

User documentaions



Google OAuth credentials

```
SOCIAL_AUTH_GOOGLE_OAUTH2_KEY = "126474618620-  
l23p7893eu8uprgcfcuon1fsqvl1087g.apps.googleusercontent.com"
```

Git : <https://github.com/anishad6/webscrapings>

Project Setup & User Guide

This project is a **Django-based web app** that supports:

-  Google OAuth Authentication (via social-auth-app-django and Google Sign-In token verification)
-  API endpoints for authentication and scraping

1. Requirements

Prerequisites

- Python **3.9+**
- pip
- Virtualenv
- Git
- Google OAuth2 Client ID (from Google Cloud Console)

2. Project Setup

Clone the Repository

git clone <https://github.com/anishad6/webscrapings>

cd project-folder

Create Virtual Environment

```
python -m venv venv
```

```
venv\Scripts\activate
```

Install Dependencies

User documentations

```
pip install -r requirements.txt
```

Main dependencies:

```
pip install django djangorestframework requests beautifulsoup4 social-auth-app-django  
google-auth
```

3. Environment Variables

Create a .env file in your project root:

```
SECRET_KEY=your-django-secret-key
```

```
DEBUG=True
```

```
ALLOWED_HOSTS=localhost,127.0.0.1
```

```
GOOGLE_CLIENT_ID=your-google-client-id.apps.googleusercontent.com
```

4. Database Setup

Run migrations:

```
python manage.py makemigrations
```

```
python manage.py migrate
```

Create a superuser (admin):

```
python manage.py createsuperuser
```

5. Run the Server

```
python manage.py runserver
```

Now open: <http://127.0.0.1:8000/>

6. API Endpoints

Authentication

Endpoint	Method	Description
/api/auth/google/	POST	Google Sign-In (with credential token)

User documentaions

Endpoint	Method	Description
/api/auth/login/google/	GET	Start Google OAuth login flow
/api/auth/logout/	POST	Logout user
/api/auth/check/	GET	Check if user is authenticated
POST /api/auth/google/		

```
{  
  "credential": "<google-id-token>"  
}
```

CSRF Token

Endpoint	Method	Description
----------	--------	-------------

/api/csrf/	GET	Get CSRF token for frontend requests
------------	-----	--------------------------------------

News Scraping

Endpoint	Method	Description
----------	--------	-------------

/api/scrape/	POST	Scrape top BBC news articles
--------------	------	------------------------------

Output {

```
"articles": [  
  {  
    "title": "Breaking News Example",  
    "url": "https://www.bbc.com/news/example",  
    "summary": "This is a test article."  
  }  
]
```