

Apache Flink

Jesús Rodríguez Heras
Roberto Muras González
Juan Pedro Rodríguez Gracia
Gabriel Fernando Sánchez Reina

30 de mayo de 2018

Resumen

Definición de Apache Flink y ejemplo de uso capturando datos de una red social con RabbitMQ como broker de mensajería.

Índice

1. Introducción	3
1.1. ¿Qué es Apache Flink?	3
1.2. ¿Qué es RabbitMQ?	3
2. Guía de instalación	3
2.1. Instalación de Java	3
2.2. Apache Flink	6
2.2.1. Prueba de Apache Flink	9
2.3. RabbitMQ	10
3. Palabras más publicadas en twitter	10
4. Problemas encontrados	12
5. Mejoras futuras	12
6. Referencias	12

1. Introducción

1.1. ¿Qué es Apache Flink?

Se trata de un motor de procesamiento de streams o flujos de datos de código abierto que proporciona capacidades de distribución de datos, comunicaciones y, muy importante, tolerancia a fallos a las computaciones.

El núcleo de Apache Flink es un motor de flujo de datos de transmisión distribuida escrito en Java y Scala. El sistema de tiempo de ejecución encadenado de Flink, permite la ejecución de programas de procesamiento por bloques y de flujo.

1.2. ¿Qué es RabbitMQ?

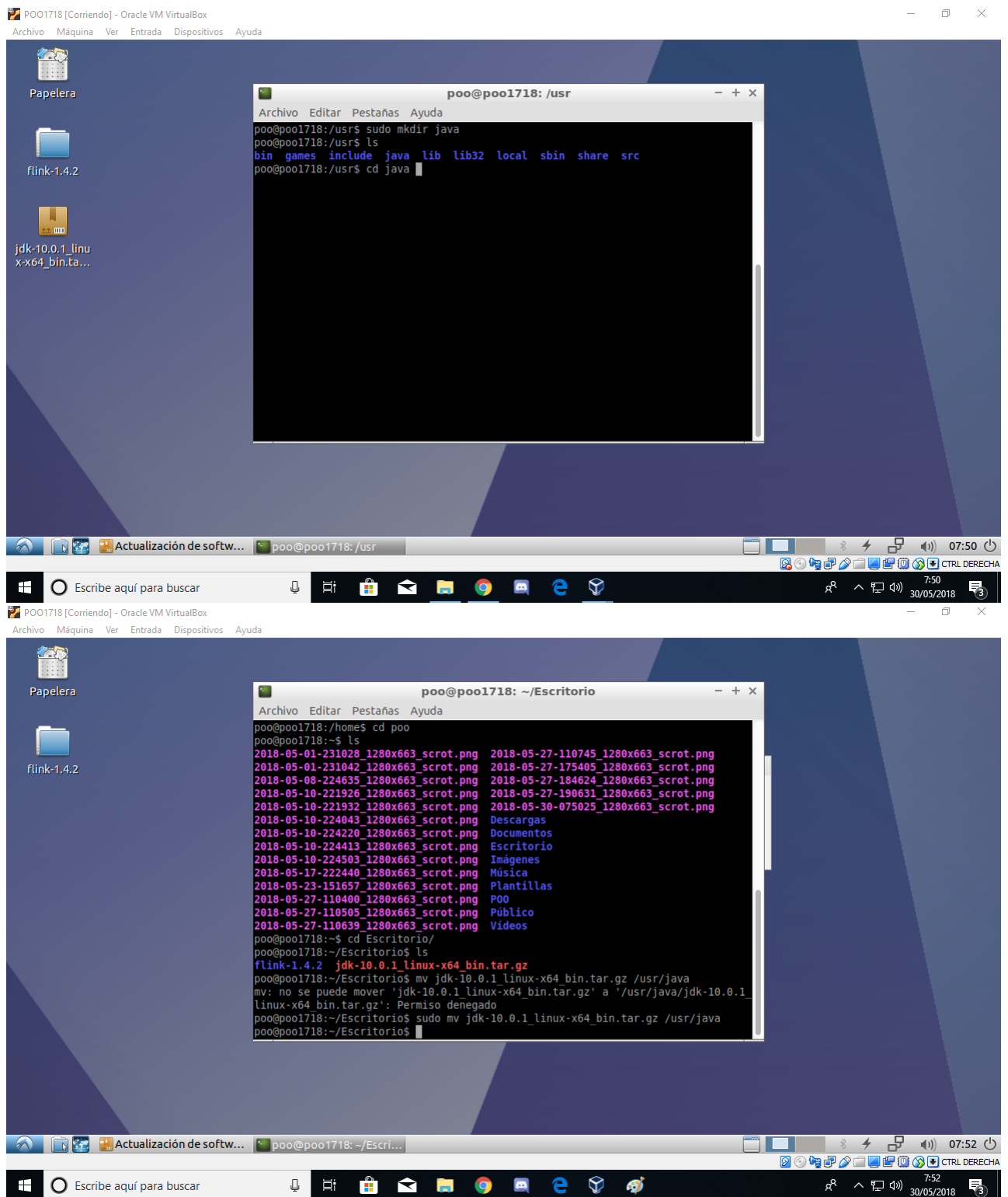
Es un software de negociación de mensajes de código abierto. Se encuentra dentro de la categoría de middleware de mensajería. Implementa el estándar “Advanced Message Queuing Protocol” (AMQP).

El servidor de RabbitMQ está escrito en Erlang y utiliza el framework “Open Telecom Platform” (OTP) para construir sus capacidades de ejecución distribuida y conmutación ante errores.

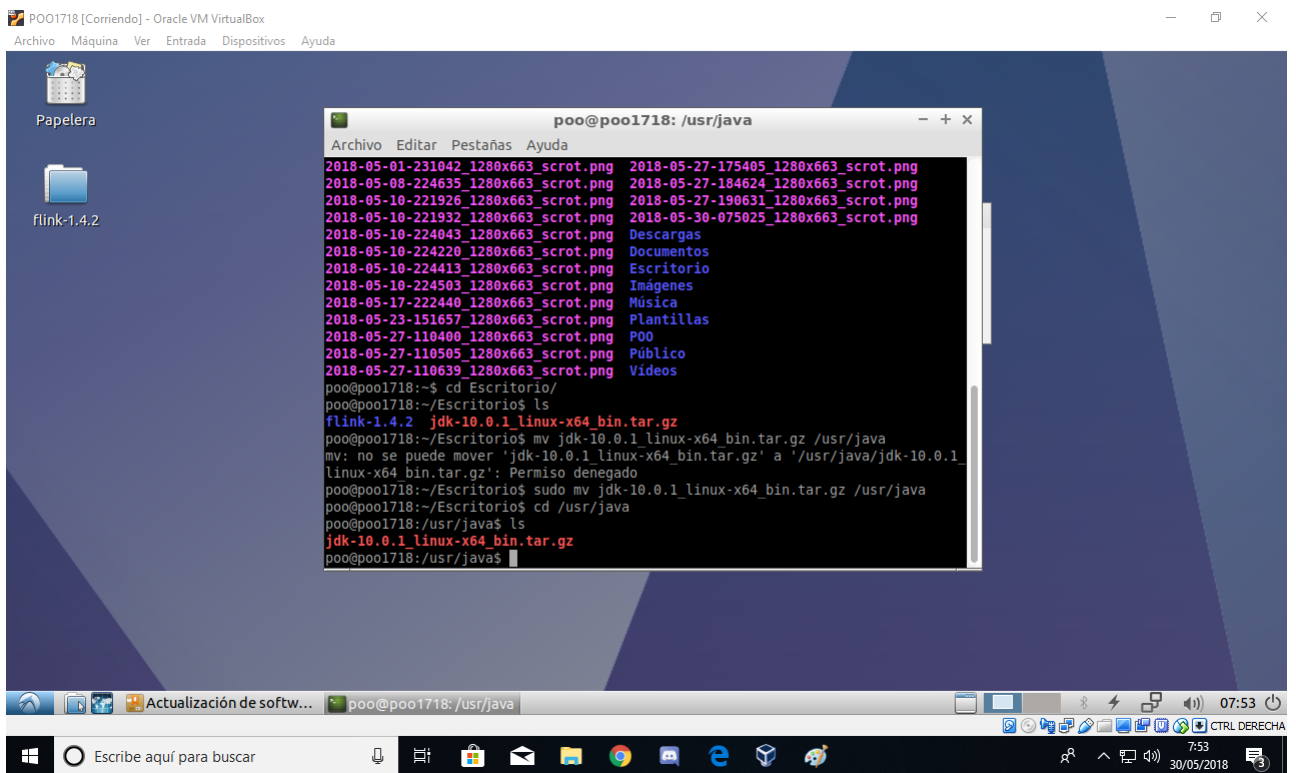
2. Guía de instalación

2.1. Instalación de Java

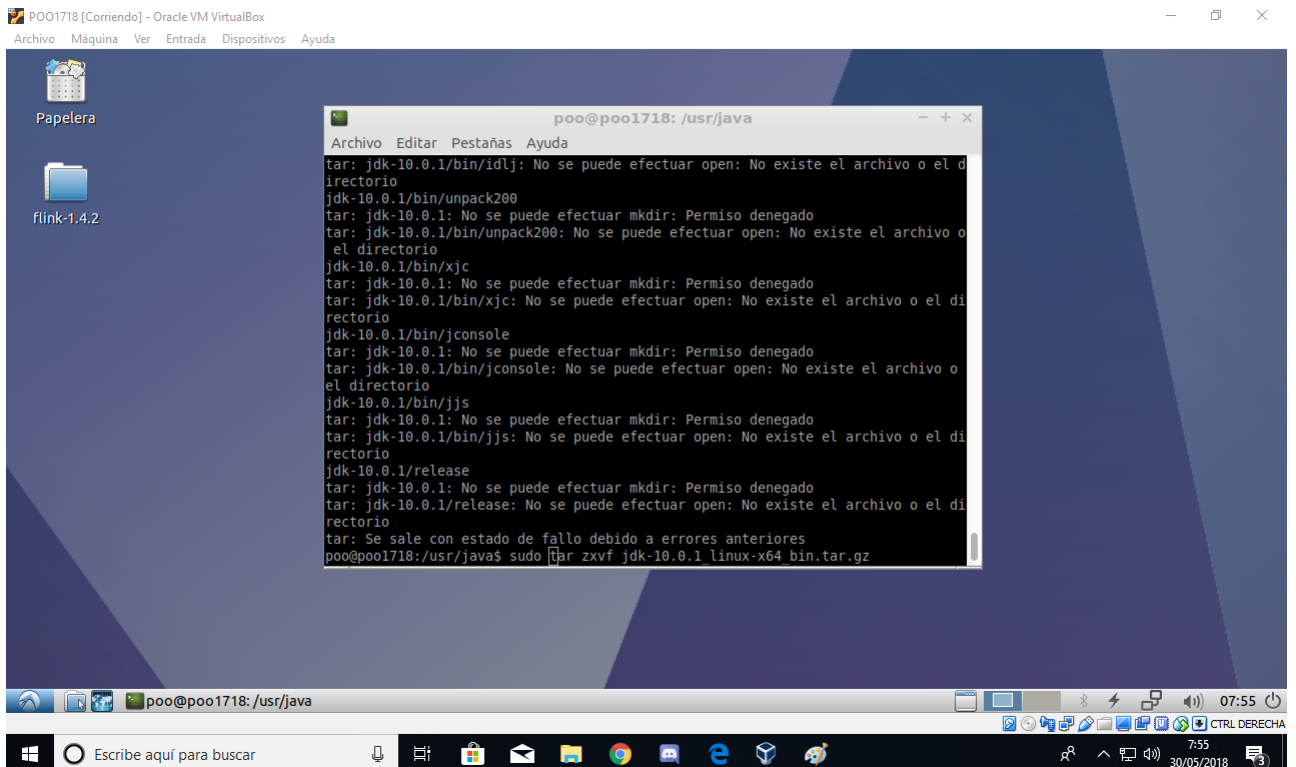
Lo primero que necesitaremos para instalar Apache Flink, será tener java instalado. Para ello, nos movemos a la carpeta en la que queramos instalarlo en nuestro caso será `/usr/java`, en caso de no tener esta carpeta haremos `mkdir java` para ello abrimos una terminal en el directorio raíz y ejecutamos la siguiente orden: `cd /usr/java`.



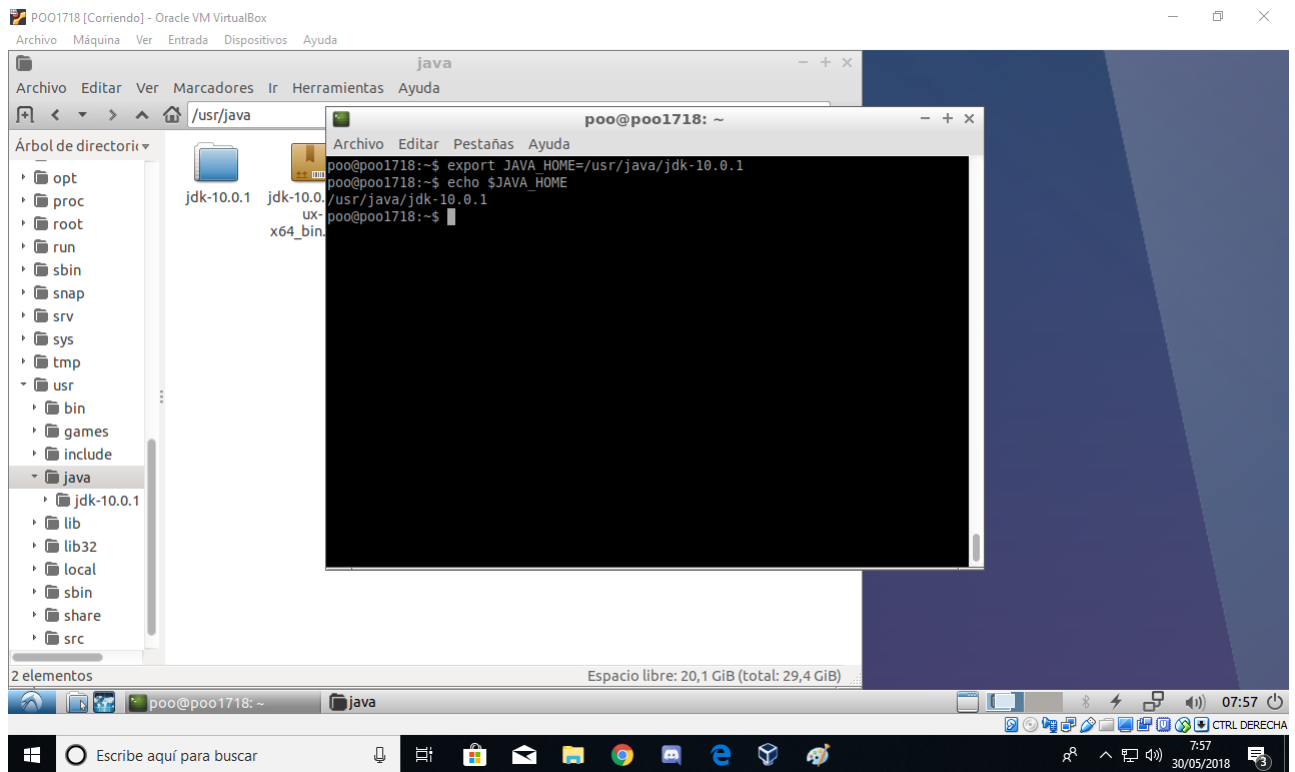
Ahora nos descargamos java de la página principal de java. https://www.java.com/es/download/help/linux_x64_install.xml El paso siguiente será mover el comprimido a dicha raíz, para ello, ejecutamos la siguiente orden, `mv NombreDescarga /usr/java`.



Tras esto, ejecutamos el siguiente comando para descomprimir e instalar el archivo previamente descargado, `tar zxvf NombreDescarga`.



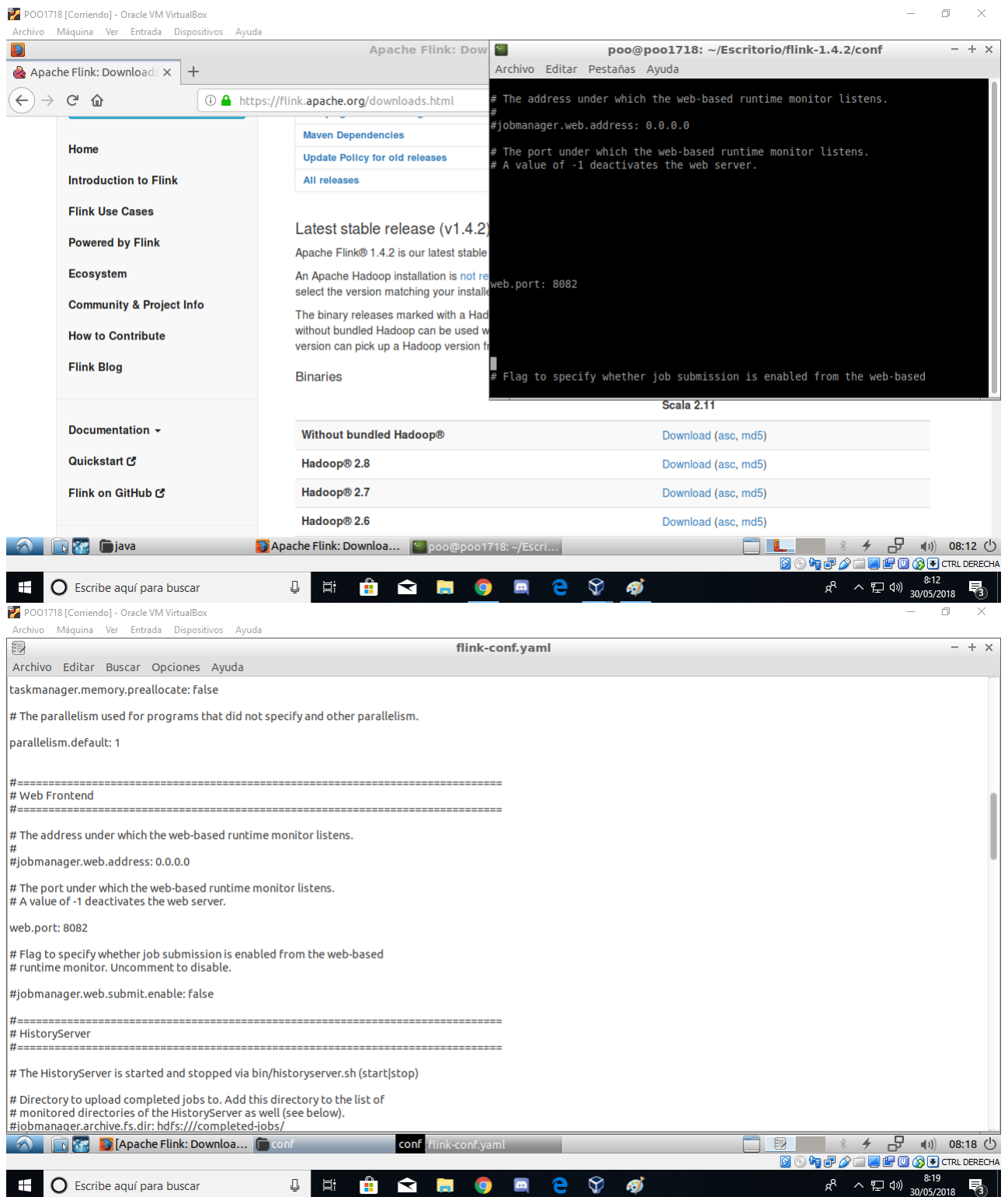
Con esto, tendríamos ya instalado java. Ahora tendríamos que asignar la variable `JAVA_HOME`, para ello, ejecutamos el siguiente comando: `export JAVA_HOME=/usr/java/VersiónInstalada`. Para comprobar que está bien ejecutado, haremos `echo $JAVA_HOME`.



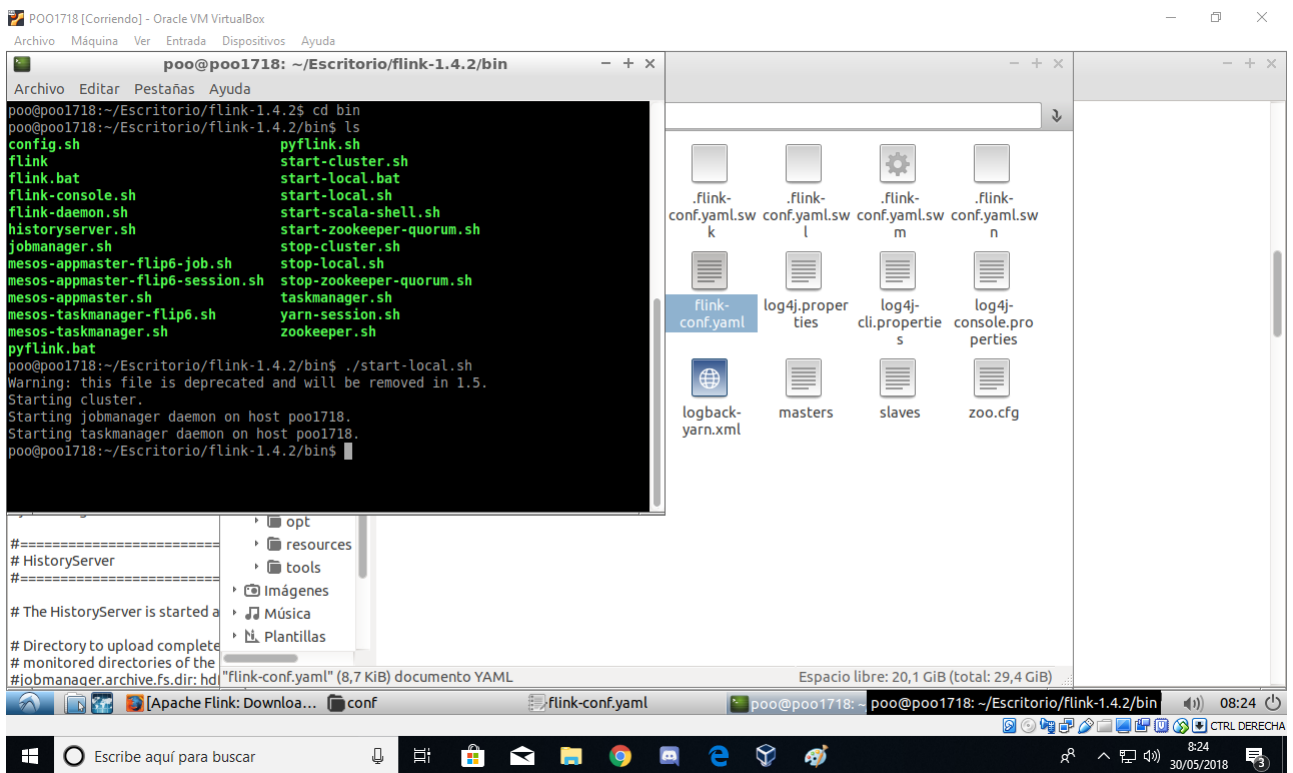
2.2. Apache Flink

Una vez hecho lo anterior, procedemos a la instalación de Apache Flink, para ello debemos descargarnos Apache Flink de la página principal de Apache: <https://flink.apache.org/downloads.html> Una vez lo tengamos descargado, lo extraemos en la carpeta que deseemos y accedemos a ella a través de la terminal.

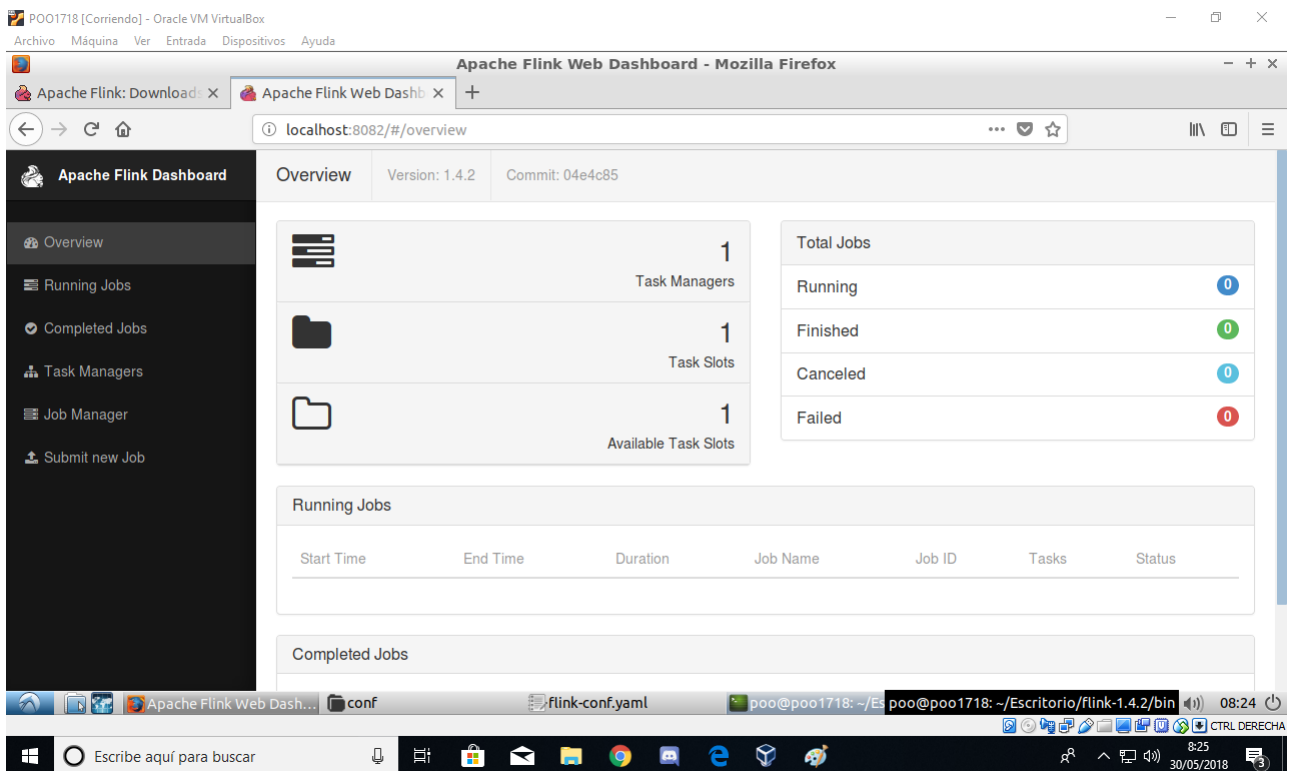
Accederemos a la configuración para cambiar el puerto al que se conecta Apache Flink, para ello hacemos `cd conf` y allí hacemos un “vi” del archivo de configuración (`flink-conf.yaml`) o directamente abrir este archivo con un editor de texto y hacer los cambios pertinentes.



Lo siguiente que haremos será ejecutar flink, para ello haremos `cd bin` y ejecutamos el archivo llamado `start-cluster.sh`.

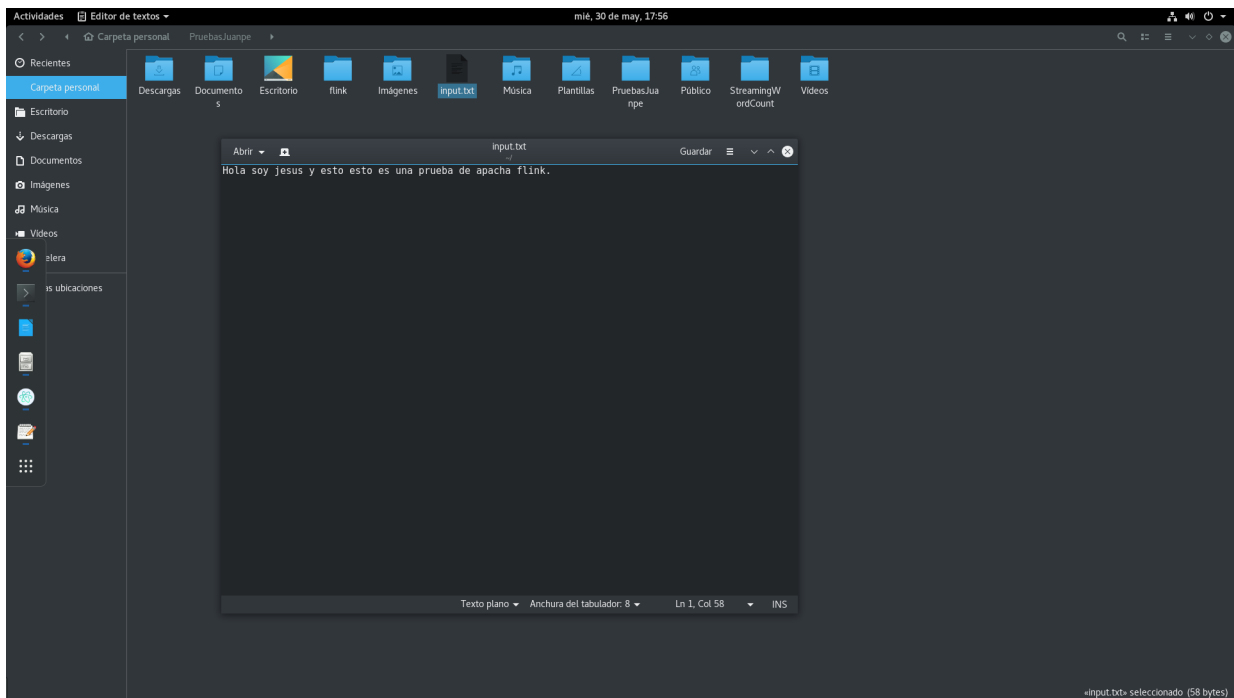


Para saber si se ha ejecutado correctamente, entramos en el navegador web y escribimos localhost : PuertoAsignado y nos mostrará lo siguiente:



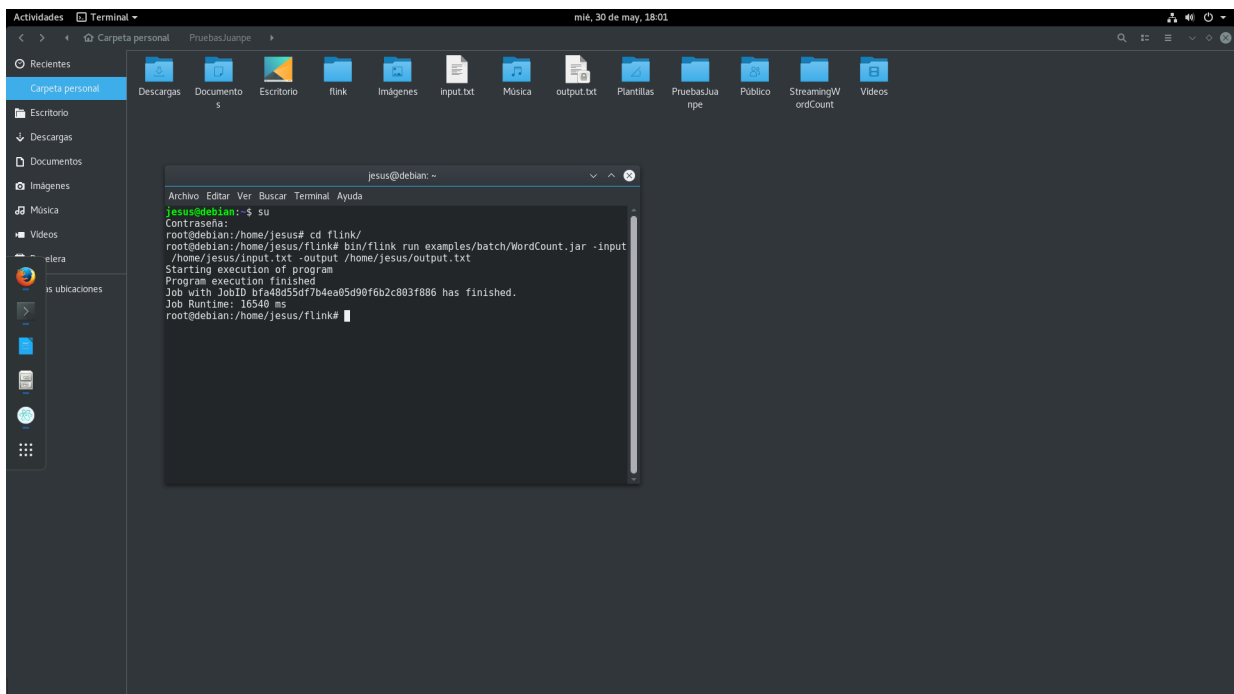
2.2.1. Prueba de Apache Flink

Pasaremos ahora a ejecutar una prueba. Para ejecutar el programa de prueba tendremos que tener creado el fichero `input.txt`.

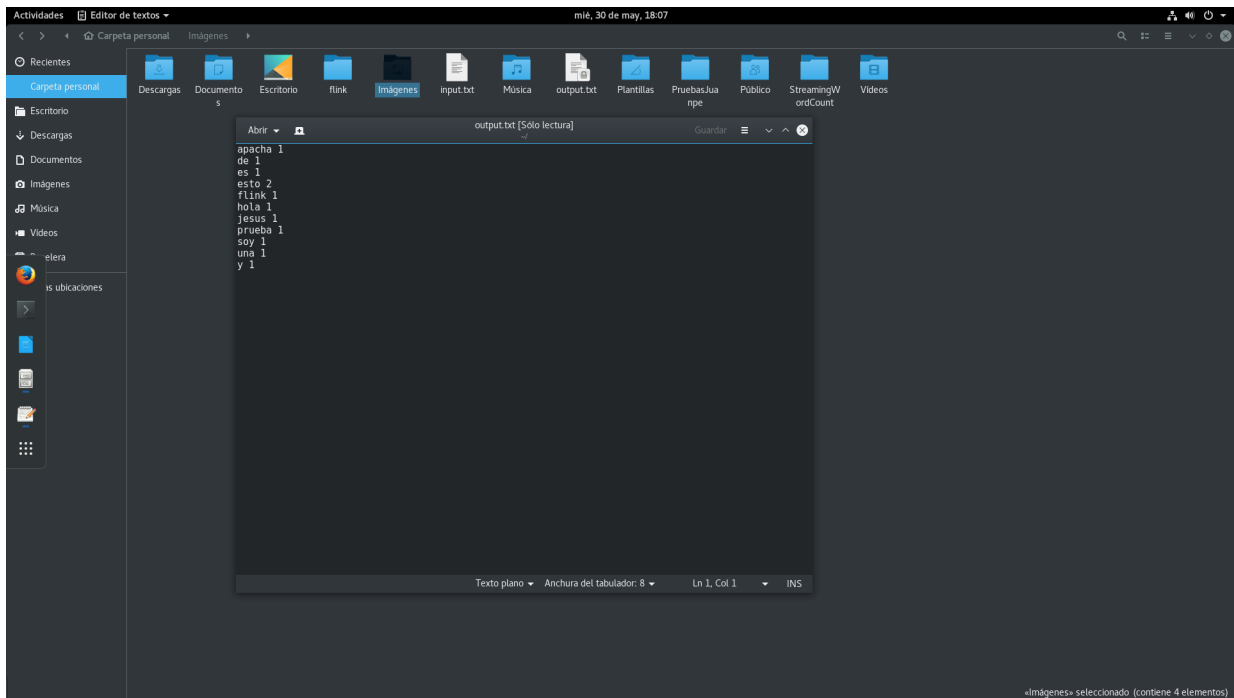


Un detalle sin importancia es la repetición de la palabra “esto” y “apacha” en lugar de “apache”.

Ahora, basta con ejecutar el siguiente comando desde la carpeta de flink `bin/flink run examples/batch/WordCount.jar -input /home/jesus/input.txt -output /home/jesus/ouput.txt`.



Y obtenemos el recuento de palabras en el fichero `output.txt` que se ha creado con la ejecución del programa.



2.3. RabbitMQ

Para la instalación de RabbitMQ lanzaremos el siguiente comando en la terminal: `sudo apt-get install rabbitmq-server`.

Una vez terminado el proceso, instalaremos la consola de administración con el comando `sudo rabbitmq-plugins enable rabbitmq_management`.

A continuación, y para ver que funciona, nos dirigimos a la siguiente dirección con nuestro navegador: `http://localhost:15672` donde entraremos con el usuario `guest` y la contraseña `guest`.

3. Palabras más publicadas en twitter

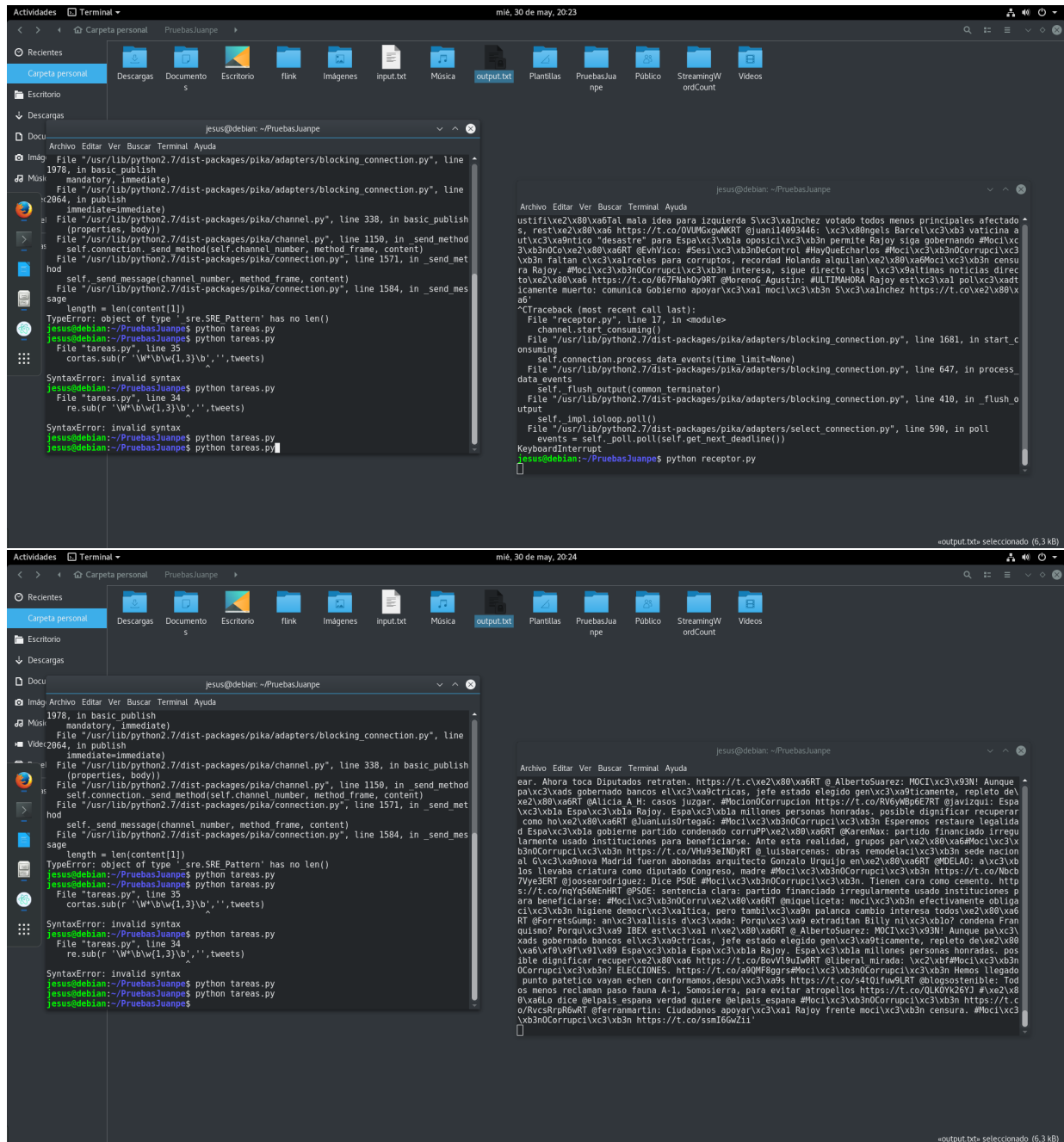
Nuestro proyecto se centra en contar las palabras más publicadas en twitter por la población española. Cuenta con dos archivos escritos en Python: `tareas.py` y `receptor.py` que se encargarán de recolectar los mensajes. Luego, el contador de palabras de prueba de Apache Flink se encargará del resto.

Lo primero de todo es iniciar Apache Flink ejecutando el fichero `start-cluster.sh` ubicado en `/home/flink/bin/` y ejecutar la consola de administración de RabbitMQ con `sudo rabbitmq-plugins enable rabbitmq_management`.

Luego ejecutamos el archivo `tareas.py` que se encarga de entrar en twitter y, de entre todos los Trending Topics españoles actuales, elige el primero, recopila los 100 ultimos tweets asociados

a dicho Trending Topic y los envía a la cola de mensajes de RabbitMQ como una sola cadena de texto.

A continuación, el archivo `receptor.py` se encarga de vaciar la cola de mensajes y escribir su contenido en el fichero `input.txt` ubicado en `/home/jesus/`.



The image consists of two screenshots of a terminal window. The top screenshot shows the execution of a Python script `receptor.py` which reads from a RabbitMQ queue and writes to `input.txt`. The terminal output shows a `TypeError: object of type 're.SRE.Pattern' has no len()` error, which is then fixed by adding `len(content[1])` to the `length` variable. The bottom screenshot shows the resulting `output.txt` file, which contains a large block of text extracted from a tweet, including a URL and a list of names.

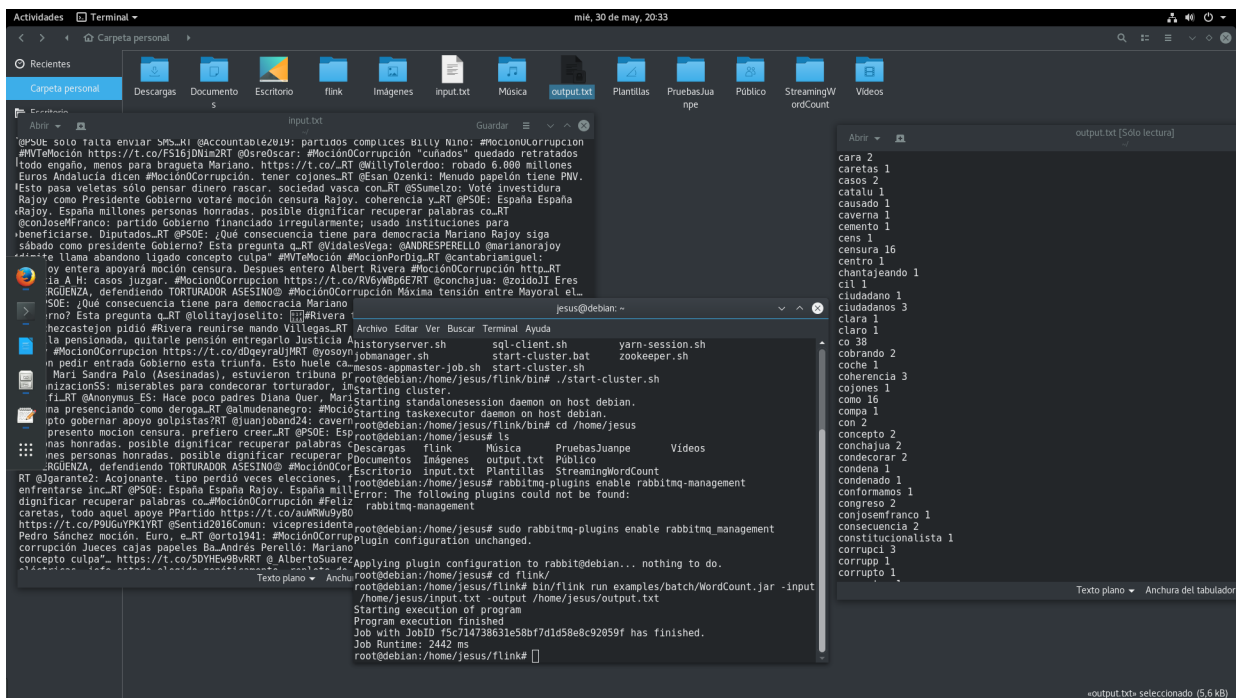
```
jesus@debian: ~/PruebasJuaque
File "/usr/lib/python2.7/dist-packages/pika/adapters/blocking_connection.py", line 1978, in basic_publish
    mandatory, immediate)
File "/usr/lib/python2.7/dist-packages/pika/adapters/blocking_connection.py", line 2064, in publish
    immediate=immediate)
File "/usr/lib/python2.7/dist-packages/pika/channel.py", line 338, in basic_publish
    (properties, body))
File "/usr/lib/python2.7/dist-packages/pika/channel.py", line 1150, in _send_method
    self.connection._send_method(self.channel_number, method_frame, content)
File "/usr/lib/python2.7/dist-packages/pika/connection.py", line 1571, in _send_method
    self._send_message(channel_number, method_frame, content)
File "/usr/lib/python2.7/dist-packages/pika/connection.py", line 1584, in _send_message
    length = len(content[1])
TypeError: object of type 're.SRE.Pattern' has no len()
jesus@debian:~/PruebasJuaque$ python tareas.py
File "tareas.py", line 35
    cortas.sub(r '\W\b\w(1,3)\b', '', tweets)
SyntaxError: invalid syntax
jesus@debian:~/PruebasJuaque$ python tareas.py
File "tareas.py", line 34
    re.sub(r '\W\b\w(1,3)\b', '', tweets)
SyntaxError: invalid syntax
jesus@debian:~/PruebasJuaque$ python tareas.py
jesus@debian:~/PruebasJuaque$ python tareas.py
jesus@debian:~/PruebasJuaque$
```

```
jesus@debian:~/PruebasJuaque
Archivo Editar Ver Buscar Terminal Ayuda
ustifi\ve2\x80\va6fal mala idea para izquierda Slvc3\vaInchez votado todos menos principales afectado
s. festi2e2\x80\va6 https://t.co/0V0K0xgwKRT @Juan14093446: vc3\x80ngels Barcelvc3\vb3 vaticina a
utvc3\va9ntico "desastre" para Espa\vc3\vb1a oposci\vc3\vb3n permite Rajoy siga gobernando #Moc1\vc3
3\vb3n0C0x2e2\x80\va6RT @EvHvico: #Sesi\vc3\vb3nDeControl #HayQueEcharlos #Moc1\vc3\vb3n0Corrupci\vc3
3\vb3n faltan c\vc3\va1rcelres para corruptos; recordad Holanda alquilan\vc2\x80\va6Moc1\vc3\vb3n censu
ra Rajoy. #Moc1\vc3\vb3n0Corrupci\vc3\vb3n interesa, sigue directo l\vc3\vb3ntinas noticias direc
to\vc2\x80\va6 https://t.co/067FNah0y9RT @MorenoG Agustín: #ULTIMAHORA Rajoy est\vc3\va1 pol\vc3\va1t
icamente muerto: comunica Gobierno apoyar\vc3\va1 moc1\vc3\vb3n Slvc3\vaInchez https://t.co\vc2\x80\va6
CTraceback (most recent call last):
  File "receptor.py", line 17, in <module>
    channel.start_consuming()
  File "/usr/lib/python2.7/dist-packages/pika/adapters/blocking_connection.py", line 1681, in start_c
onsuming
    self.connection.process_data_events(time_limit=None)
  File "/usr/lib/python2.7/dist-packages/pika/adapters/blocking_connection.py", line 647, in process_
data_events
    self._flush_output(common_terminator)
  File "/usr/lib/python2.7/dist-packages/pika/adapters/blocking_connection.py", line 418, in _flush_o
utput
    self._impl._loolop.poll()
  File "/usr/lib/python2.7/dist-packages/pika/adapters/select_connection.py", line 590, in poll
    events = self._poll.poll(self.get_next_deadline())
KeyboardInterrupt
jesus@debian:~/PruebasJuaque$ python receptor.py

```

```
jesus@debian:~/PruebasJuaque
Archivo Editar Ver Buscar Terminal Ayuda
ear. Ahora todo Diputados retratan, https://t.c\vc2\x80\va6RT @AlbertoSuares: Moc1\vc3\vb3n! Aunque
pa\vc3\va6s gobernado bancos el\vc3\va9ctricas, jefe estado elegido gen\vc3\va9nticamente, reptado de\
xe2\x80\va6RT @Alicia A.H: casos juzgar. #Mocion0Corrupcion https://t.co/RV6y8Bp6E7RT @javizqui: Espa
\vc3\vb1a Espa\vc3\vb1a Rajoy. Espa\vc3\vb1a millones personas honradas. posible dignificar recuperar
como ho\vc2\x80\va6RT @JuanLu0C0rtaga: #Moc1\vc3\vb3n0Corrupci\vc3\vb3n Esperemos restauo legaLida
d Espa\vc3\vb1a gubierne partido condenado corrup\vc2\x80\va6RT @KarenMax: partido financiado irregu
larmente usado instituciones para beneficiarse. Ante esta realidad, grupos par\vc2\x80\va6#Moc1\vc3\vb3n0Corrupci\vc3\vb3n
https://t.co/VH03e1MDyRT @ Luisbarcenas: obras remodelc1\vc3\vb3n sede nacion
al C\vc3\va9nova Madrid fueron ahondadas arquitecto Gonzalo Urquijo en\vc2\x80\va6RT @DEL40: olvc3\vb
los llevaba criatura como diputado Congreso, madre #Moc1\vc3\vb3n0Corrupci\vc3\vb3n https://t.co/NbcB
7y9e3RT @josearodriguez: Dica P50E #Moc1\vc3\vb3n0Corrupci\vc3\vb3n. Tienen cara como cemento. http
s://t.co/nq0c0N0WRT @P50E: sentencia clara: partido financiado irregularmente usado instituciones p
ara beneficiarse: #Moc1\vc3\vb3n0Corru\vc2\x80\va6RT @miquelceta: moc1\vc3\vb3n efectivamente obliga
ci\vc3\vb3n higiene democ\vc3\va1tica, pero tamb\vc3\va9n palanca cambio interes todos\vc2\x80\va6
RT @RetroGump: anlc3\va1lisis d\vc3\va6a: Porqu\vc3\va9 extraditan Billy n\vc3\vb1o7 condena Fran
quismo? Porqu\vc3\va9 IBEX est\vc3\va1 n\vc2\x80\va6RT @AlbertoSuares: Moc1\vc3\vb3n! Aunque pa\vc3\
va6s gobernados bancos el\vc3\va9ctricas, jefe estado elegido gen\vc3\va9nticamente, reptado de\vc2\x80
\va6\vc0\va9\vc0\va99 Espa\vc3\vb1a Espa\vc3\vb1a Rajoy. Espa\vc3\vb1a millones personas honradas. pos
ible dignificar recuperar\vc2\x80\va6 https://t.co/80v19u1wRT @liberal: mirads: vc2\vb#Moc1\vc3\vb3n
0Corrupci\vc3\vb3n? ELECCIONES. https://t.co/a9Q0WF8ggrs#Moc1\vc3\vb3n0Corrupci\vc3\vb3n Hemos llegado
punto patetico vayan echen conformamos, despu\vc3\va9s https://t.co/s4t0ifuw9lRT @logosostenible: Tod
os menos reclaman paso fauna A-1, Somosierra, para evitar atropellos https://t.co/0LK0YK28Y1 #xe2\x8
0\va6 dice elpais espana verdad quiere elpais espana #Moc1\vc3\vb3n0Corrupci\vc3\vb3n https://t.c
o/Rvc3Rrp6aRT @ferranmartin: Ciudadanos apoyar\vc3\va1 Rajoy frente moc1\vc3\vb3n censura. #Moc1\vc3
\vb3n0Corrupci\vc3\vb3n https://t.co/ssm16wZii'
```

Por último, ejecutando el contador de palabras del ejemplo anteriormente comentado en la prueba de Apache Flink, obtenemos el fichero `output.txt` con el número de repeticiones que tiene cada palabra. Para ello usaremos el comando `bin/flink run examples/batch/WordCount.jar -input /home/jesus/input.txt -output /home/jesus/output.txt`.



4. Problemas encontrados

El principal problema que nos encontramos a la hora de comenzar con el trabajo fue que directamente, no funcionaba en nuestro portátiles, por lo que nos pusimos manos a la obra con una máquina virtual en lubuntu, donde intentamos instalar Apache Flink pero, a la hora de ejecutar el código de prueba, saltaban muchas excepciones que no sabíamos controlar.

Visto esto, decidimos probar con una máquina virtual en Debian, en donde la instalación fue como la seda y el programa de prueba funcionó al primer intento.

Debido a que no contábamos con mucho tiempo, decidimos usar el ejemplo como idea, de tal forma que podíamos hacer una aplicación que nos diera las palabras más publicadas en twitter por la población española.

5. Mejoras futuras

En cuanto a mejorar el proyecto, podríamos filtrar las palabras recolectadas y eliminar las preposiciones y artículos, debido a que se repiten demasiado y no serían “claves” a la hora de buscar una palabra determinada si estamos haciendo un sondeo real.

Otra mejora a tener en cuenta es utilizar “jpye” que permite llamar a clases java desde python, lo que haría mucho más sencillo el trabajo de nuestro proyecto.

6. Referencias

- <https://flink.apache.org/>

- <https://www.rabbitmq.com/>
- https://en.wikipedia.org/wiki/Apache_Flink
- <https://en.wikipedia.org/wiki/RabbitMQ>
- <https://www.adictosaltrabajo.com/tutoriales/introduccion-a-apache-flink/>
- <https://data-flair.training/blogs/install-configure-apache-flink-ubuntu/>