**How to Set Up Google Authentication with Django**

### **Install Django**

$ pip install django

Create a **new Django project** using the below command.

$ django-admin startproject google\_oauth .

Create a **new Django app** using the below command.

$ python manage.py startapp google\_oauth\_app

Apply the database **migrations** using the below command:

$ python manage.py migrate

Register the app by adding it to **INSTALLED\_APPS** in **settings.py**

### **Install django-allauth**

To integrate Google OAuth using django-allauth.

$ pip install django-allauth

Register **django-allauth** by adding it to **INSTALLED\_APPS**.

Setting.py

**# Insert at bottom**

AUTHENTICATION\_BACKENDS = [

'django.contrib.auth.backends.ModelBackend',

'allauth.account.auth\_backends.AuthenticationBackend'

]

**# Google as the OAuth provider**

SOCIALACCOUNT\_PROVIDERS = {

'google': {

'SCOPE': [

'profile',

'email',],

'AUTH\_PARAMS': {

'access\_type': 'online',

}

}

}

**# Add redirect users to the base route after a successfully login or logout.**

LOGIN\_REDIRECT\_URL = '/'

LOGOUT\_REDIRECT\_URL = '/'

### **Create templates**

Create new folder templates in your app base directory. And create a file **index.html** inside the templates folder.

$ mkdir templates

$ cd templates

$ type nul> index.html

**templates/index.html**

{% load socialaccount %}

<html>

<head>

<title>Google Authentication with Django</title>

</head>

<body>

<h2>How to Set Up Google Authentication with Django</h2>

{% if user.is\_authenticated %}

<p>Welcome, {{ user.username }} !</p>

<a target="\_black" href="admin">admin</a>

<a href="{% url 'logout' %}">Logout</a>

{% else %}

{% if google\_login %}

<a href="{% provider\_login\_url 'google' %}">Login with Google</a>

{% endif %}

{% endif %}

</body>

</html>

Now register this **templates** folder in the **TEMPLATES** configurations in **settings.py** file.

### **Configure URLs**

Open the app **urls.py** file.

**app/urls.py**

from django.urls import path, include from django.contrib.auth.views import LogoutView from . import views

urlpatterns = [

path('', views.index, name='google\_login'),

path('accounts/', include('allauth.urls')),

path('logout', LogoutView.as\_view(), name="logout"),

]

And open the project **urls.py** file and import **app urls**.

path('', include('google\_oauth\_app.urls')),

**views.py**

from django.shortcuts import render

def index(request):

return render(request, 'index.html')

### **Create a new Google APIs project**

Set up an OAuth application via [**Google Developer APIs Console**](https://console.developers.google.com/apis)**.**

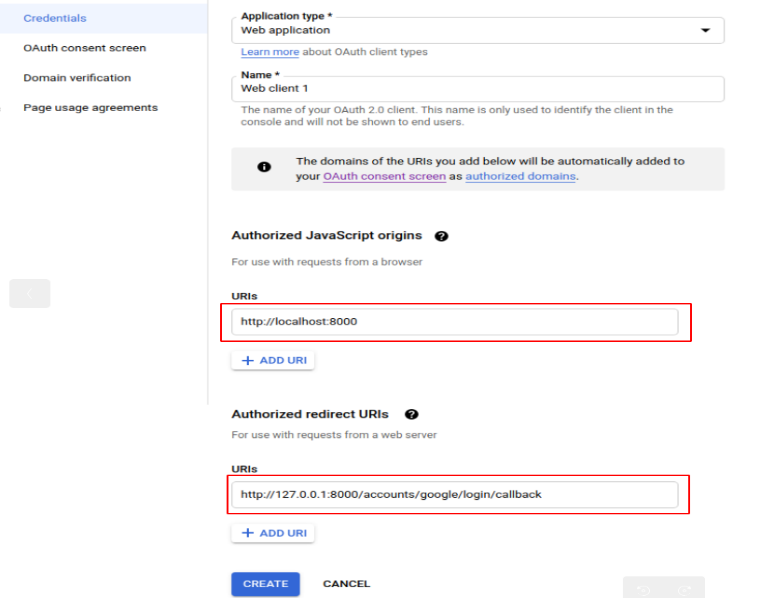
[**https://console.cloud.google.com/apis/credentials/oauthclient/628506170608-2q2788jc8kdn4v959t2vs2um9tivt5qd.apps.googleusercontent.com?project=django-auth-400105**](https://console.cloud.google.com/apis/credentials/oauthclient/628506170608-2q2788jc8kdn4v959t2vs2um9tivt5qd.apps.googleusercontent.com?project=django-auth-400105)

**Create new project**

**Add credentials**.

Click top the **dropdown**, choose **OAuth Client ID** option.

And fill the detail like below in image and hit create button



After a **successfully** OAuth create, copy Your **Client ID** and **Client Secret**.

### **Add a social app**

Migrate the database using the below command:

$ python manage.py migrate

Now create a **superuser**.

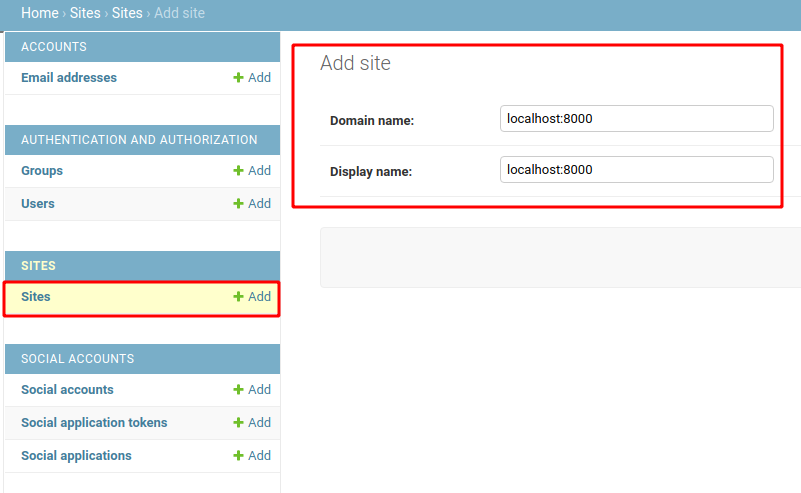
$ python manage.py createsuperuser

Run the app:

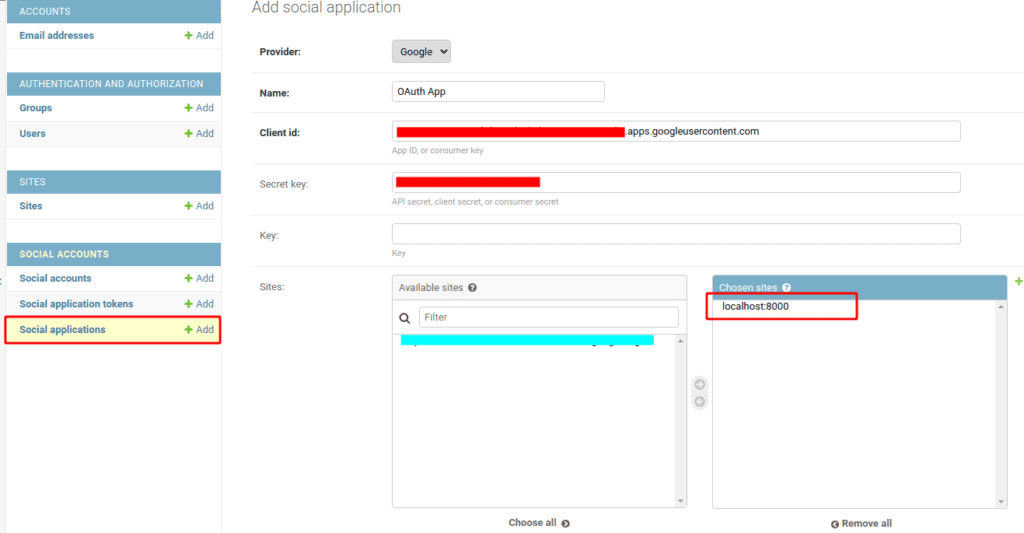
$ python manage.py runserver

Now Open [**http://localhost:8000/admin**](http://localhost:8000/admin) and log in to **Django Admin**.

Now **Sites** click Add the details and fill the detail like this:



Now **Social Applications** click Add and fill the form like below:



Now logout currently of the superuser, and log in again using your **Google account**

If you get an error: SocialApp matching query does not exist at http://127.0.0.1:8000/accounts/google/login/,

it means that the SITE\_ID of the site you created in Django admin is not the same or missing.

#### **Get site id**

$ python manage.py shell

from django.contrib.sites.models import Site

new\_site = Site.objects.create(domain='localhost:8000', name='localhost:8000')

print(new\_site.id)

Output: 1

**settings.py**

SITE\_ID = 1

You are almost done! Now navigate to (http://127.0.0.1:8000/) to see the **Login with Google**.

Further reading:

1. <https://python.plainenglish.io/proper-way-of-using-google-authentication-with-django-and-django-allauth-part-2-c47b87dd1283>

2. <https://medium.com/powered-by-django/add-social-authentication-django-allauth-google-oauth-example-d8d69a603356>

3. <https://medium.com/@20ce125/django-crud-create-retrieve-update-delete-operations-441a8a296119>