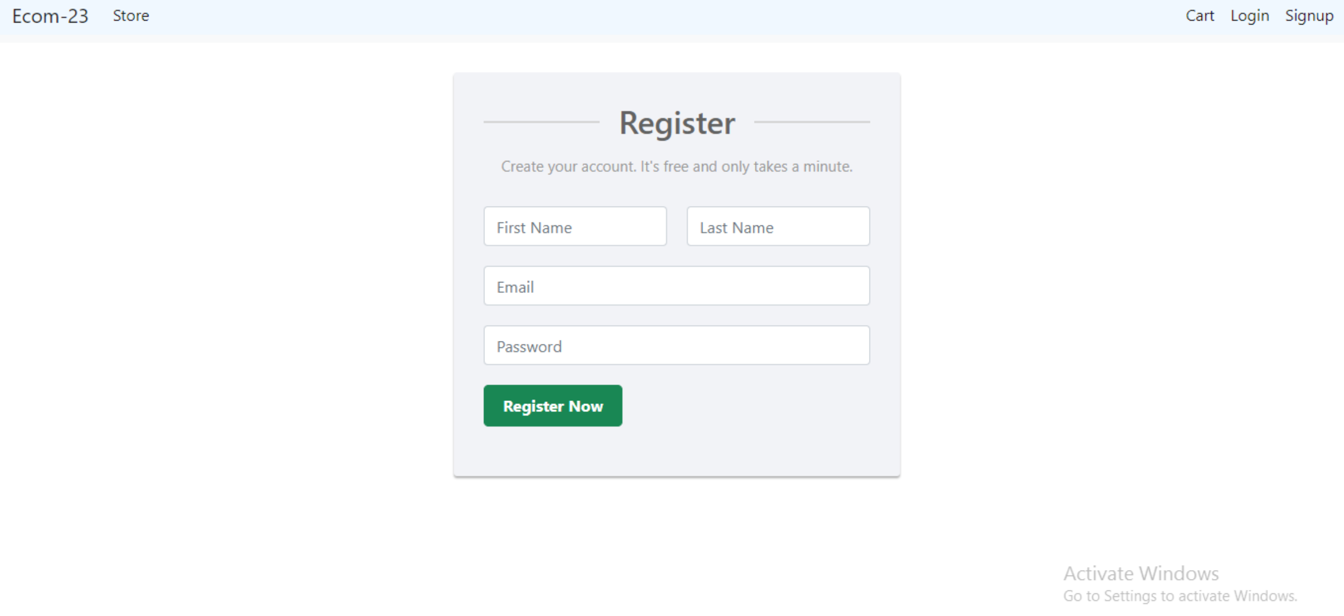
Google Pixel 4a
₹31999
Add To Cart

Realme 7 Pro
₹18995

VIVO Y31

POCO X3
₹17599

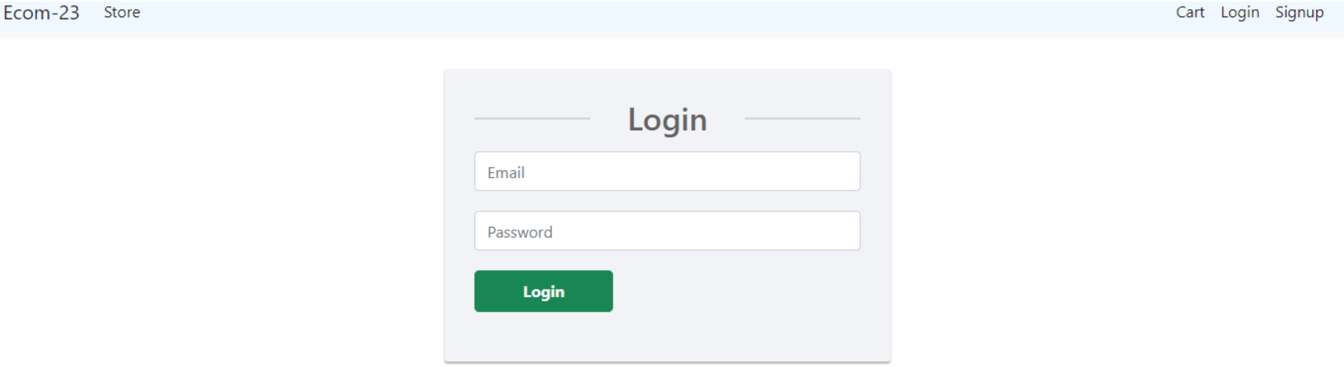
Apple iPhone 12
₹78400

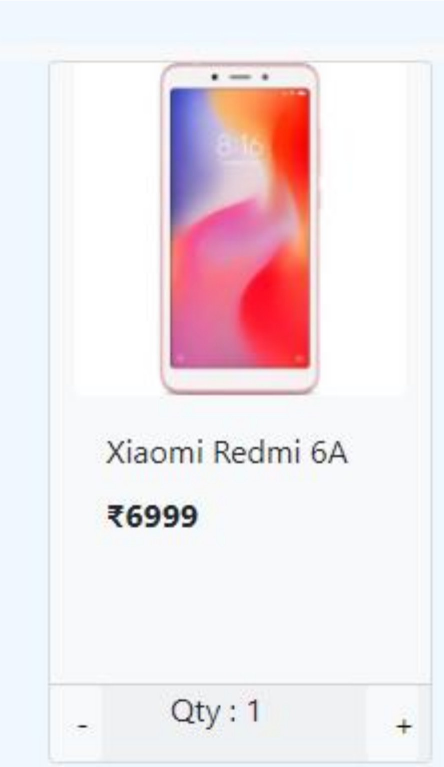


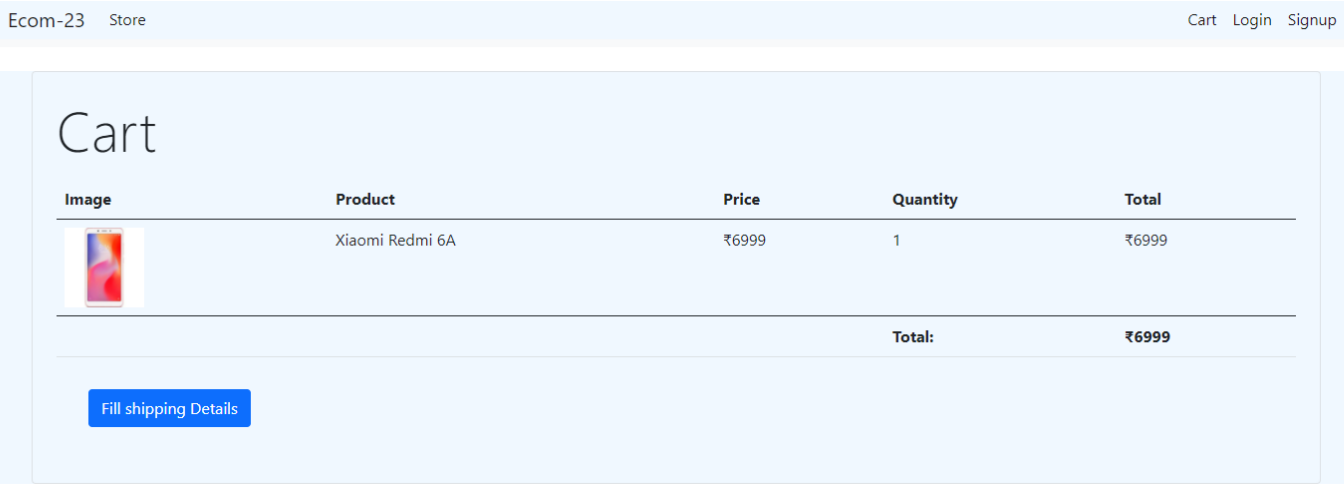
Register

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

First Name
Last Name
Email
Password

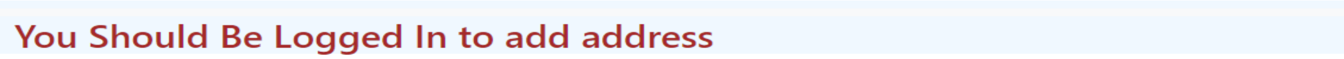

Ecom-23
Store
Cart
Login
Signup
Login
Email
Password
Login


Xiaomi Redmi 6A
₹6999
-
Qty : 1
+

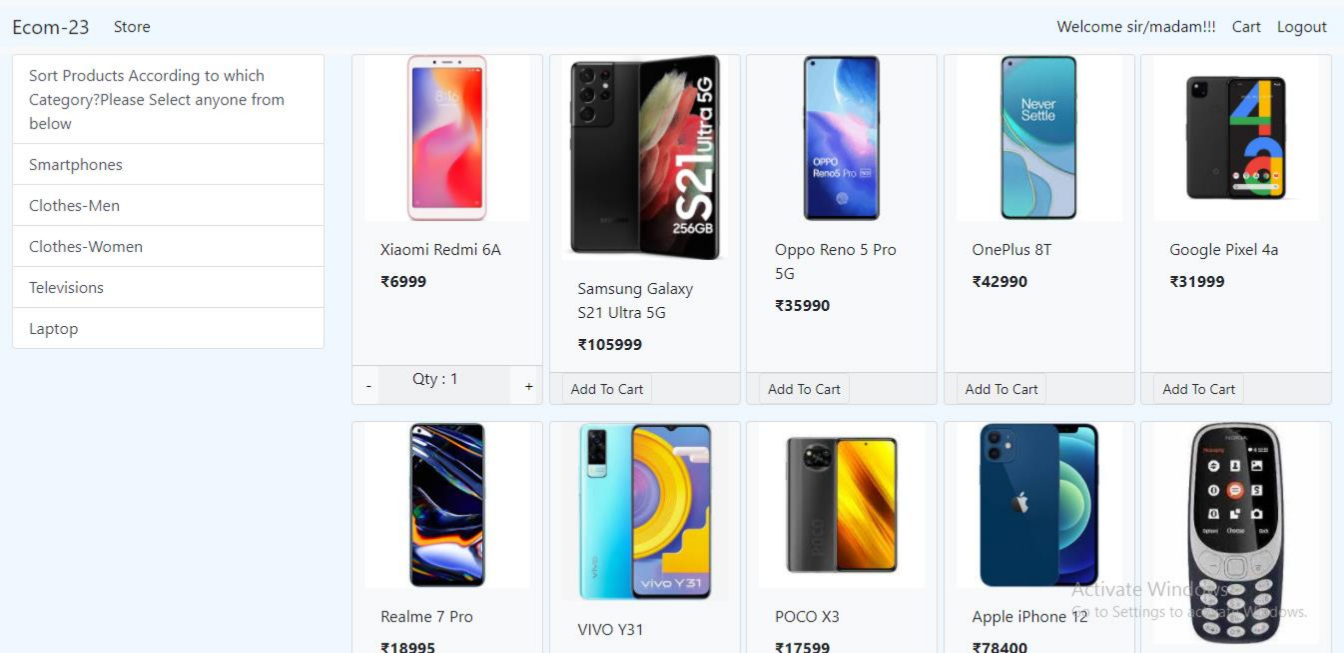


Cart

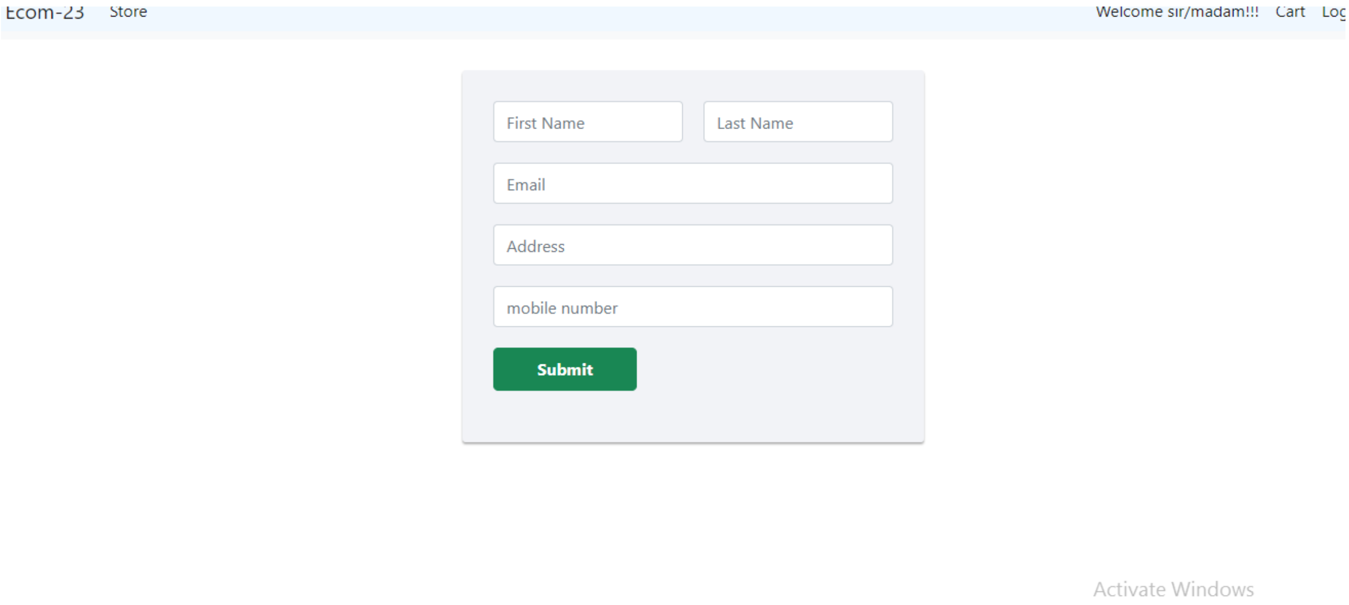
Image	Product	Price	Quantity	Total
	Xiaomi Redmi 6A	₹6999	1	₹6999
			Total:	**₹6999**

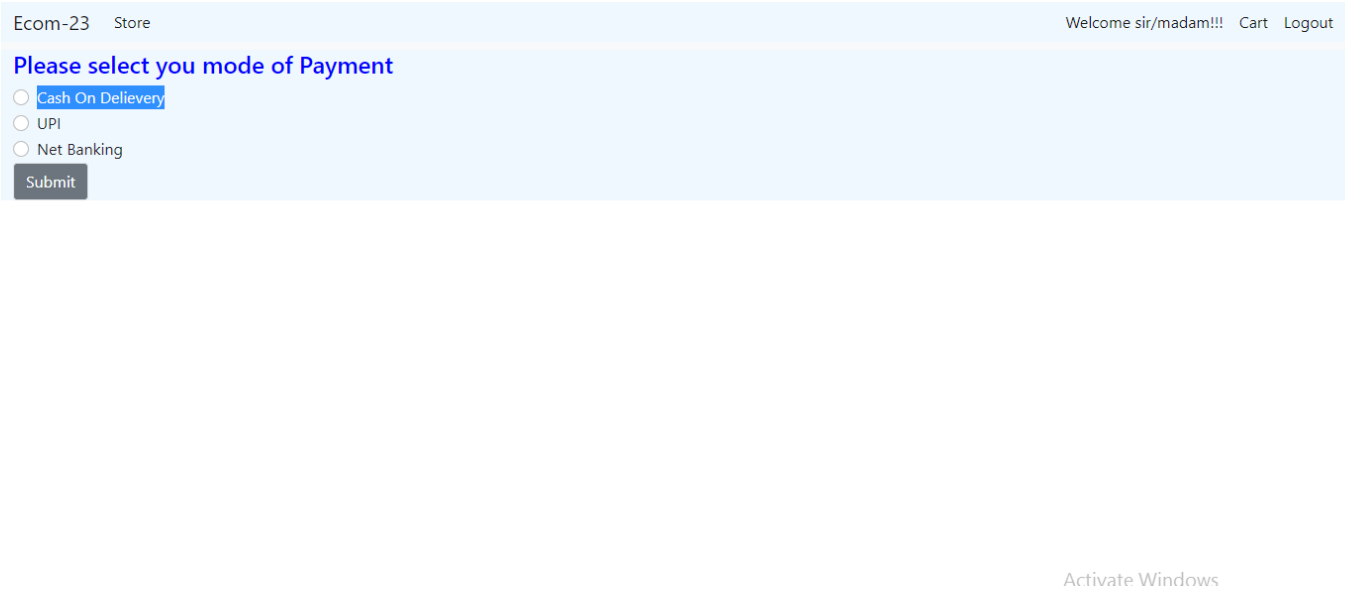

You Should Be Logged In to add address

After Login:



Shipping Details form:

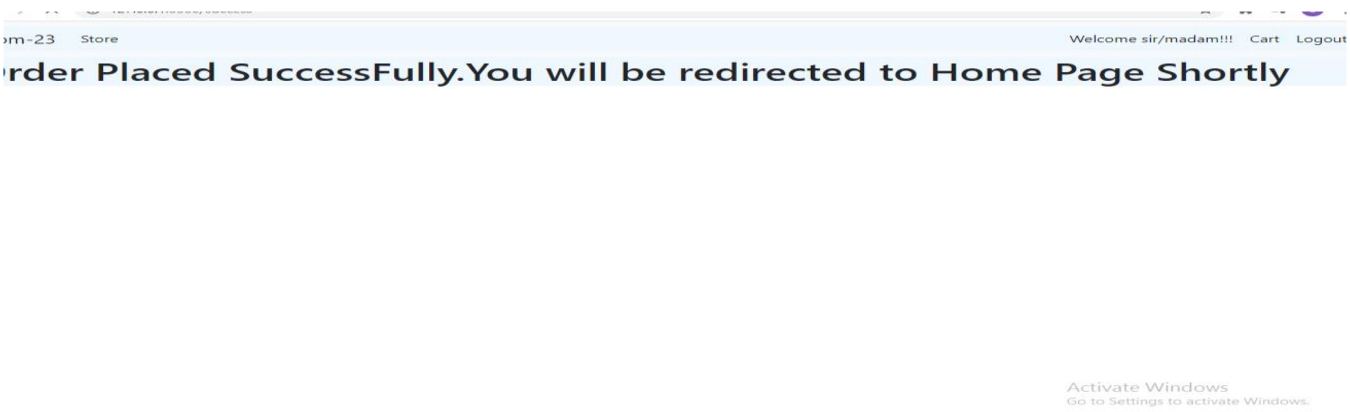




Please select you mode of Payment

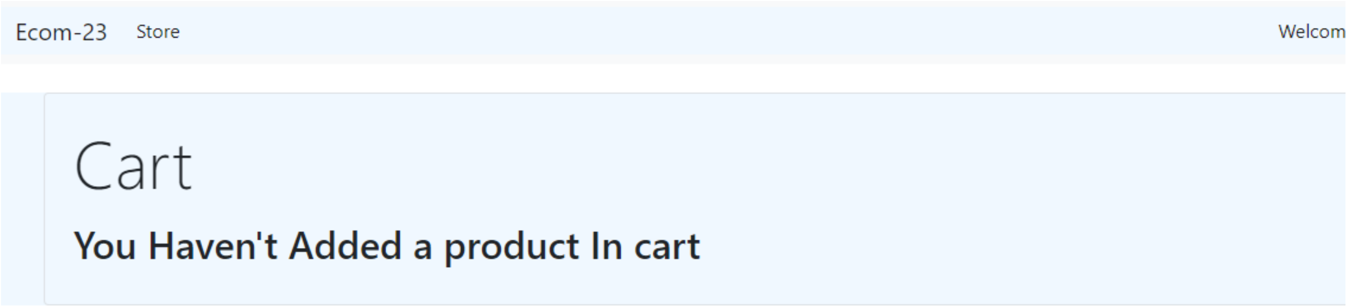
○ Cash On Delievery
○ UPI
○ Net Banking

Submit



ırder Placed SuccessFully.You will be redirected to Home Page Shortly

When cart is empty:



Cart

You Haven't Added a product In cart

8) Conclusion:

=>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

=>After that Testing was Performed to find bugs in the application and were solved.

9)Limitations and future enhancement:

=>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.

10)Reference:

1)Stackoverflow.com

2)docs.django.com

3)github.com