Web Application Development Project

President Institute of Computer Application

T.Y. BCA(2023-24) Group no : 17

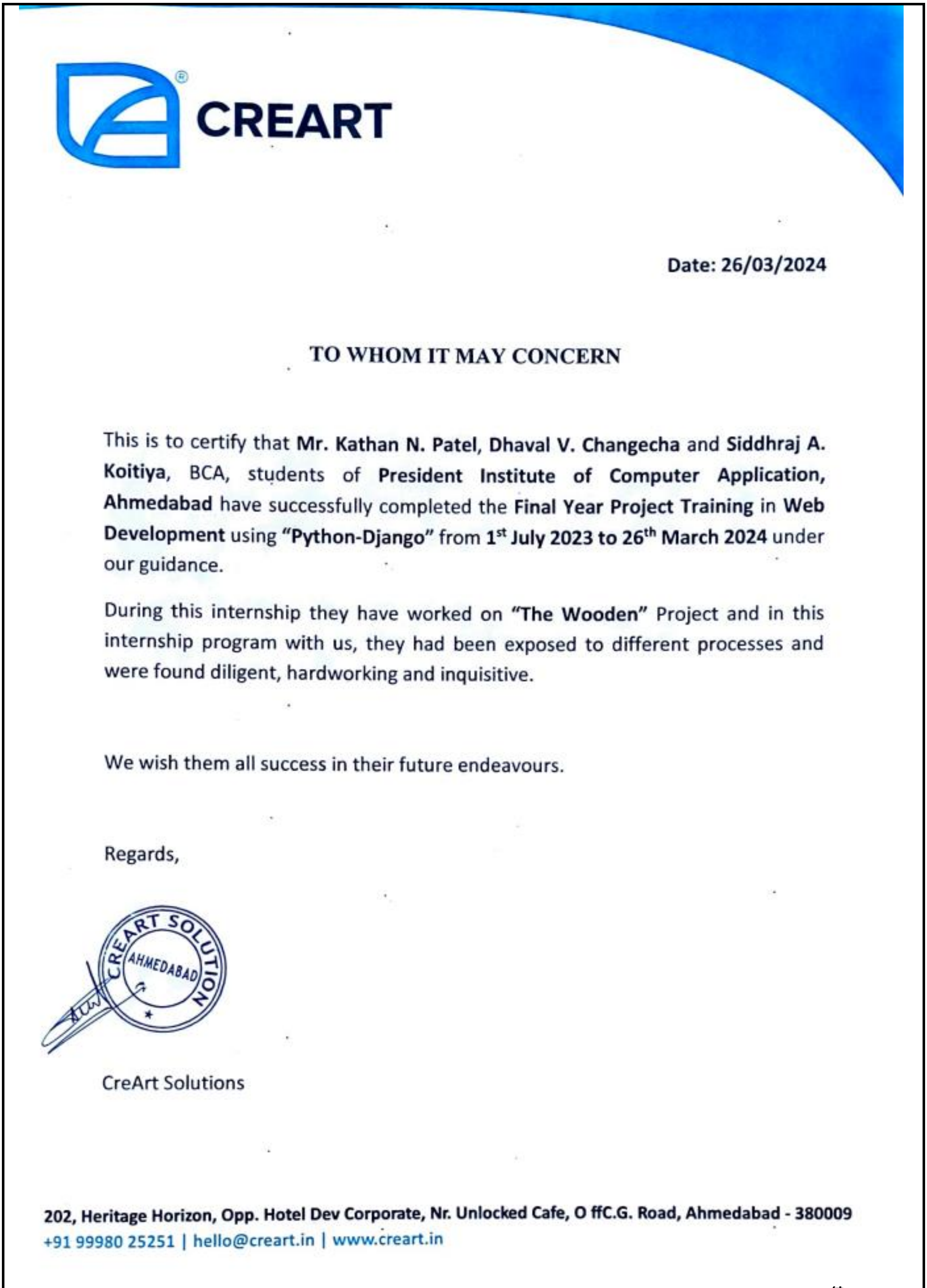
The Wooden

Group Information

Name :	Patel Kathan Navinchandra | Roll no : 6097
Enrollment no.:	202112102543
Name :	Changecha Dhaval Vijaybhai | Roll no : 6108
Enrollment no.:	202112102406
Name :	Koitiya Siddhraj Ashokbhai | Roll no : 6168
Enrollment no.:	202112102474

Index

Company Certificate



CREART

Date: 26/03/2024

TO WHOM IT MAY CONCERN

This is to certify that **Mr. Kathan N. Patel, Dhaval V. Changecha** and **Siddhraj A. Koitiya**, BCA, students of **President Institute of Computer Application, Ahmedabad** have successfully completed the **Final Year Project Training** in **Web Development** using **"Python-Django"** from **1st July 2023 to 26th March 2024** under our guidance.

During this internship they have worked on **"The Wooden"** Project and in this internship program with us, they had been exposed to different processes and were found diligent, hardworking and inquisitive.

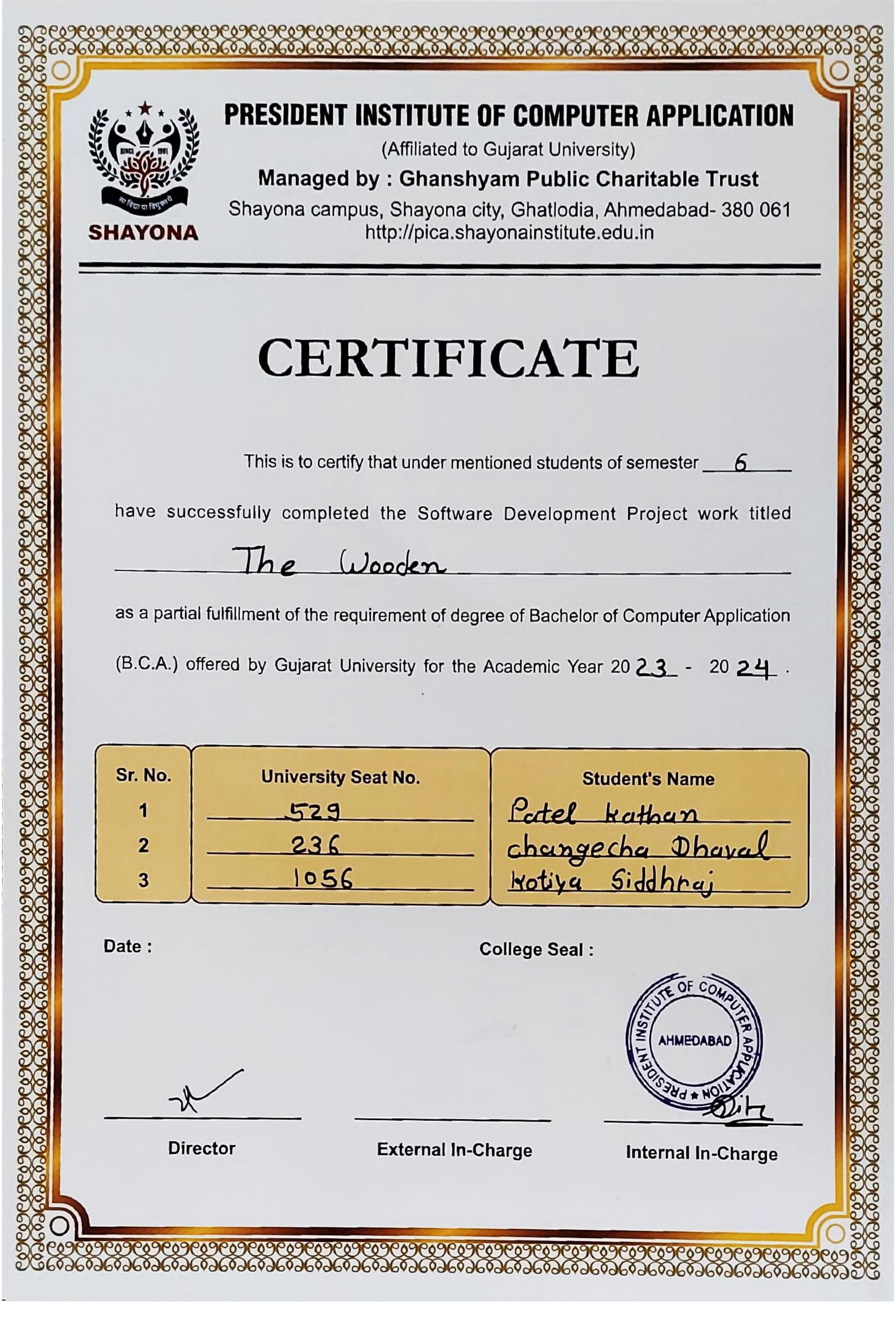
We wish them all success in their future endeavours.

Regards,

CREART SOLUTION
AHMEDABAD

CreArt Solutions

College Certificate



SHAYONA

PRESIDENT INSTITUTE OF COMPUTER APPLICATION

(Affiliated to Gujarat University)

Managed by : Ghanshyam Public Charitable Trust

Shayona campus, Shayona city, Ghatlodia, Ahmedabad- 380 061
http://pica.shayonainstitute.edu.in

CERTIFICATE

This is to certify that under mentioned students of semester 6

have successfully completed the Software Development Project work titled

The Wooden

as a partial fulfillment of the requirement of degree of Bachelor of Computer Application (B.C.A.) offered by Gujarat University for the Academic Year 2023 - 2024.

Sr. No.	University Seat No.	Student's Name
1	529	Patel kathan
2	236	changecha Dhaval
3	1056	Kotiya Siddhraj

Date :

College Seal :

PRESIDENT INSTITUTE OF COMPUTER APPLICATION ★ AHMEDABAD

Director

External In-Charge

Internal In-Charge

Acknowledgement

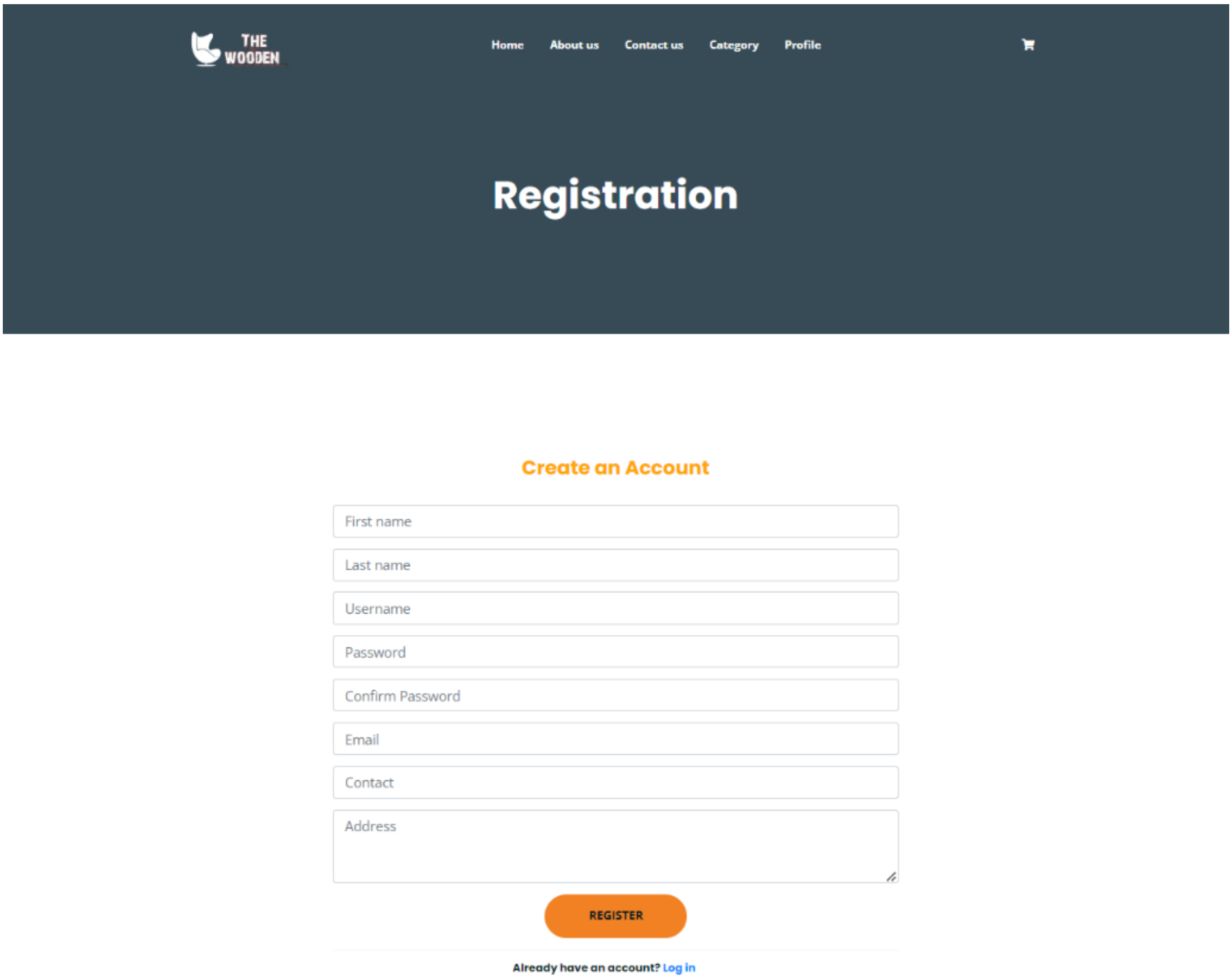
First of all, we would like to acknowledge and express our sincere appreciation for the opportunity given to us by "Alkesh Sir" and the CreArt Solutions administration for explore this project.

We are gratitude to our mentor, "Rita Mam", for her invaluable guidance and suggestions during our project on the "The Wooden". We are gratitude for the opportunity to research and learn new things, and express their deep appreciation for the mentor's invaluable assistance throughout the project.

Also, we would like to thank all our friends who supported us in one way or another.

<u>Customer Layout</u>

- ❑ Register

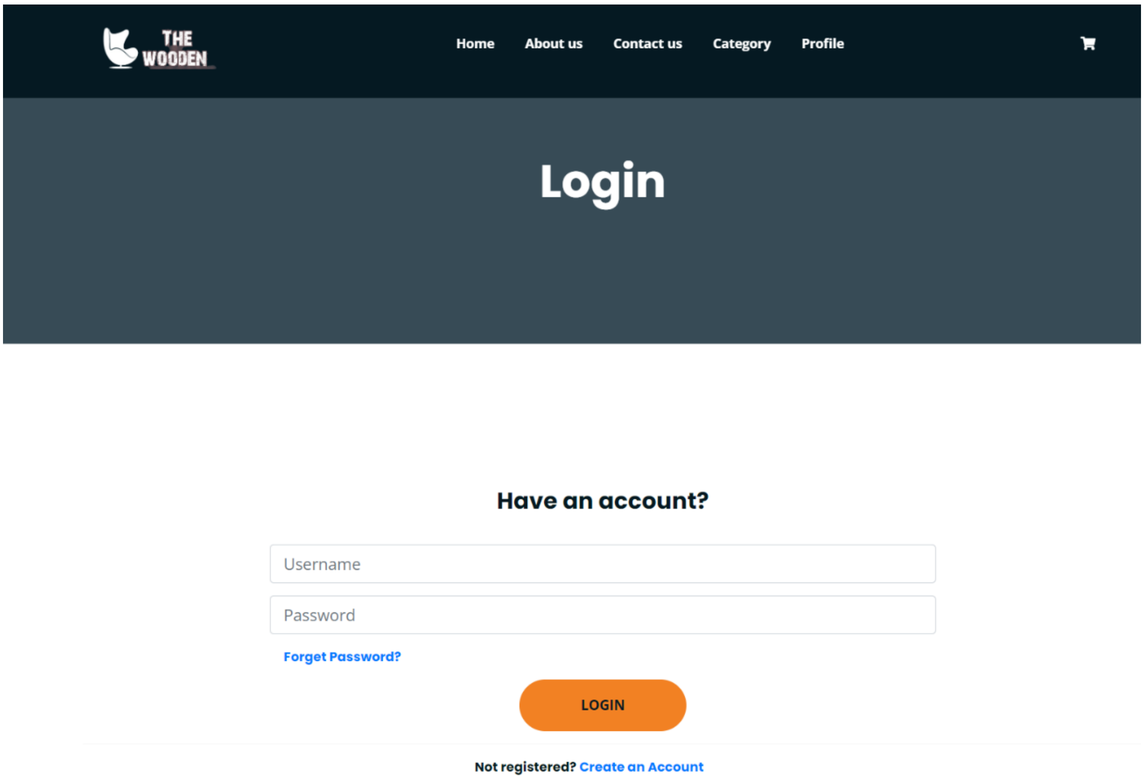


Registration

Create an Account

Already have an account? Log in

❑ Login



Login

Have an account?

Username

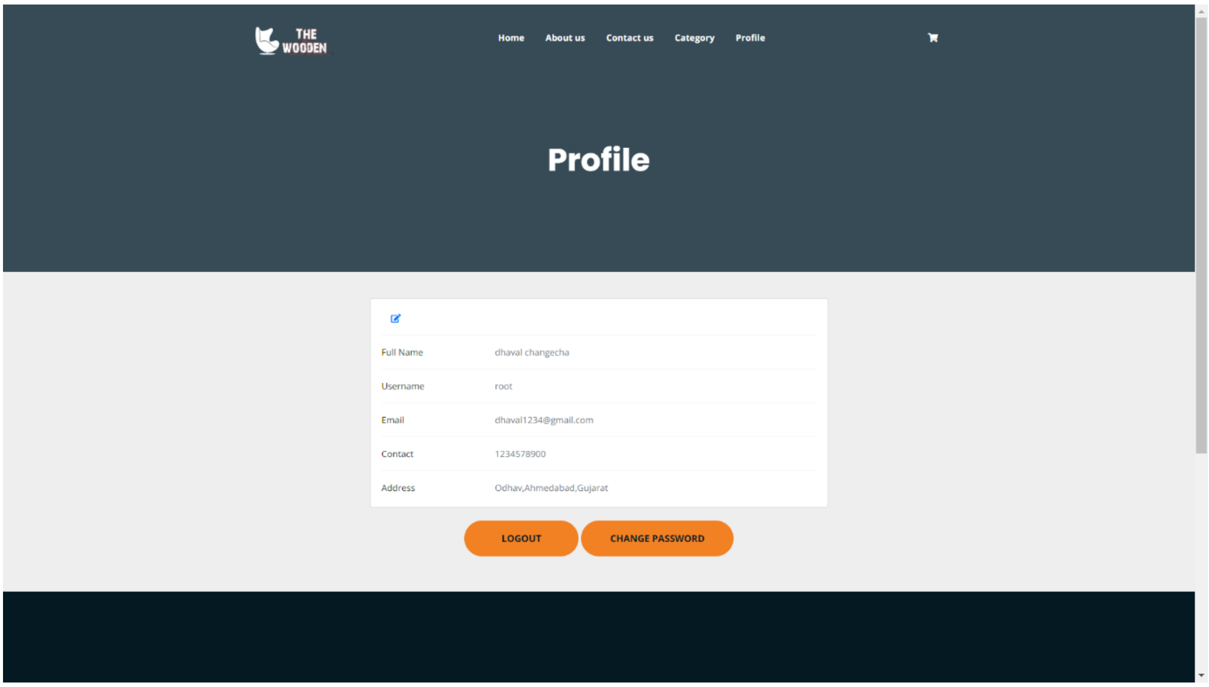
Password

Forget Password?

LOGIN

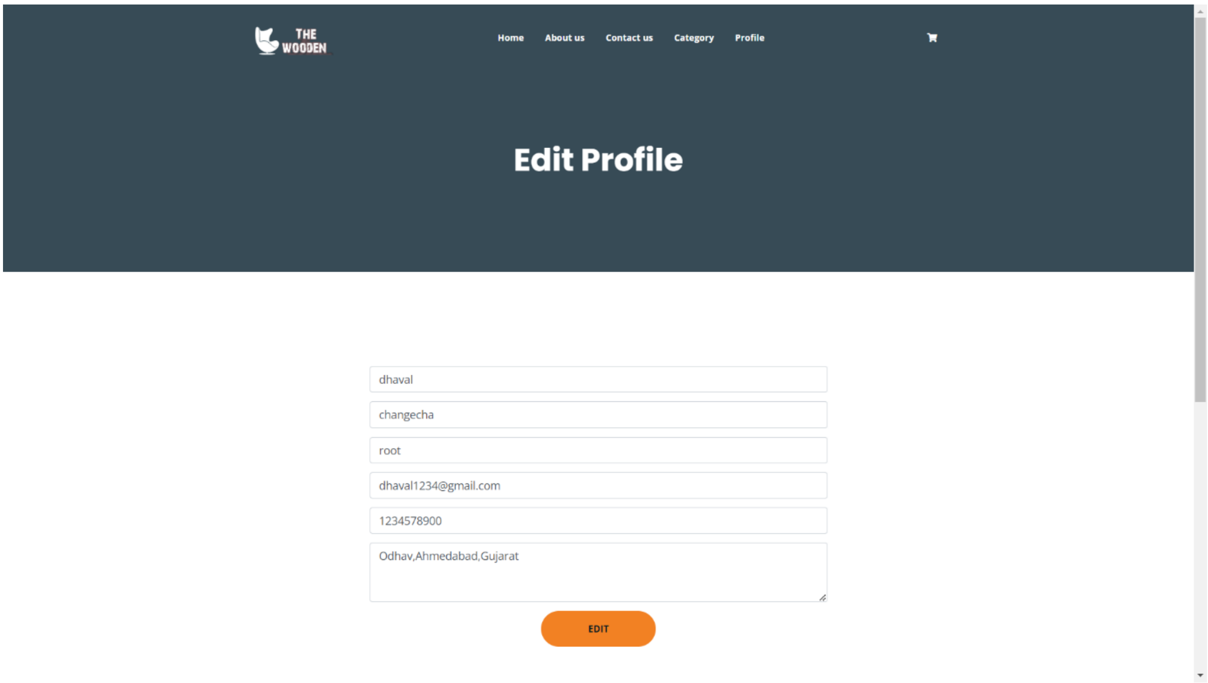
Not registered? Create an Account

- ❑ Profile



Profile

Full Name	dhaval changecha
Username	root
Email	dhaval1234@gmail.com
Contact	1234578900
Address	Odhav,Ahmedabad,Gujarat



Edit Profile

dhaval

changecha

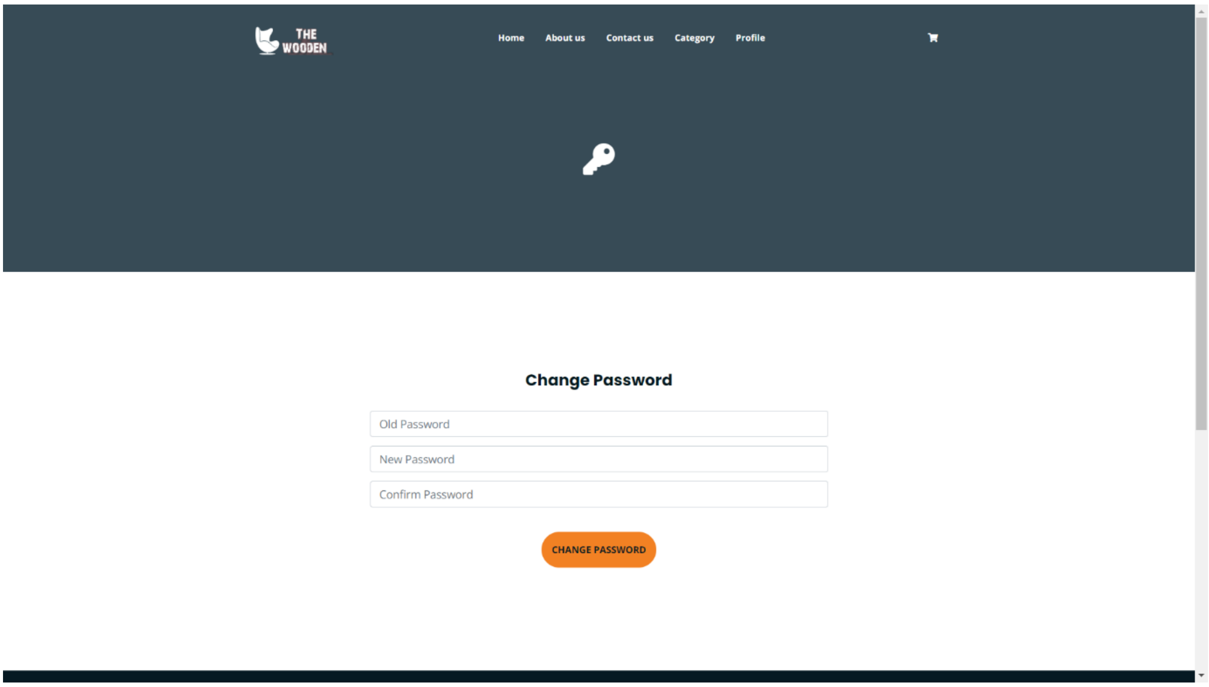
root

dhaval1234@gmail.com

1234578900

Odhav,Ahmedabad,Gujarat

- ❑ Change Password



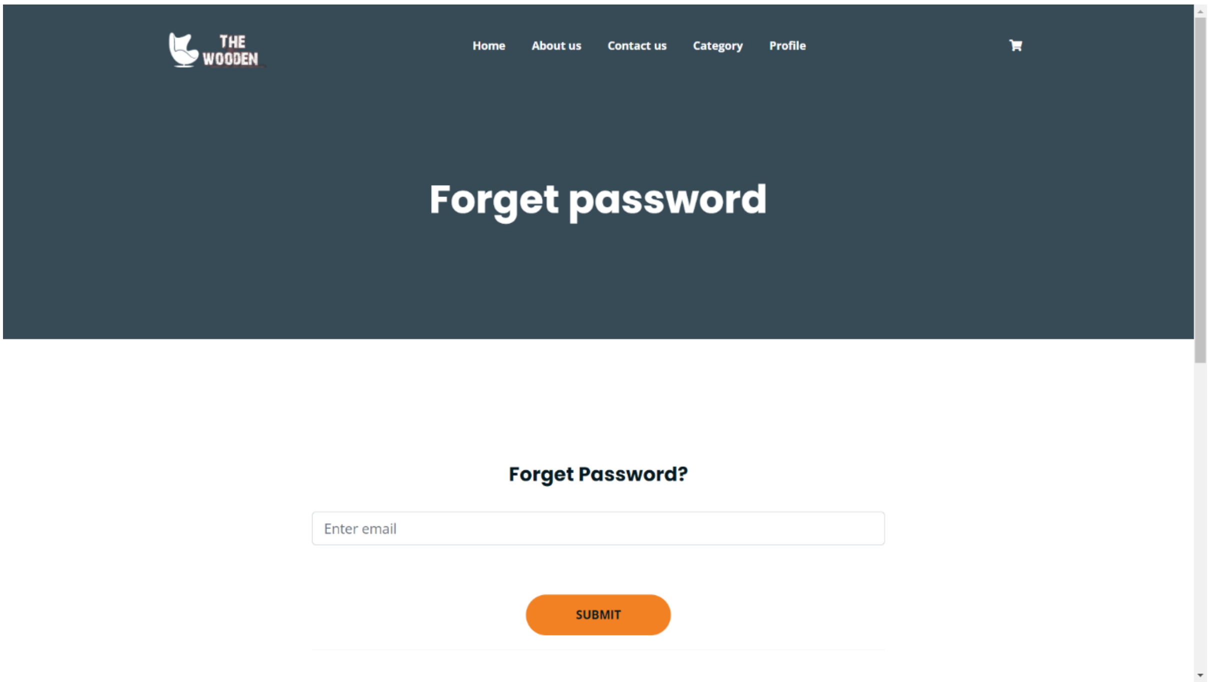
Change Password

Old Password

New Password

Confirm Password

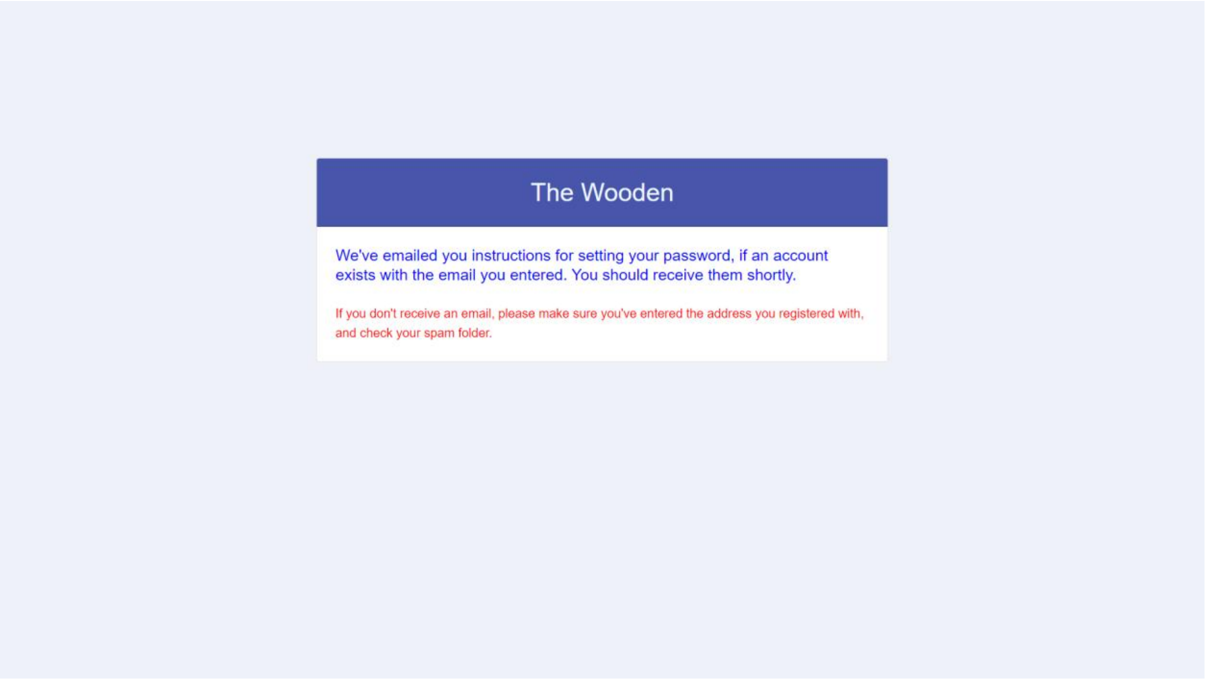
- ❑ Forget Password



Forget password

Forget Password?

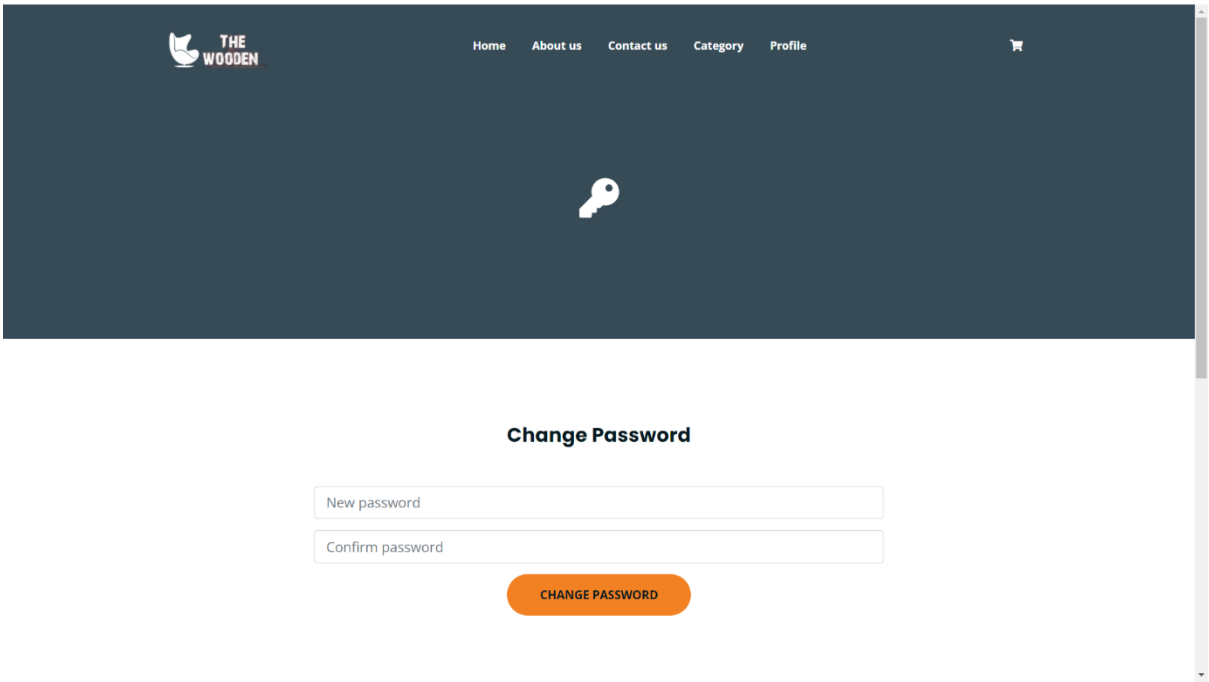
Enter email



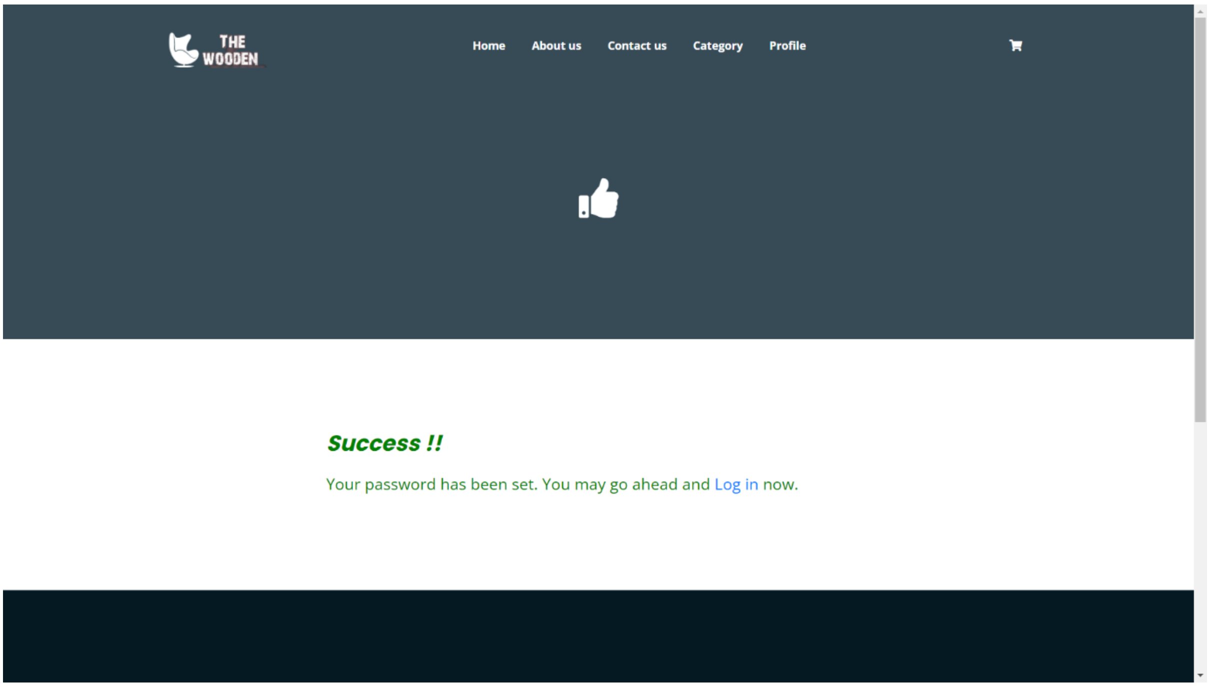
The Wooden

We've emailed you instructions for setting your password, if an account exists with the email you entered. You should receive them shortly.

If you don't receive an email, please make sure you've entered the address you registered with, and check your spam folder.



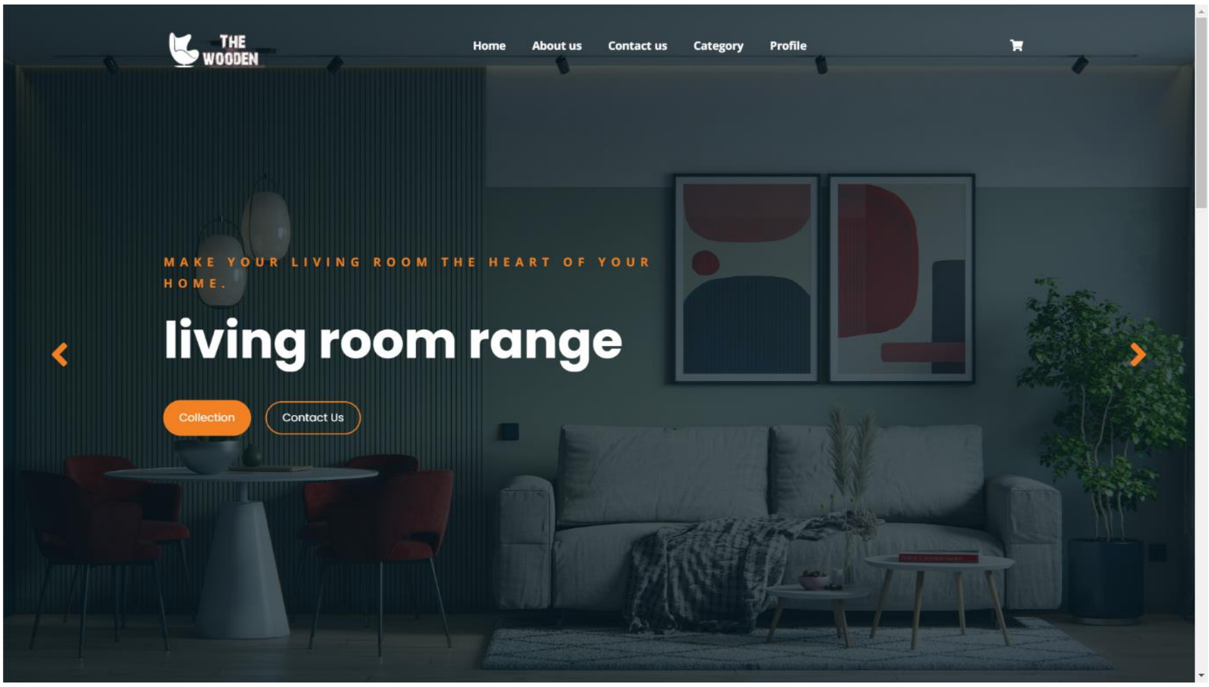
Change Password



Success !!

Your password has been set. You may go ahead and Log in now.

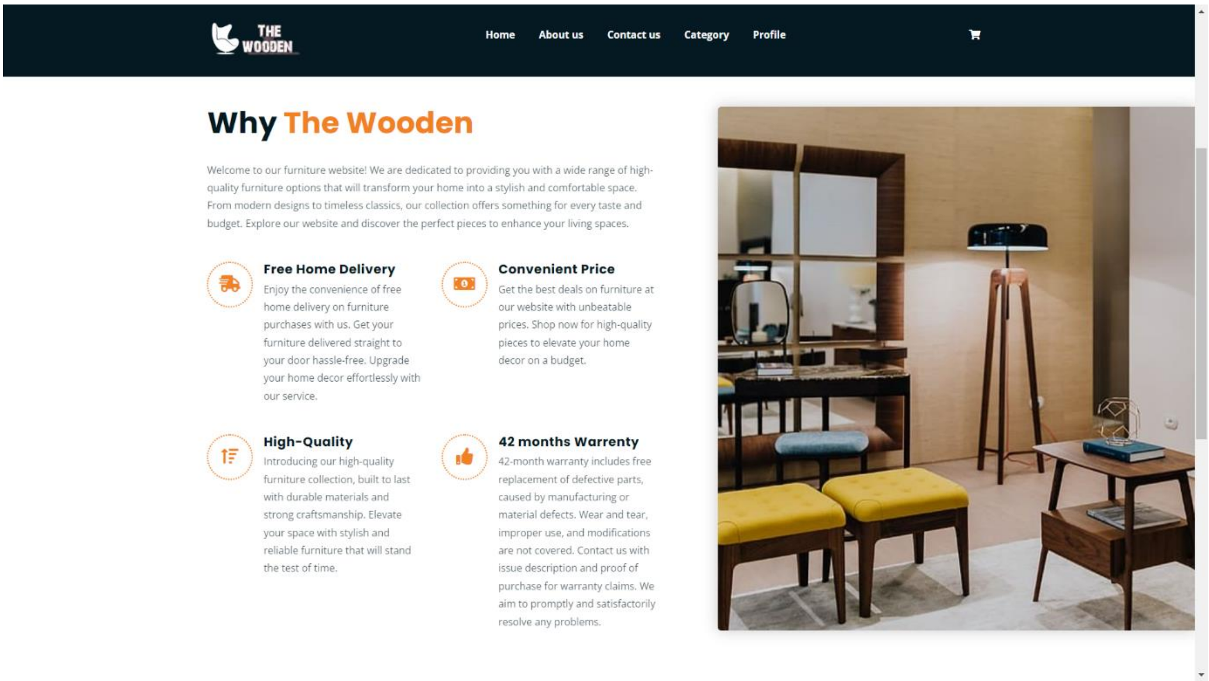
- ❑ Home Page



MAKE YOUR LIVING ROOM THE HEART OF YOUR HOME.

living room range

- ❑ About Us



Why The Wooden

Welcome to our furniture website! We are dedicated to providing you with a wide range of high-quality furniture options that will transform your home into a stylish and comfortable space. From modern designs to timeless classics, our collection offers something for every taste and budget. Explore our website and discover the perfect pieces to enhance your living spaces.

Free Home Delivery

Enjoy the convenience of free home delivery on furniture purchases with us. Get your furniture delivered straight to your door hassle-free. Upgrade your home decor effortlessly with our service.

Convenient Price

Get the best deals on furniture at our website with unbeatable prices. Shop now for high-quality pieces to elevate your home decor on a budget.

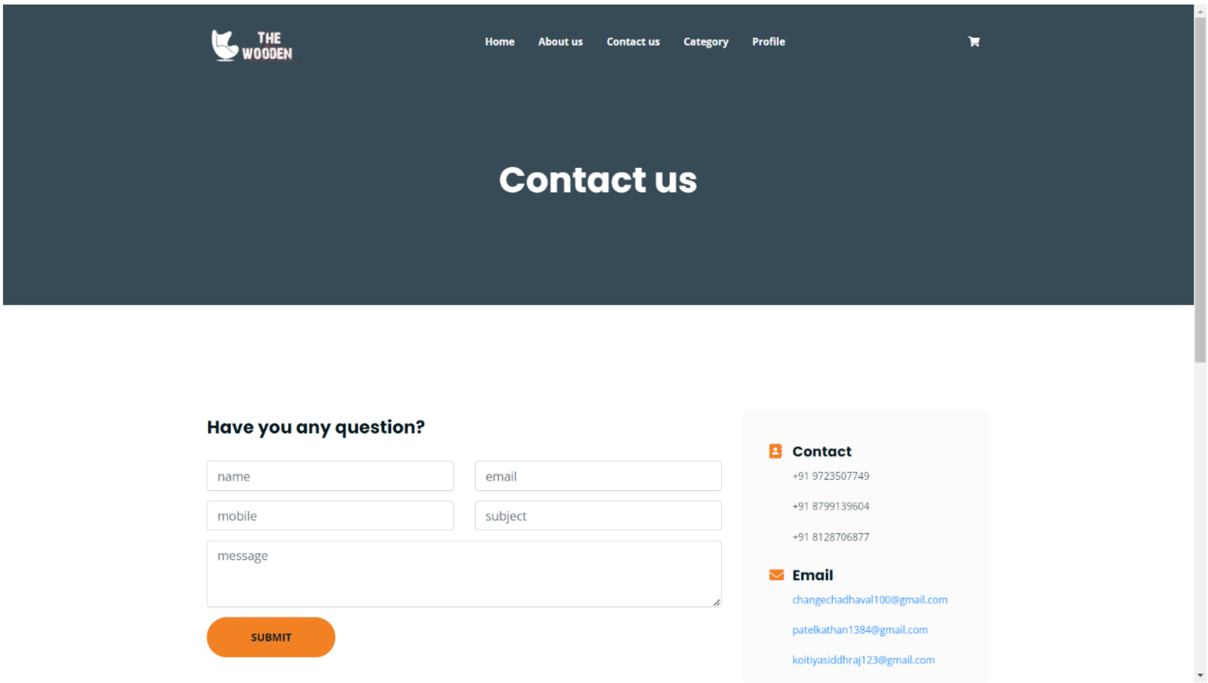
High-Quality

Introducing our high-quality furniture collection, built to last with durable materials and strong craftsmanship. Elevate your space with stylish and reliable furniture that will stand the test of time.

42 months Warrenty

42-month warranty includes free replacement of defective parts, caused by manufacturing or material defects. Wear and tear, improper use, and modifications are not covered. Contact us with issue description and proof of purchase for warranty claims. We aim to promptly and satisfactorily resolve any problems.

❑ Contact us



Contact us

Have you any question?

name

email

mobile

subject

message

SUBMIT

Contact

+91 9723507749

+91 8799139604

+91 8128706877

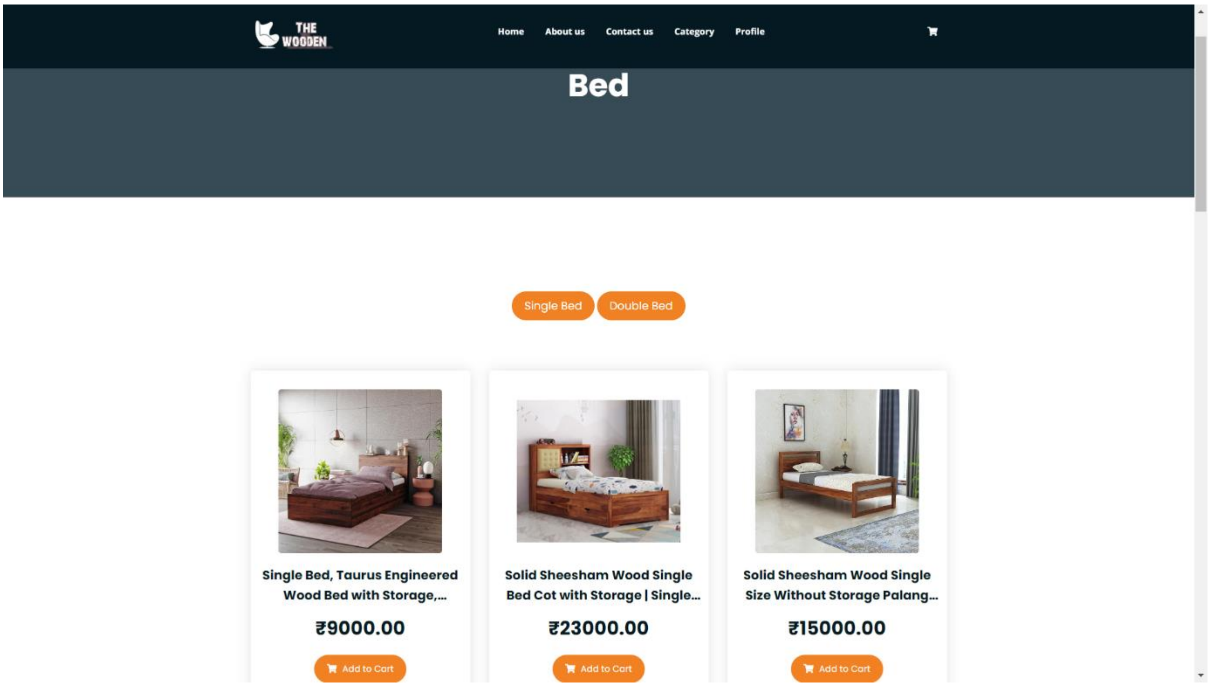
Email

changechadhaval100@gmail.com

patelkathan1384@gmail.com

koitiyasiddhraj123@gmail.com

❑ Category



Bed

Single Bed | Double Bed

Single Bed, Taurus Engineered Wood Bed with Storage,...

₹9000.00

Add to Cart

Solid Sheesham Wood Single Bed Cot with Storage | Single...

₹23000.00

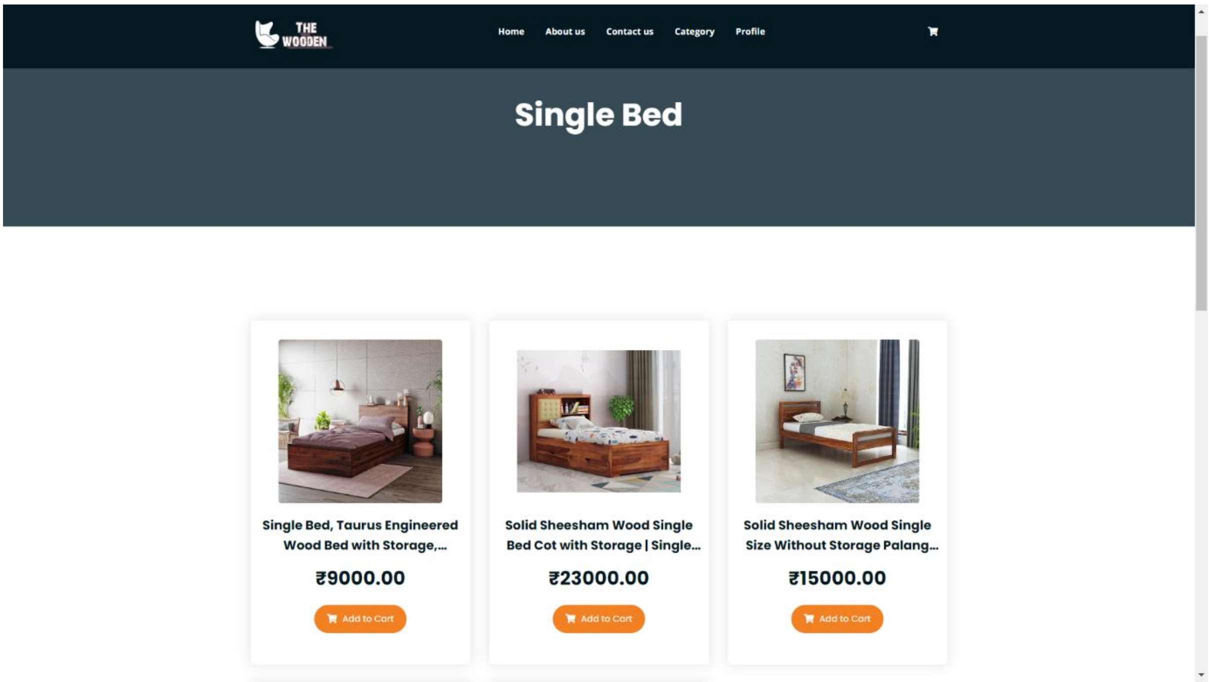
Add to Cart

Solid Sheesham Wood Single Size Without Storage Palang...

₹15000.00

Add to Cart

❑ Sub category



Single Bed

Single Bed, Taurus Engineered Wood Bed with Storage,...

₹9000.00

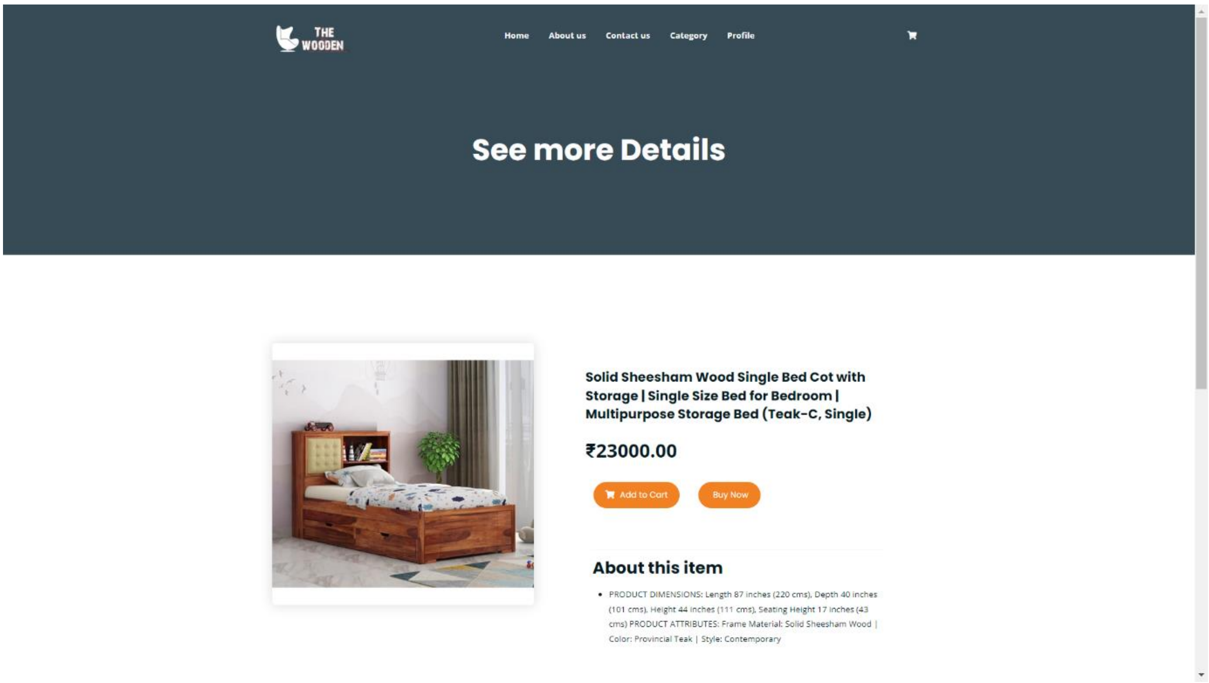
Solid Sheesham Wood Single Bed Cot with Storage | Single...

₹23000.00

Solid Sheesham Wood Single Size Without Storage Palang...

₹15000.00

❑ Single Product



See more Details

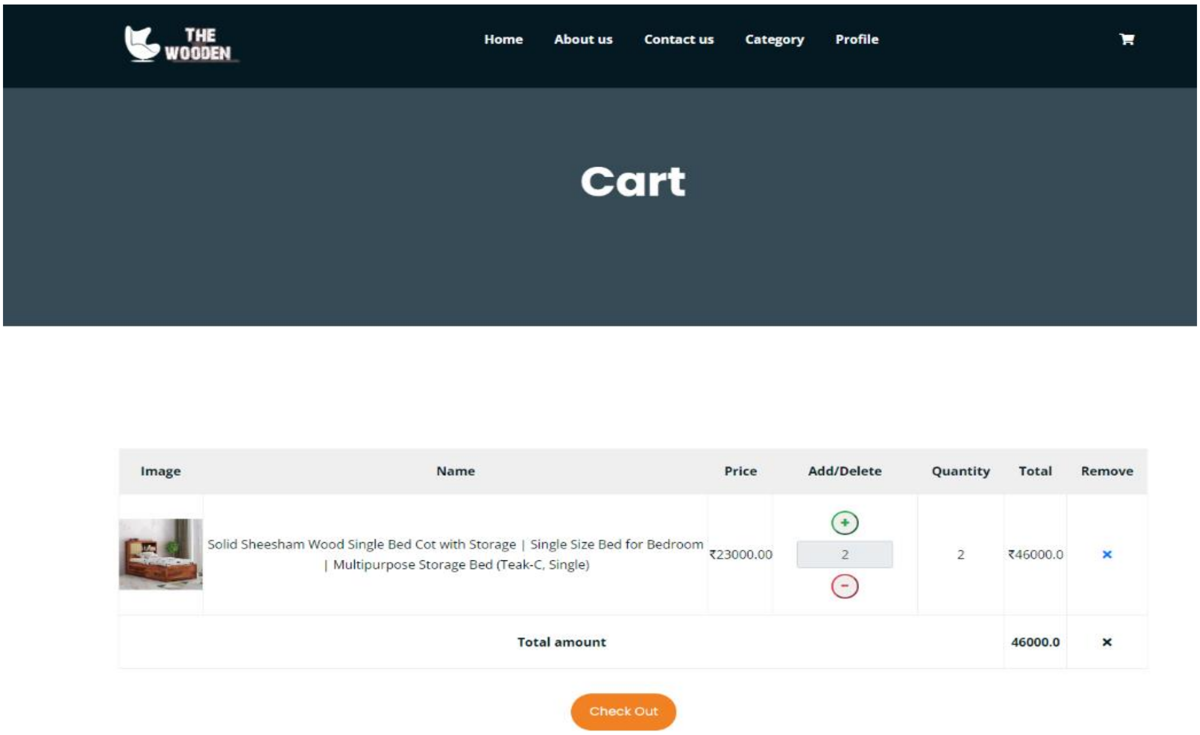
Solid Sheesham Wood Single Bed Cot with Storage | Single Size Bed for Bedroom | Multipurpose Storage Bed (Teak-C, Single)

₹23000.00

About this item

- PRODUCT DIMENSIONS: Length 87 inches (220 cms), Depth 40 inches (101 cms), Height 44 inches (111 cms), Seating Height 17 inches (43 cms) PRODUCT ATTRIBUTES: Frame Material: Solid Sheesham Wood | Color: Provincial Teak | Style: Contemporary

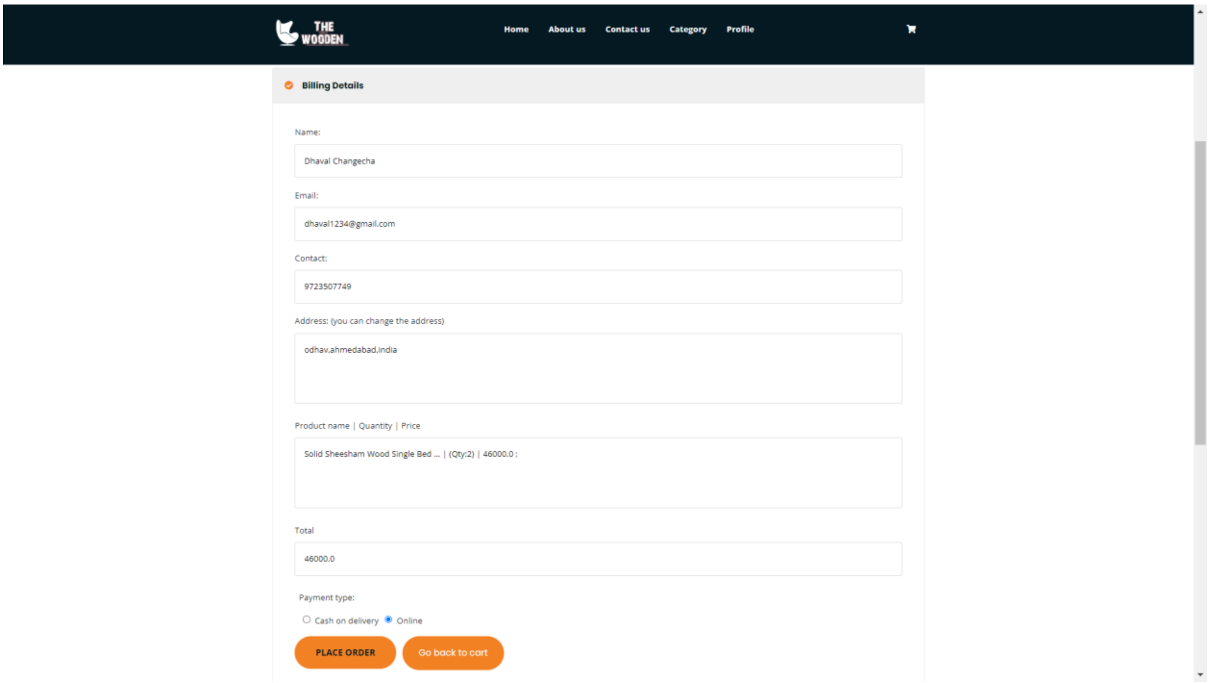
❑ Cart



Cart

Image	Name	Price	Add/Delete	Quantity	Total	Remove
	Solid Sheesham Wood Single Bed Cot with Storage | Single Size Bed for Bedroom | Multipurpose Storage Bed (Teak-C, Single)	₹23000.00	2	2	₹46000.0	×
	Total amount				46000.0	×

❑ Order



Billing Details

Name:
Dhaval Changecha

Email:
dhaval1234@gmail.com

Contact:
9723507749

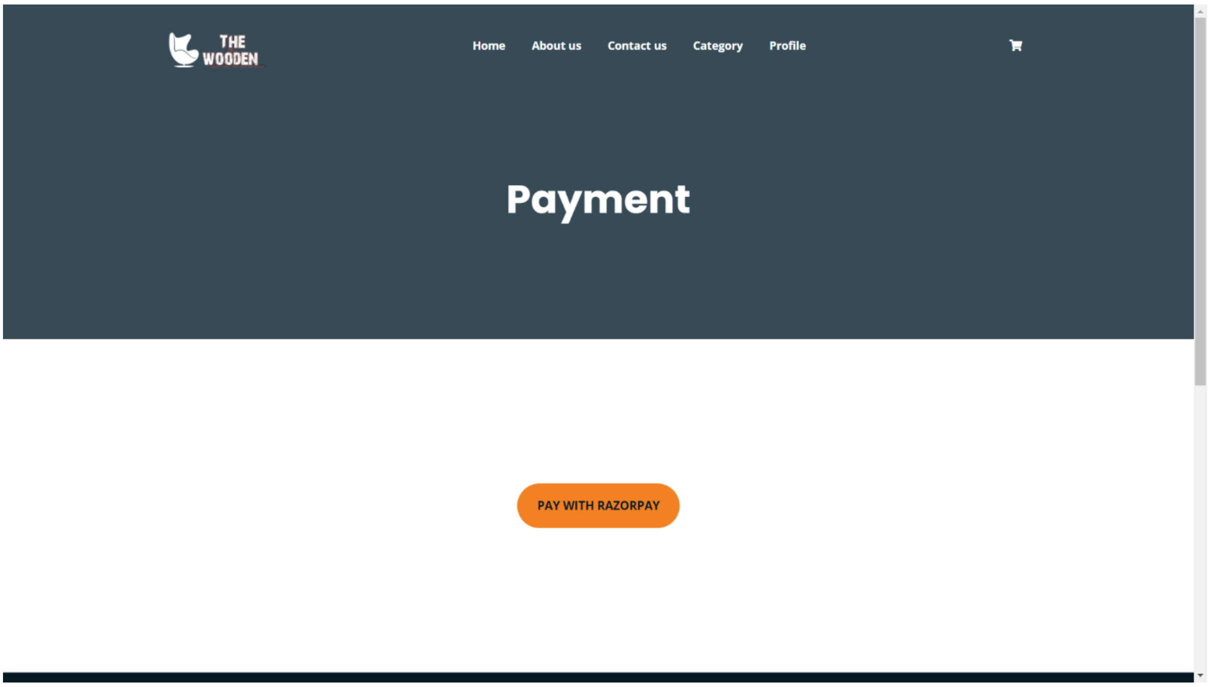
Address: (you can change the address)
odhav,ahmedabad,india

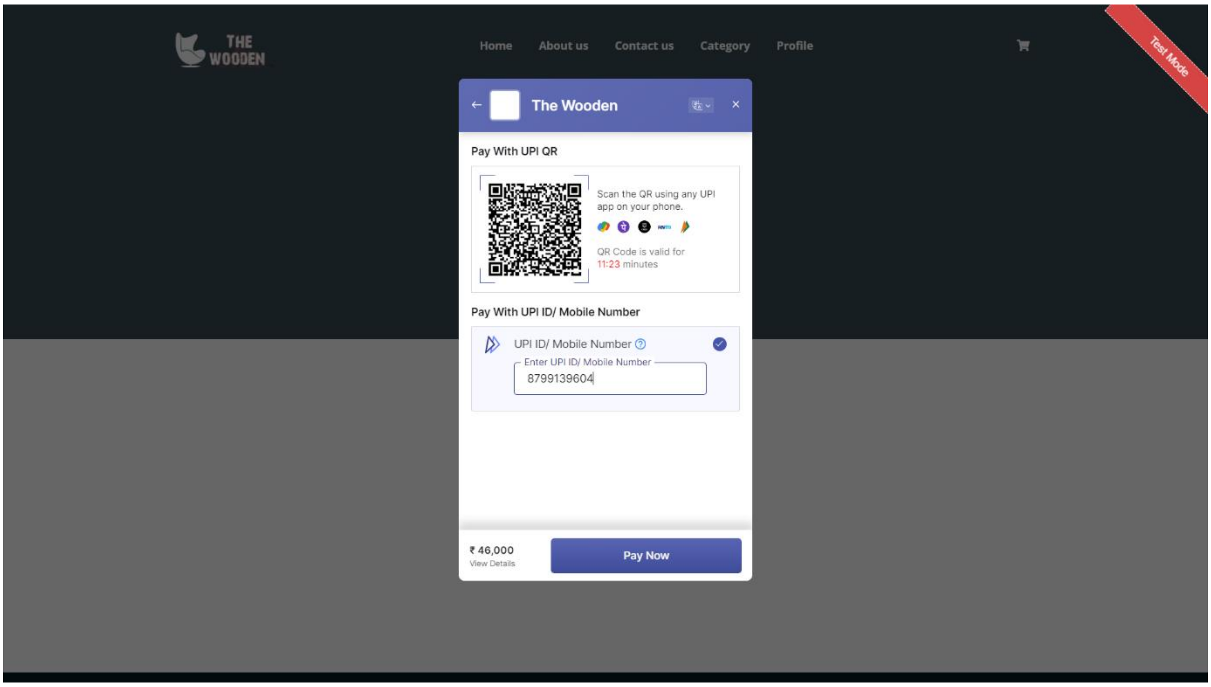
Product name | Quantity | Price
Solid Sheesham Wood Single Bed ... | (Qty:2) | 46000.0 ;

Total
46000.0

Payment type:
○ Cash on delivery ● Online

□ Payment





Pay With UPI QR

Scan the QR using any UPI app on your phone.

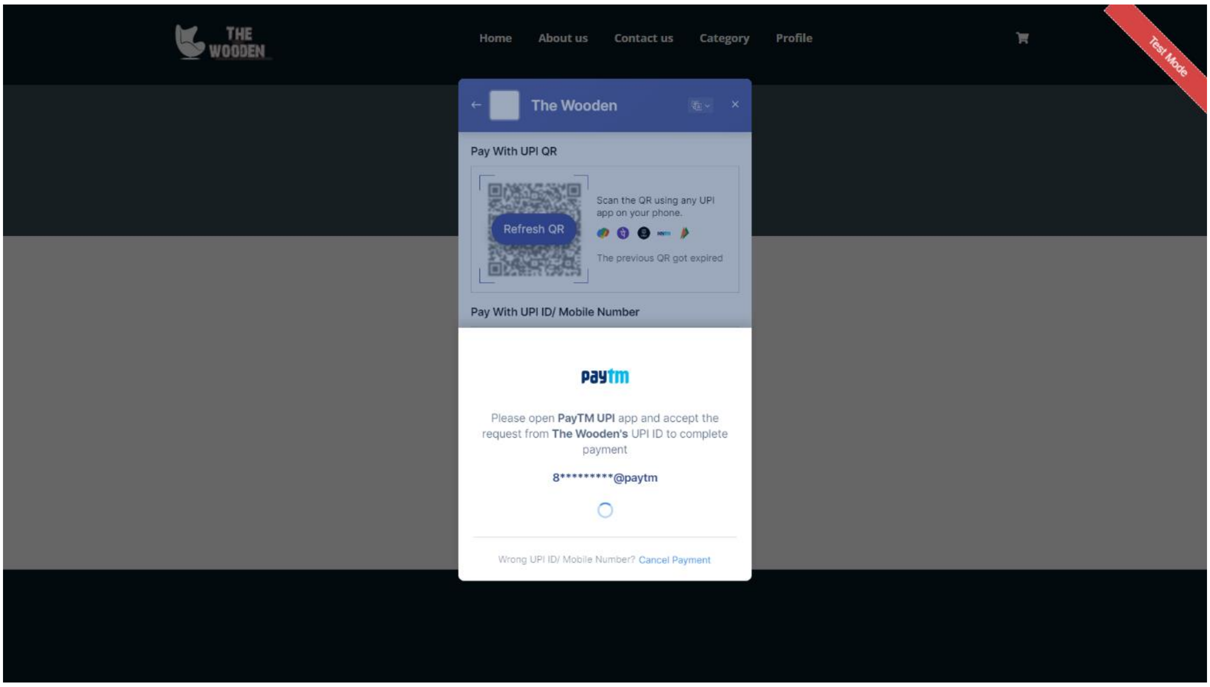
QR Code is valid for 11:23 minutes

Pay With UPI ID/ Mobile Number

UPI ID/ Mobile Number

Enter UPI ID/ Mobile Number: 8799139604

₹ 46,000
View Details



The Wooden

Pay With UPI QR

Scan the QR using any UPI app on your phone.

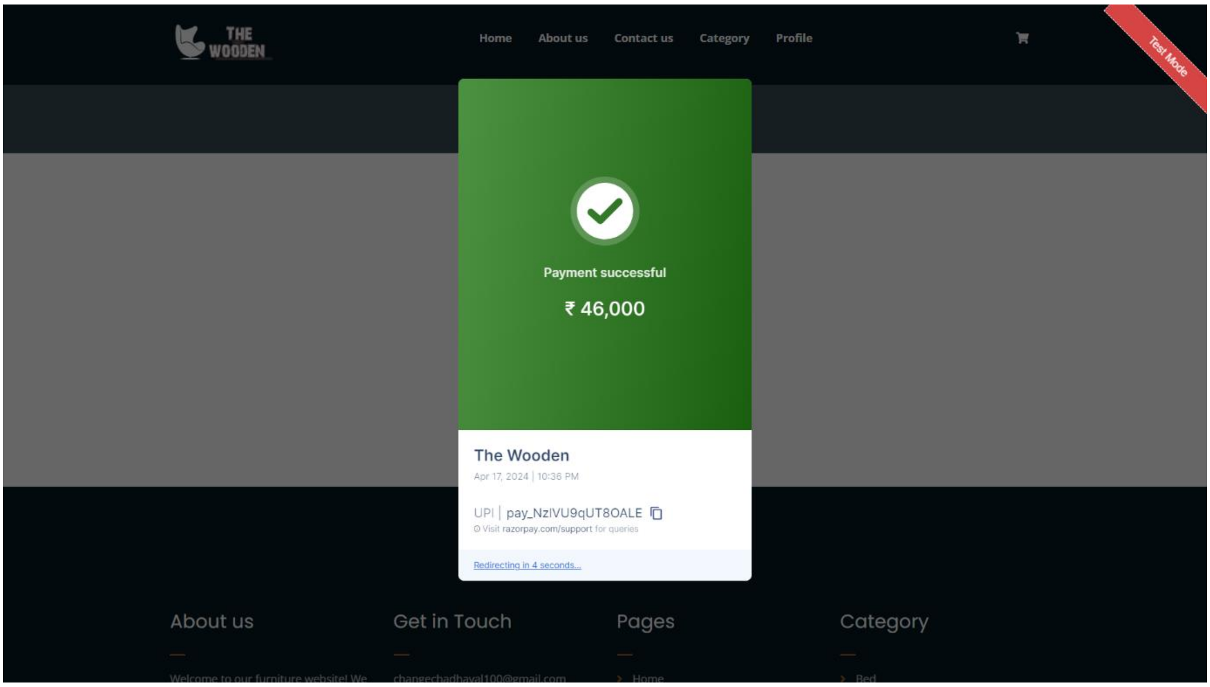
The previous QR got expired

Pay With UPI ID/ Mobile Number

Please open **PayTM UPI** app and accept the request from **The Wooden's** UPI ID to complete payment

8*******@paytm**

Wrong UPI ID/ Mobile Number? Cancel Payment



Payment successful

₹ 46,000

The Wooden

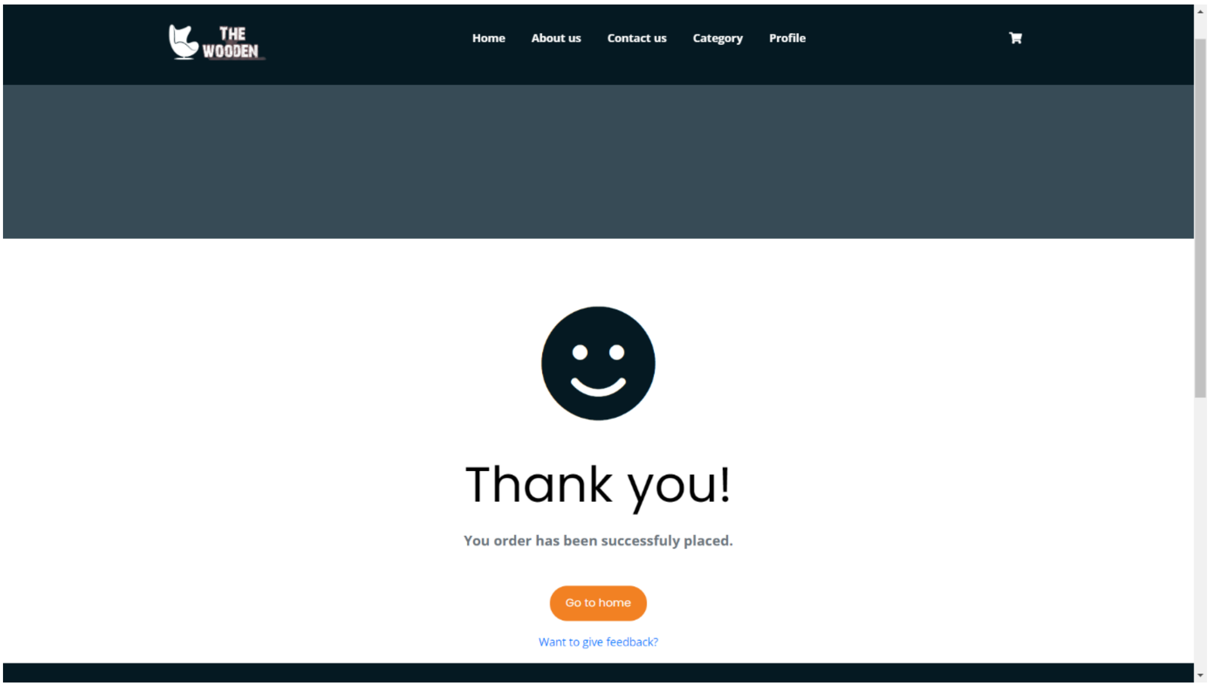
Apr 17, 2024 | 10:36 PM

UPI | pay_NzIVU9qUT8OALE

Visit razorpay.com/support for queries

Redirecting in 4 seconds...

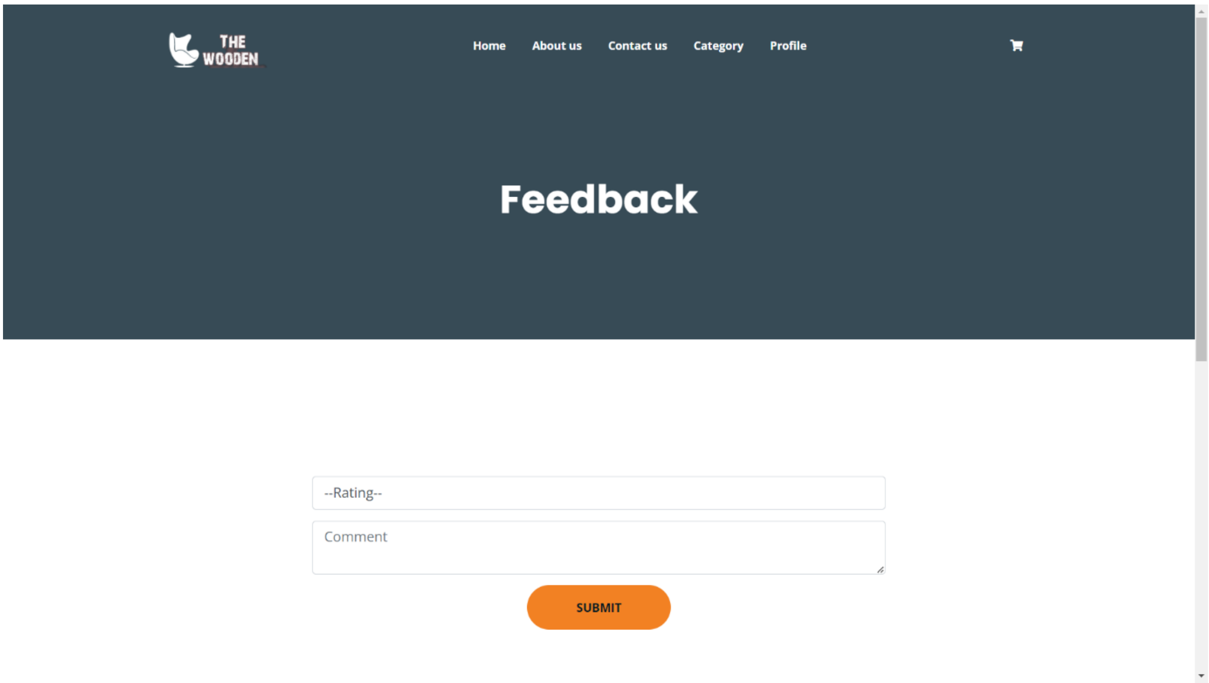
About us
Get in Touch
Pages
Category
Welcome to our furniture website! We
changechadhaval100@gmail.com
Home
Bed



Thank you!

You order has been successfuly placed.

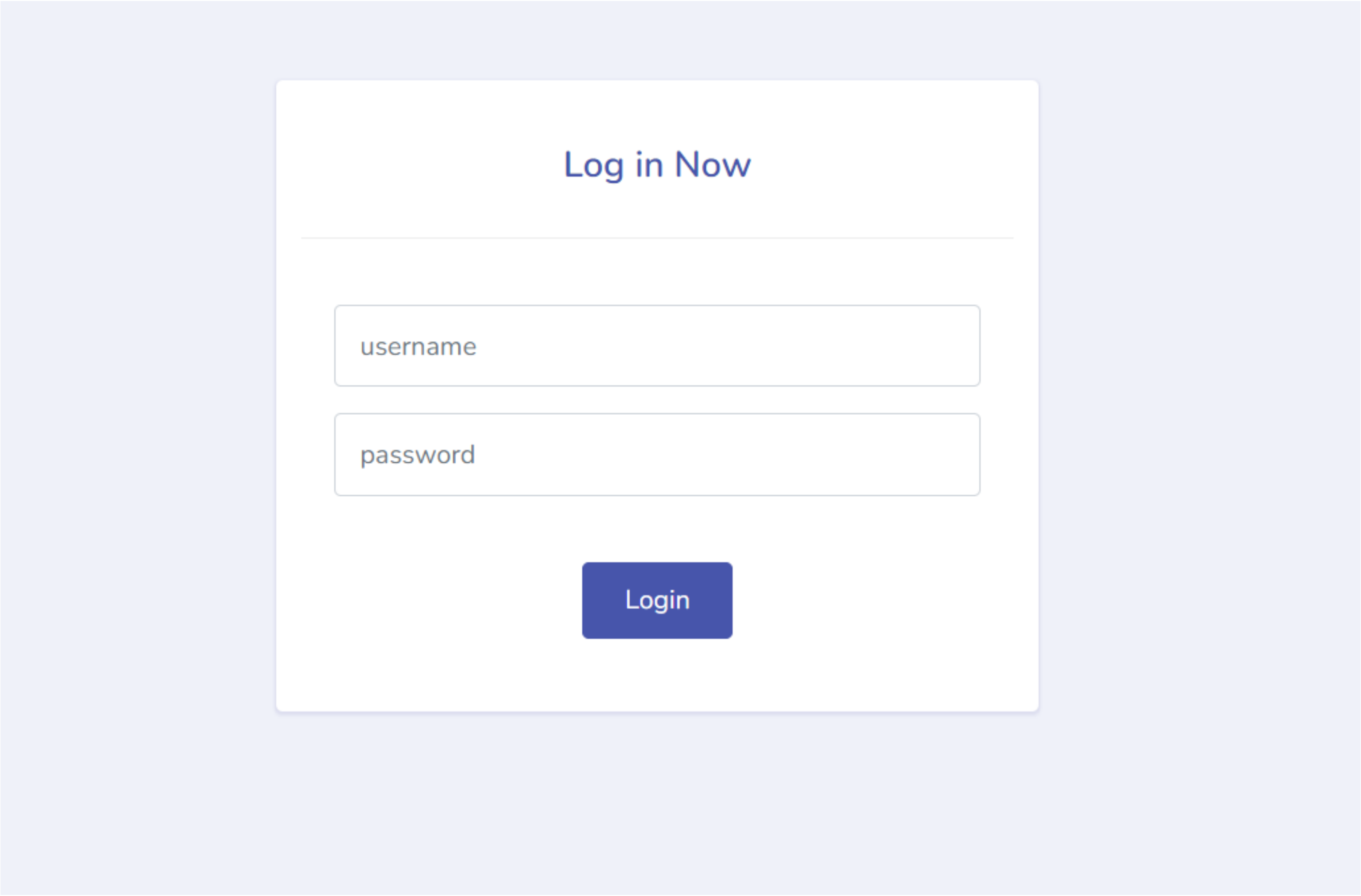
- ❑ Feedback



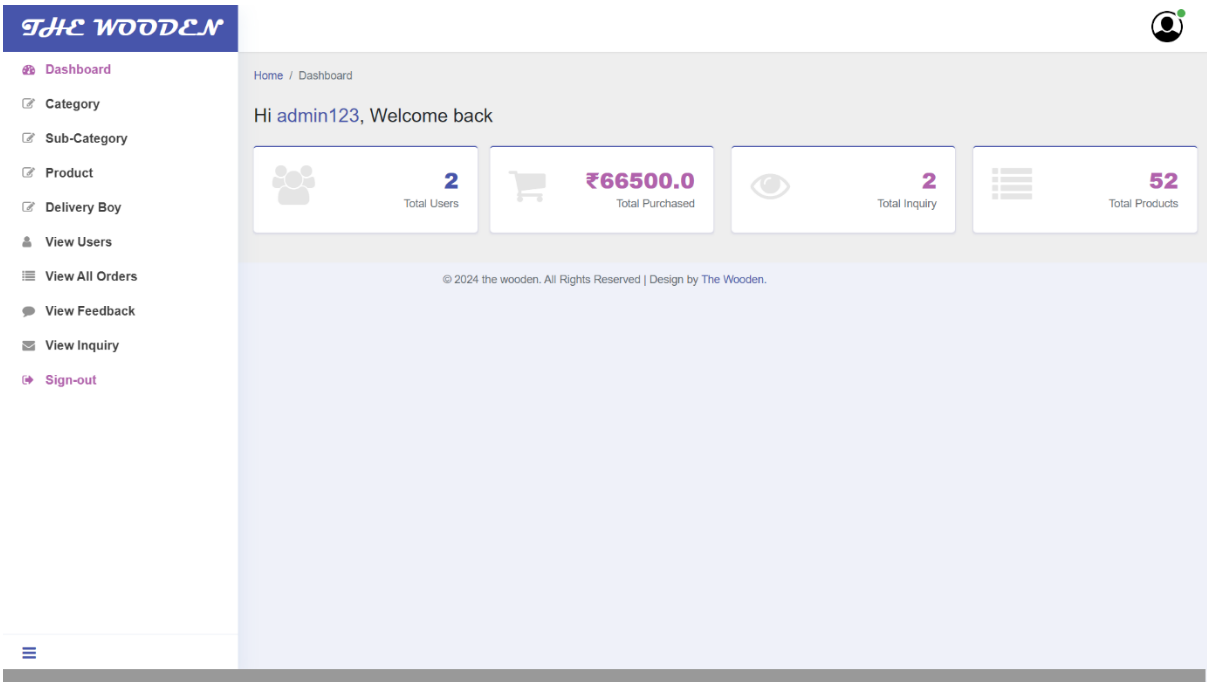
Feedback

Admin Layout

❑ Log in



- ❑ Dashboard

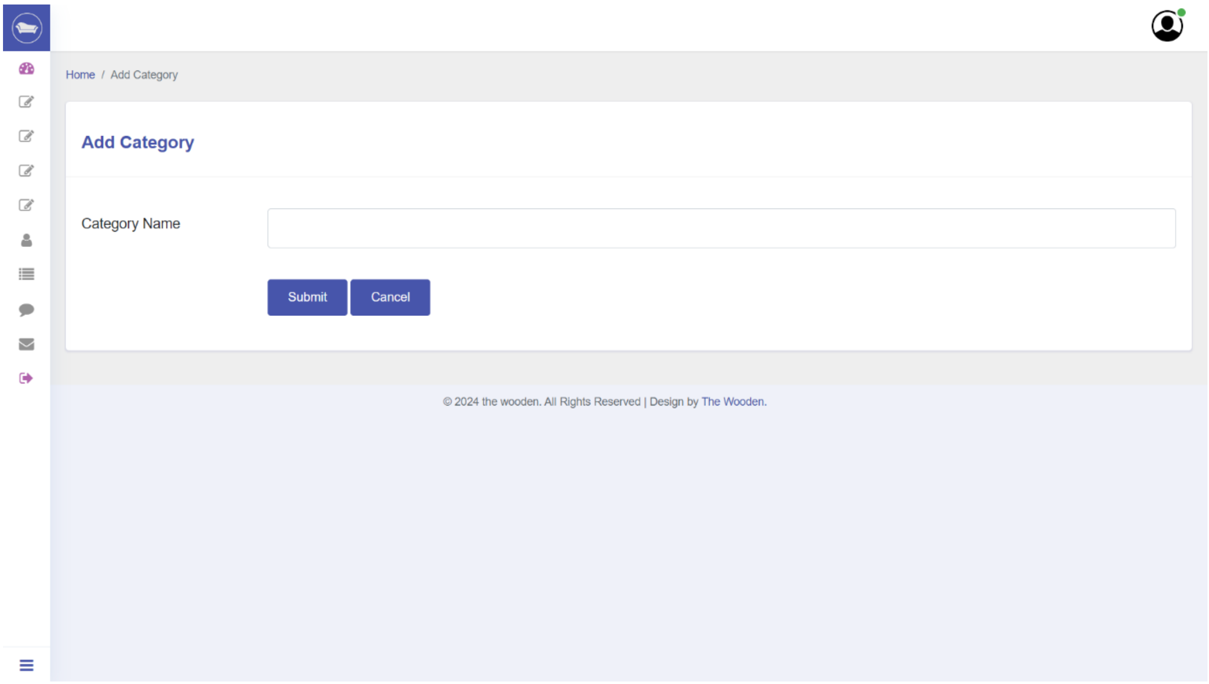


Hi admin123, Welcome back

2	₹66500.0	2	52
Total Users	Total Purchased	Total Inquiry	Total Products

© 2024 the wooden. All Rights Reserved | Design by The Wooden.

- ❑ Manage Category

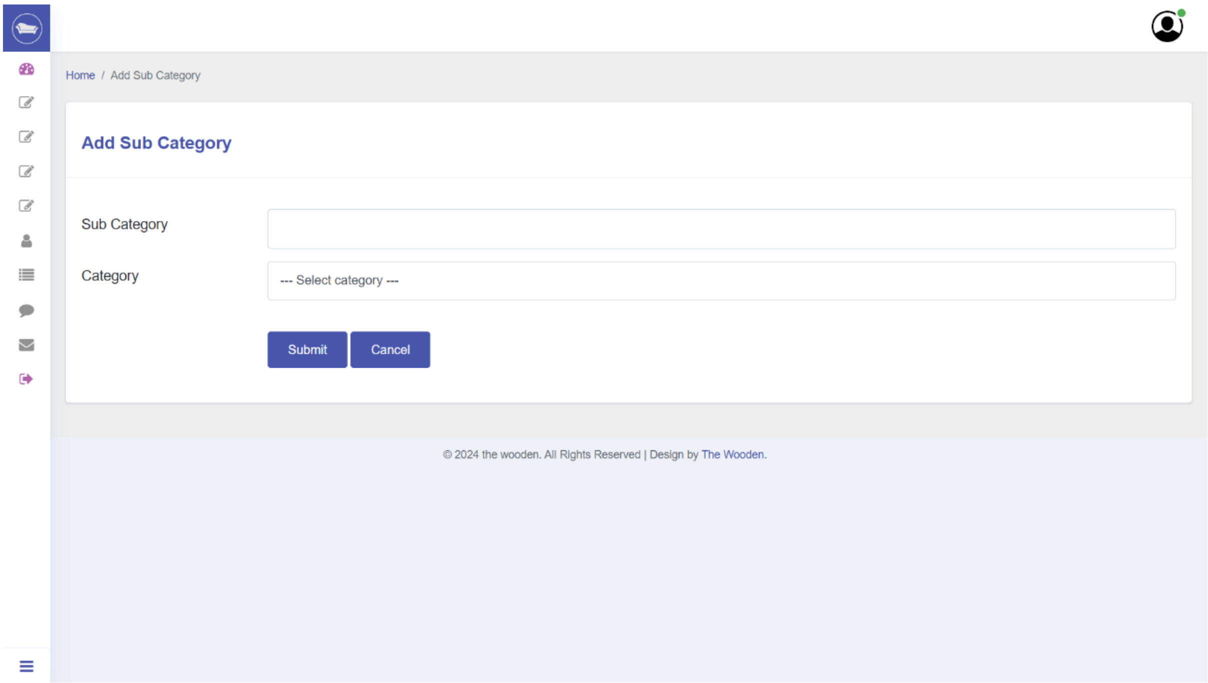


Add Category

Category Name

© 2024 the wooden. All Rights Reserved | Design by The Wooden.

❑ Manage Sub Category



Home / Add Sub Category

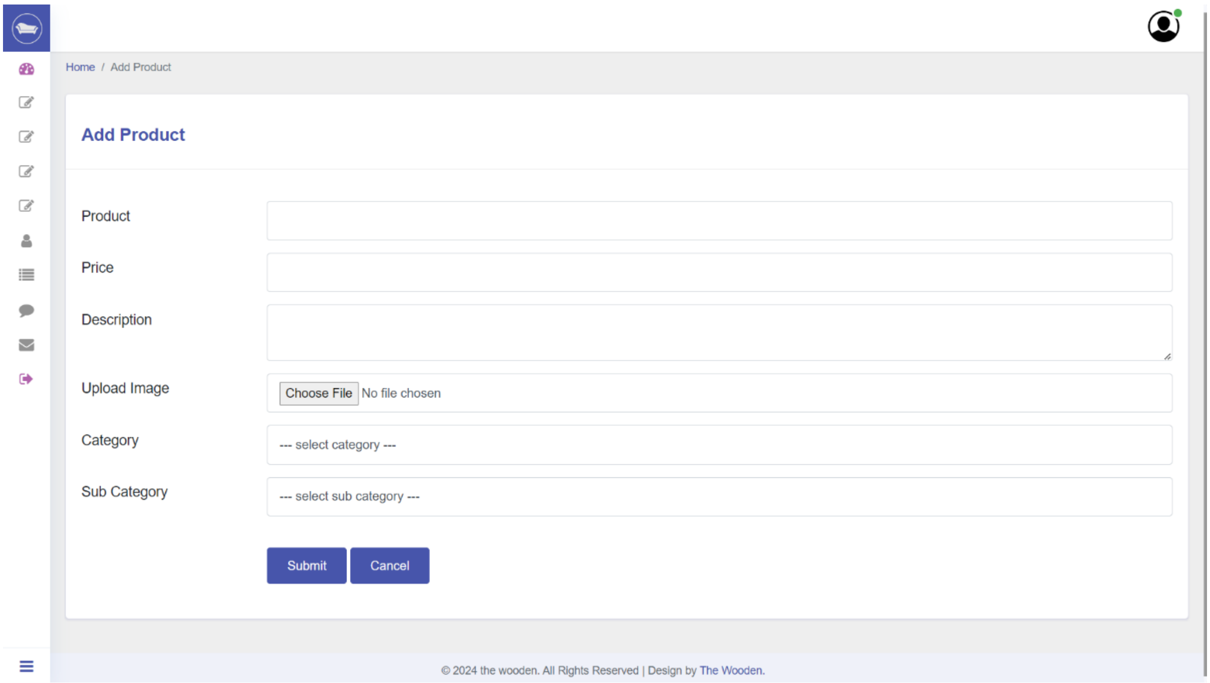
Add Sub Category

Sub Category	
Category	--- Select category ---

Submit Cancel

© 2024 the wooden. All Rights Reserved | Design by The Wooden.

❑ Manage Product



Home / Add Product

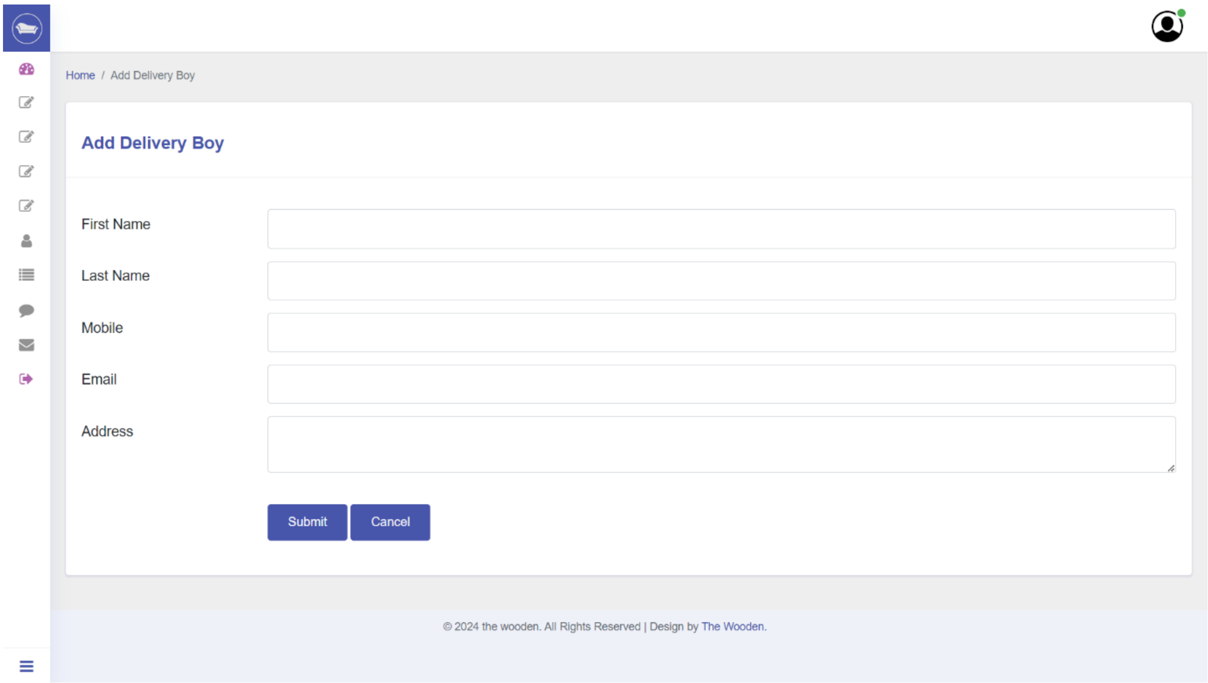
Add Product

Product	
Price	
Description	
Upload Image	Choose File No file chosen
Category	--- select category ---
Sub Category	--- select sub category ---

Submit Cancel

© 2024 the wooden. All Rights Reserved | Design by The Wooden.

❑ Manage Delivery Boy

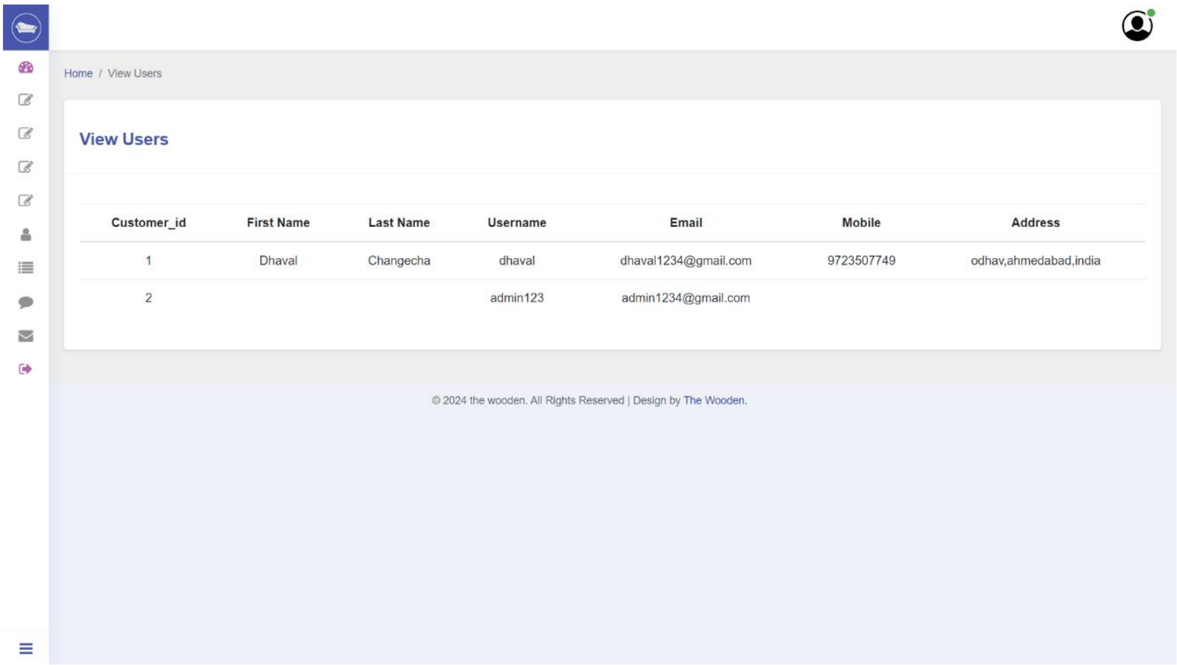


Add Delivery Boy

First Name	
Last Name	
Mobile	
Email	
Address	

© 2024 the wooden. All Rights Reserved | Design by The Wooden.

❑ Users

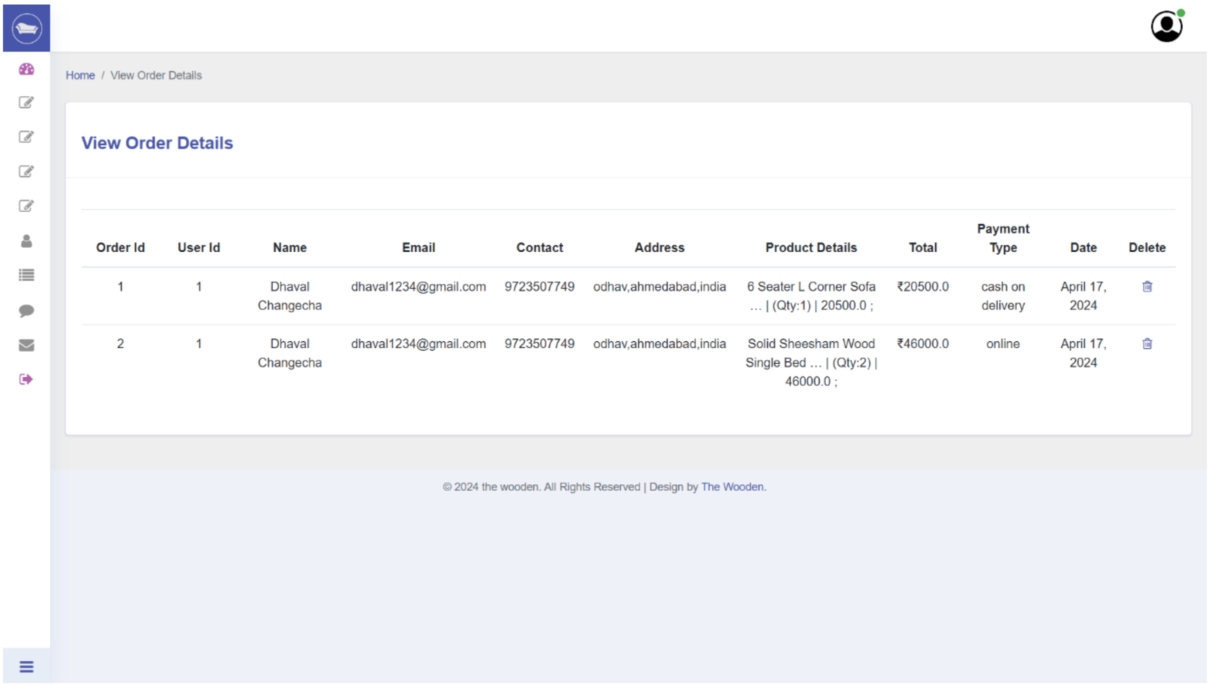


View Users

Customer_id	First Name	Last Name	Username	Email	Mobile	Address
1	Dhaval	Changecha	dhaval	dhaval1234@gmail.com	9723507749	odhav,ahmedabad,india
2			admin123	admin1234@gmail.com		

© 2024 the wooden. All Rights Reserved | Design by The Wooden.

❑ Manage Orders



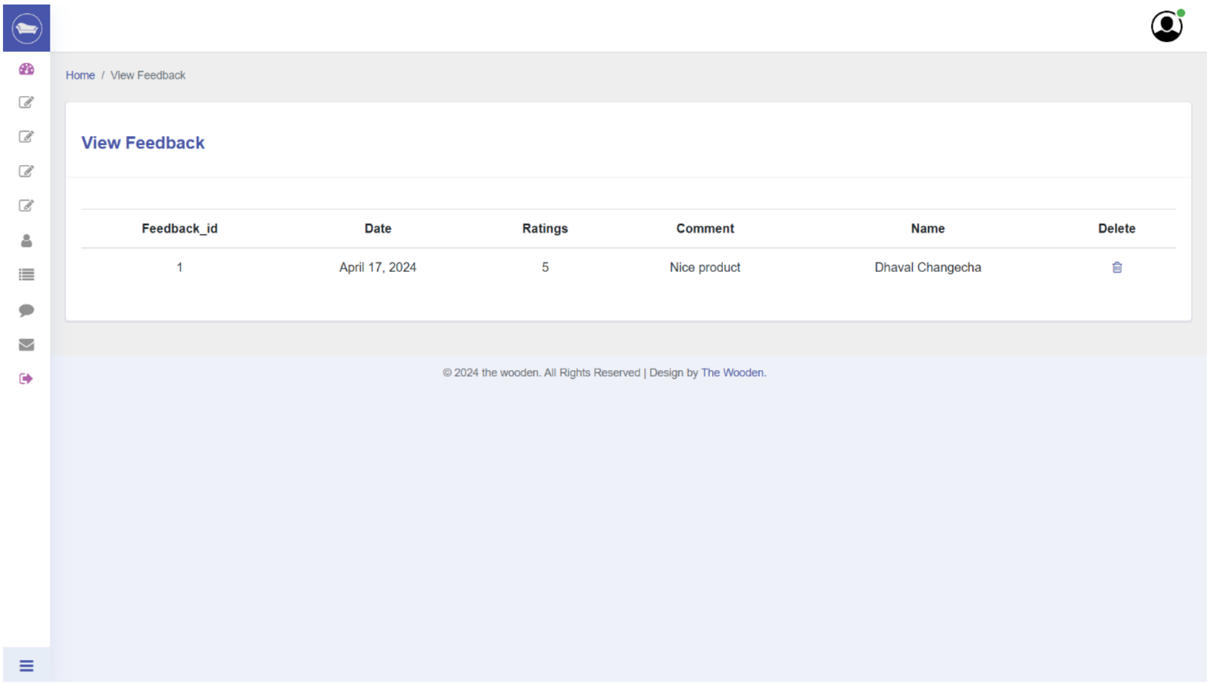
Home / View Order Details

View Order Details

Order Id	User Id	Name	Email	Contact	Address	Product Details	Total	Payment Type	Date	Delete
1	1	Dhaval Changecha	dhaval1234@gmail.com	9723507749	odhav,ahmedabad,india	6 Seater L Corner Sofa … | (Qty:1) | 20500.0 ;	₹20500.0	cash on delivery	April 17, 2024	🗑
2	1	Dhaval Changecha	dhaval1234@gmail.com	9723507749	odhav,ahmedabad,india	Solid Sheesham Wood Single Bed … | (Qty:2) | 46000.0 ;	₹46000.0	online	April 17, 2024	🗑

© 2024 the wooden. All Rights Reserved | Design by The Wooden.

❑ Manage Feedback



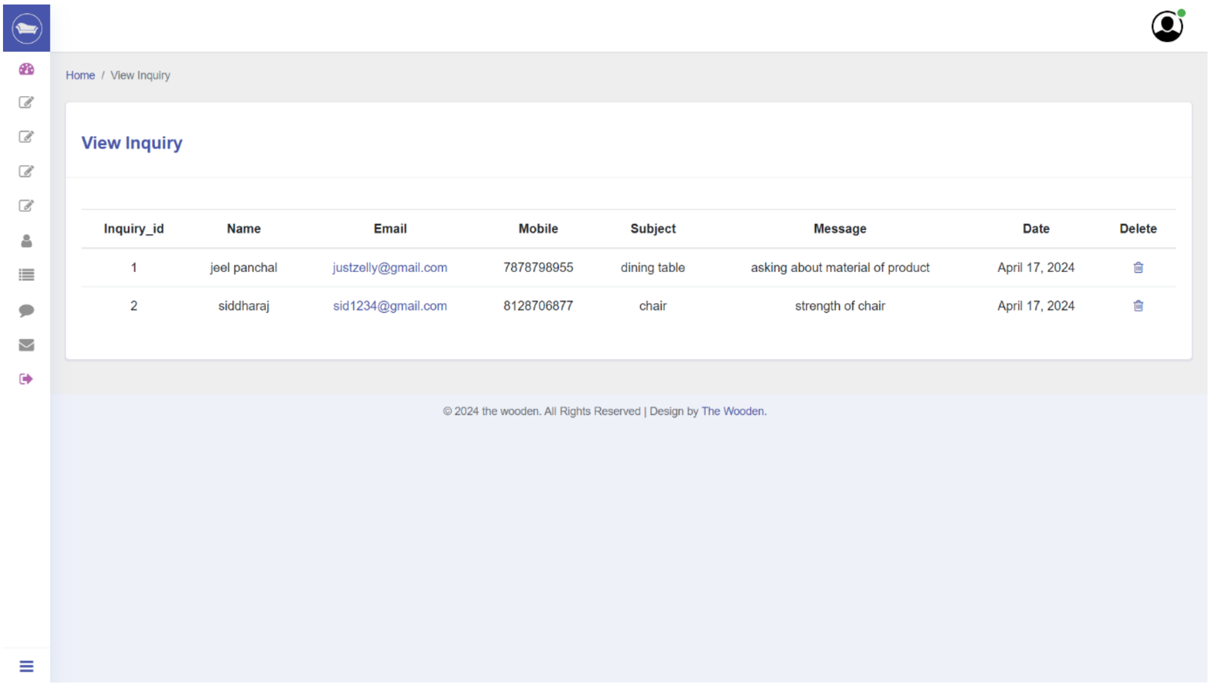
Home / View Feedback

View Feedback

Feedback_id	Date	Ratings	Comment	Name	Delete
1	April 17, 2024	5	Nice product	Dhaval Changecha	🗑

© 2024 the wooden. All Rights Reserved | Design by The Wooden.

❑ Manage Inquiry



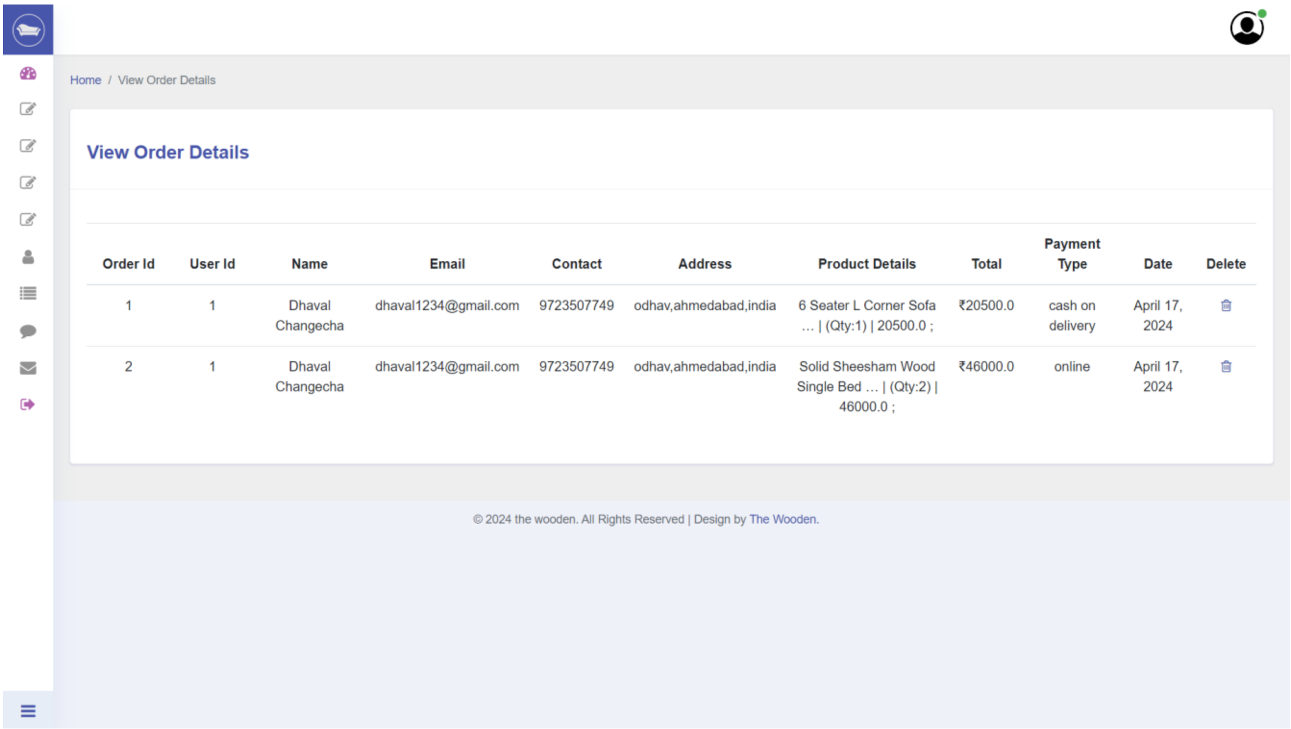
Home / View Inquiry

View Inquiry

Inquiry_id	Name	Email	Mobile	Subject	Message	Date	Delete
1	jeel panchal	justzelly@gmail.com	7878798955	dining table	asking about material of product	April 17, 2024	🗑
2	siddharaj	sid1234@gmail.com	8128706877	chair	strength of chair	April 17, 2024	🗑

© 2024 the wooden. All Rights Reserved | Design by The Wooden.

Report Layout



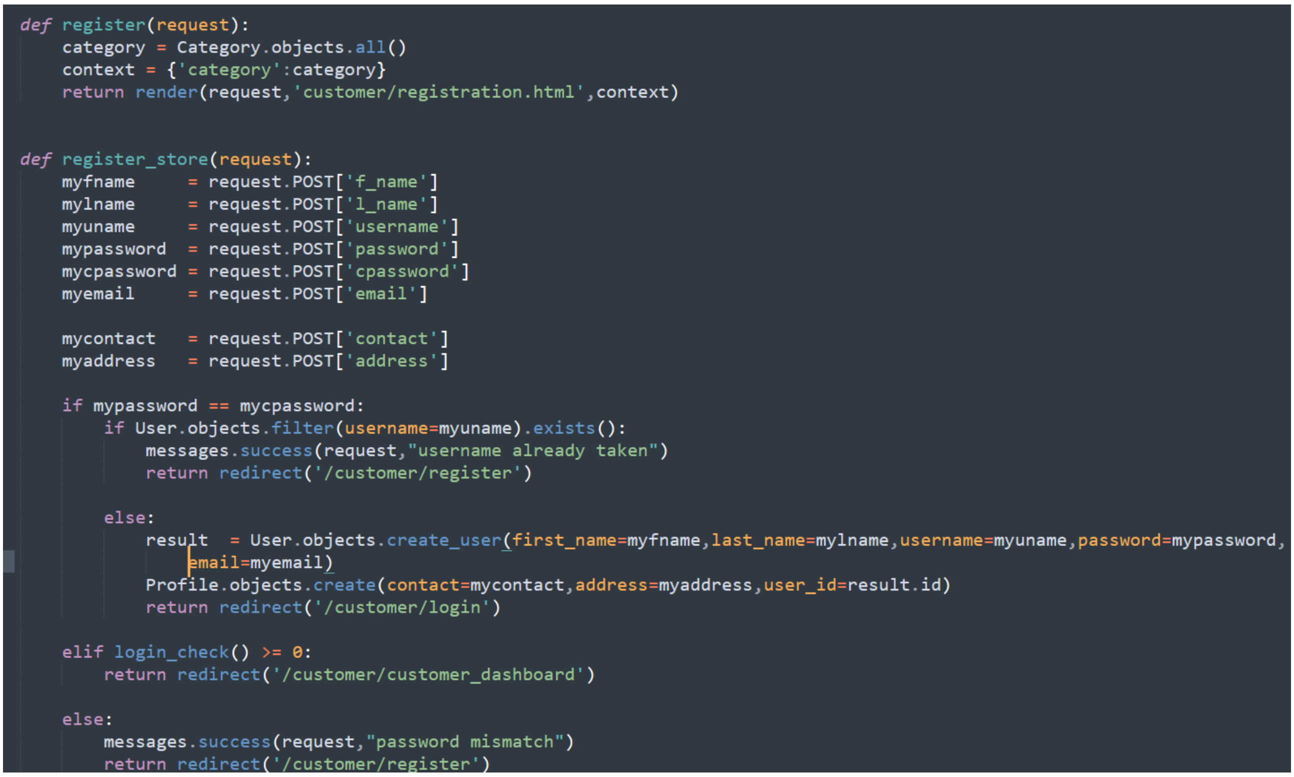
View Order Details

Order Id	User Id	Name	Email	Contact	Address	Product Details	Total	Payment Type	Date	Delete
1	1	Dhaval Changecha	dhaval1234@gmail.com	9723507749	odhav,ahmedabad,india	6 Seater L Corner Sofa … | (Qty:1) | 20500.0 ;	₹20500.0	cash on delivery	April 17, 2024	🗑
2	1	Dhaval Changecha	dhaval1234@gmail.com	9723507749	odhav,ahmedabad,india	Solid Sheesham Wood Single Bed … | (Qty:2) | 46000.0 ;	₹46000.0	online	April 17, 2024	🗑

© 2024 the wooden. All Rights Reserved | Design by The Wooden.

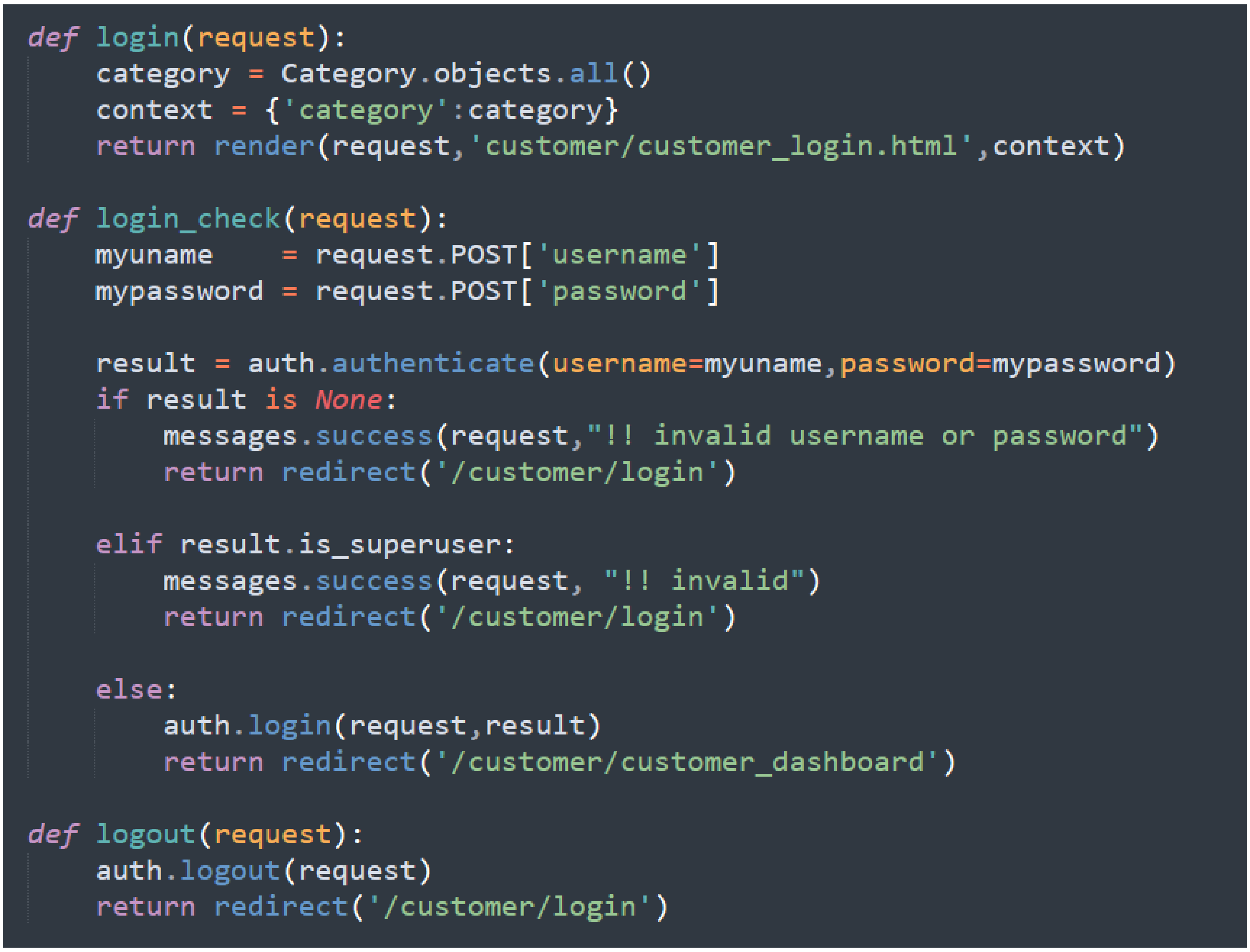
Customer Coding

- ❑ Register



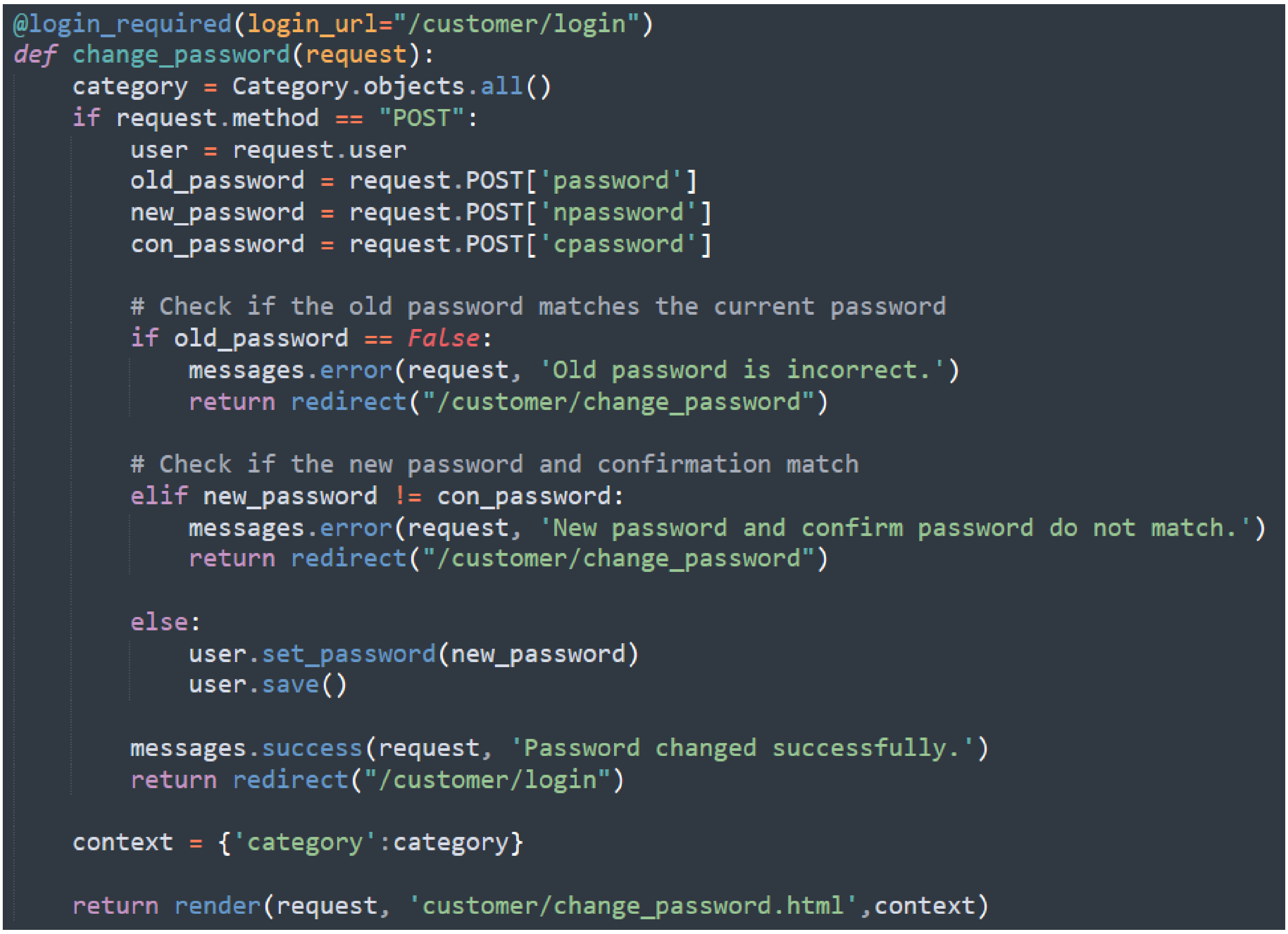
```python
def register(request):
    category = Category.objects.all()
    context = {'category':category}
    return render(request,'customer/registration.html',context)


def register_store(request):
    myfname      = request.POST['f_name']
    mylname      = request.POST['l_name']
    myuname      = request.POST['username']
    mypassword   = request.POST['password']
    mycpassword  = request.POST['cpassword']
    myemail      = request.POST['email']

    mycontact    = request.POST['contact']
    myaddress    = request.POST['address']

    if mypassword == mycpassword:
        if User.objects.filter(username=myuname).exists():
            messages.success(request,"username already taken")
            return redirect('/customer/register')

        else:
            result  = User.objects.create_user(first_name=myfname,last_name=mylname,username=myuname,password=mypassword,
                email=myemail)
            Profile.objects.create(contact=mycontact,address=myaddress,user_id=result.id)
            return redirect('/customer/login')

    elif login_check() >= 0:
        return redirect('/customer/customer_dashboard')

    else:
        messages.success(request,"password mismatch")
        return redirect('/customer/register')
```

❑ Log in



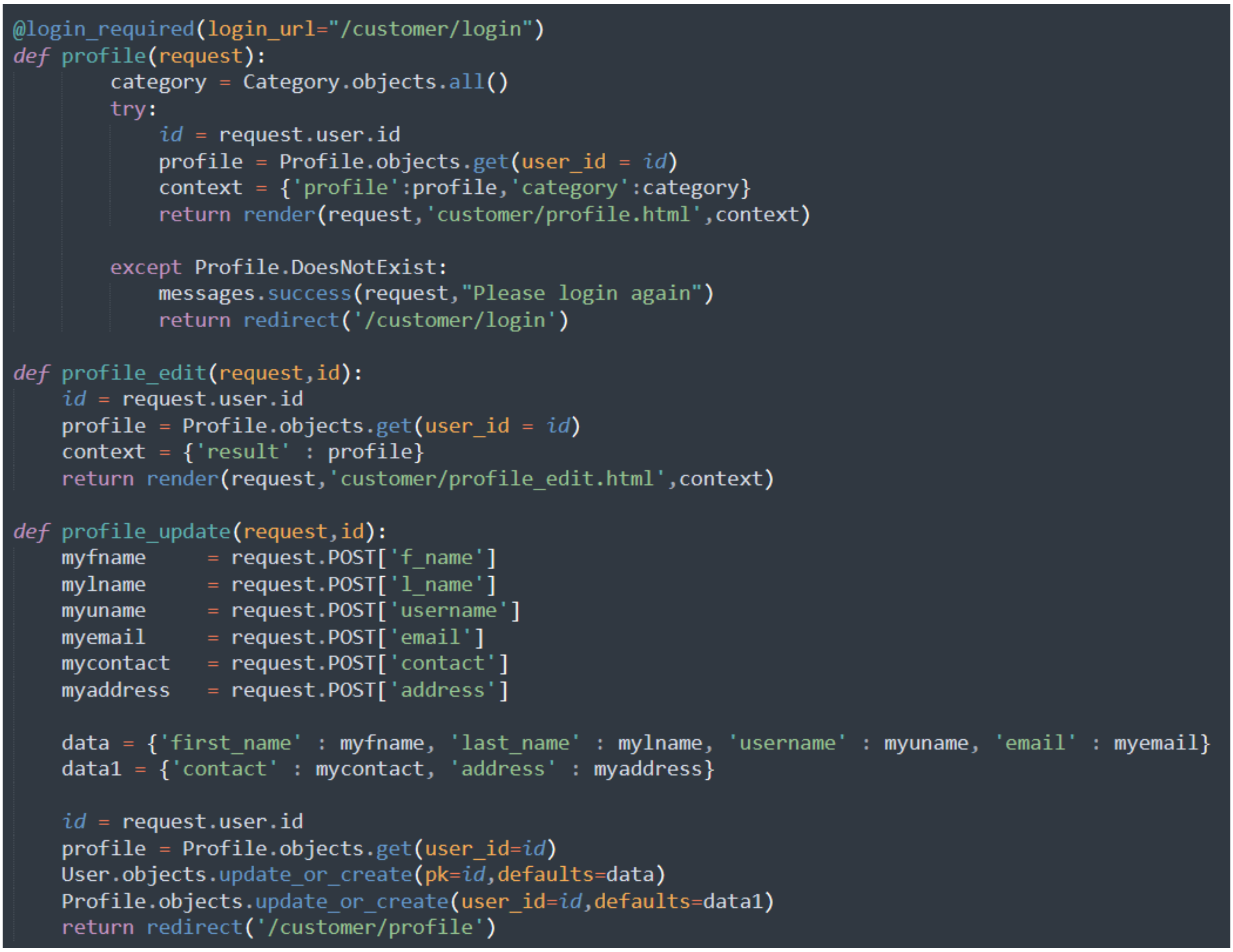
```
def login(request):
    category = Category.objects.all()
    context = {'category':category}
    return render(request,'customer/customer_login.html',context)

def login_check(request):
    myuname     = request.POST['username']
    mypassword = request.POST['password']

    result = auth.authenticate(username=myuname,password=mypassword)
    if result is None:
        messages.success(request,"!! invalid username or password")
        return redirect('/customer/login')

    elif result.is_superuser:
        messages.success(request, "!! invalid")
        return redirect('/customer/login')

    else:
        auth.login(request,result)
        return redirect('/customer/customer_dashboard')

def logout(request):
    auth.logout(request)
    return redirect('/customer/login')
```

- ❑ Change Password



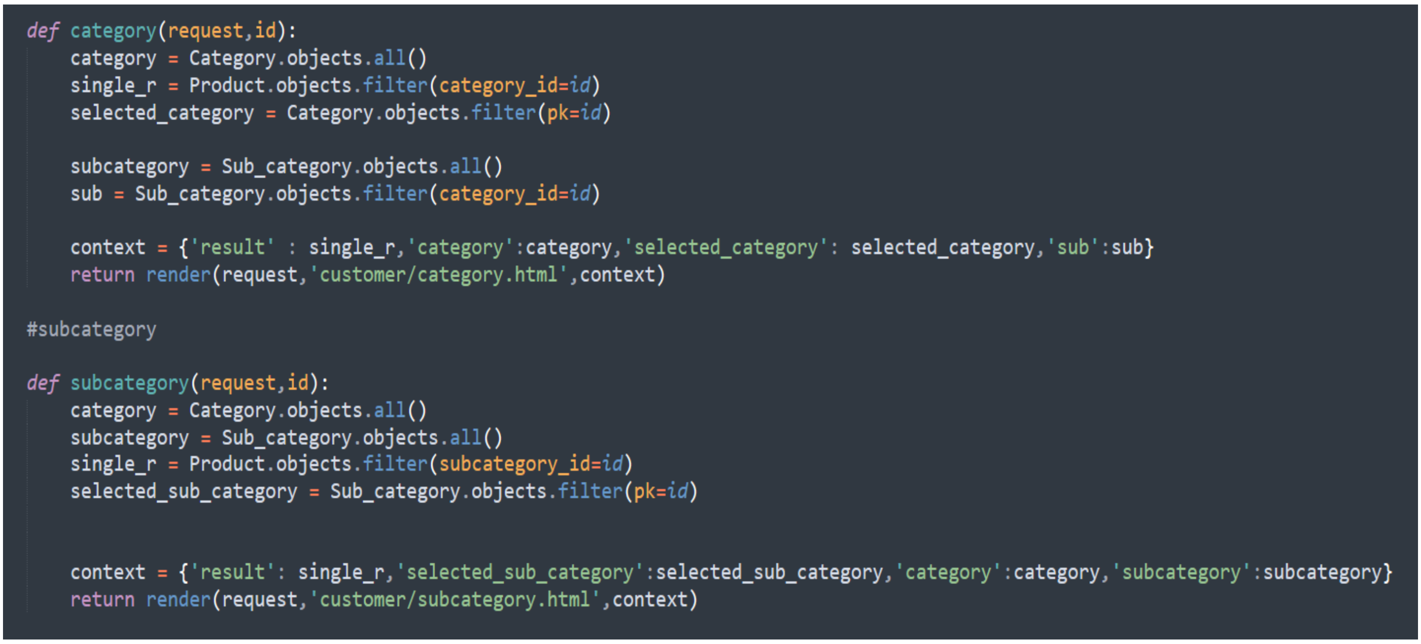
```python
@login_required(login_url="/customer/login")
def change_password(request):
    category = Category.objects.all()
    if request.method == "POST":
        user = request.user
        old_password = request.POST['password']
        new_password = request.POST['npassword']
        con_password = request.POST['cpassword']

        # Check if the old password matches the current password
        if old_password == False:
            messages.error(request, 'Old password is incorrect.')
            return redirect("/customer/change_password")

        # Check if the new password and confirmation match
        elif new_password != con_password:
            messages.error(request, 'New password and confirm password do not match.')
            return redirect("/customer/change_password")

        else:
            user.set_password(new_password)
            user.save()

        messages.success(request, 'Password changed successfully.')
        return redirect("/customer/login")

    context = {'category':category}

    return render(request, 'customer/change_password.html',context)
```

❑ Profile



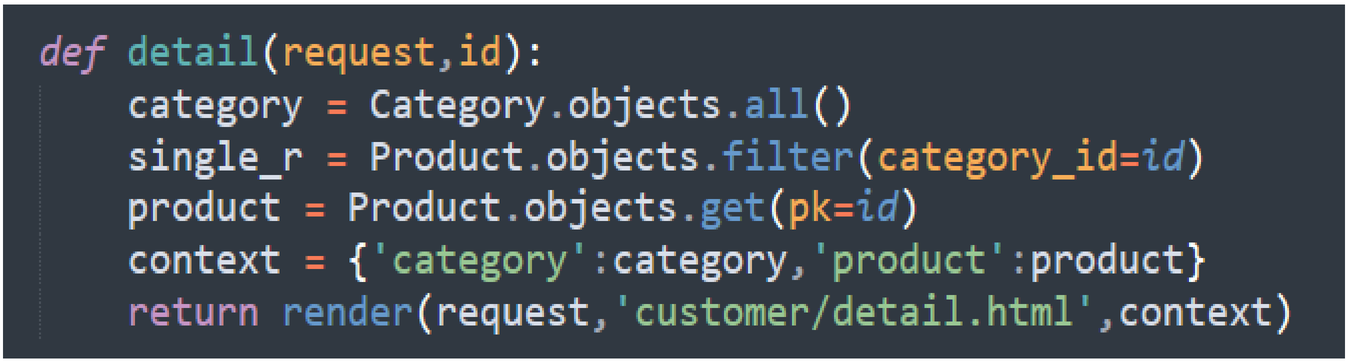
```
@login_required(login_url="/customer/login")
def profile(request):
        category = Category.objects.all()
        try:
            id = request.user.id
            profile = Profile.objects.get(user_id = id)
            context = {'profile':profile,'category':category}
            return render(request,'customer/profile.html',context)

        except Profile.DoesNotExist:
            messages.success(request,"Please login again")
            return redirect('/customer/login')

def profile_edit(request,id):
    id = request.user.id
    profile = Profile.objects.get(user_id = id)
    context = {'result' : profile}
    return render(request,'customer/profile_edit.html',context)

def profile_update(request,id):
    myfname     = request.POST['f_name']
    mylname     = request.POST['l_name']
    myuname     = request.POST['username']
    myemail     = request.POST['email']
    mycontact   = request.POST['contact']
    myaddress   = request.POST['address']

    data = {'first_name' : myfname, 'last_name' : mylname, 'username' : myuname, 'email' : myemail}
    data1 = {'contact' : mycontact, 'address' : myaddress}

    id = request.user.id
    profile = Profile.objects.get(user_id=id)
    User.objects.update_or_create(pk=id,defaults=data)
    Profile.objects.update_or_create(user_id=id,defaults=data1)
    return redirect('/customer/profile')
```

❑ Category & Sub Category



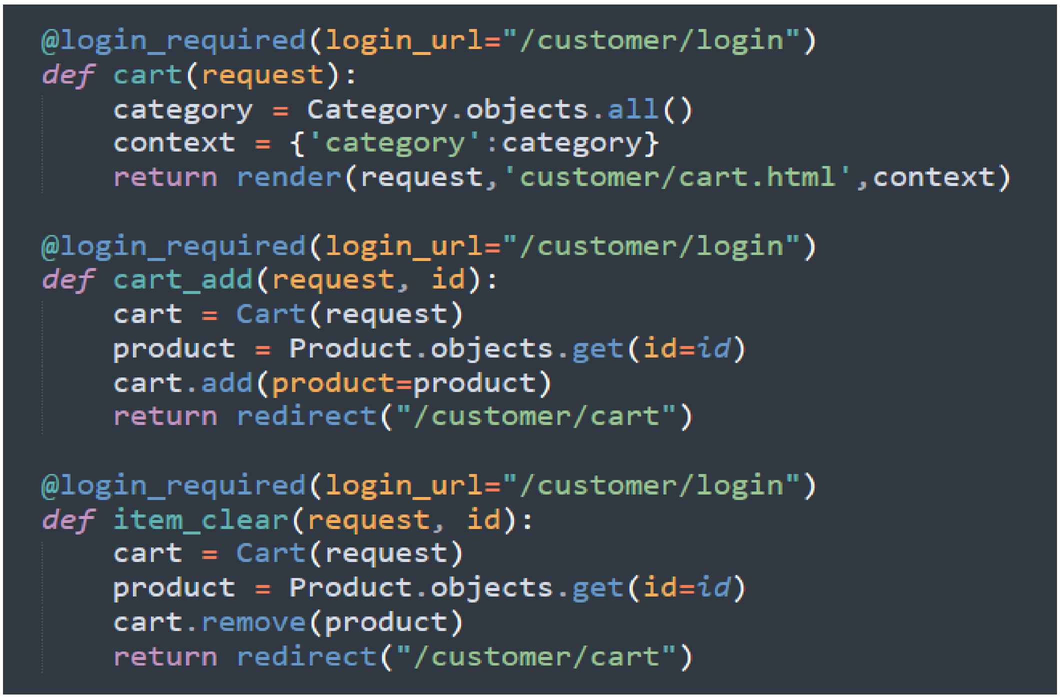
```python
def category(request,id):
    category = Category.objects.all()
    single_r = Product.objects.filter(category_id=id)
    selected_category = Category.objects.filter(pk=id)

    subcategory = Sub_category.objects.all()
    sub = Sub_category.objects.filter(category_id=id)

    context = {'result' : single_r,'category':category,'selected_category': selected_category,'sub':sub}
    return render(request,'customer/category.html',context)

#subcategory

def subcategory(request,id):
    category = Category.objects.all()
    subcategory = Sub_category.objects.all()
    single_r = Product.objects.filter(subcategory_id=id)
    selected_sub_category = Sub_category.objects.filter(pk=id)


    context = {'result': single_r,'selected_sub_category':selected_sub_category,'category':category,'subcategory':subcategory}
    return render(request,'customer/subcategory.html',context)
```

❑ Single Product

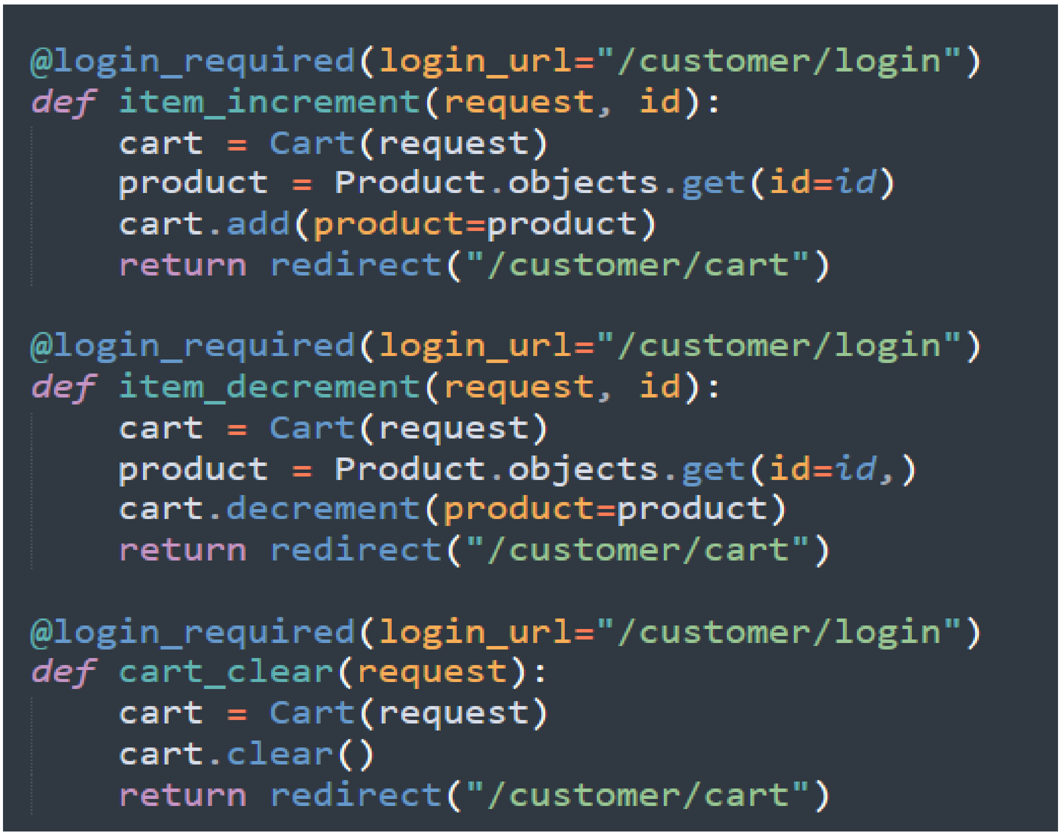


```python
def detail(request,id):
    category = Category.objects.all()
    single_r = Product.objects.filter(category_id=id)
    product = Product.objects.get(pk=id)
    context = {'category':category,'product':product}
    return render(request,'customer/detail.html',context)
```

- ❑ Shopping Cart

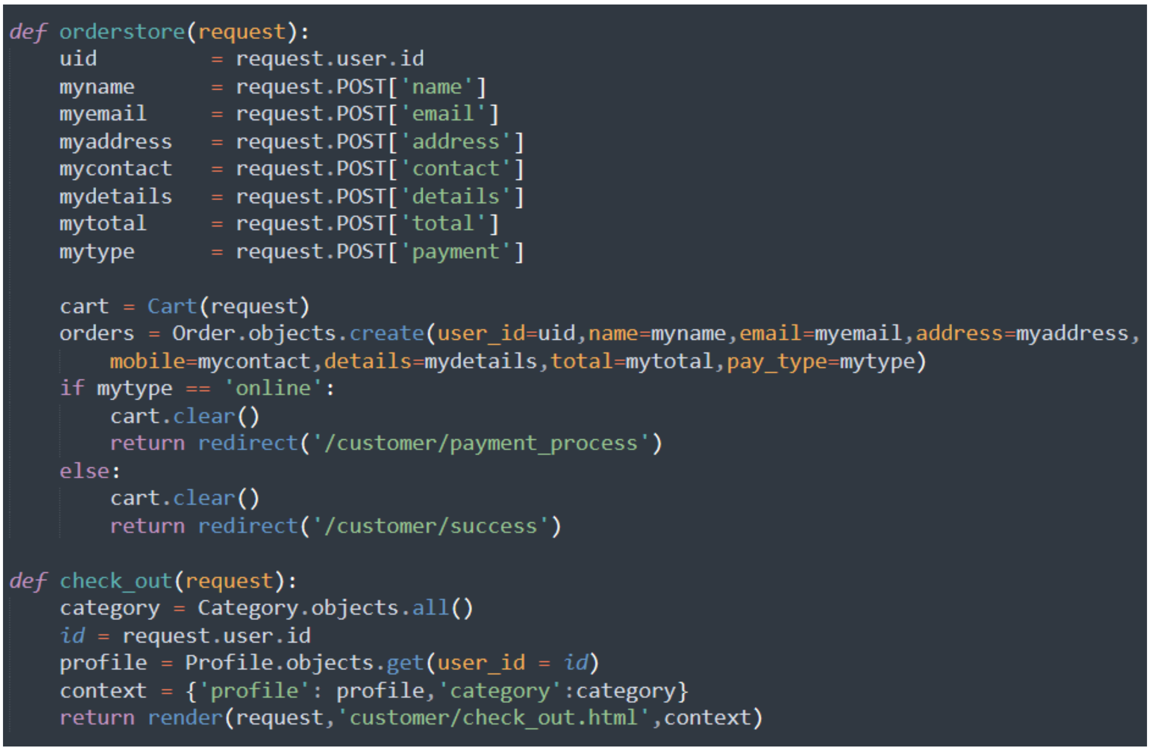


```
@login_required(login_url="/customer/login")
def cart(request):
    category = Category.objects.all()
    context = {'category':category}
    return render(request,'customer/cart.html',context)

@login_required(login_url="/customer/login")
def cart_add(request, id):
    cart = Cart(request)
    product = Product.objects.get(id=id)
    cart.add(product=product)
    return redirect("/customer/cart")

@login_required(login_url="/customer/login")
def item_clear(request, id):
    cart = Cart(request)
    product = Product.objects.get(id=id)
    cart.remove(product)
    return redirect("/customer/cart")
```



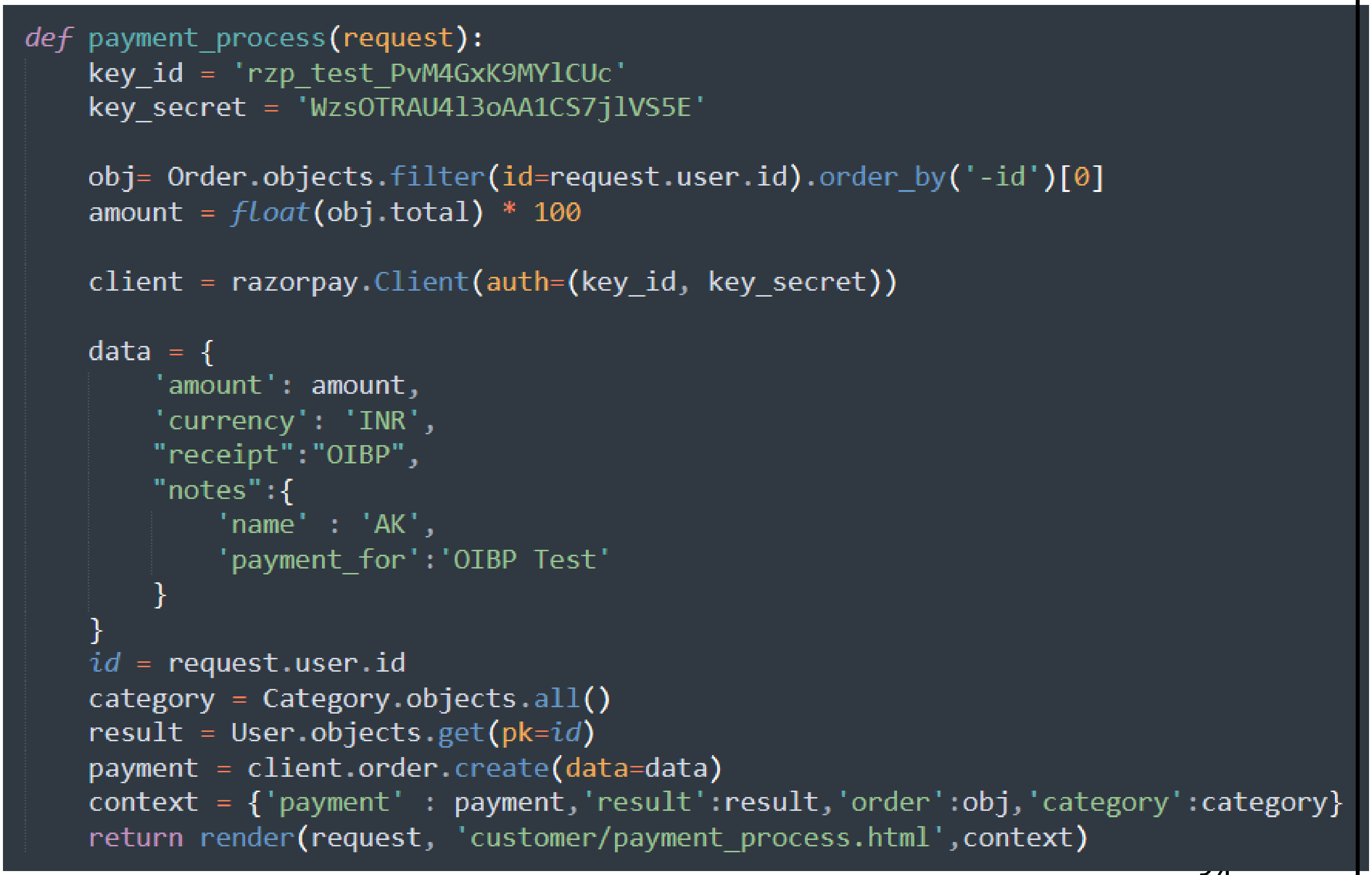
```
@login_required(login_url="/customer/login")
def item_increment(request, id):
    cart = Cart(request)
    product = Product.objects.get(id=id)
    cart.add(product=product)
    return redirect("/customer/cart")

@login_required(login_url="/customer/login")
def item_decrement(request, id):
    cart = Cart(request)
    product = Product.objects.get(id=id,)
    cart.decrement(product=product)
    return redirect("/customer/cart")

@login_required(login_url="/customer/login")
def cart_clear(request):
    cart = Cart(request)
    cart.clear()
    return redirect("/customer/cart")
```

❑ Order



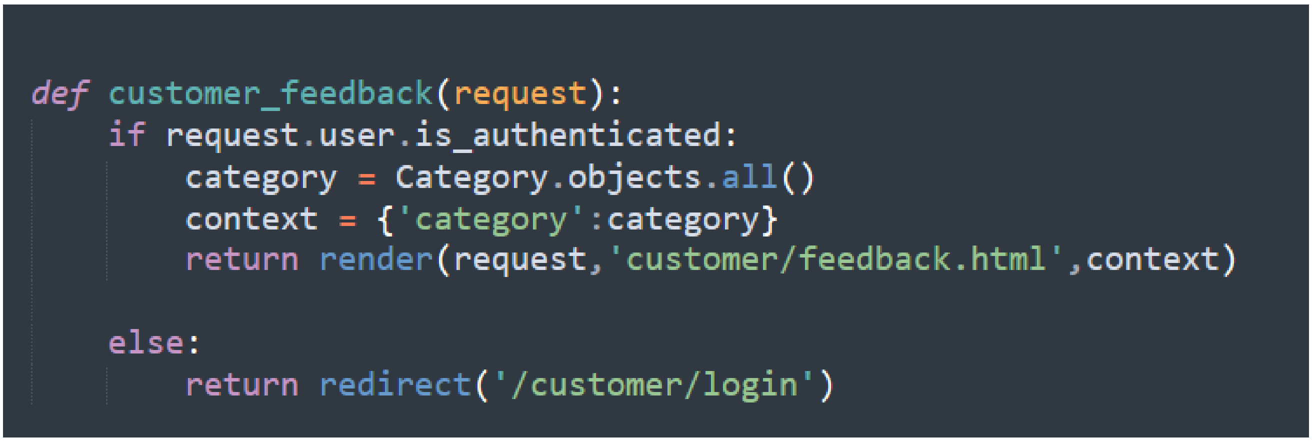
```
def orderstore(request):
    uid         = request.user.id
    myname      = request.POST['name']
    myemail     = request.POST['email']
    myaddress   = request.POST['address']
    mycontact   = request.POST['contact']
    mydetails   = request.POST['details']
    mytotal     = request.POST['total']
    mytype      = request.POST['payment']

    cart = Cart(request)
    orders = Order.objects.create(user_id=uid,name=myname,email=myemail,address=myaddress,
        mobile=mycontact,details=mydetails,total=mytotal,pay_type=mytype)
    if mytype == 'online':
        cart.clear()
        return redirect('/customer/payment_process')
    else:
        cart.clear()
        return redirect('/customer/success')

def check_out(request):
    category = Category.objects.all()
    id = request.user.id
    profile = Profile.objects.get(user_id = id)
    context = {'profile': profile,'category':category}
    return render(request,'customer/check_out.html',context)
```

❑ Payment



```
def payment_process(request):
    key_id = 'rzp_test_PvM4GxK9MYlCUc'
    key_secret = 'WzsOTRAU4l3oAA1CS7jlVS5E'

    obj= Order.objects.filter(id=request.user.id).order_by('-id')[0]
    amount = float(obj.total) * 100

    client = razorpay.Client(auth=(key_id, key_secret))

    data = {
        'amount': amount,
        'currency': 'INR',
        "receipt":"OIBP",
        "notes":{
            'name' : 'AK',
            'payment_for':'OIBP Test'
        }
    }
    id = request.user.id
    category = Category.objects.all()
    result = User.objects.get(pk=id)
    payment = client.order.create(data=data)
    context = {'payment' : payment,'result':result,'order':obj,'category':category}
    return render(request, 'customer/payment_process.html',context)
```

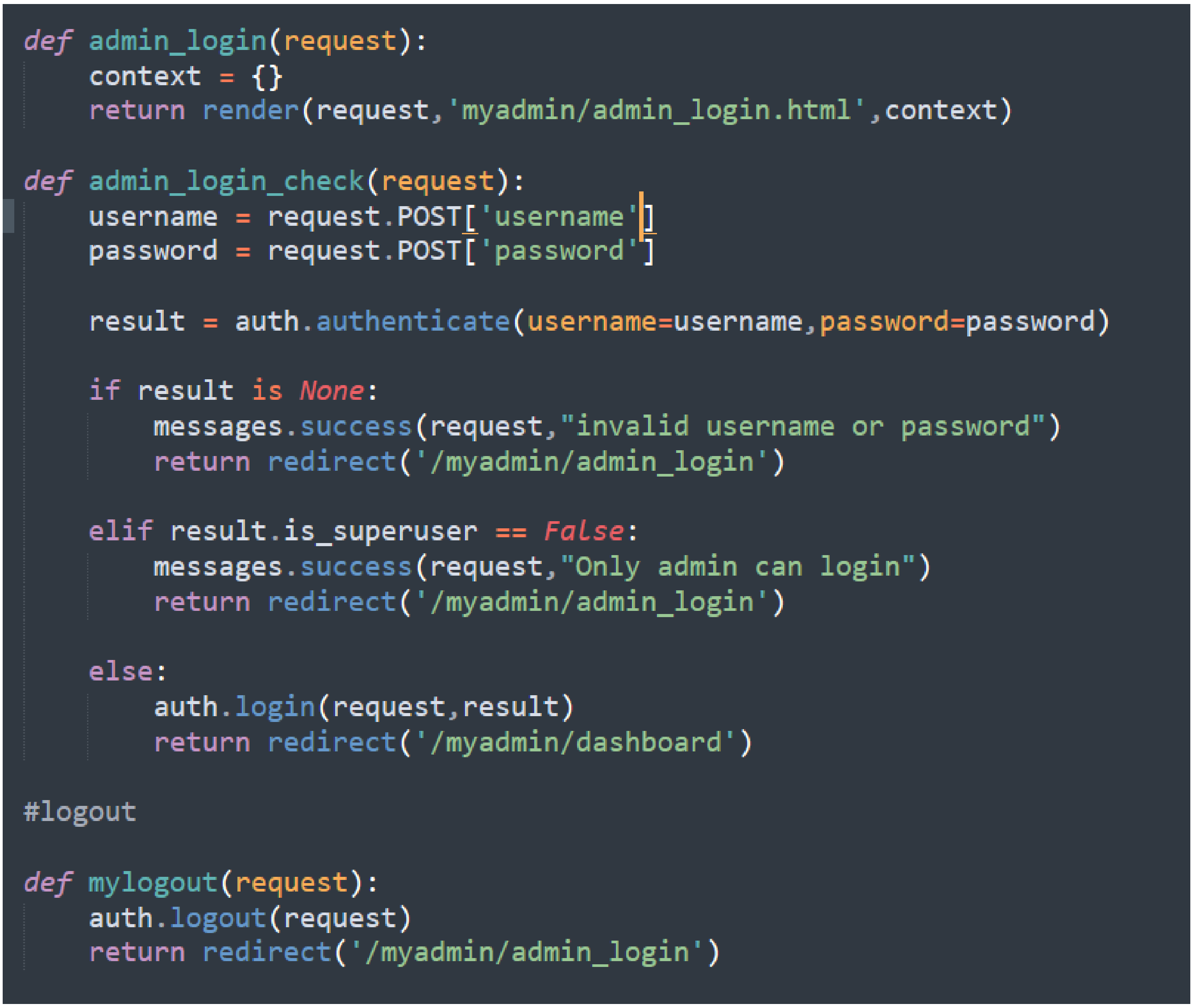
❑ Feedback



```python
def customer_feedback(request):
    if request.user.is_authenticated:
        category = Category.objects.all()
        context = {'category':category}
        return render(request,'customer/feedback.html',context)

    else:
        return redirect('/customer/login')
```

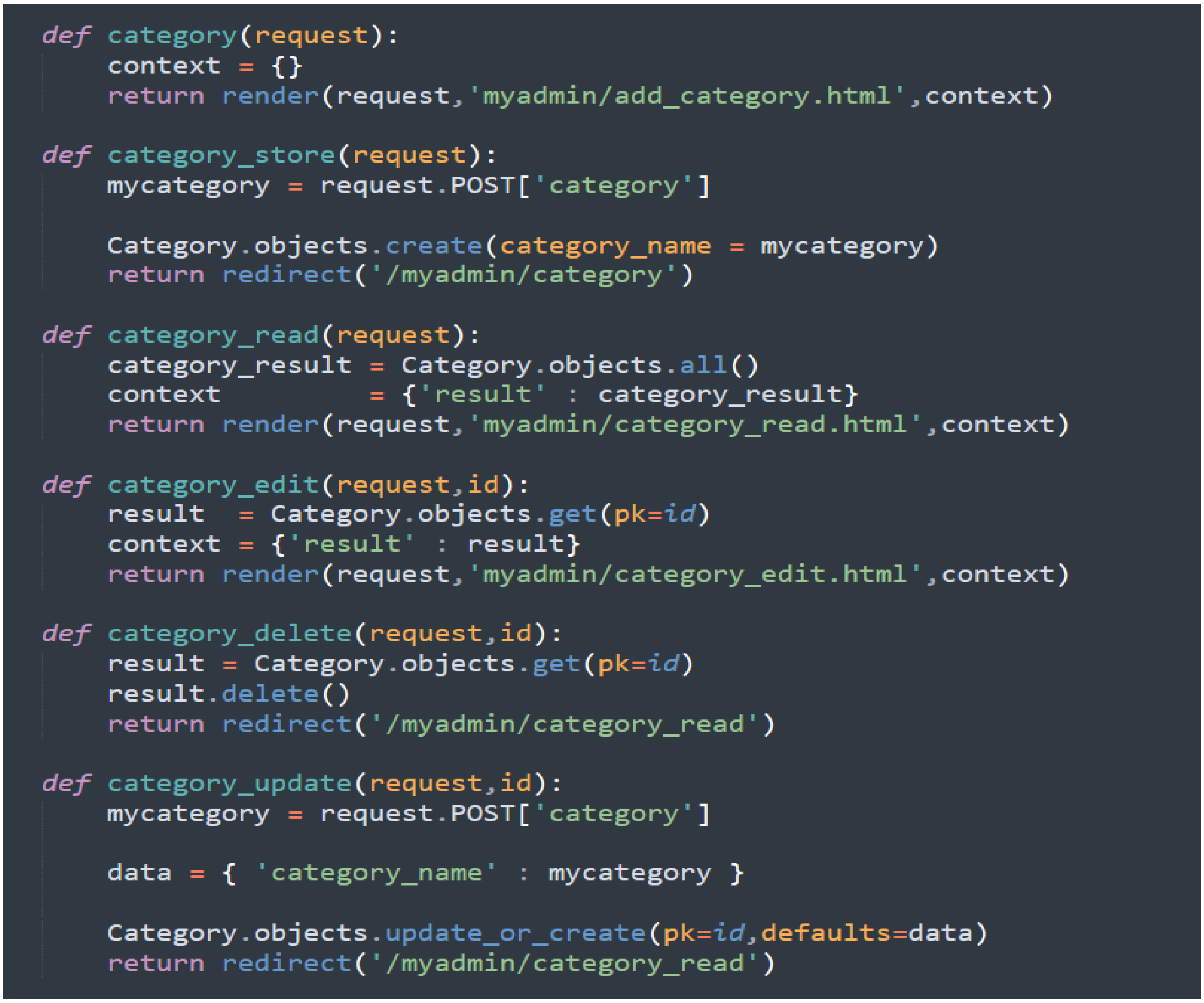
Admin Coding

❑ Log in



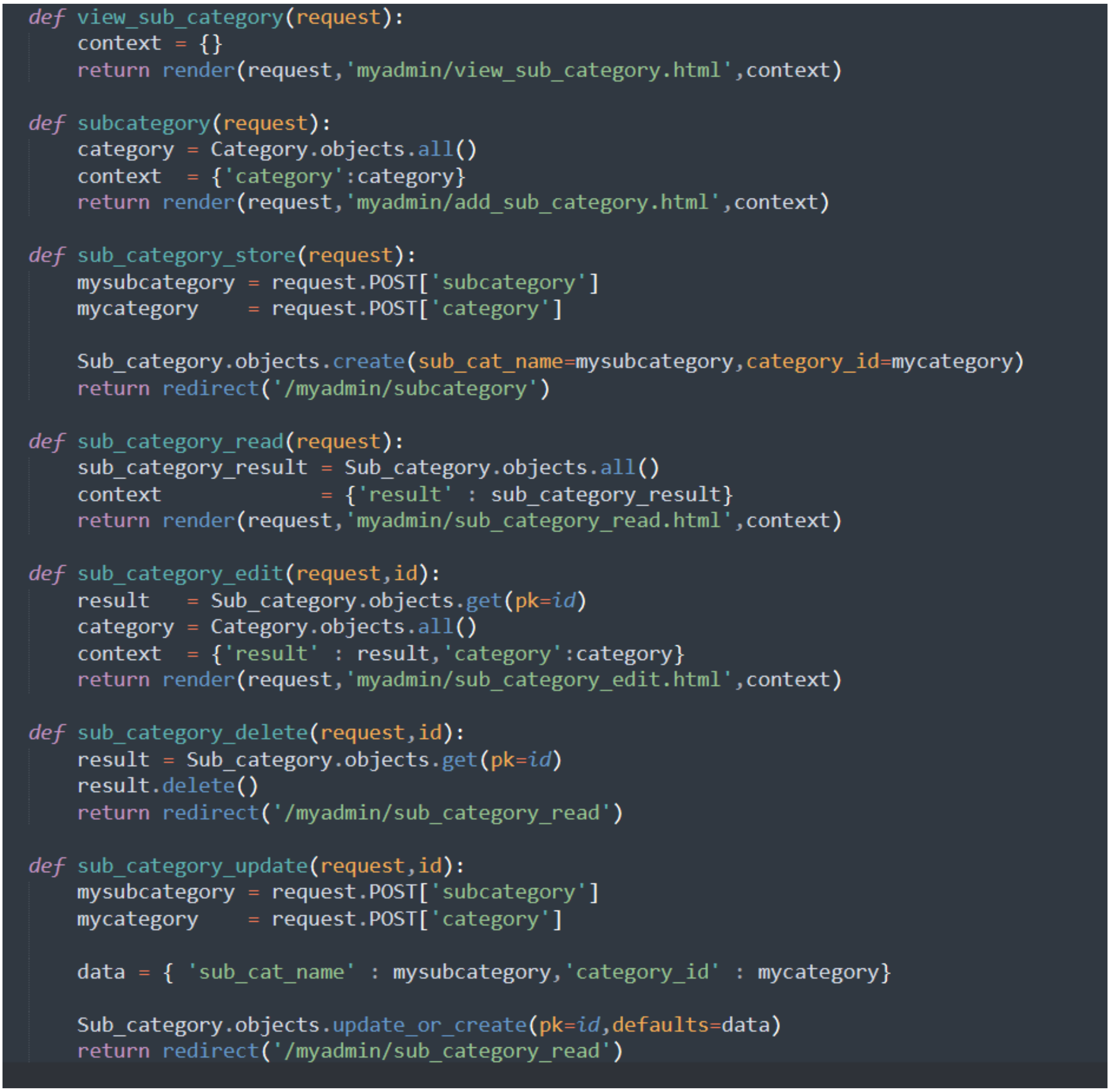
```
def admin_login(request):
    context = {}
    return render(request,'myadmin/admin_login.html',context)

def admin_login_check(request):
    username = request.POST['username']
    password = request.POST['password']

    result = auth.authenticate(username=username,password=password)

    if result is None:
        messages.success(request,"invalid username or password")
        return redirect('/myadmin/admin_login')

    elif result.is_superuser == False:
        messages.success(request,"Only admin can login")
        return redirect('/myadmin/admin_login')

    else:
        auth.login(request,result)
        return redirect('/myadmin/dashboard')

#logout

def mylogout(request):
    auth.logout(request)
    return redirect('/myadmin/admin_login')
```

❑ Manage Category



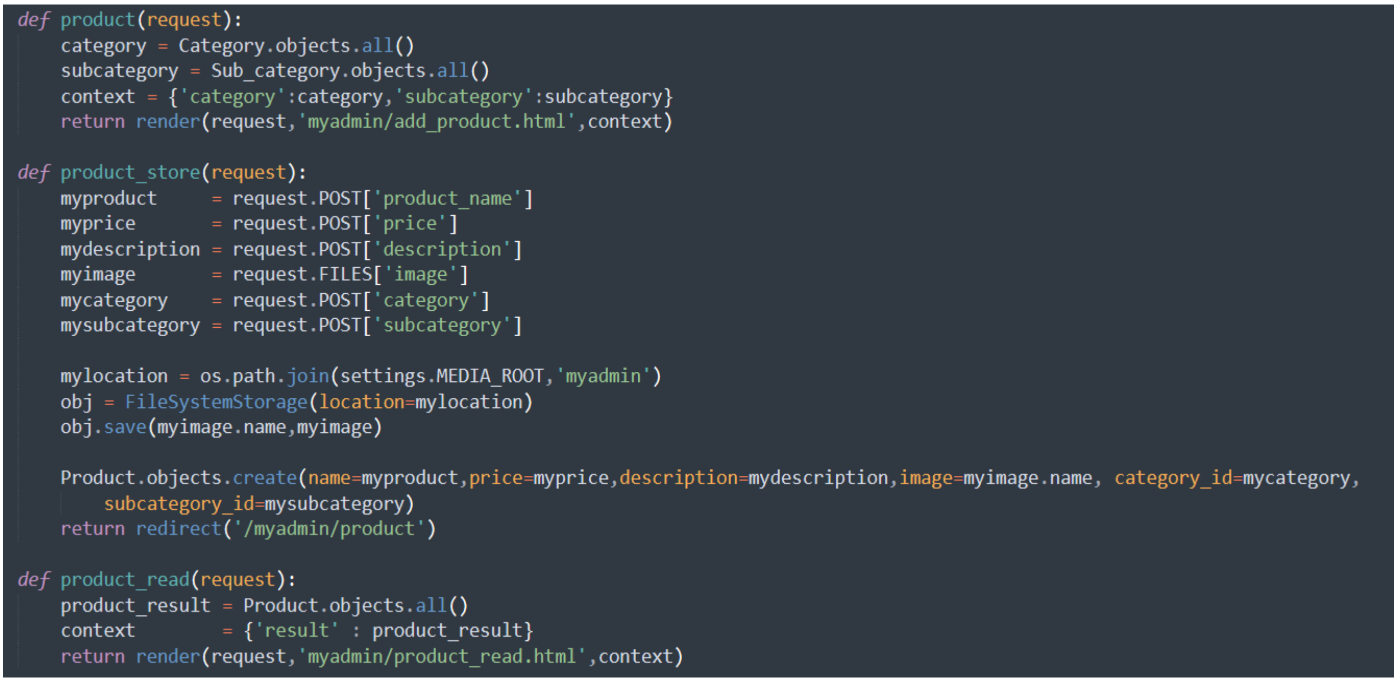
```
def category(request):
    context = {}
    return render(request,'myadmin/add_category.html',context)

def category_store(request):
    mycategory = request.POST['category']

    Category.objects.create(category_name = mycategory)
    return redirect('/myadmin/category')

def category_read(request):
    category_result = Category.objects.all()
    context          = {'result'  : category_result}
    return render(request,'myadmin/category_read.html',context)

def category_edit(request,id):
    result   = Category.objects.get(pk=id)
    context = {'result'  : result}
    return render(request,'myadmin/category_edit.html',context)

def category_delete(request,id):
    result = Category.objects.get(pk=id)
    result.delete()
    return redirect('/myadmin/category_read')

def category_update(request,id):
    mycategory = request.POST['category']

    data = { 'category_name'  : mycategory }

    Category.objects.update_or_create(pk=id,defaults=data)
    return redirect('/myadmin/category_read')
```

- ❑ Manage Sub Category

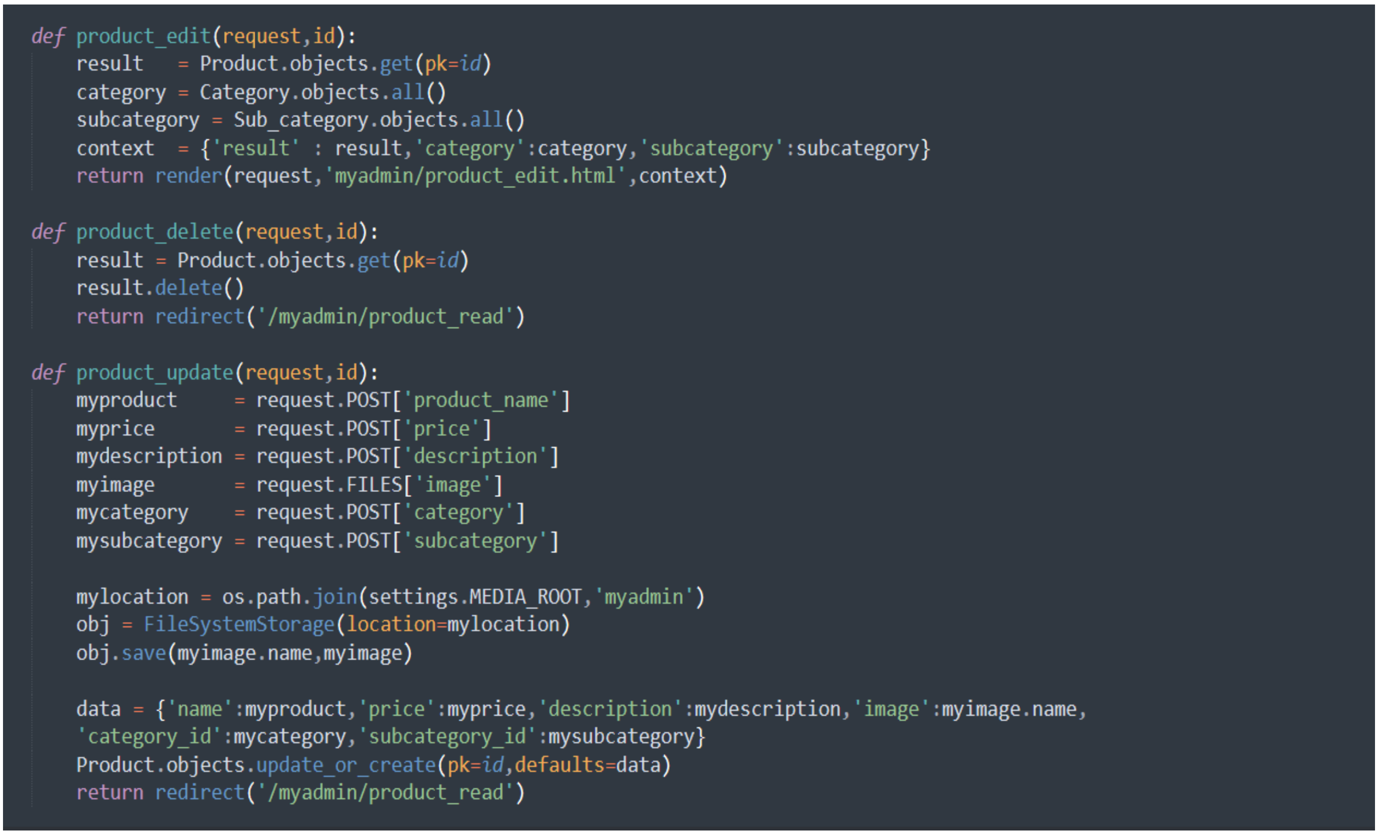


```python
def view_sub_category(request):
    context = {}
    return render(request,'myadmin/view_sub_category.html',context)

def subcategory(request):
    category = Category.objects.all()
    context  = {'category':category}
    return render(request,'myadmin/add_sub_category.html',context)

def sub_category_store(request):
    mysubcategory = request.POST['subcategory']
    mycategory    = request.POST['category']

    Sub_category.objects.create(sub_cat_name=mysubcategory,category_id=mycategory)
    return redirect('/myadmin/subcategory')

def sub_category_read(request):
    sub_category_result = Sub_category.objects.all()
    context             = {'result' : sub_category_result}
    return render(request,'myadmin/sub_category_read.html',context)

def sub_category_edit(request,id):
    result   = Sub_category.objects.get(pk=id)
    category = Category.objects.all()
    context  = {'result' : result,'category':category}
    return render(request,'myadmin/sub_category_edit.html',context)

def sub_category_delete(request,id):
    result = Sub_category.objects.get(pk=id)
    result.delete()
    return redirect('/myadmin/sub_category_read')

def sub_category_update(request,id):
    mysubcategory = request.POST['subcategory']
    mycategory    = request.POST['category']

    data = { 'sub_cat_name' : mysubcategory,'category_id' : mycategory}

    Sub_category.objects.update_or_create(pk=id,defaults=data)
    return redirect('/myadmin/sub_category_read')
```

- ❑ Manage Product

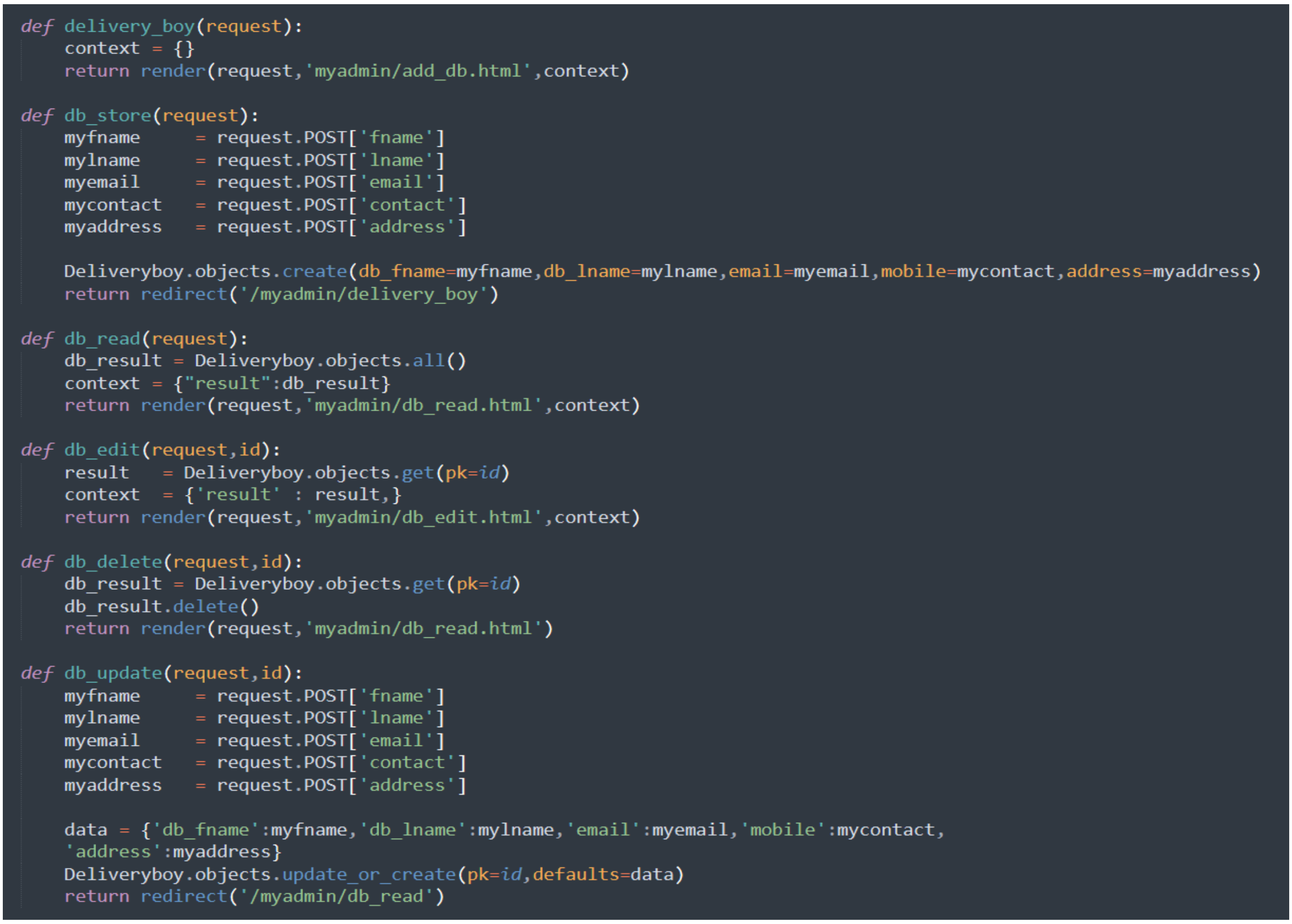


```
def product(request):
    category = Category.objects.all()
    subcategory = Sub_category.objects.all()
    context = {'category':category,'subcategory':subcategory}
    return render(request,'myadmin/add_product.html',context)

def product_store(request):
    myproduct     = request.POST['product_name']
    myprice       = request.POST['price']
    mydescription = request.POST['description']
    myimage       = request.FILES['image']
    mycategory    = request.POST['category']
    mysubcategory = request.POST['subcategory']

    mylocation = os.path.join(settings.MEDIA_ROOT,'myadmin')
    obj = FileSystemStorage(location=mylocation)
    obj.save(myimage.name,myimage)

    Product.objects.create(name=myproduct,price=myprice,description=mydescription,image=myimage.name, category_id=mycategory,
        subcategory_id=mysubcategory)
    return redirect('/myadmin/product')

def product_read(request):
    product_result = Product.objects.all()
    context        = {'result' : product_result}
    return render(request,'myadmin/product_read.html',context)
```



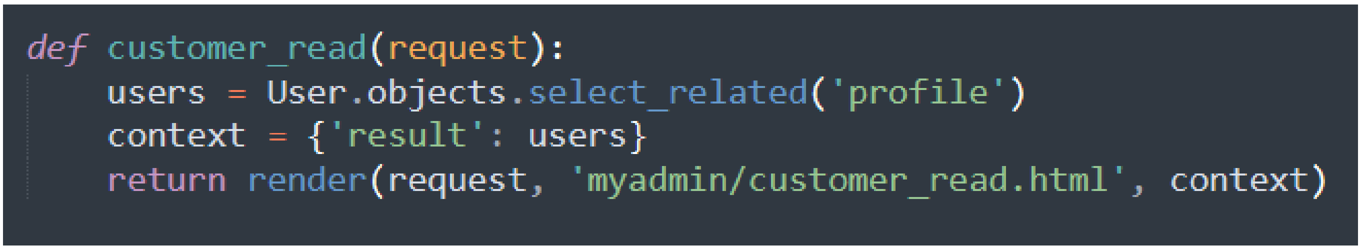
```
def product_edit(request,id):
    result    = Product.objects.get(pk=id)
    category = Category.objects.all()
    subcategory = Sub_category.objects.all()
    context   = {'result' : result,'category':category,'subcategory':subcategory}
    return render(request,'myadmin/product_edit.html',context)

def product_delete(request,id):
    result = Product.objects.get(pk=id)
    result.delete()
    return redirect('/myadmin/product_read')

def product_update(request,id):
    myproduct     = request.POST['product_name']
    myprice       = request.POST['price']
    mydescription = request.POST['description']
    myimage       = request.FILES['image']
    mycategory    = request.POST['category']
    mysubcategory = request.POST['subcategory']

    mylocation = os.path.join(settings.MEDIA_ROOT,'myadmin')
    obj = FileSystemStorage(location=mylocation)
    obj.save(myimage.name,myimage)

    data = {'name':myproduct,'price':myprice,'description':mydescription,'image':myimage.name,
    'category_id':mycategory,'subcategory_id':mysubcategory}
    Product.objects.update_or_create(pk=id,defaults=data)
    return redirect('/myadmin/product_read')
```

❑ Manage Delivery Boy



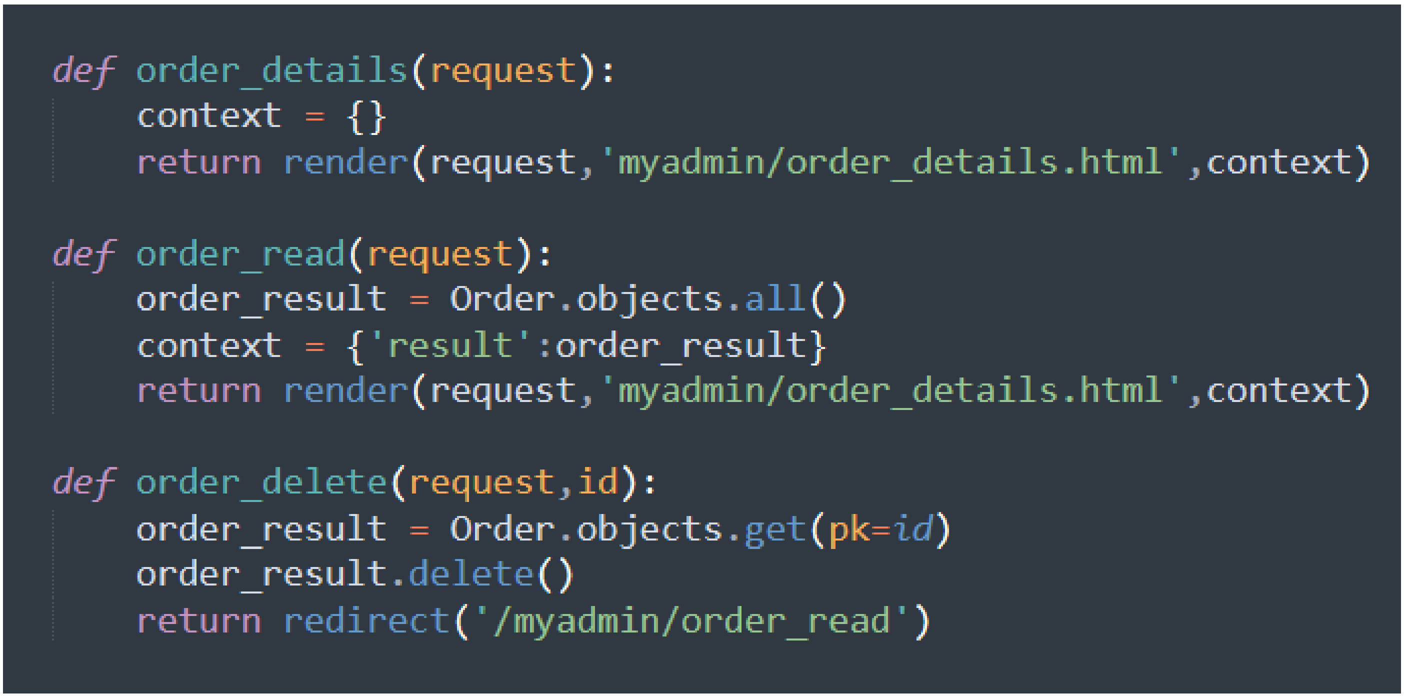
```
def delivery_boy(request):
    context = {}
    return render(request,'myadmin/add_db.html',context)

def db_store(request):
    myfname      = request.POST['fname']
    mylname      = request.POST['lname']
    myemail      = request.POST['email']
    mycontact    = request.POST['contact']
    myaddress    = request.POST['address']

    Deliveryboy.objects.create(db_fname=myfname,db_lname=mylname,email=myemail,mobile=mycontact,address=myaddress)
    return redirect('/myadmin/delivery_boy')

def db_read(request):
    db_result = Deliveryboy.objects.all()
    context = {"result":db_result}
    return render(request,'myadmin/db_read.html',context)

def db_edit(request,id):
    result    = Deliveryboy.objects.get(pk=id)
    context   = {'result' : result,}
    return render(request,'myadmin/db_edit.html',context)

def db_delete(request,id):
    db_result = Deliveryboy.objects.get(pk=id)
    db_result.delete()
    return render(request,'myadmin/db_read.html')

def db_update(request,id):
    myfname      = request.POST['fname']
    mylname      = request.POST['lname']
    myemail      = request.POST['email']
    mycontact    = request.POST['contact']
    myaddress    = request.POST['address']

    data = {'db_fname':myfname,'db_lname':mylname,'email':myemail,'mobile':mycontact,
    'address':myaddress}
    Deliveryboy.objects.update_or_create(pk=id,defaults=data)
    return redirect('/myadmin/db_read')
```

❑ Users



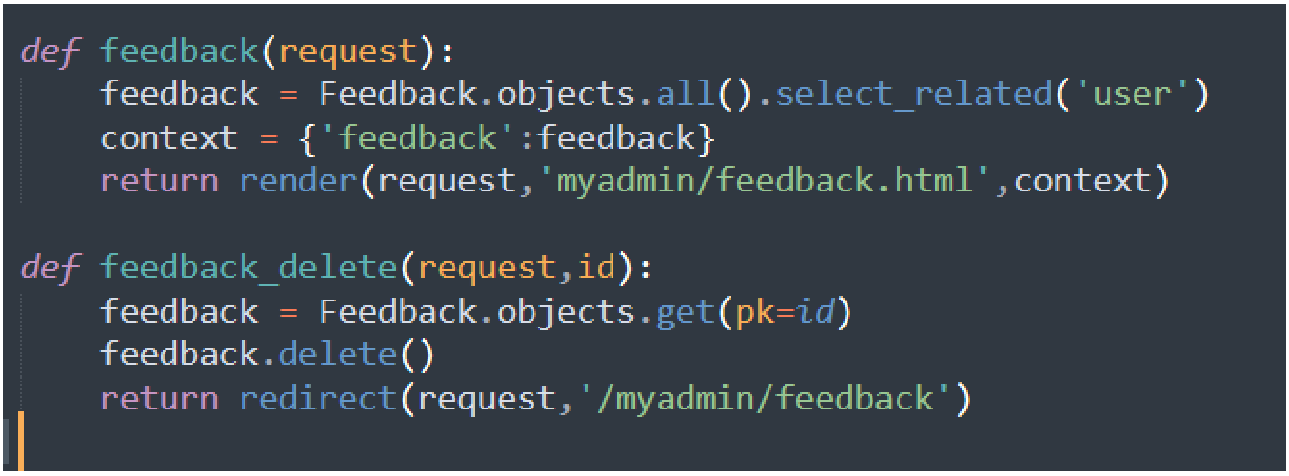
```python
def customer_read(request):
    users = User.objects.select_related('profile')
    context = {'result': users}
    return render(request, 'myadmin/customer_read.html', context)
```

❑ Manage Order



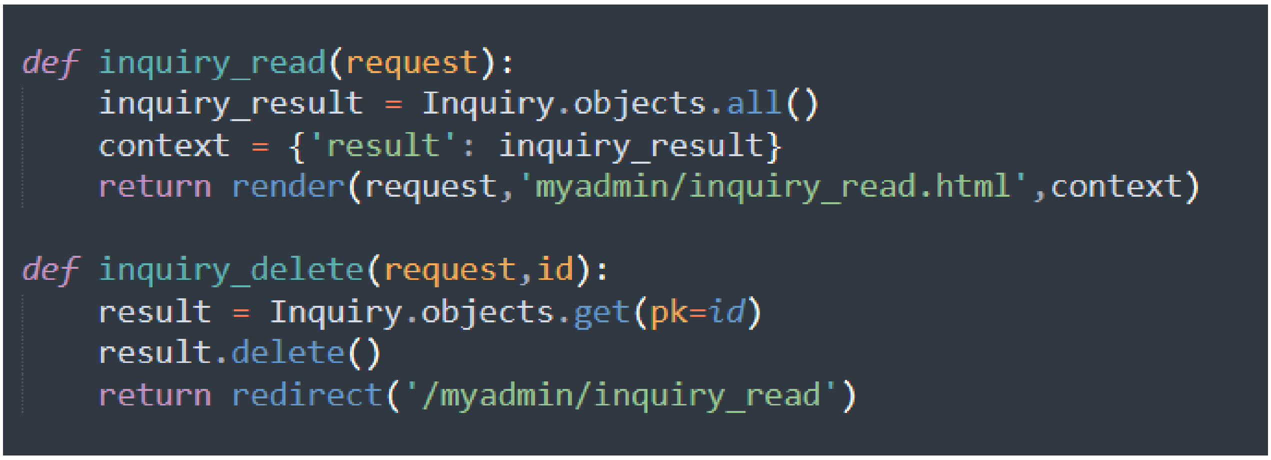
```python
def order_details(request):
    context = {}
    return render(request,'myadmin/order_details.html',context)

def order_read(request):
    order_result = Order.objects.all()
    context = {'result':order_result}
    return render(request,'myadmin/order_details.html',context)

def order_delete(request,id):
    order_result = Order.objects.get(pk=id)
    order_result.delete()
    return redirect('/myadmin/order_read')
```

❑ Manage Feedback



```python
def feedback(request):
    feedback = Feedback.objects.all().select_related('user')
    context = {'feedback':feedback}
    return render(request,'myadmin/feedback.html',context)

def feedback_delete(request,id):
    feedback = Feedback.objects.get(pk=id)
    feedback.delete()
    return redirect(request,'/myadmin/feedback')
```

❑ Manage Inquiry



```python
def inquiry_read(request):
    inquiry_result = Inquiry.objects.all()
    context = {'result': inquiry_result}
    return render(request,'myadmin/inquiry_read.html',context)

def inquiry_delete(request,id):
    result = Inquiry.objects.get(pk=id)
    result.delete()
    return redirect('/myadmin/inquiry_read')
```

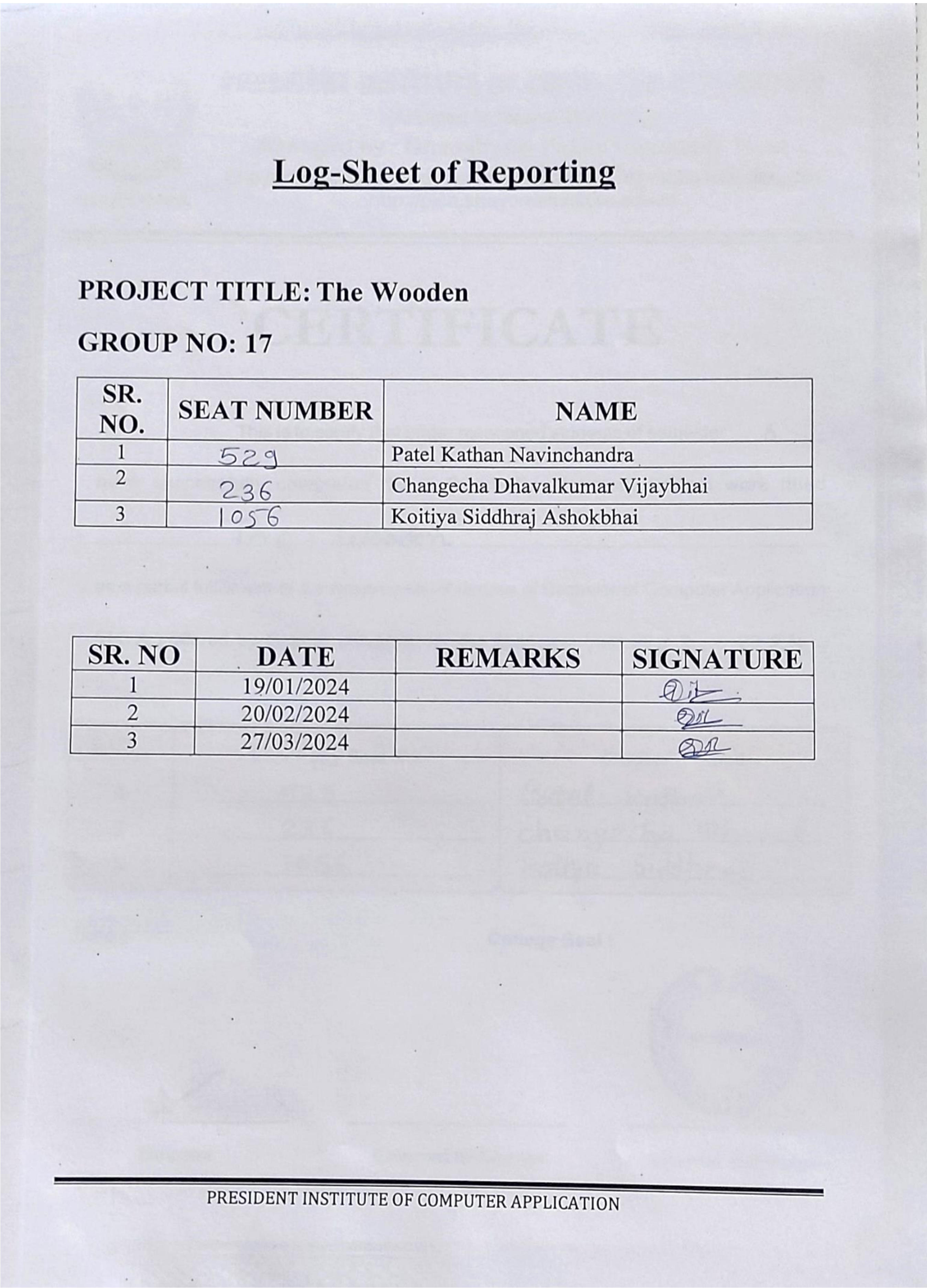
Bibliography

For successfully completing our project file we have taken help from the following website links :-

- www.google.com
- www.w3layout.com
- www.w3schools.com
- www.djangoproject.com
- www.getbootstrap.com
- https://dev.to/

Conclusion

- The online furniture website offers a wide variety of high-quality furniture options for customers to choose from.
- Customers can easily browse through different categories and styles of furniture to find the perfect pieces for their homes.
- The website provides detailed product descriptions, images, and customer reviews to help shoppers make informed purchasing decisions.
- With convenient online ordering and delivery options, customers can enjoy a seamless shopping experience from start to finish.
- Overall, the online furniture website is a convenient and reliable platform for customers to find and purchase furniture for their homes.



Log-Sheet of Reporting

PROJECT TITLE: The Wooden

GROUP NO: 17

SR. NO.	SEAT NUMBER	NAME
1	529	Patel Kathan Navinchandra
2	236	Changecha Dhavalkumar Vijaybhai
3	1056	Koitiya Siddhraj Ashokbhai

SR. NO	DATE	REMARKS	SIGNATURE
1	19/01/2024		
2	20/02/2024		
3	27/03/2024		

Thank you