

'''

Experiment 09 : Creating web application using Django web framework to demonstrate functionality of user login and registration (also validating user detail using regular expression).

Name : Khan Arshad Abdulla

Roll No : 20CO24

Academic Year : 2021-22

'''

```
from django.shortcuts import render ,redirect
from django.http import HttpResponse
from django.contrib.auth.models import User,auth
from .models import Faculty, student_profile
from django.contrib.auth import authenticate,login,logout
from django.contrib.auth.mixins import LoginRequiredMixin
from django.contrib.auth.decorators import login_required
from .decorators import allowed_users
from django.contrib import messages

# Create your views here.

# Home page

@login_required()
def home(request):
    if request.user.is_staff:
        return HttpResponse("Since you are faculty you cannot fill this form")
    else:

        if request.method == 'POST':
            full_name=request.POST.get('full_name')
            Roll_no = request.POST.get('Roll_no')
            birthdate = request.POST.get('birthdate')
            image = request.FILES.get('image')
```

```

annualin = request.POST.get('annualin')
mobil_no = request.POST.get('mobile_no')
address = request.POST.get('address')
bloodgroup = request.POST.get('bloodgroup')
familymember = request.POST.get('familymember')
fatheroccupation = request.POST.get('fatheroccupation')
maths = request.POST.get('maths')
algo = request.POST.get('algo')
operating = request.POST.get('operating')
micro = request.POST.get('micro')
dbms = request.POST.get('dbms')
health = request.POST.get('health')
# dept = request.POST.get('dept')
# clas = request.POST.get('clas')
mentor = request.POST.get('mentor')
travel = request.POST.get('travel')
cgpa = request.POST.get('cgpa')
language = request.POST.get('language')
areaofinterest = request.POST.get('areaofinterest')
internship = request.POST.get('internship')
user_id = request.POST.get('user')
tempuser = User.objects.get(id=user_id)

```

```

student=student_profile(full_name=full_name, Roll_no=Roll_no, birthdate=birthdate, image=image, annualin=annualin, mobile_no=mobil_no, address=address, bloodgroup=bloodgroup, familymember=familymember, fatheroccupation=fatheroccupation, maths=maths, algo=algo, operating=operating, micro=micro, dbms=dbms, health=health, mentor=mentor, travel=travel, cgpa=cgpa, language=language, areaofinterest=areaofinterest, internship=internship, connect=tempuser)

```

```

if student_profile.objects.filter(Roll_no=Roll_no).exists():

```

```

    messages.info(request, "roll no already exists")

```

```

    return redirect('/')

```

else:

student.save()

print('request is', request.POST)

messages.success(request,"data successfully uploded ")

return redirect('student_profiler', id=user_id)

else:

if student_profile.objects.filter(connect_id= request.user.id).exists():

return redirect('student_profiler/' + str(request.user.id))

else:

return render(request,'home.html')

faculty page where he will see is roll number

login required is django inbuild decorator

@login_required()

@allowed_users(allowed_roles=['admin','staff'])

def faculty(request):

profile= student_profile.objects.all().order_by('Roll_no')

return render(request,'faculty.html', {'profile':profile})

this functions is used to view student personal profile

@login_required()

@allowed_users(allowed_roles=['admin','staff'])

def faculty_view(request, Roll_no):

profile= student_profile.objects.filter(Roll_no=Roll_no).first()

return render(request,'faculty_view.html', {'profile': profile})

```

# profiler
@login_required()
def student_profiler(request, id):
    if request.user.is_staff:
        return HttpResponse("Since you are faculty you cannot view this page")
    else:
        profile= student_profile.objects.filter(connect_id=id).first()
        # user=User.objects.get(id=id)
        if profile.connect_id==request.user.id:
            print(profile)
            return render(request,'student_profiler.html', {'profile': profile })
        else:
            return HttpResponse("you are not authorized")

```

```

def logout_user(request):
    logout(request)
    return redirect('login')

```

Login page (student)

@unauthenticated_user

```

def login1(request):
    if request.user.is_authenticated:
        if request.user.is_staff:
            return redirect('faculty')
        else:
            return redirect('/')
    else:
        if request.method == 'POST':
            username = request.POST['username']
            password = request.POST['password']

```

```
user = authenticate(request,username=username,password=password)
```

```
if user is None:
```

```
    messages.error(request,'Invalid Credentials')
```

```
    return redirect('login')
```

```
else :
```

```
    if user.is_staff:
```

```
        auth.login(request,user)
```

```
        return redirect('faculty')
```

```
    else:
```

```
        auth.login(request,user)
```

```
        return redirect("/")
```

```
else:
```

```
    return render(request,'login.html')
```

```
def register(request):
```

```
    if request.user.is_authenticated:
```

```
        return redirect('/')
```

```
    else:
```

```
        if request.method == 'POST':
```

```
            first_name = request.POST['first_name']
```

```
            last_name = request.POST['last_name']
```

```
            username = request.POST['username']
```

```
            password1 = request.POST['password1']
```

```
            password2 = request.POST['password2']
```

```
            email = request.POST['email']
```

```
            if password1 == password2:
```

```
                if User.objects.filter(username=username).exists():
```

```
                    messages.info(request,"Username Already taken")
```

```
                    return redirect('register')
```

```
elif User.objects.filter(email=email).exists():
    messages.info(request,"Email Already Taken")
    return redirect('register')
else:
    user =
User.objects.create_user(username=username,password=password1,email=email,first_name=first_
name,last_name=last_name)
    user.save();
    messages.success(request,'User created')
    print("user creator")
    return redirect('login')
else:
    messages.info(request,"Password not matching")
    return redirect('register')
return redirect('/')
else:
    return render(request,'register.html')
```