**Source Code**

**url.py:**

path('userlogin/',user.userlogin,name='userlogin'),

path('userregister/',user.userregister,name='userregister'),

path('userlogincheck/',user.userlogincheck,name='userlogincheck'),

path('pagerank',user.pagerank,name='pagerank'),

path('search/',user.search, name="search"),

path('search1/',user.search1, name="search1"),

path('usersearchresult/',user.usersearchresult, name="usersearchresult"),

path('usersearchresult1/',user.usersearchresult1, name="usersearchresult1"),

path('weight/', user.weight, name="weight"),

path('logout/',user.logout,name='logout'),

path('managerlogin/',manager.managerlogin,name='managerlogin'),

path('managerregister/',manager.managerregister,name='managerregister'),

path('managerlogincheck/',manager.managerlogincheck,name='managerlogincheck'),

path('fileupload/', manager.fileupload, name='fileupload'),

path('admin1/',search.adminlogin,name='admin1'),

path('adminloginentered/',search.adminloginentered,name='adminloginentered'),

path('userdetails/',search.userdetails,name='userdetails'),

path('Managerdetails/',search.managerdetails,name='Managerdetails'),

path('activateuser/',search.activateuser,name='activateuser'),

path('activatemanager/',search.activatemanager,name='activatemanager'),

**views.py:**

def managerlogin(request):

return render(request,'manager/managerlogin.html')

def managerregister(request):

if request.method=='POST':

form1=managerForm(request.POST)

if form1.is\_valid():

form1.save()

print("succesfully saved the data")

return render(request, 'manager/managerlogin.html')

#return HttpResponse("registreration succesfully completed")

else:

print("form not valied")

return HttpResponse("form not valied")

else:

form=managerForm()

return render(request,"manager/managerregister.html",{"form":form})

def managerlogincheck(request):

if request.method == 'POST':

sname = request.POST.get('email')

print(sname)

spasswd = request.POST.get('upasswd')

print(spasswd)

try:

check = managerModel.objects.get(email=sname,passwd=spasswd)

# print('usid',usid,'pswd',pswd)

print(check)

# request.session['name'] = check.name

# print("name",check.name)

status = check.status

print('status',status)

if status == "Activated":

request.session['email'] = check.email

return render(request, 'manager/managerpage.html')

else:

messages.success(request, 'manager is not activated')

return render(request, 'manager/managerlogin.html')

except Exception as e:

print('Exception is ',str(e))

pass

messages.success(request,'Invalid name and password')

return render(request,'manager/managerlogin.html')

**models.py:**

from django.db import models

class userModel(models.Model):

name = models.CharField(max\_length=50)

email = models.EmailField()

passwd = models.CharField(max\_length=40)

cwpasswd = models.CharField(max\_length=40)

mobileno = models.CharField(max\_length=50, default="", editable=True)

status = models.CharField(max\_length=40, default="", editable=True)

def \_\_str\_\_(self):

return self.email

class Meta:

db\_table='userregister'

class weightmodel(models.Model):

filename = models.CharField(max\_length=100)

file = models.FileField(upload\_to='files/pdfs/')

weight=models.CharField(max\_length=100)

rank=models.CharField(max\_length=100,default="", editable=False)

label=models.CharField(max\_length=100,default="", editable=False)

def \_\_str\_\_(self):

return self.filename

class Meta:

db\_table='weight'

**forms.py:**

from django import forms

from user.models import \*

from django.core import validators

class userForm(forms.ModelForm):

name = forms.CharField(widget=forms.TextInput(), required=True, max\_length=100,)

passwd = forms.CharField(widget=forms.PasswordInput(), required=True, max\_length=100)

cwpasswd = forms.CharField(widget=forms.PasswordInput(), required=True, max\_length=100)

email = forms.CharField(widget=forms.TextInput(),required=True)

mobileno= forms.CharField(widget=forms.TextInput(), required=True, max\_length=10,validators=[validators.MaxLengthValidator(10),validators.MinLengthValidator(10)])

status = forms.CharField(widget=forms.HiddenInput(), initial='waiting', max\_length=100)

def \_\_str\_\_(self):

return self.email

class Meta:

model=userModel

fields=['name','passwd','cwpasswd','email','mobileno','status']

**userdetails.html:**

{% extends 'adminbase.html' %}

{% load static %}

{% block contents %}

<div class="modal fade" id="login" role="dialog">

<div class="modal-dialog modal-sm">

<!-- Modal content no 1-->

<div class="modal-content">

<div class="modal-header">

<button type="button" class="close" data-dismiss="modal">&times;</button>

<h4 class="modal-title text-center form-title">Login</h4>

</div>

<div class="modal-body padtrbl">

<div class="login-box-body">

<p class="login-box-msg">Sign in to start your session</p>

<div class="form-group">

<form name="" id="loginForm">

<div class="form-group has-feedback">

<!----- username -------------->

<input class="form-control" placeholder="Username" id="loginid" type="text" autocomplete="off" />

<span style="display:none;font-weight:bold; position:absolute;color: red;position: absolute;padding:4px;font-size: 11px;background-color:rgba(128, 128, 128, 0.26);z-index: 17; right: 27px; top: 5px;" id="span\_loginid"></span>

<!---Alredy exists ! -->

<span class="glyphicon glyphicon-user form-control-feedback"></span>

</div>

<div class="form-group has-feedback">

<!----- password -------------->

<input class="form-control" placeholder="Password" id="loginpsw" type="password" autocomplete="off" />

<span style="display:none;font-weight:bold; position:absolute;color: grey;position: absolute;padding:4px;font-size: 11px;background-color:rgba(128, 128, 128, 0.26);z-index: 17; right: 27px; top: 5px;" id="span\_loginpsw"></span>

<!---Alredy exists ! -->

<span class="glyphicon glyphicon-lock form-control-feedback"></span>

</div>

<div class="row">

<div class="col-xs-12">

<div class="checkbox icheck">

<label>

<input type="checkbox" id="loginrem" > Remember Me

</label>

</div>

</div>

<div class="col-xs-12">

<button type="button" class="btn btn-green btn-block btn-flat" onclick="userlogin()">Sign In</button>

</div>

</div>

</form>

</div>

</div>

</div>

</div>

</div>

</div>

<!--/ Modal box-->

<!--Banner-->

<div class="banner">

<div class="bg-color">

<div class="container">

<div class="row">

<div class="banner-text text-center">

<div class="text-border">

<!--<h2 class="text-dec"></h2>-->

</div>

<div class="intro-para text-center quote">

<p>

Welcome admin page...

</p>

<center><h3>

<table border="2px solid red" align="left">

<tr><th style="color:green">Id</th>

<th style="color:green">name</th>

<th style="color:green">email</th>

<th style="color:green">mobileno</th>

<th style="color:green">status</th>

<th style="color:green">activate</th>

</tr>

{% for x in qs %}

<tr>

<td style="color:red">{{x.id}}</td>

<td style="color:red">{{x.name}}</td>

<td style="color:red">{{x.email}}</td>

<td style="color:red">{{x.mobileno}}</td>

<td style="color:red">{{x.status}}</td>

{% if x.status == 'waiting' %}

<td style="color:orange"> <a href="/activateuser/?pid={{ x.id }}" >Activate</a></td>

{% else %}

<td style="color:orange"> Activated</td>

{% endif %}

</tr>

{% endfor %}

</table>

</h3>

</center>

</div>

</a>

</div>

</div>

</div>

</div>

</div>-->

{% endblock %}