In 5 mins: Set up Google login to sign up users on Django

[Zoe Chew 🇲🇾👩‍💻📕](https://medium.com/@whizzoe?source=post_page-----e71d5c38f5d5----------------------)

Follow

[Jul 27, 2019](https://medium.com/@whizzoe/in-5-mins-set-up-google-login-to-sign-up-users-on-django-e71d5c38f5d5?source=post_page-----e71d5c38f5d5----------------------) · 6 min read

Note: This article is part of my [**toolkit newsletters**](https://whizzoe.substack.com/subscribe)↗️ where I share resources about building things.

I work with Python and Django as my tech stacks in building [**web apps**](https://ilazytoread.herokuapp.com/)↗️

Today, I want to share about building user registration through Google OAuth. I also vlog this in [**video**](https://twitter.com/whizzzoe/status/1155361813904494594?s=20)↗️

I will cover the basic setup and keep the context specific to Django app & social login with Google. Therefore, this post is ideal for any beginners.

**1. How to sign up users on your site ❤️**

Users registration is part of the customer onboarding process for tech-based products.

If you’re a beginner in programming, building a custom login might be complex to start.

Luckily, Django comes with a third-party django-allauth package that helps you set up social login in few minutes.

Google login is fast because it doesn’t require to create a new User account.

Once users access your app via Gmail, you’ll be able to see their email addresses in your backend:

**2. Create a django project 😻**

Before that happens, let’s create a new directory on desktop and call it “google\_login”.

In the new directory, you will need to set up virtual environment, install django, create django project (my\_project) and create an app (social\_app).

On terminal, use these command lines to proceed:

virtualenv .  
./Scripts/activate  
pip install django  
django-admin.py startproject my\_project .  
python manage.py startapp social\_app  
python manage.py runserver  
python manage.py migrate  
python manage.py runserver

To check if your setup works, go to (http://127.0.0.1:8000/) in your browser. You should see the default Django page:

**3. Install django-allauth 🌲**

The django-allauth package will handle authentication that has to do with user’s social account access.

Quit your server with CTRL+C in the terminal to continue next step.

On terminal, use this command to install:

pip install django-allauth

**4. Configure settings.py 💎**

Next, you’ll need to configure settings.py file in the my\_project folder.

* First, update the INSTALLED\_APPS section:

INSTALLED\_APPS = [  
 ‘django.contrib.admin’,  
 ‘django.contrib.auth’,  
 ‘django.contrib.contenttypes’,  
 ‘django.contrib.sessions’,  
 ‘django.contrib.messages’,  
 ‘django.contrib.staticfiles’,  
 ‘django.contrib.sites’, *# <--*  
 ‘social\_app’, *# <--*  
   
 ‘allauth’, *# <--*  
 ‘allauth.account’, *# <--*  
 ‘allauth.socialaccount’, *# <--*  
 ‘allauth.socialaccount.providers.google’, *# <--*  
]

* At the bottom of settings.py we need to specify the allauth backend:

AUTHENTICATION\_BACKENDS = (  
 ‘django.contrib.auth.backends.ModelBackend’,  
 ‘allauth.account.auth\_backends.AuthenticationBackend’,  
 )

* Continue to add site\_id and specify redirect URL upon successful Google login:

SITE\_ID = 1LOGIN\_REDIRECT\_URL = ‘/’

* Finally, enable email scope to receive user’s email addresses after successful social login:

SOCIALACCOUNT\_PROVIDERS = {  
 'google': {  
 'SCOPE': [  
 'profile',  
 'email',  
 ],  
 'AUTH\_PARAMS': {  
 'access\_type': 'online',  
 }  
 }  
}

**5. Create templates folder 📂**

You’ll need to display a homepage with a link that shows “Sign in with Google”.

* Set up the Django template folders following the tree level below. You’ll also need to create a new html file index.html:

── social\_app  
 ── \_\_pycache\_\_  
 ── migrations  
 ── templates *# <-- New Folder*  
 ── social\_app *# <-- New Folder* ── index.html *# <-- New File*

* Edit index.html file by adding necessary title, links and tags:

{% load socialaccount %}  
<html>  
 <head>  
 <title>Google Registration</title>  
 </head>  
 <body>  
 <h1>My Google Login Project</h1>  
 {% if user.is\_authenticated %}  
 <p>Welcome, {{ user.username }} !</p>  
   
 {% else %}  
 <h1>My Google Login Project</h1>  
 <a href="{% provider\_login\_url 'google' %}">Login with Google</a>  
 {% endif %}  
  
 </body>  
</html>

**6. Configure urls.py 🔁**

Go to urls.py file of your my\_project directory.

* Add the allauth urls and specify “include” on top of import:

from django.contrib import admin  
from django.urls import path, include *# <--*urlpatterns = [  
 path(‘admin/’, admin.site.urls),  
 path(‘accounts/’, include(‘allauth.urls’)), *# <--*  
]

* Configure Django templates and link to the urls:

from django.contrib import admin  
from django.urls import path, include  
from django.views.generic import TemplateView *# <--*urlpatterns = [  
 path(‘’, TemplateView.as\_view(template\_name=”social\_app/index.html”)), *# <--*  
 path(‘admin/’, admin.site.urls),  
 path(‘accounts/’, include(‘allauth.urls’)),  
]

* Now, make all the necessary migrations for the changes you’ve made:

(my\_project) $ python manage.py makemigrations  
(my\_project) $ python manage.py migrate

**7. Google API console🚀**

To add Google login on your app, you’ll need to set up OAuth application via [Google Developers Console](https://console.developers.google.com/)

**Getting started**

* Go to Dashboard, create a NEW PROJECT
* Name your new project, preferably your website or app name. User will be able to see this project name when we redirect them to Google login page.
* Click “CREATE” to proceed.

**APIs Credentials**

* Back to “Dashboard”, go to “Credentials” on left panel.
* Create credentials. On the dropdown, choose “OAuth Client ID” option.

**OAuth consent screen**

* Make sure you fill out the “OAuth Consent Screen” form.
* You’ll only need to provide “Application name”, “Email” and click “SAVE”.

**Create OAuth client ID**

* Now, you can create your OAuth Client ID by filling out these details;
* Authorized Javascript origins (http://127.0.0.1:8000)
* Authorized redirect URL (http://127.0.0.1:8000/accounts/google/login/callback/)

**Obtain OAuth client**

* Once you click “CREATE”, you will be able to obtain your “client ID” and “client Secret”.
* You’ll need this information to proceed the next steps

**8. Django admin 👮‍♀️**

We’ll need to configure Django admin. To access the admin panel, you’ll need to create a superuser to login. Type this command on terminal:

(my\_project) $ python manage.py createsuperuser

You’re required to provide “username”, “email” and “password” in the terminal. Once you’re done, proceed to start the server:

(my\_project) $ python manage.py runserver

Go to (http://127.0.0.1:8000/admin) to access your admin page. Make sure you provide the credentials to login.

After successful login, your admin page should look like this:

**Add a site**

On the SITES section, click “sites” and fill out the details and click “Save”:

* **Domain name:**127.0.0.1:8000
* **Display name:**127.0.0.1:8000

**Add social applications**

Back to admin homepage, under “SOCIAL ACCOUNTS” section, click “Social applications” to fill out these settings:

* **Provider:**Google
* **Name:**Google API
* **Client id:**(refer step 7, your OAuth details)
* **Secret key:**(refer step 7, your OAuth details)

**9. See the result 👏**

You’re almost done! Now navigate to (http://127.0.0.1:8000/) to see the “Sign in with Google” page.

Try to access the Google login, you should be able to see the redirect screen:

After successfully signed in with Google, you should be able to see your Google account name on the homepage:

Try to sign in with another Google account via (http://127.0.0.1:8000/accounts/google/login/), you’ll see the user name changed accordingly:

**10. Where to find me? 👉**

**[1]** If you love this article & feeling generous, [**send me a coffee**](https://www.buymeacoffee.com/whizzoe?source=post_page---------------------------)? ☕

**[2]** Find me on [**Personal Site**](https://whizzoe.com/) / [**Twitter**](https://twitter.com/whizzzoe/) / [**LinkedIn**](https://www.linkedin.com/in/whizzoe/) 🔥

**[3]** In case you miss out, I can send my upcoming resources to your inbox — [**Click here to join my newsletter**](https://whizzoe.substack.com/subscribe?source=post_page---------------------------) ↗️