

SPRINT-2

Date	16-11-2022
Team ID	PNT2022TMID17383
Project Name	Smart Farmer - IoT Smart farming Application

1 . A form is created for the user registration (register.html):

```
<form action="/register/" method="POST"> {% csrf_token %}
  <div class="container mt-5">
    <div class="mb-3">
      <label for="username" class="form-label">Username</label>
      <input type="text" class="form-control" id="username"
        name="username">
    </div>
    <div class="mb-3">
      <label for="first_nam" class="form-label">First Name</label>
      <input type="text" class="form-control" id="first_name"
        name="first_name">
    </div>
    <div class="mb-3">
      <label for="last_name" class="form-label">Last Name</label>
      <input type="text" class="form-control" id="last_name"
        name="last_name">
    </div>
```

```

<div class="mb-3">
    <label for="email" class="form-label">Email address</label>
    <input type="email" class="form-control" id="email" name="email">
</div>

<div class="mb-3">
    <label for="password1" class="form-label">Password</label>
    <input type="password" class="form-control" id="password1"
    name="password1">
</div>

<div class="mb-3">
    <label for="password2" class="form-label">Confirm Password</label>
    <input type="password" class="form-control" id="password2"
    name="password2">
</div>

<button type="submit" class="btn btn-primary">Submit</button>
</div>
</form> Views.py :

```

```

def register(request):
    if request.method=="POST":
username =
request.POST['username'] email
= request.POST['email']
first_name=request.POST['first_
name']

```

```

last_name=request.POST['last_n
ame'] password1 =
request.POST['password1']
password2 =
request.POST['password2']

if password1 != password2:
    messages.error(request, "Passwords
do not match.") return
    redirect('/register')

user = User.objects.create_user(username, email,
password1) user.first_name = first_name
user.last_name = last_name
user.save()

return render(request,
'login.html') return
render(request,
"register.html")

```

Code Explanation:

We create a form with a post request to take the required details from the user. These are then stored in the Django User model. These details will be further required to identify the user while logging in.

2.Then a form is created for the user login (login.html):

```
<form action="/login/" method="POST"> {% csrf_token %}
<div class="container mt-5">
<div class="mb-3">
  <label for="username" class="form-label">Username</label>
  <input type="text" class="form-control" id="username"
    name="username">
</div>
<div class="mb-3">
  <label for="password" class="form-label">Password</label>
  <input type="password" class="form-control" id="password"
    name="password">
</div>
<br>
<button type="submit" class="btn btn-primary">Submit</button>
</div>
</form>
```

Views.py :

```
def
loggedin(request)
:
if
request.user.is_authenticated
```

```
d: return redirect("/")
```

```
else:
```

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

```
        username =
```

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

```
        password =
```

```
        request.POST['password']
```

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

```
        if user is not None:
```

```
            login(request, user)
```

```
            messages.success(requ
```

```
                est, "Successfully
```

```
                Logged In") return
```

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

```
        else:
```

```
            messages.error(request,
```

```
                "Invalid Credentials") return
```

```
            render(request, 'college.html')
```

```
    return render(request,
```

```
        "login.html")
```

Code Explanation:

If the user wants to fill the application form, then the user has to register and then login. While login if the username and password is wrong then “Invalid Credentials” message will be shown else the user will get successfully logged in.