

Diseño de Aplicaciones para Internet (2014-2015)

Guión de Prácticas 4(c):

Utilización de Servicios On-line:

Twitter

S. Alonso, J.M. Guirao
zerjioi@ugr.es, jmguirao@ugr.es

Resumen

Twitter [1] es el famoso sistema que permite comunicar ideas, frases, situaciones, etc. utilizando (fundamentalmente) mensajes de un máximo de 140 caracteres. En esta práctica vamos a utilizar las herramientas que nos proporciona este servicio para integrarlo en nuestra página web.

1. Introducción e Integración Sencilla

Antes de lanzarnos a programar e implementar algún servicio de Twitter en nuestra página web debemos familiarizarnos un poco con los servicios que ofrecen, así como darnos de alta como usuarios de la plataforma.

También es interesante leer un poco acerca de la plataforma Twitter [2]. Debemos familiarizarnos con los conceptos: *Tweet*, *User*, *Entity* y *Place*.

Twitter ofrece una manera muy simple con la que podemos empezar a incrustar el servicio en nuestra página web. En la página *Twitter for Websites* [3] se nos ofrecen algunas herramientas muy sencillas con las que podemos incrustar el “típico” botón de Twitter o incluso una caja de Tweets dinámicos:



Compruebe que efectivamente puede incrustar ambos tipos de objetos en su página web simplemente copiando el código HTML en el sitio oportuno.

2. Python Twitter

Tweepy [4] es una biblioteca que permite acceder a la API de *Twitter* desde el lenguaje de programación Python.

En esta práctica (muy abierta) debemos utilizar esta biblioteca para acceder a contenidos publicados en Twitter y mostrarlos de la manera que consideremos oportuna. Por ejemplo, podemos mostrar los tweets que respondan a una búsqueda, los que se hayan producido cerca de unas coordenadas concretas, etc.

Para utilizar la biblioteca debemos instalarla:

```
> sudo apt-get install python-pip  
> sudo pip install tweepy
```

Ejemplo mínimo:

```
# Ejemplo basico de uso de la biblioteca Tweepy  
  
import tweepy  
  
# Consumer keys and access tokens, used for OAuth  
consumer_key = 'PONER CREDENCIALES'  
consumer_secret = 'PONER CREDENCIALES'  
access_token = 'PONER CREDENCIALES'  
access_token_secret = 'PONER CREDENCIALES'  
  
# OAuth process, using the keys and tokens  
auth = tweepy.OAuthHandler(consumer_key, consumer_secret)  
auth.set_access_token(access_token, access_token_secret)  
  
# Creation of the actual interface, using authentication  
api = tweepy.API(auth)  
  
# https://dev.twitter.com/docs/api/1.1/get/search/tweets  
tweets = api.search(q='Granada', count=1)  
  
# Mostramos los campos del objeto Tweet  
print dir(tweets[0])  
  
# Mostramos los campos del objeto author del Tweet  
print dir(tweets[0].author)  
  
# Mostramos el nombre del Autor del Tweet.  
print tweets[0].author.name
```

3. Mashupeando

Una vez que sabemos utilizar distintos servicios, podemos hacer un Mashup [5] de los mismos. Por ejemplo, podemos mostrar en un mapa la localización de unos tweets (¿útil para seguir eventos masivos?), mostrar una gráfica estadística sobre los últimos tweets relacionados con un tema, etc.

En esta sección de la práctica se pide pensar y realizar un mashup utilizando los servicios vistos (o algún otro que conozcamos).

Referencias

[1] *Twitter*: <https://twitter.com/>

[2] *Getting Started with Twitter*: <https://dev.twitter.com/docs>

- [3] *Twitter for Websites*: <https://dev.twitter.com/docs/twitter-for-websites>
- [4] *Tweepy*: <http://pythonhosted.org/tweepy/html/>
- [5] *Mashup (aplicación web híbrida)*: http://es.wikipedia.org/wiki/Mashup_%28aplicaci%C3%B3n_web_h%C3%ADbrida%29