

USER AUTHENTICATION

1. PROJECT : CREATE A WEB APPLICATION PROJECT WITH USER AUTHENTICATION

1. Create Project and Application.

```
C:\Windows\system32\cmd.exe
Microsoft Windows [Version 10.0.18363.1556]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\Reception1>d:
D:\>cd Django
D:\Django>django-admin startproject userauth1
D:\Django>cd userauth1
D:\Django\userauth1>python manage.py startapp userapp
D:\Django\userauth1>
```

2. Add Application and Templates name in Settings file.

```
settings.py
from pathlib import Path
import os
# Build paths inside the project like this: BASE_DIR / 'subdir'.
BASE_DIR = Path(__file__).resolve().parent.parent
T_dir=os.path.join(BASE_DIR,'templates')

# Quick-start development settings - unsuitable for production
# See https://docs.djangoproject.com/en/4.0/howto/deployment/checklist/

# SECURITY WARNING: keep the secret key used in production secret!
SECRET_KEY = 'django-insecure-a^rxofhbhx#se6ikph+kfd28a#4ibf_+dh7i68v06rt35(yp(h'

# SECURITY WARNING: don't run with debug turned on in production!
DEBUG = True

ALLOWED_HOSTS = []

INSTALLED_APPS = [
    'django.contrib.admin',
    'django.contrib.auth',
    'django.contrib.contenttypes',
    'django.contrib.sessions',
    'django.contrib.messages',
    'django.contrib.staticfiles',
    'userapp'
]
```

Activate
Quick Settings

----- WEB DEVELOPMENT -----

```
TEMPLATES = [
{
    'BACKEND': 'django.template.backends.django.DjangoTemplates',
    'DIRS': [T_dir],
    'APP_DIRS': True,
    'OPTIONS': {
        'context_processors': [
            'django.template.context_processors.debug',
            'django.template.context_processors.request',
            'django.contrib.auth.context_processors.auth',
            'django.contrib.messages.context_processors.messages',
        ],
    },
},
]
```

3. Create 5 html pages and create template inheritance.
4. Create Views file.

The screenshot shows a Sublime Text window with the following details:

- Title Bar:** D:\Django\userauth1\userapp\views.py • (userauth1) - Sublime Text (UNREGISTERED)
- Menu Bar:** File Edit Selection Find View Goto Tools Project Preferences Help
- Folders List:** A sidebar on the left shows the project structure:
 - userauth1
 - templates
 - apps
 - contact.html
 - gallery.html
 - home.html
 - index.html
 - services.html
 - userapp
 - migrations
 - __init__.py
 - admin.py
 - apps.py
 - models.py
 - tests.py
 - views.py
 - userauth1
 - __pycache__
 - __init__.py
 - asgi.py
 - settings.py
 - urls.py
 - wsgi.py
 - manage.py
 - File Content:** The main pane displays the content of views.py:

```
from django.shortcuts import render
def index(request):
    return render(request, 'apps/index.html')
def home(request):
    return render(request, 'apps/home.html')
def service(request):
    return render(request, 'apps/services.html')
def gallery(request):
    return render(request, 'apps/gallery.html')
def contact(request):
    return render(request, 'apps/contact.html')
```

----- WEB DEVELOPMENT -----

5. Create url file.

```
urls.py  
from django.contrib import admin  
from django.urls import path  
from userapp import views  
urlpatterns = [  
    path('admin/', admin.site.urls),  
    path('index/',views.index),  
    path('home/',views.home),  
    path('service/',views.service),  
    path('gallery/',views.gallery),  
    path('contact/',views.contact)  
]
```

6. Run the server and Verify the result.

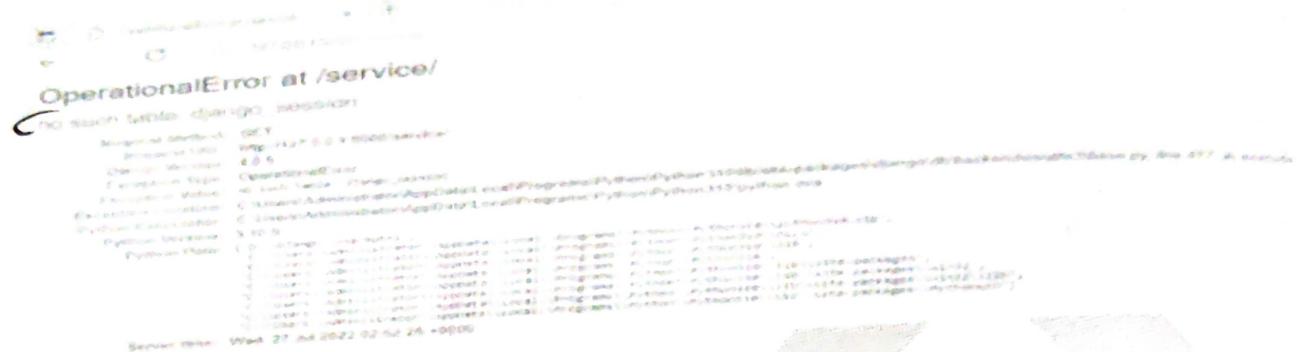
7. Type command: **from django.contrib.auth.decorators import login_required** (User auth module)

@login_required

```
from django.shortcuts import render  
from django.contrib.auth.decorators import login_required  
def index(request):  
    return render(request,'apps/index.html')  
def home(request):  
    return render(request,'apps/home.html')  
@login_required  
def service(request):  
    return render(request,'apps/services.html')  
def gallery(request):  
    return render(request,'apps/gallery.html')  
def contact(request):  
    return render(request,'apps/contact.html')
```

WEB DEVELOPMENT

8. Run the server.



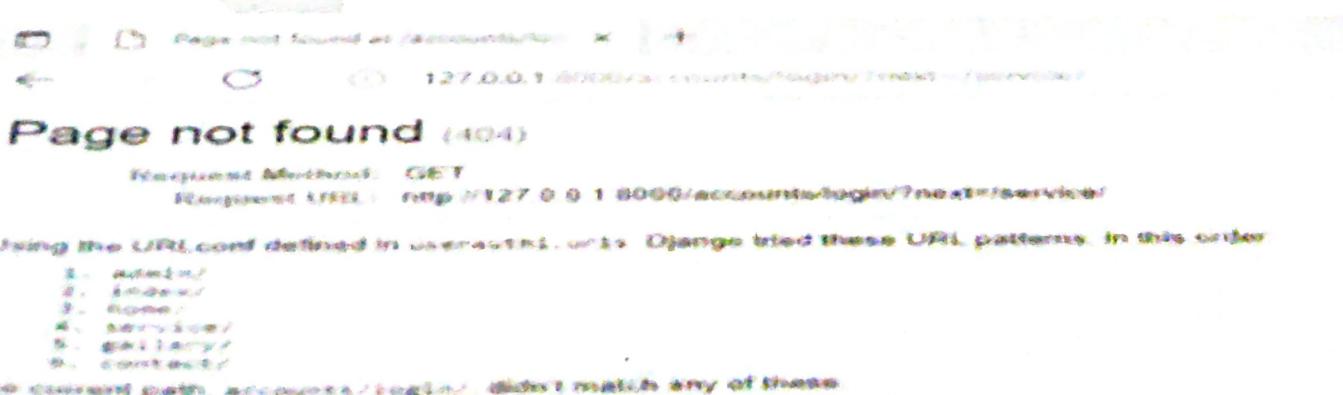
The screenshot shows a browser window with the URL `http://127.0.0.1:8000/service/`. The page displays an error message: **OperationalError at /service/**, specifically stating "no session table: django_session". It provides a detailed stack trace, mentioning `Django 3.2.10`, `django 3.2.10`, `psycopg2 2.9.3`, and `psycopg2-binary 2.9.3`. The error occurred in `django/contrib/auth/backends.py` at line 147. The error code is `OperationalError`.

9. Type commands: `python manage.py migrate`

```
C:\Windows\system32\cmd.exe
D:\Django\userauth1>python manage.py migrate
Operations to perform:
  Apply all migrations: admin, auth, contenttypes, sessions
Running migrations:
  Applying contenttypes.0001_initial... OK
  Applying auth.0001_initial... OK
  Applying admin.0001_initial... OK
  Applying admin.0002_logentry_remove_auto_add... OK
  Applying admin.0003_logentry_add_action_flag_choices... OK
  Applying contenttypes.0002_remove_content_type_name... OK
  Applying auth.0002_alter_permission_name_max_length... OK
  Applying auth.0003_alter_user_email_max_length... OK
  Applying auth.0004_alter_user_username_opts... OK
  Applying auth.0005_alter_user_last_login_null... OK
  Applying auth.0006_require_contenttypes_0002... OK
  Applying auth.0007_alter_validators_add_error_messages... OK
  Applying auth.0008_alter_user_username_max_length... OK
  Applying auth.0009_alter_user_last_name_max_length... OK
  Applying auth.0010_alter_group_name_max_length... OK
  Applying auth.0011_update_proxy_permissions... OK
  Applying auth.0012_alter_user_first_name_max_length... OK
  Applying sessions.0001_initial... OK

D:\Django\userauth1>
```

10. Run the server.



11. Type command: `from django.urls import path,include`

```
path('accounts/', include('django.contrib.auth.urls'))
```

```
from django.contrib import admin
from django.urls import path
from userapp import views
```

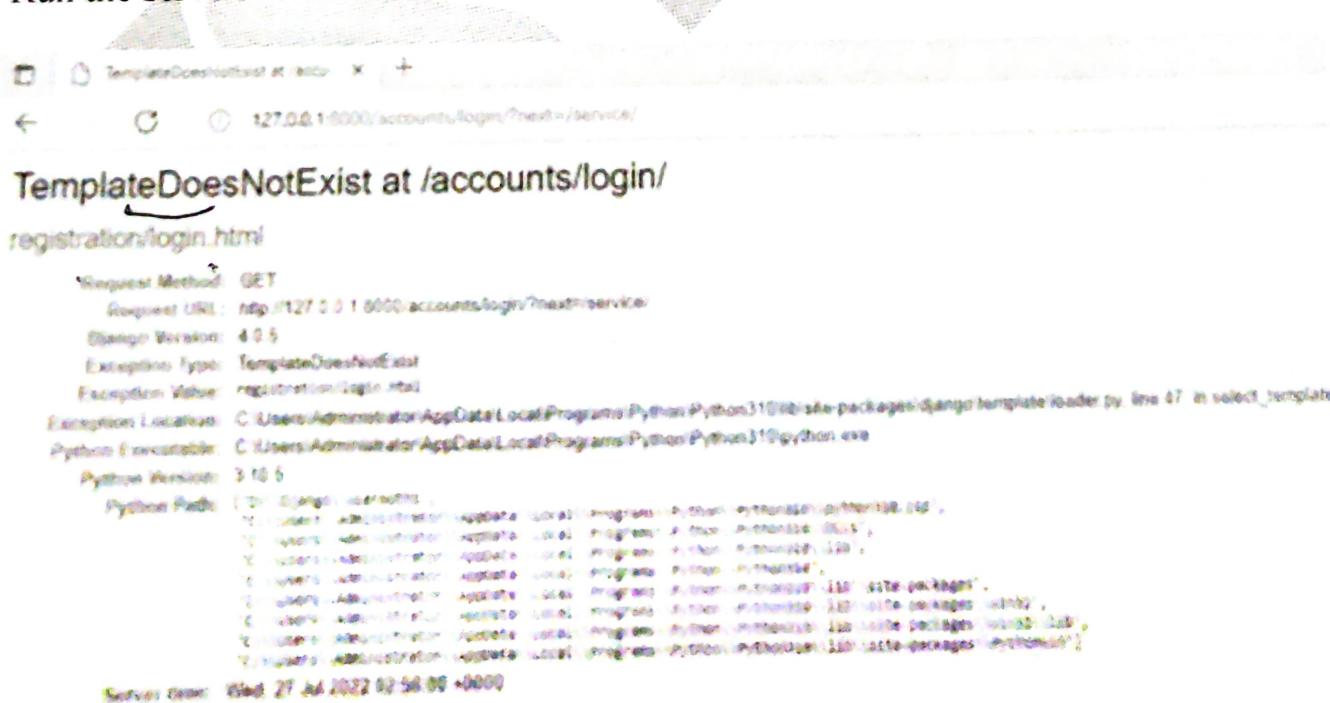
① `from django.urls import path,include`
`urlpatterns = [`

```
    path('admin/', admin.site.urls),
    path('index/',views.index),
    path('home/',views.home),
    path('service/',views.service),
    path('gallery/',views.gallery),
    path('contact/',views.contact),
```

② `path('accounts/', include('django.contrib.auth.urls'))`

]

12. Run the server.



13. Create registration folder inside templates folder.



14. Create Login.html file

```
login.html x
FolderName: registration
Line Count: 10
Title Size: 4 Plain Text

```
{%extends 'apps/index.html'%}
{%block body_block%}
<center>
 <h3>LOGIN PAGE</h3>
 <form method="POST">
 {{form.as_p}}
 {%csrf_token%}
 <button type="submit">Login</button>
 </form>
</center>
{%endblock%}
```
```

15. Type commands: `python manage.py createsuperuser`

```
C:\Windows\system32\cmd.exe

D:\Django\userauth1>python manage.py createsuperuser
Username (leave blank to use 'reception1'): admin
Email address: admin@gmail.com
Password:
Password (again):
This password is too short. It must contain at least 8 characters.
This password is too common.
This password is entirely numeric.
Bypass password validation and create user anyway? [y/N]: y
Superuser created successfully.

D:\Django\userauth1>
```

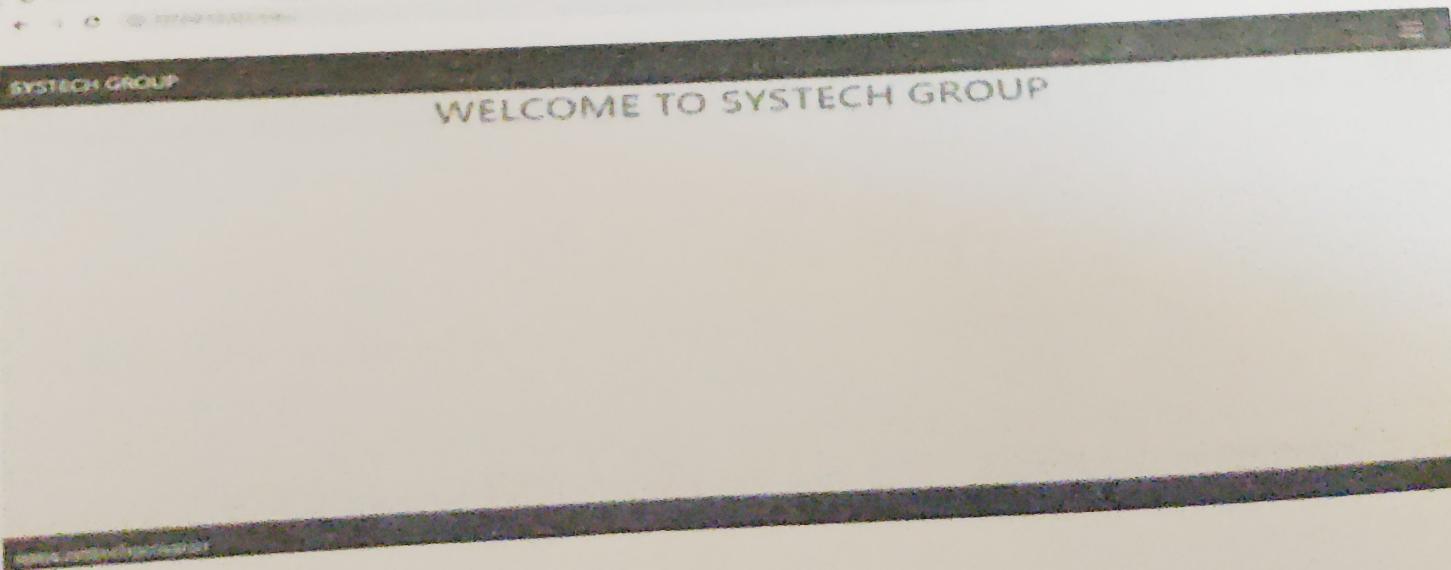
16. Run the server

```
C:\Windows\system32\cmd.exe - python manage.py runserver

D:\Django\userauth1>python manage.py runserver
Watching for file changes with StatReloader
Performing system checks...

System check identified no issues (0 silenced).
April 24, 2022 - 20:53:15
Django version 4.0.4, using settings 'userauth1.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
```

OUTPUT:



WEB DEVELOPMENT

SYSTECH GROUP

WELCOME TO SYSTECH GROUP
TRAINING COURSES
100% PRACTICAL TRAINING

DIANGO

PYTHON

C

C++

SQL

SYSTECH GROUP

WELCOME TO SYSTECH GROUP

Aruvi block,
St paul's complex,
Gharathiyar salai,
Cantonment,
Trichy - 61.

CONTACT:

www.systechgroup.in

2. PROJECT : CREATE A WEB APPLICATION PROJECT WITH USER AUTHENTICATION

1. Create Project and Application

C:\Windows\system32\cmd.exe

D:\Django>django-admin startproject userauth2

D:\Django>cd userauth2

D:\Django\userauth2>python manage.py startapp userapp

D:\Django\userauth2>

2. Add Application and Template name in Settings file

```
settings.py

from pathlib import Path
import os
# Build paths inside the project like this: BASE_DIR / 'subdir'.
BASE_DIR = Path(__file__).resolve().parent.parent
T_dir=os.path.join(BASE_DIR,'templates')

# Quick-start development settings - unsuitable for production
# See https://docs.djangoproject.com/en/4.0/howto/deployment/checklist/

# SECURITY WARNING: keep the secret key used in production secret!
SECRET_KEY = 'django-insecure-t-7j_u9uv+1m$lh=cnkr9+2ro0si4q*xp@+*vm)@dityi$|i*'

# SECURITY WARNING: don't run with debug turned on in production!
DEBUG = True

ALLOWED_HOSTS = []
INSTALLED_APPS = [
    'django.contrib.admin',
    'django.contrib.auth',
    'django.contrib.contenttypes',
    'django.contrib.sessions',
    'django.contrib.messages',
    'django.contrib.staticfiles',
    'templates'
]
```

Activate Win
Go to Settings

```
TEMPLATES = [
{
    'BACKEND': 'django.template.backends.django.DjangoTemplates',
    'DIRS': [T_dir],
    'APP_DIRS': True,
    'OPTIONS': {
        'context_processors': [
            'django.template.context_processors.debug',
            'django.template.context_processors.request',
            'django.contrib.auth.context_processors.auth',
            'django.contrib.messages.context_processors.messages',
        ],
    },
},
]
```

3. Create Views file



```
views.py      x
from django.shortcuts import render
def index(request):
    return render(request, 'apps/index.html')
def home(request):
    return render(request, 'apps/home.html')
def service(request):
    return render(request, 'apps/services.html')
def gallery(request):
    return render(request, 'apps/gallery.html')
def contact(request):
    return render(request, 'apps/contact.html')
```

4. Create url file.

```
urls.py
```

```
from django.contrib import admin
from django.urls import path
from userapp import views
urlpatterns = [
    path('admin/', admin.site.urls),
    path('index/',views.index),
    path('home/',views.home),
    path('service/',views.service),
    path('gallery/',views.gallery),
    path('contact/',views.contact)
]
```

5. Add login_required command in views.py file.

```
views.py
```

```
from django.shortcuts import render
from django.contrib.auth.decorators import login_required
def index(request):
    return render(request,'apps/index.html')
def home(request):
    return render(request,'apps/home.html')
@login_required
def service(request):
    return render(request,'apps/services.html')
def gallery(request):
    return render(request,'apps/gallery.html')
def contact(request):
    return render(request,'apps/contact.html')
```

WEB DEVELOPMENT

6. Type command: `python manage.py migrate`

```
C:\Windows\system32\cmd.exe
D:\Django\userauth2>python manage.py migrate
Operations to perform:
  Apply all migrations: admin, auth, contenttypes, sessions
Running migrations:
  Applying contenttypes.0001_initial... OK
  Applying auth.0001_initial... OK
  Applying admin.0001_initial... OK
  Applying admin.0002_logentry_remove_auto_add... OK
  Applying admin.0003_logentry_add_action_flag_choices... OK
  Applying contenttypes.0002_remove_content_type_name... OK
  Applying auth.0002.Alter_permission_name_max_length... OK
  Applying auth.0003.Alter_user_email_max_length... OK
  Applying auth.0004.Alter_user_username_opts... OK
  Applying auth.0005.Alter_user_last_login_null... OK
  Applying auth.0006.Require_contenttypes_0002... OK
  Applying auth.0007.Alter_validators_add_error_messages... OK
  Applying auth.0009.Alter_user_username_max_length... OK
  Applying auth.0010.Alter_group_name_max_length... OK
  Applying auth.0011.Update_proxy_permissions... OK
  Applying auth.0012.Alter_user_first_name_max_length... OK
  Applying sessions.0001_initial... OK
D:\Django\userauth2>
```

7. Create registration folder.



8. Create login.html file

```
login.html
{%extends 'apps/index.html'%}
{%block body_block%}
<center>
    <h3>LOGIN PAGE</h3>
<form method="POST">
    {{form.as_p}}
    {%csrf_token%}
    <button type="submit">Login</button>
</form>
</center>
{%endblock%}
```

9. Type command: **python manage.py createsuperuser**

```
C:\Windows\system32\cmd.exe
D:\Django\userauth2>python manage.py createsuperuser
Username (leave blank to use 'reception1'): admin
Email address: admin@gmail.com
Password:
Password (again):
This password is too short. It must contain at least 8 characters.
This password is too common.
This password is entirely numeric.
Bypass password validation and create user anyway? [y/N]: y
Superuser created successfully.

D:\Django\userauth2>
```

10. Add accounts path in url file

```
urls.py
from django.contrib import admin
from django.urls import path, include
from userapp import views
urlpatterns = [
    path('admin/', admin.site.urls),
    path('index/', views.index),
    path('home/', views.home),
    path('service/', views.service),
    path('gallery/', views.gallery),
    path('contact/', views.contact),
    path('accounts/', include('django.contrib.auth.urls'))
]
```

11. Type command: **python manage.py runserver**

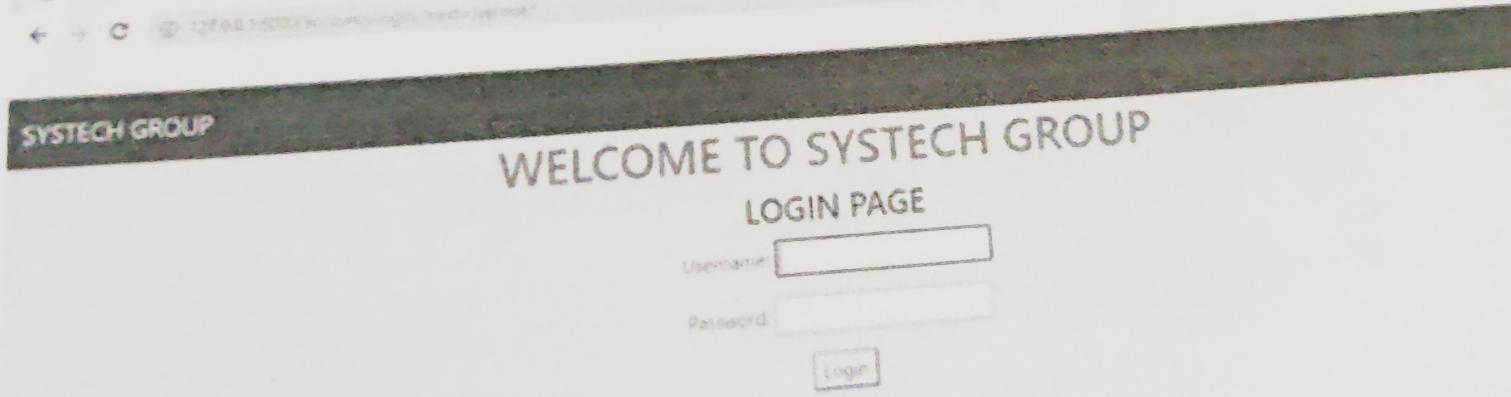
```
C:\Windows\system32\cmd.exe - python manage.py runserver
```

```
D:\Django\userauth2>python manage.py runserver
Watching for file changes with StatReloader
Performing system checks...
```

```
System check identified no issues (0 silenced).
April 24, 2022 - 21:20:28
Django version 4.0.4, using settings 'userauth2.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
```

WEB DEVELOPMENT

12. Verify the result in browser.



WELCOME TO SYSTECH GROUP

TRAINING COURSES

100% PRACTICAL TRAINING

DJANGO

PYTHON

C

C++

SQL

13. Add signup, login and logout links in index file.

```
index.html
<ul>
    </li>
    <li class="nav-item">
        <a class="nav-link" href="/gallery">Gallery</a>
    </li>
    <li class="nav-item">
        <a class="nav-link" href="/contact">Contact Us</a>
    </li>
    <li class="nav-item">
        <a class="nav-link" href="#">Signup</a>
    </li>
    <li class="nav-item">
        <a class="nav-link" href="#">Login</a>
    </li>
    <li class="nav-item">
        <a class="nav-link" href="#">Logout</a>
    </li>
</ul>
</div>
</div>
</nav>
</head>
<body style="background-color: #d5f6f7">
    <h1 style="text-align: center; size: 20; color: red">WELCOME TO SYSTECH</h1>
```



WELCOME TO SYSTECH GROUP

Hi Everyone

Systech is a leading Training Development and Placement company in Trichy. We are known for our practical approach towards training that enables students to gain real-time exposure to competitive technologies. Training is offered by employees from MNCs to give real corporate exposure to the students.

DJANGO WEB DEVELOPMENT COURSE

www.systechgroup.in

14. Add Login page in index.html

```
index.html
•
</li>
<li class="nav-item">
    <a class="nav-link" href="/gallery">Gallery</a>
</li>
<li class="nav-item">
    <a class="nav-link" href="/contact">Contact Us</a>
</li>
<li class="nav-item">
    <a class="nav-link" href="#">Signup</a>
</li>
<li class="nav-item">
    <a class="nav-link" href="/accounts/login">Login</a>
</li>
<li class="nav-item">
    <a class="nav-link" href="#">Logout</a>
</li>
</ul>
</div>
</div>
</nav>
</head>
<body style="background-color: #d5f6f7">
    <h1 style="text-align: center; size: 20; color: red">WELCOME TO SYSTECH

```

Activate Window

15. Add the command **LOGIN_REDIRECT_URL = '/home'** (home- home page)

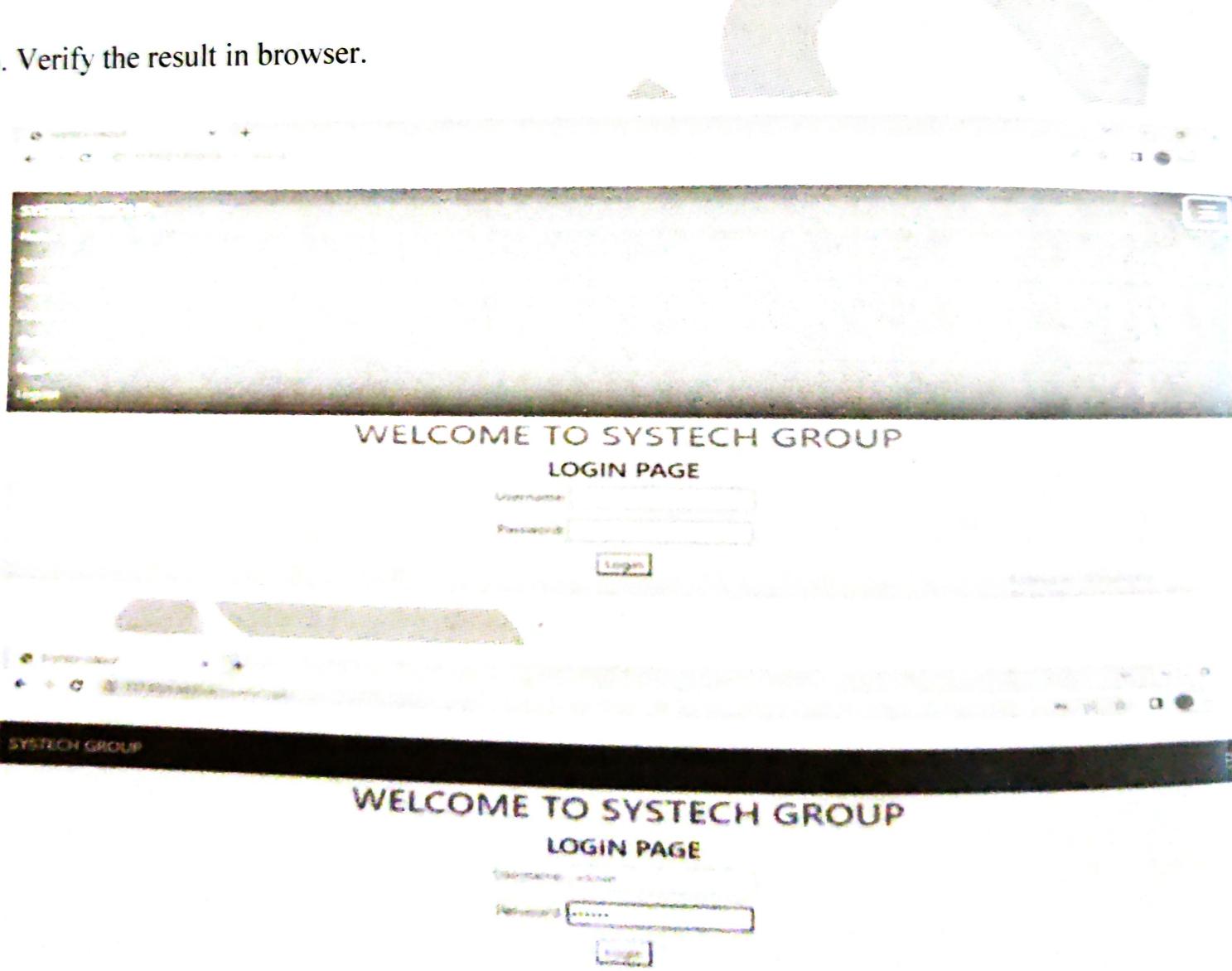
```
settings.py
STATIC_URL = 'static/'

# Default primary key field type
# https://docs.djangoproject.com/en/4.0/ref/settings/#default-auto-field

DEFAULT_AUTO_FIELD = 'django.db.models.BigAutoField'

LOGIN_REDIRECT_URL = '/home'
```

16. Verify the result in browser.



① systechgroup ② + https://systechgroup.com

SYSTECH GROUP

WELCOME TO SYSTECH GROUP

Hi Everyone

Systech is a leading Training Development and Placement company in Trichy. We are known for our practical approach towards training that enables students to gain real-time exposure to competitive technologies. Training is offered by employees from MNCs to give real corporate exposure to the students.

DJANGO WEB DEVELOPMENT COURSE

Django is a Web Development course provided by systech Trichy. It is the best framework for developing a Web application framework in Python programming language. Django follows Model View Template (MVT) Architecture which is based on the Model View Controller(MVC) Architecture. The demand for Django remains is high at all times because of its fast development feature. It can be used to build any type of website and web application.

17. Add logout page in index.html.

```
index.html
<ul>
    <li class="nav-item">
        <a class="nav-link" href="/contact">Contact Us</a>
    </li>
    <li class="nav-item">
        <a class="nav-link" href="#">Signup</a>
    </li>
    <li class="nav-item">
        <a class="nav-link" href="/accounts/login">Login</a>
    </li>
    <li class="nav-item">
        <a class="nav-link" href="/accounts/logout">Logout</a>
    </li>
</ul>
</div>
</div>
</nav>
</head>
<body style="background-color: #d5f6f7">
    <h1 style="text-align: center; size: 20; color: red">WELCOME TO SYSTECH
    GROUP </h1>
</body>
<footer style="position: fixed; padding: 10px 10px 0px 10px; bottom: 0; width: 100%">
```

Activate Windows

----- WEB DEVELOPMENT -----

18. Add Login and Logout url in Settings file.

```
settings.py
```

```
USE_TZ = True

# Static files (CSS, JavaScript, Images)
# https://docs.djangoproject.com/en/4.0/howto/static-files/

STATIC_URL = 'static/'

# Default primary key field type
# https://docs.djangoproject.com/en/4.0/ref/settings/#default-auto-field

DEFAULT_AUTO_FIELD = 'django.db.models.BigAutoField'

LOGIN_REDIRECT_URL = '/home'
LOGOUT_REDIRECT_URL = '/logout'
```

19. Create Logout.html file in templates folder

```
logout.html
```

```
{%extends 'apps/index.html'%}
{%block body_block%}
<br></br>
<center><h2>SUCCESSFULLY LOGOUT</h2>
    <br></br>
    <a href="/accounts/login">Login Here</a>
</center>

{%endblock%}
```

----- WEB DEVELOPMENT -----

20. Configure Views.py file

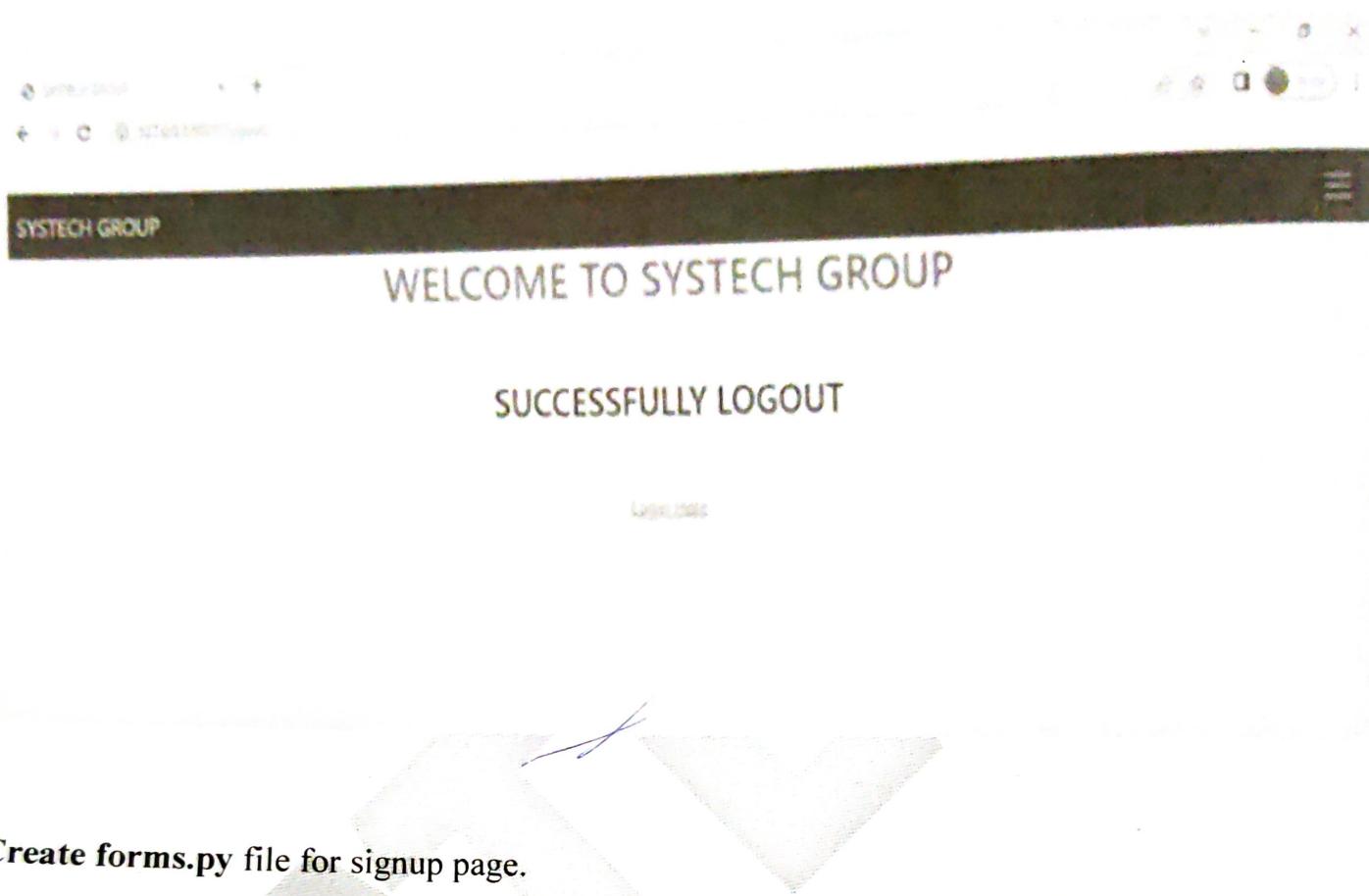
```
views.py
from django.shortcuts import render
from django.contrib.auth.decorators import login_required
def index(request):
    return render(request, 'apps/index.html')
def home(request):
    return render(request, 'apps/home.html')
@login_required
def service(request):
    return render(request, 'apps/services.html')
def gallery(request):
    return render(request, 'apps/gallery.html')
def contact(request):
    return render(request, 'apps/contact.html')
def logout(request):
    return render(request, 'apps/logout.html')
```

21. Configure url file.

```
urls.py
from django.contrib import admin
from django.urls import path, include
from userapp import views
urlpatterns = [
    path('admin/', admin.site.urls),
    path('index/', views.index),
    path('home/', views.home),
    path('service/', views.service),
    path('gallery/', views.gallery),
    path('contact/', views.contact),
    path('accounts/', include('django.contrib.auth.urls')),
    path('logout/', views.logout)
]
```

WEB DEVELOPMENT

22. Verify logout page in browser.



23. Create forms.py file for signup page.

```
forms.py
from django import forms
from django.contrib.auth.models import User #import auth module

class SignUpForm(forms.ModelForm):
    class Meta:
        model=User
        fields=['username','password','email','first_name','last_name']
```

WEB DEVELOPMENT

24. Create signup function in Views file.

```
views.py
from django.shortcuts import render
from django.contrib.auth.decorators import login_required
from userapp.forms import SignUpForm
from django.http import HttpResponseRedirect
def index(request):
    return render(request, 'apps/index.html')
def home(request):
    return render(request, 'apps/home.html')
@login_required
def service(request):
    return render(request, 'apps/services.html')
def gallery(request):
    return render(request, 'apps/gallery.html')
def contact(request):
    return render(request, 'apps/contact.html')
def logout(request):
    return render(request, 'apps/logout.html')
def signup_view(request):
    form=SignUpForm()
    if request.method=="POST":
        form=SignUpForm(request.POST)
        user=form.save()
        user.set_password(user.password)
        user.save()
        return HttpResponseRedirect('/accounts/login')
    return render(request, 'apps/signup.html', {'form':form})
```

25. Create signup.html file.

```
signup.html
{% extends 'apps/index.html'%}
{% block body_block %}
<center>
    <h2>SignUp Here</h2>
    <form method="POST">
        {{form.as_p}}
        {%csrf_token%}
        <input type="submit" value="SignUp">
    </form>
</center>
{%endblock%}
```

26. Configure url.py file.

```
urls.py
from django.contrib import admin
from django.urls import path, include
from userapp import views
urlpatterns = [
    path('admin/', admin.site.urls),
    path('index/',views.index),
    path('home/',views.home),
    path('service/',views.service),
    path('gallery/',views.gallery),
    path('contact/',views.contact),
    path('accounts/', include('django.contrib.auth.urls')),
    path('logout/',views.logout),
    path('signup/', views.signup_view)
]
```

27. Type command: **python manage.py runserver**.

```
C:\Windows\system32\cmd.exe - python manage.py runserver
```

```
D:\Django\userauth2>python manage.py runserver
Watching for file changes with StatReloader
Performing system checks...

System check identified no issues (0 silenced).
April 24, 2022 - 21:51:53
Django version 4.0.4, using settings 'userauth2.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
```

WEB DEVELOPMENT

28. Create Login.html file

```
login.html
•
{%extends 'apps/index.html'%}
{%block body_block%}
<center>
    <h3>LOGIN PAGE</h3>
<form method="POST">
    {{form.as_p}}
    {%csrf_token%}
    <button type="submit">Login</button>
</form>
<a href="apps/signup">REGISTER HERE</a>
</center>
{%endblock%}
```

29. Add signup link in index file

```
index.html
x
<a class="nav-link active" aria-current="page" href="/home">Home</a>
<a class="nav-link" href="/service">Services</a>
<a class="nav-link" href="/gallery">Gallery</a>
<a class="nav-link" href="/contact">Contact Us</a>
<a class="nav-link" href="/signup">Signup</a>
<a class="nav-link" href="/accounts/login">Login</a>
<a class="nav-link" href="/accounts/logout">Logout</a>
```

Activate Windows
Go to Settings to activate Win

WEB DEVELOPMENT

OUTPUT:

The screenshot shows a web browser window with a title bar "SYSTECH GROUP". The main content area displays a sign-up form titled "WELCOME TO SYSTECH GROUP". The form includes fields for "Username", "Password", "Email Address", "First name", and "Last name", followed by a "SignUp" button.

The screenshot shows a web browser window with a title bar "SYSTECH GROUP". The main content area displays a login page titled "WELCOME TO SYSTECH GROUP". It features fields for "Username" and "Password", a "Login" button, and a "REGISTER HERE" link.

HTML
CSS
Oracle
Ten
model
ORM
CRUD
CSV
Fern &

new
oride
sess -
osseal
preser -
[preser]