



UNIVERSIDAD DE CASTILLA-LA MANCHA
ESCUELA SUPERIOR DE INFORMÁTICA

Proyecto Final

Marcos López Sobrino
Alfonso Barragán Carmona
Roberto Plaza Romero

Aplicaciones Dsitribuidas en Internet
25/01/2019

PARTICIPACIÓN

Alfonso Barragán Carmona	33%
Marcos López Sobrino	33%
Roberto Plaza Romero	33%

API REST: Servidor

- **/collect** [POST]: La función de este *endpoint* consiste en el inicio de la recolección de *tweets* por parte de *flume*, en el *backend* de nuestra aplicación.
- **/no_collect** [POST]: La función de este *endpoint* consiste en la parada de la recolección de *tweets* por parte de *flume*, en el *backend* de nuestra aplicación.
- **/list_word** [POST]: La función de este *endpoint* trata de filtrar los *tweets* mediante PIG dependiendo de los argumentos insertados en el cliente *web*. En este caso muestra los *tweets* que contienen la palabra indicada.
- **/show_likes** [POST]: La función de este *endpoint* consiste en realizar un filtrado mediante PIG y seleccionar aquellos *tweets* que tengan menos “me gusta” de los indicados en el cliente *web*.
- **/show_rts** [POST]: La función de este *endpoint* consiste en realizar un filtrado mediante PIG y seleccionar aquellos *tweets* que tengan menos “retweets” de los indicados en el cliente *web*.
- **/webhook** [POST]: La función de este *endpoint* es agregar a una lista interna del servidor las direcciones de terceros de tal manera que estos clientes sean informados de todas las operaciones realizadas por el *backend*.

API REST: Cliente

- **/login**: Este *endpoint* iniciará la petición de autorización mediante *Twitter*, pues utilizamos OAuth de terceros.
- **/logout**: Este *endpoint* se encarga de cerrar la sesión del usuario actual, sería la operación contraria a la anterior.
- **/oauthorized**: Este *endpoint* intermedio es necesario para la gestión del OAuth de terceros.

- **/:** Este *endpoint* cargará el *index.html* del cliente *web* para dar el servicio.
- **/operations:** Este *endpoint* se ocupa de la carga de la interfaz del cliente *web* para realizar las distintas operaciones disponibles.
- **/collect** [POST]: La función de este *endpoint* es comunicar al *backend* el inicio de la recolección de *tweets* cuando sea indicada en el cliente.
- **/no_collect** [POST]: La función de este *endpoint* es comunicar al *backend* la parada de la recolección de *tweets* cuando sea indicada en el cliente.
- **/collect** [POST]: La función de este *endpoint* es comunicar al *backend* el inicio de la recolección de *tweets* cuando sea indicada en el cliente.
- **/list_word** [POST]: La función de este *endpoint* es comunicar al *backend* la palabra con la que realizar el filtrado. Una vez finalizado, se comunica al cliente *web* el resultado para que lo muestre.
En caso de error se lanza un mensaje *flash* indicándolo, y en caso de éxito se renderizan los *tweets* para su visualización.
- **/show_likes** [POST]: La función de este *endpoint* es comunicar al *backend* el número de “me gustas” con los que realizar el filtrado. Una vez finalizado, se comunica al cliente *web* el resultado para que lo muestre.
En caso de error se lanza un mensaje *flash* indicándolo, y en caso de éxito se renderizan los *tweets* para su visualización.
- **/show_rts** [POST]: La función de este *endpoint* es comunicar al *backend* el número de “retweets” con los que realizar el filtrado. Una vez finalizado, se comunica al cliente *web* el resultado para que lo muestre.
En caso de error se lanza un mensaje *flash* indicándolo, y en caso de éxito se renderizan los *tweets* para su visualización.