

DIGIT.B4 – Big Data PoC

DIGIT 01 – Social media topics

D04.01.Information System

everis Spain S.L.U

Table of contents

1 Introduction	4
1.1 Context of the project	4
1.2 Objective	4
2 Technologies used	5
2.1 Python	5
2.2 Django	5
2.3 JavaScript + D3.js	5
2.4 HTML & CSS	5
2.5 Amazon EC2	5
3 How to run the tool	7
3.1 Import source and data	7
3.2 Project Structure	7
3.2.1 The app structure	7
3.2.2 Application logic	8
3.2.3 The application templates	8
3.2.4 The application static files	8
3.3 Start and stop the tool	9
3.3.1 Start the tool	9
3.3.2 Stop the tool	9

Table of figures

Figure 1: Project structure.....	8
Figure 2: App structure	8

1 INTRODUCTION

1.1 Context of the project

This proof of concept shall also demonstrate the use of text mining techniques on large amounts of tweets as a means to identify areas of interest for the 2016 ICT conferences.

1.2 Objective

The purpose of this document is to reflect the technologies used and the tool structure.

2 TECHNOLOGIES USED

In this project, multiple technologies have been used for the publication of the results. These technologies are the ones mostly used in the data analysis and visualization fields of knowledge and are widely supported by different software communities around the world.

2.1 Python

Python is a high level multi-paradigm programming language available for most operating systems. Its philosophy emphasizes code readability and its syntax allow users to express complex concepts in fewer lines of code than in other languages such as Java or C, which is one of the main reasons to choose this language as the base for the project.

It is also a very 'hot' technology in the data science world, and it is widely used for the implementation of different machine learning algorithms and text mining techniques with very popular libraries such as *numpy*, *scipy* or *scikit-learn* among others.

2.2 Django

Django is a web framework designed for python that allows rapid development and prototyping, taking care for the most common task performed by web developers so they can focus on writing the app instead of dealing with repetitive work.

It supports the most popular web application servers (as Apache and Nginx) and databases (MySQL, PostgreSQL).

2.3 JavaScript + D3.js

One of the most important tasks of this project is the data visualization.

D3.js is a JavaScript library that provides ways to make beautiful, dynamic interactive data visualizations in web browsers making use of technologies widely implemented such as SVG, HTML5 and CSS, providing great control over the final result.

2.4 HTML & CSS

These technologies have been used as well as Django uses HTML templates and CSS style sheets for the website presentation layer,

These are probably the most used web technologies and their benefits and support are beyond doubt.

Django also provides its own template language to work with HTML that makes the work easier for developers as it makes the presentation of the application more dynamic.

2.5 Amazon EC2

One of the objectives of the project is to be as agile as possible in the creation of prototypes and versions of the application.

The Amazon EC2 (Elastic Compute Cloud) service provides a very fast and easy server deployment process, allowing the developers to ignore the details of preparing a specific machine or machines for the web application.

These web servers as a service also provide great scalability and security, and the possibility of choose the operating system that best suit the needs of the project.

3 HOW TO RUN THE TOOL

3.1 Import source and data

The objective of this section is to show how to import the source code and the data in a linux. The tool is composed by two folders to import:

- Source: This folder contains all source code of the tool
- Data: This folder contains all data of the tool (BBDD and csv)



Social Media.zip

The user must follow the next steps to import the tool:

- 1) Open the “linux” console
- 2) Copy the two folders in the directory
- 3) Execute the following sentences in this order:

- a. To enter in the directory “Source:

```
> cd Source
```

- b. To create the structure of the database

```
> python manage.py migrate
```

- 4) Copy the file “db” situated at the folder “Data” into the structure created.
 - ✓ This step is to update the database created previously empty, with the real information of the PoC.

3.2 Project Structure

Once the user has executed the previous steps the structure of the project has been generated

Django offers a very intuitive structure for a project, separating each part of the project in different areas.

3.2.1 The app structure

A Django project may consist in several apps, each one with its specific directory inside the main project. In this case, only one app has been developed: the one for the visualization of the social networks analysis result, so the directory structure is as follows:

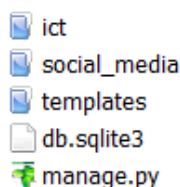


Figure 1: Project structure

Where ICT is the app, social_media is a folder that belongs to the project, the database file is where the data is stored, and the anage.py file is a Django script to run several tasks not related with the development.

The ICT app is structured as shown in the following image:

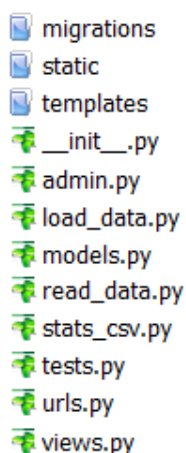


Figure 2: App structure

3.2.2 Application logic

The application logic can be found in the root directory for the app. Here, the views.py file is where the functionality of each web page is implemented, and where all the custom libraries (such as stats_csv.py) are located.

Here is also where the URLs for each web page are defined (urls.py), and the models for the database, which contains the structure and fields of the database tables, created (models.py and read_data.py).

3.2.3 The application templates

The web application uses HTML templates to show the results and the information gathered from the analysis. These templates are located I the “templates” directory, and are associated to the views.

3.2.4 The application static files

Web applications usually use some kind of static files for the presentation layer. In this case, the static folder contains:

- Images of the application (logos, icons, etc.)
- JavaScript visualizations: the JavaScript visualizations are static script files that read from a data source to show the results of the analysis.
- CSV files: some visualizations use CSV files as a source of information.

- CSS files: the CSS style sheets are also located in this folder, as they won't dynamically change in the application.

3.3 Start and stop the tool

Once the structure of the project is created, the user can execute the following sentences for start and stop the tool.

3.3.1 Start the tool

- 1) Execute the next command lines in the "linux" console

- a. To enter in the directory "Source:

```
> cd Source
```

- b. To start the tool

```
> Python manage.py runserver 0.0.0.0:8080
```

- 2) Copy the URL in the navigator (chrome, explorer or firefox) with the next URL:
 - a. **"XX.XX.XX.XX:8080/ict/home/"** → where XX.XX.XX.XX is the IP or the machine

3.3.2 Stop the tool

- 1) Execute the next command lines in the "linux" console
 - a. To stop the tool:

```
> Ctrl+C
```