

## Application for the analysis of the received tweets

By Mironela Pirnau with the collaboration with Horia-Nicolai Teodorescu

Please cite this application as:

**Horia-Nicolai L. TEODORESCU, Mironela PIRNAU, Analysis of requirements for SN monitoring applications in disasters – A case study. ECAI 2016, IEEE International Conference on Electronics, Computers and Artificial Intelligence, 8th Edition, 30 June - 02 July, 2016, Ploiesti, ROMÂNIA**

The application is executed simultaneously with the extraction of tweets  
We extracted tweets using <https://github.com/jublonet/codebird-php> and <https://dev.twitter.com/docs/api/1/get/search>.  
See the library used at <https://www.jublo.net/projects/codebird/php>.

The Tweets are collected in a table with the structure in Fig 1.

| Field            | Type       | Null | Key | Default | Extra          |
|------------------|------------|------|-----|---------|----------------|
| id               | int(11)    | NO   | PRI | NULL    | auto_increment |
| twitter_tweet_id | bigint(20) | NO   |     | NULL    |                |
| request_id       | int(11)    | NO   |     | NULL    |                |
| field_key        | text       | NO   |     | NULL    |                |
| field_value      | text       | YES  |     | NULL    |                |
| data             | text       | NO   |     | NULL    |                |

Figure 1. The structure of the table where tweets are collected

Then with must create table **alerta\_flag** and **index** tables with defined structure in Fig 2 and Fig 3 and corresponding code as shown below.

```
create table alerta_flag with structure
"CREATE TABLE IF NOT EXISTS `alerta_flag`
(`id` int(11) NOT NULL AUTO INCREMENT,
`id_tweet` BIGINT(11) NOT NULL,
`request_id` int(11) NOT NULL,
`flag` text CHARACTER SET utf8 COLLATE utf8_bin NOT NULL,
`tweets` text CHARACTER SET utf8 COLLATE utf8_bin NOT NULL,
`keys` text CHARACTER SET utf8 COLLATE utf8_bin NOT NULL,
`created_at` text CHARACTER SET utf8 COLLATE utf8_bin NOT NULL,
`data_identificare` text CHARACTER SET utf8 COLLATE utf8_bin NOT NULL,
`data_prelucaire` text CHARACTER SET utf8 COLLATE utf8_bin NOT NULL,
`created_at` text CHARACTER SET utf8 COLLATE utf8_bin NOT NULL,
`id_tw` int(11) NOT NULL,
PRIMARY KEY (`id`))");
```

```
create table index with structure
"CREATE TABLE IF NOT EXISTS `alerta_flag`
(`id` int(11) NOT NULL AUTO INCREMENT,
`id_tw` int(11) NOT NULL,
PRIMARY KEY (`id`))");
```

| + Options        |            |      |     |         |                |
|------------------|------------|------|-----|---------|----------------|
| Field            | Type       | Null | Key | Default | Extra          |
| id               | int(11)    | NO   | PRI | NULL    | auto_increment |
| id_tweet         | bigint(20) | NO   |     | NULL    |                |
| request_id       | int(11)    | NO   |     | NULL    |                |
| flag             | text       | NO   |     | NULL    |                |
| tweets           | text       | NO   |     | NULL    |                |
| keys             | text       | NO   |     | NULL    |                |
| created_at       | text       | NO   |     | NULL    |                |
| data_indenticare | text       | NO   |     | NULL    |                |
| data_prelucreare | text       | NO   |     | NULL    |                |
| id_tw            | int(11)    | NO   |     | 0       |                |

Fig 2. The structure of **alerta\_flag** table

| Field | Type    | Null | Key | Default | Extra          |
|-------|---------|------|-----|---------|----------------|
| id    | int(11) | NO   | PRI | NULL    | auto_increment |
| id_tw | int(11) | NO   |     | 0       |                |

Fig 3. The structure of index table

The sequence of PHP code of the application, which find extracted tweets containing the words “mort”, “killed”, “dead” and “victim” is the following:

```
<?php
include("parametri.php");
include("config.php");//file of configuration
include("lib/db.php");// file for connecting to Data Base
$sql = 'SELECT max(`id_tw`) as max from `index`';
$result = query_mysql($sql, $link);
$z = mysql_fetch_assoc($result);
$id_max=$z["max"];
$sql = "SELECT max(id) as maxim from c_twitter_tweetmetadata";
$result = query_mysql($sql, $link);
$z = mysql_fetch_assoc($result);
$id_maxim=$z["maxim"];
echo "incept1";
//add your conditions
$sql1 = "SELECT * from c_twitter_tweetmetadata where id >'".$id_max."' and field_key
like 'text' and
(lower(field_value) like lower('% mort%') or
lower(field_value) like lower('%killed%') or
lower(field_value) like lower('%dead%') or
lower(field_value) like lower('%victim%'))";
$result = query_mysql($sql1, $link);
while ($z = mysql_fetch_assoc($result))
```

```

{
  $id_tw=$z["id"];
  $id_tweet=$z["twitter_tweet_id"];
  $request_id=$z["request_id"];
  $tweets=$z["field_value"];
  $data_identificare=$z["data"];
  $flag="de_rezolvat";
  //cleaning RT @
  $mes=strstr($tweets,"RT @");
  $lng=strlen($mes);
  if($lng>1)
  {
    $a=strstr($mes,": ");
    if($a)
    {
      $lung=strlen($a);
      $pos1 = stripos($mes, ": ");
      $x1=substr($mes,$pos1+2,$lung);
    }
    else
    {
      $a=strstr($mes,":");
      if($a)
      {
        $pos1 = stripos($mes, ":");
        $x1=substr($mes,$pos1+1,$lung);
      }
    }
  }
  else {$x1=$tweets;}

  $tweets=$x1;
  //find morti, morți, mort
  $mes=strstr(strtolower($tweets),"morti");
  $lng=strlen($mes);
  if($lng>=1) $key=$key." morti, ";
  else
  {
    $mes=strstr(strtolower($tweets),"morți");
    $lng=strlen($mes);
    if($lng>=1) $key=$key." morți, ";
    else
    {
      $mes=strstr(strtolower($tweets),"mort");
      $lng=strlen($mes);
      if($lng>=1) {$key=$key." mort, ";}
    }
  }
  //find victim
  $mes=strstr(strtolower($tweets),"victim");
  $lng=strlen($mes);
  if($lng>=1) $key=$key." victim ";
  //find killed
  $mes=strstr(strtolower($tweets),"killed");

```

```

$lng=strlen($mes);
if($lng>=1) $key=$key." killed,";
//find dead
$mes=strstr(strtolower($tweets),"dead");
$lng=strlen($mes);
if($lng>=1) $key=$key." dead.";
//memory created_at tweets
$creat = "SELECT field_value as created_at from c_twitter_tweetmetadata where
twitter_tweet_id=$id_tweet and field_key like 'created_at'";
$rez = query_mysql($creat, $link);
$z1 = mysql_fetch_assoc($rez);
$created_at=$z1["created_at"];
$y = 'INSERT INTO `alerta_flag`
(`id_tw`,`keys`,`id_tweet`,`request_id`,`flag`,`tweets`,`created_at`,
`data_identificare`,`data_prelucreare`)
VALUES('.$id_tw.','.$key.','.$id_tweet.','.$request_id.','.$flag.','.$tweets.','
'.$created_at.','.$data_identificare.','.$date("Y-m-d H:i:s", time()).)';
$k++;
$key="";
if(query_mysql($y, $link))
{
    //echo "Insert OK";
}
else
{
    echo "Error-- ".$id."err";
}
//echo "End of tweets analyzing";
}
//memory id_tw
$index = 'INSERT INTO `index` (`id_tw`) VALUES (.$id_maxim.)';
query_mysql($index, $link);
?>

```

#### **File parametri.php**

```

<?php
$id_max=0;
$k=0; $id_tw=""; $id_tweet="";
$request_id="";
$flag=""; $tweets="";
$key=""; $created_at="";
$data_identificare="";
$data_prelucreare="";
?>

```

**The archive of this application can be downloaded from this link (16.06.2016)**

[http://www.totalschool.ro/descarca/analiza\\_simultan\\_tweets.zip](http://www.totalschool.ro/descarca/analiza_simultan_tweets.zip)