Motores de Búsqueda

Construcción

Rogelio Ferreira Escutia

Profesor / Investigador
Tecnológico Nacional de México
Campus Morelia

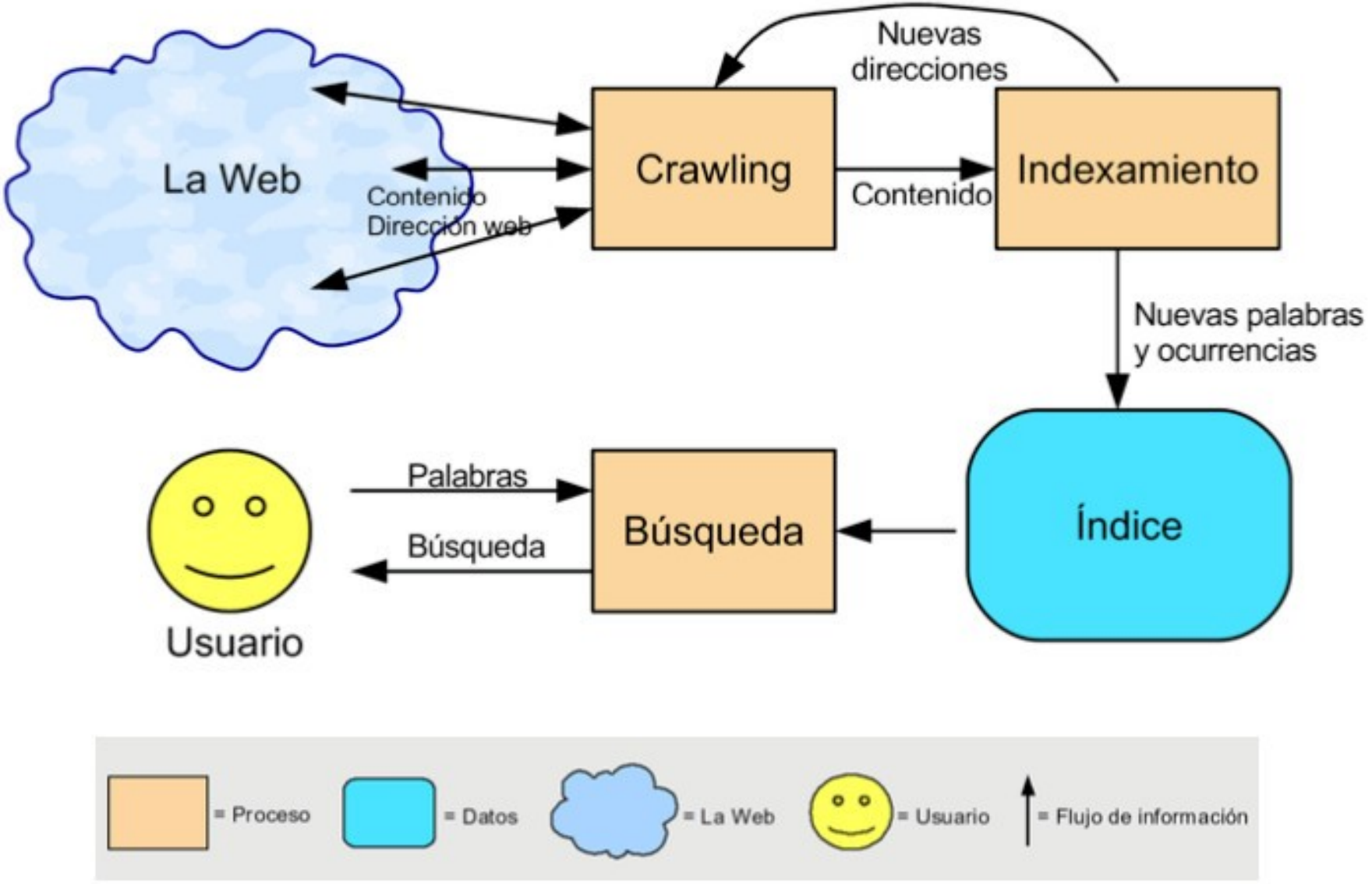
Arquitectura de un Motor de Búsqueda

Construcción de un Motor

- **1) Crawling:**
  - **Buscar páginas**
- **2) Indexado:**
  - **Análisis y creación de un índice de páginas**
- **3) Ranking:**
  - **Asignar importancia a las páginas**
- **4) Búsqueda**
  - **Interfaz y búsqueda del usuario final**

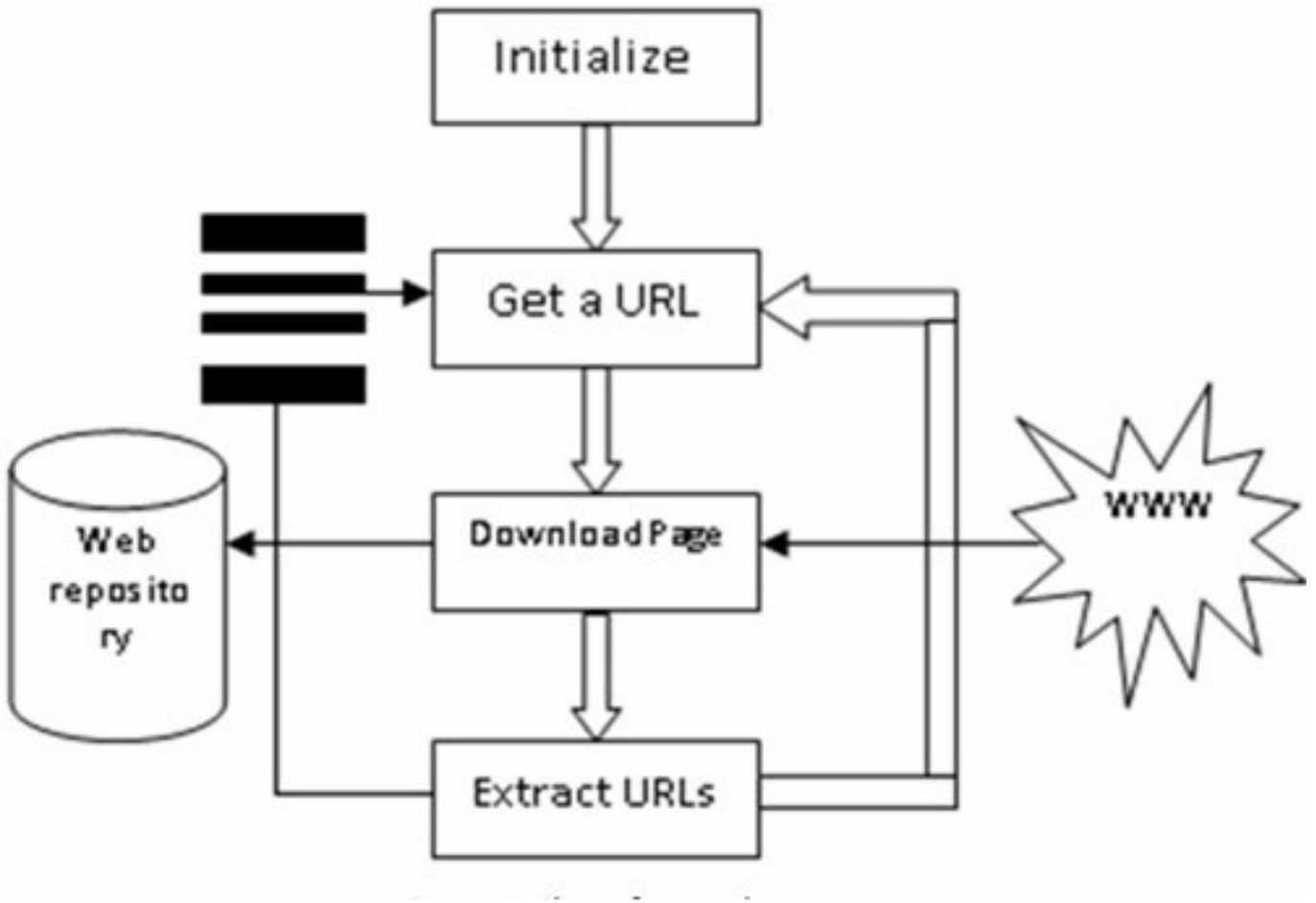
”Cómo funciona La Web”, http://www.ciw.cl/libroweb, 2008

Arquitectura de un Motor



”Cómo funciona La Web”, http://www.ciw.cl/libroweb, 2008

1) Crawling



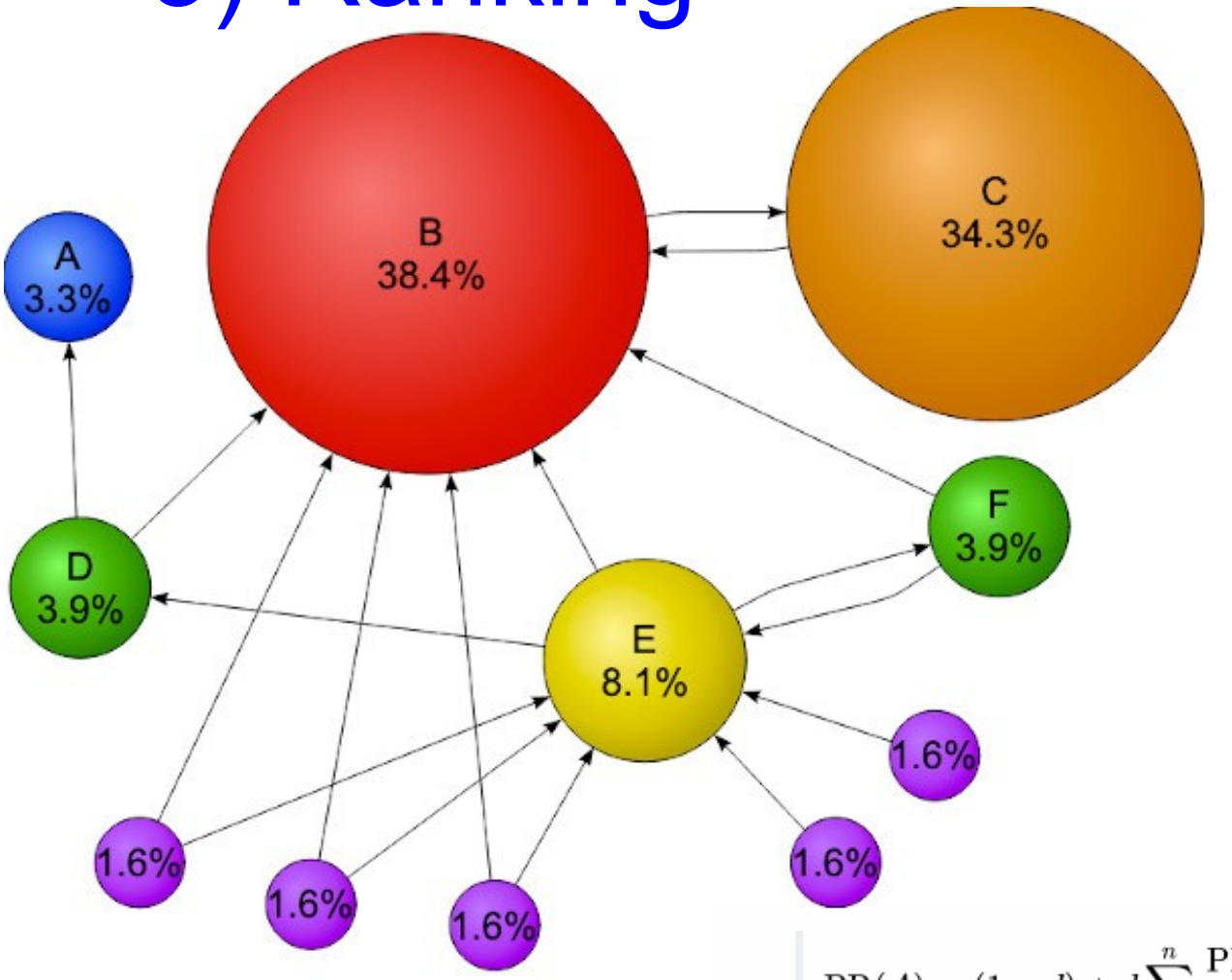
”How to build a Crawler”, https://www.octoparse.com/blog/how-to-build-a-crawler-to-extract-web-data-without-coding-skills-in-10-mins, marzo 2021

2) Indexado

	Índice invertido
	centro ----------> www.ciw.cl ecuador ----------> www.midiario.cl información ---------> www.viajes.com investigación -------> www.ciw.cl, www.midiario.cl policial ----------> www.midiario.cl presidenta ----------> www.midiario.cl viaje ----------> www.viajes.com, www.midiario.cl web ----------> www.ciw.cl

”Cómo funciona La Web”, http://www.ciw.cl/libroweb, 2008

3) Ranking



$$\mathrm{PR}(A) = (1-d) + d\sum_{i=1}^{n}\frac{\mathrm{PR}(i)}{C(i)}$$

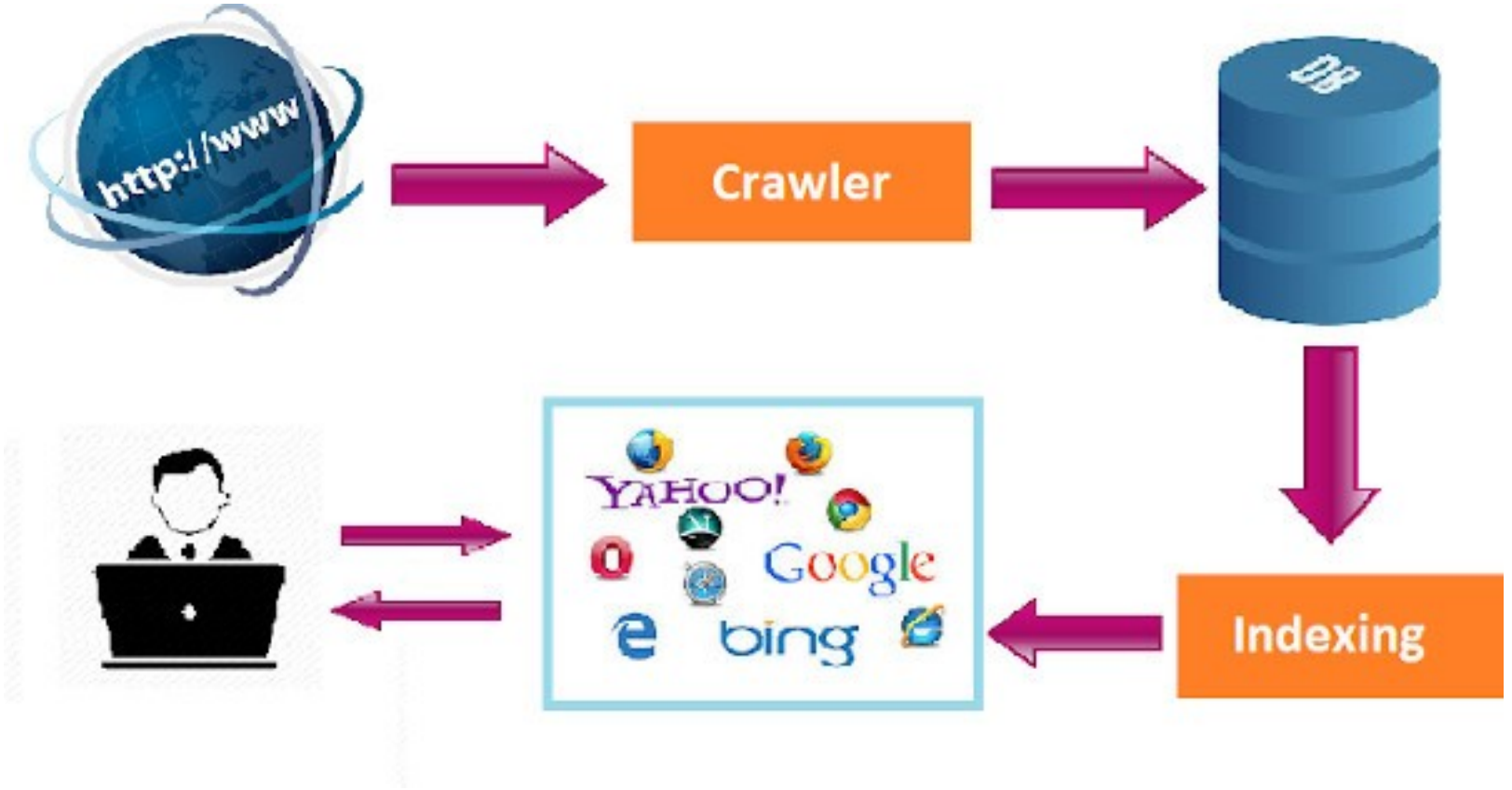
Donde:

- $\mathrm{PR}(A)$ es el PageRank de la página A.
- **d** es un factor de amortiguación que tiene un valor entre 0 y 1.
- $\mathrm{PR}(i)$ son los valores de PageRank que tienen cada una de las páginas **i** que enlazan a A.
- **C(i)** es el número total de enlaces salientes de la página i (sean o no hacia A).

”Page Rank”,

https://es.wikipedia.org/wiki/PageRank, marzo 2021

4) Búsqueda (interfaz)



”What is SEO?”, http://www.digitalabhyasa.com/what-is-seo/, marzo 2021

Construcción de un Motor de Búsqueda

Construcción de un Motor

- **A) Crawling**
- **B) Indexado**
- **C) Búsqueda de resultados**

Fase A) Crawling

1) Definir base de datos

- **Script para SQL:**

```
create database motor;
use motor;
create table indice (url text, analizado bit, palabra1 text, palabra2 text, palabra3 text);
```

2) Definir página inicial de arranque

- **Script para SQL:**

```
insert into indice values("http://www.xumarhu.net/", 0, "", "", "");
```

Base de datos

- **Estado actual de la tabla::**

url	analizado	palabra1	palabra2	palabra3
http://www.xumarhu.net/	0			

3) Buscar enlaces y guardar (1)

- **Iniciar con el programa en Python y crear una función para conectarse al DBMS:**

```
def conectar_dbms():
    import mysql.connector
    servidor = "localhost"
    usuario = "adriana"
    clave = "123"
    base = "motor"
    print("Conectándose al DBMS con los siguientes datos:")
    print("Servidor=",servidor,"- Usuario=",usuario,"- Clave=",clave,"- Base=",base)
    mi_conexion = mysql.connector.connect(
        host=servidor,
        user=usuario,
        passwd=clave,
        database=base
    )
    return mi_conexion
```

3) Buscar enlaces y guardar (2)

- **En el programa principal mandamos llamar a la función:**

```
# Conectarse a la base de datos
mi_conexion = conectar_dbms()
```

3) Buscar enlaces y guardar (3)

- **Instalar el Conector (en consola):**

```
MYSQL-CONNECTOR-PYTHON
Instalar el ODBC para conectar Python con MySQL
Sitio oficial: https://pypi.org/project/mysql-connector-python/
Para su instalación, ejecutar en la terminal:
> pip3 install mysql-connector-python
```

- **Si hay problemas de conexión se recomienda instalar una versión mas antigua del conector:**

```
> pip3 install mysql-connector-python==8.0.29
```

3) Buscar enlaces y guardar (4)

- **Crear una función para leer el primer enlace:**

```
def leer_primer_enlace(mi_conexion):
    try:
        if mi_conexion.is_connected():
            operacion = mi_conexion.cursor()
            operacion.execute( "SELECT * FROM indice WHERE analizado=0 LIMIT 1" )
            for url, analizado, palabra1, palabra2, palabra3 in operacion.fetchall()
            :
                primer_enlace = url
    except Error as e:
        print("Error al conectarse a MySQL: ", e)
    return primer_enlace
```

3) Buscar enlaces y guardar (5)

- **Llamar a la función para leer el primer enlace:**

```
# Extraer el primer enlace
primer_enlace = leer_primer_enlace(mi_conexion)
print("Primer Enlace: ", primer_enlace)
```

3) Buscar enlaces y guardar (6)

- **Función para extraer los enlaces de una página:**

```
def extraer_enlaces(enlace):
    from urllib.request import urlopen
    from bs4 import BeautifulSoup
    url = urlopen(enlace)
    print("\nExtraer los enlaces de la página Web: ", enlace)
    bs = BeautifulSoup(url.read(), 'html.parser')
    enlaces_encontrados = []
    contador = 0
    for enlaces in bs.find_all("a"):
        enlace = format(enlaces.get("href"))
        if enlace[-3:]!="jpg":
            enlaces_encontrados.append("{}".format(enlaces.get("href")))
            print("{}".format(enlaces.get("href")))
            contador = contador + 1
    print("\nEnlaces encontrados: ", contador)
    return enlaces_encontrados
```

3) Buscar enlaces y guardar (7)

- **Llamar a la Función para extraer los enlaces de una página:**

```
# Encontrar y extraer los enlaces encontrados
enlaces_encontrados = extraer_enlaces(primer_enlace)
```

3) Buscar enlaces y guardar (8)

- **Función para almacenar los enlaces:**

```
def almacenar_enlaces(mi_conexion, enlaces_encontrados):
    try:
        if mi_conexion.is_connected():
            operacion = mi_conexion.cursor()
            for i in range(len(enlaces_encontrados)):
                sql = "INSERT INTO indice (url, analizado, palabra1, palabra2,
                palabra3) VALUES('"+enlaces_encontrados[i]+"',0,'','','')"
                print(sql)
                operacion.execute("INSERT INTO indice (url, analizado, palabra1,
                palabra2, palabra3) VALUES('"+enlaces_encontrados[i]+"',0,'','','')
                ")
                mi_conexion.commit()
    except Error as e:
        print("Error al conectarse a MySQL: ", e)
    return
```

3) Buscar enlaces y guardar (9)

- **Llamar a la función para almacenar los enlaces:**

```
# Almacenar enlaces
almacenar_enlaces(mi_conexion, enlaces_encontrados)
```

3) Buscar enlaces y guardar (10)

- **Crear una función para marcar los enlaces ya revisados:**

```
def marcar_enlace_revisado(mi_conexion, enlace_a_marcar):
    try:
        if mi_conexion.is_connected():
            operacion = mi_conexion.cursor()
            sql = "UPDATE indice SET analizado=1 where url='"+enlace_a_marcar+"'
            LIMIT 1"
            operacion.execute(sql)
            mi_conexion.commit()
    except Error as e:
        print("Error al conectarse a MySQL: ", e)
    return
```

3) Buscar enlaces y guardar (11)

- **Llamar a la función para marcar los enlaces ya revisados:**

```
# Marcar enlace como revisado
enlace_a_marcar = primer_enlace
marcar_enlace_revisado(mi_conexion, enlace_a_marcar)
```

Resultados (terminal)

```
ARRANCANDO PONYGOOGLE

Leyendo primer enlace de la base de datos...
Conectándose al DBMS con los siguientes datos:
Servidor= localhost - Usuario= adriana - Clave= 123 - Base= motor
Primer Enlace:  http://www.xumarhu.net/

Extraer los enlaces de la página Web:  http://www.xumarhu.net/
index.html
bib_inic.htm
cur_inic.htm
dsc_inic.htm
ewe_inic.htm
pro_inic.htm
rfe_inic.htm
https://www.facebook.com/xumarhu.net
https://twitter.com/rogeplus
http://sagitario.itmorelia.edu.mx/~rogelio/
http://www.xumarhu.net/

Enlaces encontrados:  11
INSERT INTO indice (url, analizado, palabra1, palabra2, palabra3) VALUES('index.html',0,'','','')
INSERT INTO indice (url, analizado, palabra1, palabra2, palabra3) VALUES('bib_inic.htm',0,'','','')
INSERT INTO indice (url, analizado, palabra1, palabra2, palabra3) VALUES('cur_inic.htm',0,'','','')
INSERT INTO indice (url, analizado, palabra1, palabra2, palabra3) VALUES('dsc_inic.htm',0,'','','')
INSERT INTO indice (url, analizado, palabra1, palabra2, palabra3) VALUES('ewe_inic.htm',0,'','','')
INSERT INTO indice (url, analizado, palabra1, palabra2, palabra3) VALUES('pro_inic.htm',0,'','','')
INSERT INTO indice (url, analizado, palabra1, palabra2, palabra3) VALUES('rfe_inic.htm',0,'','','')
INSERT INTO indice (url, analizado, palabra1, palabra2, palabra3) VALUES('https://www.facebook.com/xumarhu.net',0,'','','')
INSERT INTO indice (url, analizado, palabra1, palabra2, palabra3) VALUES('https://twitter.com/rogeplus',0,'','','')
INSERT INTO indice (url, analizado, palabra1, palabra2, palabra3) VALUES('http://sagitario.itmorelia.edu.mx/~rogelio/',0,'','
,'')
INSERT INTO indice (url, analizado, palabra1, palabra2, palabra3) VALUES('http://www.xumarhu.net/',0,'','','')
```

Resultados (tabla)

url	analizado	palabra1	palabra2	palabra3
http://www.xumarhu.net/	1			
index.html	0			
bib_inic.htm	0			
cur_inic.htm	0			
dsc_inic.htm	0			
ewe_inic.htm	0			
pro_inic.htm	0			
rfe_inic.htm	0			
https://www.facebook.com/xumarhu.net	0			
https://twitter.com/rogeplus	0			
http://sagitario.itmorelia.edu.mx/~rogelio/	0			
http://www.xumarhu.net/	0			

Fase B) Indexado

4) Leer enlaces sin analizar

- **? ? ?**

5) Descargar la página de Internet

- **? ? ?**

6) Extraer texto (eliminar etiquetas)

- **? ? ?**

7) Eliminar signos y puntuaciones

- **? ? ?**

8) Eliminar StopWords

- **? ? ?**

9) Contar palabras

- **? ? ?**

10) Almacenar palabras mas repetidas

- **? ? ?**

Fase C) Búsqueda de Resultados

11) Crear formulario en HTML

- **? ? ?**

11) Página PHP para mostrar resultados

- **? ? ?**

Rogelio Ferreira Escutia

Profesor / Investigador
Tecnológico Nacional de México
Campus Morelia

rogelio.fe@morelia.tecnm.mx

rogeplus@gmail.com

xumarhu.net

@rogeplus

https://www.youtube.com/channel/UC0on88n3LwTKxJb8T09sGjg

rogelioferreiraescutia