



Universidad De Guayaquil

FACULTAD DE CIENCIAS MATEMÁTICAS Y FÍSICA

# PROYECTO (WORDCLOUD)

*Proyecto Final del Parcial*

Autores:

- Vanessa Ronquillo
- Melissa Plaza
- Emily González

Agosto 2020

# 1 INTRODUCCIÓN

El presente proyecto tiene como propósito la creación de una nube de palabras de etiquetas de los usuarios de la pagina web Stack Overflow (Español), estará codificado en el lenguaje de programación python haciendo uso de la plataforma Google Colab, referente al proyecto a realizar se va a aplicar un modelo de desarrollo de software y en este caso se eligió el modelo cascada por la claridad de los requerimientos y por tratarse de un sistema pequeño y poco complejo, además por su fácil implementación. Se debe tomar en cuenta cada una de las etapas del modelo elegido y realizar su respectiva fase correctamente, es necesario realizar este conjunto de actividades para transformar los requerimientos del usuario en un sistema de software.

El requerimiento principal de este proyecto es que se pueda ingresar el ID del usuario de la página Stack Overflow (Español), y automáticamente ésta genere una nube de palabras con las etiquetas del usuario que se ha elegido. Se ha usado el método de web scraping para explorar la pagina y luego generar la nube de palabras, ya que esto nos permite a través de los algoritmos de búsqueda poder rastrear centenares de webs para solo extraer la información necesaria en este caso solo necesitábamos la base de datos donde están almacenados los usuarios.

El objetivo de esta investigación es poder llevar un control en el desarrollo de software a través de actividades que son partes de los procesos y así se garantice el éxito del proyecto y entregar un software netamente de calidad.

## 2 FASE DE ANÁLISIS

### DATOS DE LA PÁGINA



StackOverFlow es un sitio de preguntas y respuestas para programadores profesionales y aficionados. Se creó para ser una alternativa más abierta a sitios previos de preguntas y respuestas como Experts-Exchange. El nombre del sitio web (en español "desbordamiento de pila") fue elegido por votación en abril de 2008 por los lectores de Coding Horror, el popular blog de programación de Atwood.

En la página principal se identifican 5 bloques principales:

1. Inicio
2. Preguntas
3. Etiquetas
4. Usuarios
5. Sin responder

En el proyecto se uso principalmente el bloque de usuarios el cual nos daba acceso a otras dependencias de la pagina como: etiquetas de preguntas y respuestas, puntuaciones, numero de etiquetas.

La página SOes tiene aproximadamente 149.622 usuarios y el usuario con mas interacciones tiene aproximadamente 15 páginas de etiquetas. Concluyendo que la página cuenta con una gran base de datos la cuál se podría procesar de diversas maneras dependiendo al tipo de análisis que se quiere realizar, y a través de la aplicación de nube de palabras se puede simplificar un poco esta información de una forma interesante y creativa.

## DESCRIPCIÓN GENERAL

El desarrollo del proyecto se realiza a través de una secuencia simple de fases, cada fase tiene un conjunto de metas y actividades bien definidas.

Este modelo de software es el enfoque metodológico que ordena rigurosamente las etapas del ciclo de vida del software, de tal forma que el inicio de cada etapa debe esperar a la finalización de la inmediatamente anterior.

De esta forma, cualquier error de diseño detectado en la etapa de prueba conduce necesariamente al rediseño y nueva programación del código afectado, aumentando los costos del desarrollo.[4]

Esos requisitos son la base sobre la que se construye el éxito. Por lo tanto, se lleva a cabo una descripción de los requisitos del software, y se acuerda entre el cliente y la empresa desarrolladora lo que el producto deberá hacer. Disponer de una especificación de los requisitos permite estimar de forma rigurosa las necesidades del software antes de su diseño. Además, permite tener una base a partir de la cual estimar el coste del producto, los riesgos y los plazos.

## ANÁLISIS DE REQUERIMIENTOS

Se hace un breve análisis de los requerimientos que necesitara el usuario para desarrollar el sistema solicitado ya que esto permitirá que los Analistas determinen los requerimientos mas relevantes que facilitarán el correcto desarrollo de software.

- El software debe ser escrito en un lenguaje compatible con el existente.
- Se usará la base de datos de la pagina Stack Overflow (Español).
- El sistema permitirá ingresar un ID de usuario de la pagina Stack Overflow(Español).
- El sistema mostrara la coincidencia de este ID y mostrará directamente el nombre del usuario.
- El sistema generará una nube de palabras con las etiquetas obtenidas.

### 3 DISEÑO

Se utilizó los elementos y modelos obtenidos en la etapa de análisis para transformarlos en mecanismos que puedan ser utilizados en el entorno de desarrollo que se uso para realizar este proyecto.

La etapa de diseño y la de análisis son la mejor estructura para producir un software de calidad.

Se proyecta implementar un sistema que permita elaborar una nube de palabras como datos las etiquetas de preguntas y respuestas de un usuario escogido previamente. Se identifica un solo módulo en la aplicación que seria el ingreso de la ID del usuario y visualización de nube de palabras.

MÓDULO	DESCRIPCIÓN
INGRESO	Este módulo permitirá que el usuario pueda digitar el ID de su elección.
VISUALIZACIÓN	Permite consultar información con respecto al ID ingresado tal como: ID, nombre de usuario, etiquetas. Permite representar las etiquetas del usuario ingresado en una nube de palabras.

### 4 CODIFICACIÓN

La codificación se la realiza en el lenguaje Python3, por compatibilidad con el entorno de Desarrollo que se utilizará, el cual es Google Collab, una plataforma que beneficia al desarrollo del proyecto

Se usan las librerías:

```
1 import matplotlib.pyplot as plt
2 %matplotlib inline
3 from wordcloud import WordCloud, STOPWORDS
4
5 from bs4 import BeautifulSoup
6 import requests
7 import pandas as pd
```

- **Matplotlib:** Esta librería nos ayuda a la generación de gráficos a partir de datos contenidos en listas o arrays en el lenguaje de programación Python.[2]

- **Wordcloud:** Análisis de palabras frecuentes, generar nube de palabras. [1]
- **BeautifulSoup:** BeautifulSoup es una biblioteca de Python para analizar documentos HTML. Esta biblioteca crea un árbol con todos los elementos del documento y puede ser utilizado para extraer información. Se utiliza para el web scraping. [3]
- **Requests:** La librería requests nos permite enviar solicitudes HTTP con Python sin necesidad de tanta labor manual, haciendo que la integración con los servicios web sea mucho más fácil.[5]
- **Pandas:** Es una librería para el análisis de datos que cuenta con las estructuras de datos que necesitamos para limpiar los datos en bruto y que sean aptos para el análisis (por ejemplo, tablas).

Se podría dividir en dos partes la codificación, la primera scraping del sitio StackOverflow en Español y la segunda, el procesamiento de estos datos en la nube de palabras.

## PRIMERA PARTE: WEB SCRAPING

Con la importación de las librerías ya expuestas, después de pedir al usuario el ID a buscar, se empieza con las solicitudes a la web de StackOverflow en Español, se realizan multiples solicitudes a la pagina de las cuales se obtiene: nombre de usuario, etiquetas de respuestas y preguntas y el numero de etiquetas del usuario elegido Una vez obtenido la información de las etiquetas, en primera instancia se crean dos listas, la primera para almacenar las etiquetas, la segunda para el almacenamiento del numero de las mismas.

A través del primer ciclo for se recorren todas las paginas de etiquetas posibles de un usuario, y por medio de los siguientes se recorren todas las etiquetas y números de etiquetas y cada una se va agregando a la lista.

En el fragmento final de esta parte, se crea una lista nueva que servirá para que el nombre etiquetas se guarden tantas veces indique el numero de las mismas, esto a través de varios ciclos for y usando una función para agregar cada uno a la lista.

```

10 print("Ingrese un usuario: ")
11 usuario = input()
12 print("ID DE USUARIO:" +usuario)
13
14 #Implementa ID de usuario
15 page = requests.get("https://es.stackoverflow.com/users/"+usuario)
16 soup = BeautifulSoup(page.content,'html.parser')
17
18 #obtener nombre usuario
19
20 try:
21     Nusuario = soup.find('div',class_="grid--cell fw-bold").text
22     print("NOMBRE: "+Nusuario)
23     print("NOMBRE: "+Nusuario)
24     print("Procesando...")
25
26     #Implementa Nombre de usuario
27     url = "https://es.stackoverflow.com/users/"+usuario+"/"+Nusuario+"?tab=tags" + "&sort=votes&page="
28
29
30     #Definir listas de tags y Numero de Tags
31     #Se guardan los Tags obtenidos de la URL en las listas correspondientes
32     tags = list()
33
34     Ntags = list()
35
36     #Se crea una lista nueva definida en conjunto relacionando los valores anteriores
37     listacomplit = list()
38     cont = list()

```

```

36 #Se crea una lista nueva definida en conjunto relacionando los valores anteriores
37 listacomplit = list()
38 cont = list()
39
40 #conseguir el numero de rango
41 for page in range(1,15):
42
43     r = requests.get(url + str(page))
44
45     soup = BeautifulSoup(r.content, "html.parser")
46     td = soup.find_all("td")
47
48     #Tags
49     for i in td:
50         tags.append(i.find('a', class_='post-tag').text)
51
52     #Ntags
53     for i in td:
54         Ntags.append(i.find('div', class_='answer-votes').text)
55
56     #Procesa Tags con Numeros
57     for i in range(len(tags)):
58         if (int(Ntags[i])> 0):
59             if Ntags[i] == "3k" or "2k" or "1k":
60                 Ntags[i]=2000
61             else:
62                 Ntags[i]= Ntags
63
64             for k in range(int(Ntags[i])):
65                 listacomplit.append(tags[i])
66
67 print("Las etiquetas del usuario "+ Nusuario+" son: ")
68 print(tags)
69 print("Creando nube de palabras...")

```



## SEGUNDA PARTE: NUBE DE PALABRAS

Se define una función para poder esbozar la nube de palabras, luego se la crea y define en los argumentos de la función WORDCLOUD, se le da parámetros tales como: tamaño, colores de letras y fondo, palabras repetidas, exclusión de palabras conectoras, inclusión de caracteres especiales

```
71 #Nube de palabras
72 # Define a function to plot word cloud
73 try:
74     def plot_cloud(wordcloud):
75         # Set figure size
76         plt.figure(figsize=(20, 10))
77         # Display image
78         plt.imshow(wordcloud)
79         # No axis details
80         plt.axis("off");
81
82     # Generate word cloud
83     wordcloud = WordCloud(width = 3000, height = 2000, random_state=1, background_color='salmon', colormap='Pastell',
84     | | | | | | | | | | collocations=False, stopwords = STOPWORDS, regexp= r"\\S[\\S']+" ).generate(' '.join(listacomplit))
85
86     # Plot
87     plot_cloud(wordcloud)
88 except ValueError:
89     print("No se puede generar nube de palabras, el usuario no registra etiquetas")
90 except AttributeError:
91     print("ID NO ENCONTRADO \\n\\n VUELVA A EJECUTAR EL PROGRAMA E INGRESE UN ID CORRECTO")
```

En cuanto a la interfaz del usuario, no es muy compleja, se trabajará con la consola de Google Collab, se le pedirá que el usuario ingrese un Id y empezara con la búsqueda, mostrará los resultados de la misma y generará la nube de palabras

```
... Ingrese un usuario:
|
```

```
... Ingrese un usuario:
32292
ID DE USUARIO:32292
NOMBRE: F.Igor
Procesando...
```

**D**: Ingrese un usuario:  
32292  
**ID DE USUARIO:** 32292  
**NOMBRE:** F-Igor  
Procesando...  
Las etiquetas del usuario F.Igor son:  
['javascript', 'postgresql', 'excel', 'class', 'php', 'google-chrome', 'angular', 'constructor', 'html', 'c#', 'ionic', 'css', 'mysql', 'ubuntu', 'phpmyadmin', 'navegador-web', 'sql', 'base-de-datos', 'react-native', 'oracle', 'java']  
Creando nube de palabras...

Conforme se implementaban funciones al software se procedía a probar, para comprobar la correcta implementación de las mismas, lo cual llevo a colocar condiciones o excepciones para evitar el mayor numero de errores, en la prueba final se procede a probar con diversos usuarios, con diferentes numero de etiquetas, o sin ninguna de ellas, que se recorra todas las páginas de etiquetas, que los datos sean correctos, se comprueba que muestre los datos necesarios y que la creación de la nube sea correcta con respecto a los parámetros establecidos.

```

➔ Ingrese un usuario:
107
ID DE USUARIO:107
NOMBRE: DavidGu
Procesando...
Las etiquetas del usuario DavidGu son:
[]
Creando nube de palabras...
No se puede generar nube de palabras, el usuario no registra etiquetas

```

← → ↻

stackoverflow.com/users/2217248/davidgu

🔍 user:2217248


Log in Sign up

stackoverflow

AboutProductsFor Teams

ProfileActivityDeveloper Story

HomePUBLICStack OverflowTagsUsersFIND A JOBJobsCompaniesTEAMSWhat's this?Free 30 Day Trial



1 REPUTATION

Communities (3)

Area 5151

Stack Overflow1

Stack Overflow en español1

Top network posts

We respect a laser-like focus on one topic.

Keeping a low profile.

This user hasn't posted yet.

davidgu.net

Member for 7 years, 4 months

1 profile view

Last seen Feb 11 '16 at 17:52

network profile

By using our site, you acknowledge that you have read and understand our [Cookie Policy](#), [Privacy Policy](#), and our [Terms of Service](#).

Ingrese un usuario:  
181183  
ID DE USUARIO:181183  
NOMBRE: mah.cr  
Procesando...  
Las etiquetas del usuario mah.cr son:  
['angular', 'html5', 'formularios', 'graphql', 'typescript', 'html', 'tooltip', 'service-workers', 'angular-material', 'validator', 'javascript', 'pwa']  
Creando nube de palabras...

typescript

angular-material

angular

html5

html

es.stackoverflow.com/users/181183/mah-cr?tab=tags

**stackoverflow**

user: 181183

Inicio Preguntas Etiquetas **Usuarios** Sin responder

Perfil **Actividad**

**REPUTACIÓN**

177

111 JUL 17 AGO 01 AGO 16

Etiqueta principal angular

Siguiente privilegio See reduced ads

Reputación de 200

**MEDALLAS**

7

Más reciente Resurgimiento

Siguiente medalla 0/1 Informado

**IMPACTO**

~125 personas alcanzadas

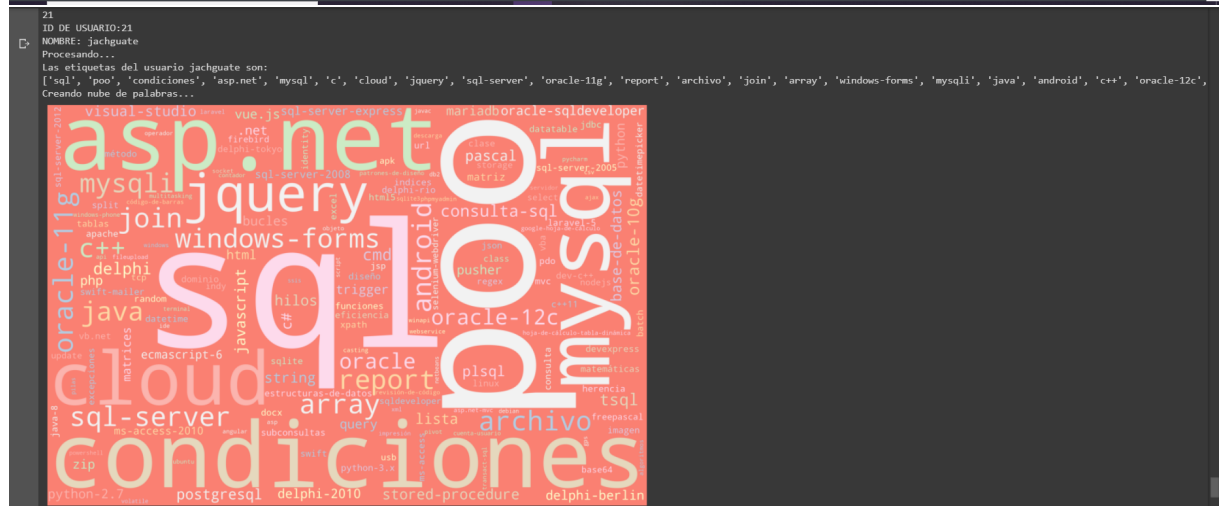
- 3 ediciones
- 0 reportes útiles
- 4 votos emitidos

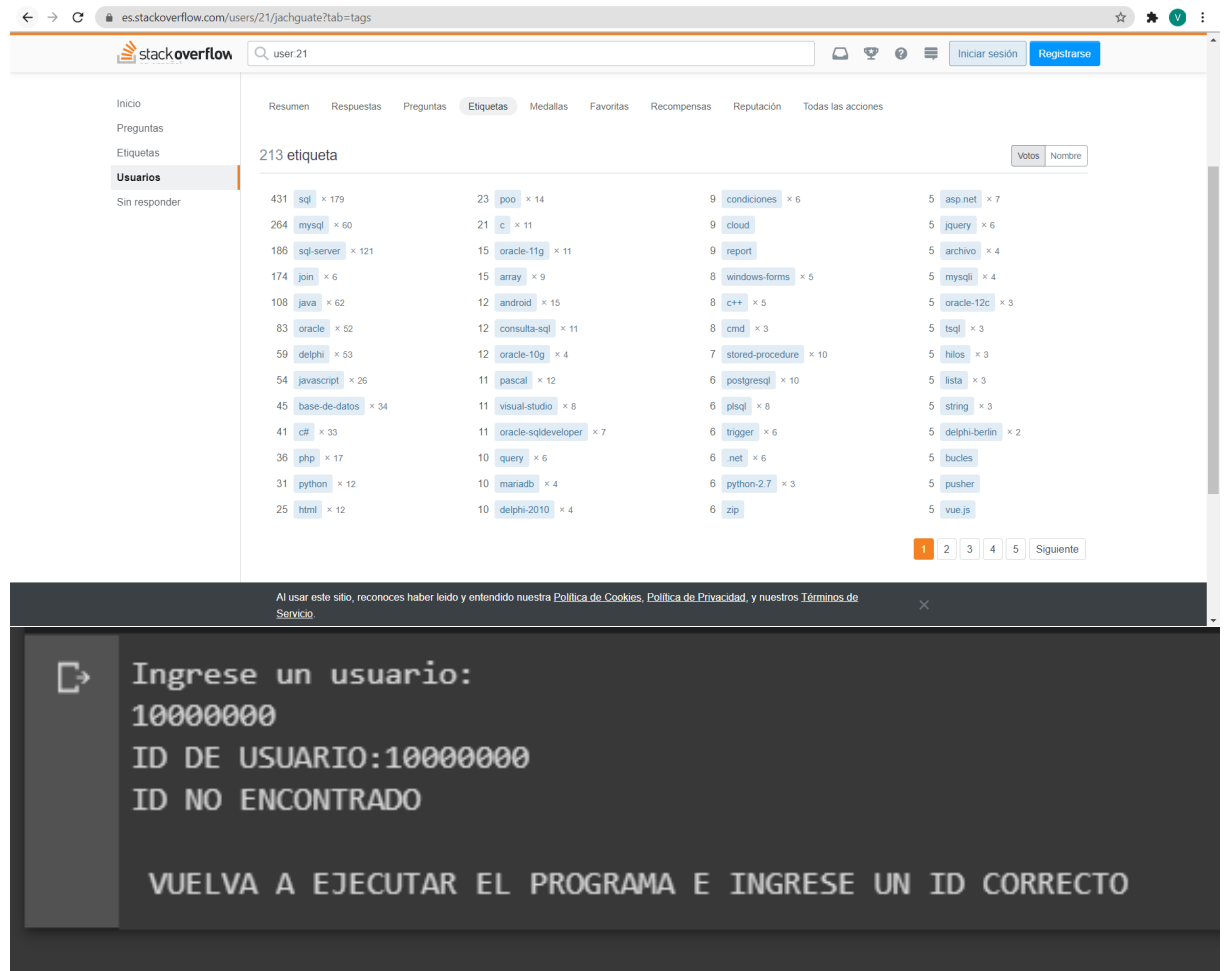
Resumen Respuestas Preguntas **Etiquetas** Medallas Favoritas Recompensas Reputación Todas las acciones

12 etiqueta

Votos Nombre

4 angular × 11	1 html5 × 2	0 formularios	0 graphql
4 typescript × 6	1 html	0 tooltip	0 service-workers
1 angular-material × 2	0 validator	0 javascript	0 pwa





## 6 MANTENIMIENTO

A partir de tener el software final se ha seguido revisando el código para comprobar su calidad y ha estado funcionando correctamente según los requerimientos planteados en un principio, hasta ahora no se agregado nuevos requerimientos, conforme el uso de la aplicación en un futuro se seguirá planteado posibles modificaciones, mantenimientos y actualizaciones.

## 7 CONCLUSIÓN

Se puede concluir que la realización e implementación del proyecto ha sido exitosa, se han cumplido todos los objetivos planteados en el principio, los requerimientos se materializaron, se cumple con el diseño propuesto, el código es funcional y las pruebas permitieron la detección de errores y mejoramiento de las diferentes partes del programa,

el mantenimiento se seguirá realizado para detectar posibles errores futuros, modificación de requerimientos o adaptación a nuevos entornos

El modelo de ciclo de vida de software escogido ha dado buenos resultados con este tipo de sistema, durante el desarrollo no se presentaron iteraciones o incrementos imprevistos, la planificación y análisis permitió que se realice todo de una forma ordenada, los métodos usados para la extracción de los datos y la generación de la nube también resultaron muy convenientes e interesantes para el cumplimiento del proyecto

[https://github.com/985emily/WORD\\_CLOUD\\_PROYECT.git](https://github.com/985emily/WORD_CLOUD_PROYECT.git)

## References

- [1] Educacion 3.0. *Cómo crear una nube de tags y sus usos en el aula*. Sept. 2018. URL: <https://www.educaciontrespuntocero.com/recursos/crear-una-nube-tags-las-palabras-mas-usadas-texto/>.
- [2] José Antonio García. *Representacion grafica 2D*. Mar. 2005. URL: <http://www.linux-magazine.es/issue/11/Matplotlib.pdf>.
- [3] Leonard Richardson. *Beautiful Soup is licensed under the same terms as Python itself*. Apr. 2012. URL: <https://www.crummy.com/software/BeautifulSoup/#Download>.
- [4] Daniel Tapia. *Modelo Cascada*. 2016. URL: <https://openclassrooms.com/en/courses/4309151-gestiona-tu-proyecto-de-desarrollo/4538221-en-que-consiste-el-modelo-en-cascada>.
- [5] unipython. *SOLICITUDES HTTP EN PYTHON CON REQUESTS*. Nov. 2018. URL: <https://unipython.com/solicitudes-http-en-python-con-requests/>.