



API TUMBLR

Stefany González Véliz



Descripción General

- **Tumblr** es una plataforma de microblogueo que permite a sus usuarios publicar textos, imágenes, vídeos, enlaces, citas y audio a manera de tumblelog.
- Con sede en Manhattan (Nueva York, Estados Unidos), fue fundada por David Karp en 2007. Sus usuarios pueden «seguir» a otros usuarios registrados y ver las entradas de estos conjuntamente con las suyas, por lo cual *Tumblr* es considerada ser una *red social*.
- Su servicio enfatiza la facilidad de uso y la personalización; es muy sencilla de personalizar y ofrece diversas opciones de temas, así como la opción de poder ponerle reproductor de música, gracias a que permite incrustar código HTML y Javascript.

Licencias

- Como Suscriptor de los Servicios, Tumblr concede una licencia de ámbito mundial, revocable, no exclusiva, no sublicenciable y no transferible para descargar el contenido, almacenarlo, verlo, mostrarlo, representarlo, redistribuirlo y crear obras derivativas de él, únicamente en asociación con el uso que hagas de los Servicios, y en estricto acuerdo con las funcionalidades y restricciones de estos (lo que incluye, entre otros, los Servicios de pago). Esto significa que, por ejemplo, se concede licencia sobre el Contenido para rebloguearlo.
- COSTOS: Ofrece un servicio gratuito, pero con publicidad.

Detalles técnicos

La API Web se accede a los recursos de datos mediante solicitudes HTTPS estándar en formato UTF-8. Siempre que sea posible, la API se esfuerza por utilizar verbos HTTP apropiados para cada acción:

- POST, se utiliza para recuperar recursos.
- GET, se utiliza para crear recursos.
- PUT, se utiliza para cambiar/reemplazar recursos o colecciones.

Descripción App

- App simple, para buscar imágenes (gif, jpg o png) dentro de tumblr sin necesidad de cargar todos los blogs e información de estos.

Código python

```
# coding: utf-8
from flask import Flask, render_template , request , url_for, g, flash, redirect
import json, pyumblr, time

search=""
data=[]

app = Flask(__name__)
client = pyumblr.TumblrRestClient('3woTLqDJZ5zXB1RJphQbT96W75gz4jKR0E6LookITY1fXCjhAu')
# API KEY para acceder a la información

@app.route("/", methods=['GET', 'POST'])
def home():
    global search,data
    if request.method=="POST":
        search = request.form.get("search", None)
        dic = client.tagged(search,before=time.time())
        dic=dic+client.tagged(search,before=time.time()-100000)
        dic=dic+client.tagged(search,before=time.time()-200000)
        dic=dic+client.tagged(search,before=time.time()-300000)
        dic=dic+client.tagged(search,before=time.time()-400000)
        print ("len(dic)")
        data=[]
        for each in dic:
            if "photos" in each:
                data.append(each["photos"][0]['original_size']['url'])
        return redirect(url_for('results'))
    return render_template("home.html")
```

```
@app.route("/results")
def results():
    global search,data
    l=len(data)
    d1=[]
    d2=[]
    d3=[]
    d4=[]
    count=-1
    for each in data:
        if count<(l/4):
            d1.append(each)
        elif count<(l/2):
            d2.append(each)
        elif count<(l*3/4):
            d3.append(each)
        else:
            d4.append(each)
        count+=1
    return render_template("resultados.html",search=search,d1=d1,d2=d2,d3=d3,d4=d4)
if __name__ == "__main__":
    app.run()
```

Screenshots de aplicación web

