



Jutta Steiner de Toruño

Alumnos:

- Johnny Alejandro Caballero Hernández
- Jorge Maximiliano Herrera Valladares
- Cesar Alexander Montes Martínez

Materia:

-MODULO

Año:

2022

Maestra: Caleb López.

Manual del Programador

INDICE

1.INTRODUCCION.	3
2.OBJETIVOS.	4
3.REQUERIMIENTOS TECNICOS DE INSTALACION.	5
4.EXPLICACION GENERAL DE LAS FASES DEL SISTEMA.	6
5.LA DESCRIPCION DE MODULOS O PROCESOS DETALLANDO.	12
ENTRADA.....	12
PROCESO-SALIDA.	12
6. DIAGRAMA DE CLASES.....	13
7. DIAGRAMA ENTIDAD RELACION.	14
8. DIAGRAMA CASO DE USO.....	15
9.DICCIONARIO DE DIAGRAMA CASO DE USO.	16
10.DICCIONARIO DE DATOS.....	17
11. DEFINICION Y DESCRIPCION DE VARIABLES USADAS EN EL SISTEMA.	19
12.INICIACION O IMPLEMENTACION DEL SISTEMA.	20

1.INTRODUCCION.

En este documento o Manual de Programador describe los aspectos técnicos informáticos del sistema de información. El documento familiariza al personal técnico especializado encargado de las actividades de mantenimiento, revisión, solución de problemas, instalación y configuración del sistema Centauri.

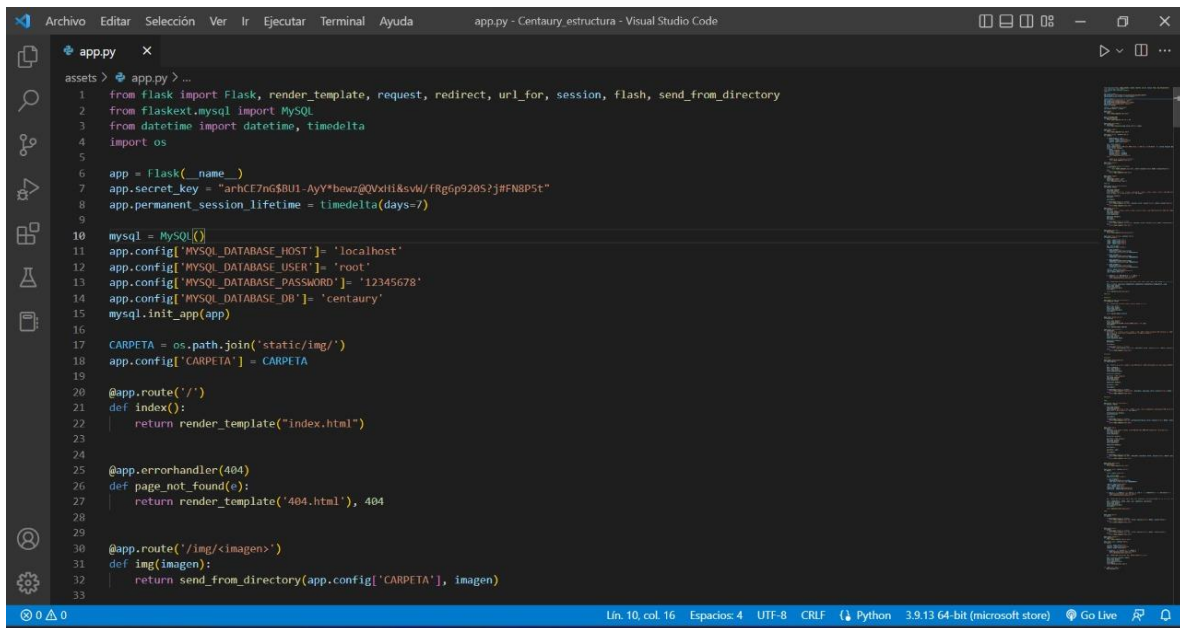
2.OBJETIVOS.

Instruir el uso adecuado del Sistema de Información, para el acceso oportuno y adecuado en la instalación de este, mostrando los pasos a seguir en el proceso de navegar en el sitio web, así como la descripción de los archivos relevantes del sistema los cuales nos orienten en la configuración y soporte de este.

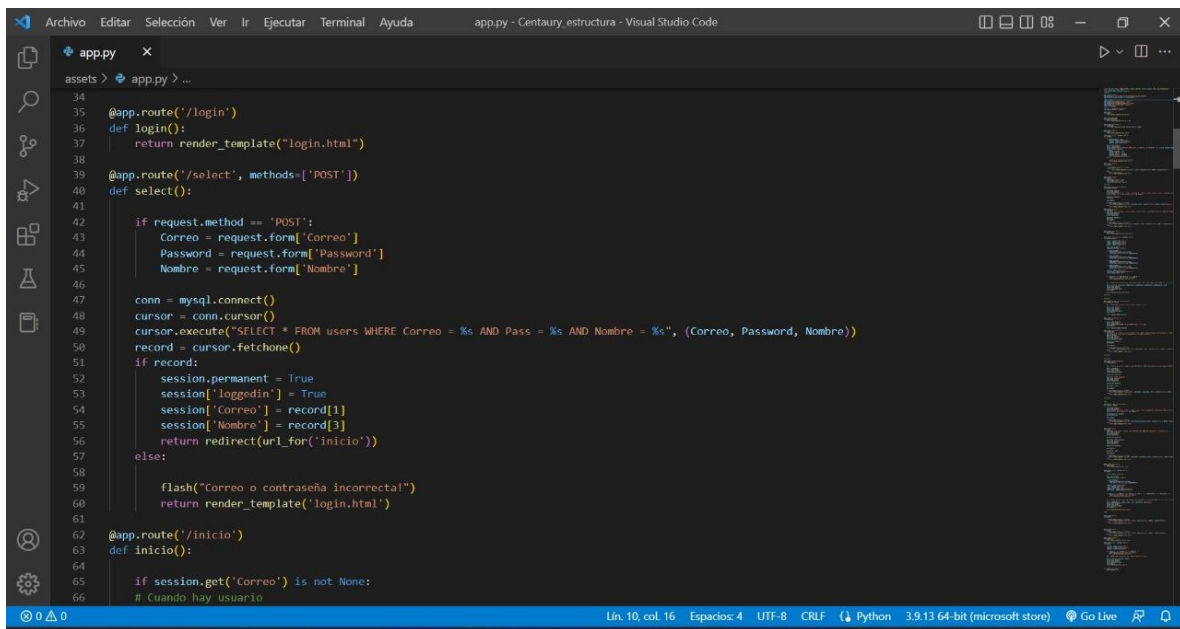
3.REQUERIMIENTOS TECNICOS DE INSTALACION.

- Sistema Operativo: Windows 7
- Procesador: Intel Core Celeron
- Memoria RAM: 1GB
- Disco Duro: 1GB
- Resolución de pantalla: 1280 x 720 pixeles
- Periféricos: Teclado, ratón, Bocinas (Opcional)

4.EXPLICACION GENERAL DE LAS FASES DEL SISTEMA.



```
1 from flask import Flask, render_template, request, redirect, url_for, session, flash, send_from_directory
2 from flaskext.mysql import MySQL
3 from datetime import datetime, timedelta
4 import os
5
6 app = Flask(__name__)
7 app.secret_key = "arhCE7nG$BU1-AyY*bewz@VvHl&svW/fRg6p920S?j#FN8P5t"
8 app.permanent_session_lifetime = timedelta(days=7)
9
10 mysql = MySQL()
11 app.config['MYSQL_DATABASE_HOST'] = 'localhost'
12 app.config['MYSQL_DATABASE_USER'] = 'root'
13 app.config['MYSQL_DATABASE_PASSWORD'] = '12345678'
14 app.config['MYSQL_DATABASE_DB'] = 'centaury'
15 mysql.init_app(app)
16
17 CARPETA = os.path.join('static/img/')
18 app.config['CARPETA'] = CARPETA
19
20 @app.route('/')
21 def index():
22     return render_template("index.html")
23
24
25 @app.errorhandler(404)
26 def page_not_found(e):
27     return render_template("404.html"), 404
28
29
30 @app.route('/img/<imagen>')
31 def img(imagen):
32     return send_from_directory(app.config['CARPETA'], imagen)
33
```



```
34
35 @app.route('/login')
36 def login():
37     return render_template("login.html")
38
39 @app.route('/select', methods=['POST'])
40 def select():
41
42     if request.method == 'POST':
43         Correo = request.form['Correo']
44         Password = request.form['Password']
45         Nombre = request.form['Nombre']
46
47         conn = mysql.connect()
48         cursor = conn.cursor()
49         cursor.execute("SELECT * FROM users WHERE Correo = %s AND Pass = %s AND Nombre = %s", (Correo, Password, Nombre))
50         record = cursor.fetchone()
51         if record:
52             session.permanent = True
53             session['logged_in'] = True
54             session['Correo'] = record[1]
55             session['Nombre'] = record[3]
56             return redirect(url_for('inicio'))
57         else:
58             flash("Correo o contraseña incorrecta!")
59             return render_template("login.html")
60
61
62 @app.route('/inicio')
63 def inicio():
64
65     if session.get('Correo') is not None:
66         # Cuando hay usuario
67
```

```
Archivo  Editar  Selección  Ver  Ir  Ejecutar  Terminal  Ayuda  app.py - Centaury_estructura - Visual Studio Code

assets > app.py > ...
67     return render_template('inicio.html', Correo = session['Correo'], Nombre = session['Nombre'])
68     else:
69         # Cuando no hay usuario
70         return render_template("index.html")
71
72
73 @app.route('/logout')
74 def logout():
75     session.pop('logged_in', None)
76     session.pop('Correo', None)
77     return redirect(url_for('login'))
78
79 #Noticias
80
81 @app.route('/news/<int:id>/<titulo>')
82 def news(id, titulo):
83
84     conn= mysql.connect()
85     cursor=conn.cursor()
86     cursor.execute("SELECT n.id , n.titulo, n.descripcion, n.foto1, n.foto2, n.foto3, n.foto4, c.tipo FROM noticias AS n INNER JOIN categoria AS c ON n.
87                       tipo=c.id WHERE n.id = %s AND n.titulo = %s", (id, titulo))
88
89     news=cursor.fetchall()
90     print(news)
91
92     conn.commit()
93
94     if session.get('Correo') is not None:
95         return render_template('news.html', news=news, Correo = session['Correo'], Nombre = session['Nombre'])
96     else:
97         return render_template("index.html")
98
99 @app.route('/noticias')
```

```
Archivo  Editar  Selección  Ver  Ir  Ejecutar  Terminal  Ayuda  app.py - Centaury_estructura - Visual Studio Code

assets > app.py > ...
99 def noticias():
100     sql="SELECT n.id , n.titulo, n.foto1, n.foto2, n.foto3, n.foto4, c.tipo FROM noticias AS n INNER JOIN categoria AS c ON n.tipo=c.id;"
101     conn= mysql.connect()
102     cursor=conn.cursor()
103     cursor.execute(sql)
104
105     news=cursor.fetchall()
106     print(news)
107     conn.commit()
108
109     if session.get('Correo') is not None:
110         return render_template('noticias.html', news=news, Correo = session['Correo'], Nombre = session['Nombre'])
111     else:
112         return render_template("index.html")
113
114
115
116 @app.route("/data_news")
117 def data_news():
118     return render_template('dato_noticias.html')
119
120
121 @app.route('/datos_noticias', methods=['POST'])
122 def datos_noticias():
123
124     _foto1 = request.files["foto1"]
125     _foto2 = request.files["foto2"]
126     _foto3 = request.files["foto3"]
127     _foto4 = request.files["foto4"]
128
129     now = datetime.now()
130     tiempo=now.strftime('%Y-%m-%s')
131
```

```
Archivo  Editar  Selección  Ver  Ir  Ejecutar  Terminal  Ayuda  app.py - Centaury.estructura - Visual Studio Code

assets > app.py > ...
132 if _foto1.filename!='':
133     NewNamePhoto1=tiempo+_foto1.filename
134     _foto1.save('assets/static/img/'+NewNamePhoto1)
135
136 if _foto2.filename!='':
137     NewNamePhoto2=tiempo+_foto2.filename
138     _foto2.save('assets/static/img/'+NewNamePhoto2)
139
140 if _foto3.filename!='':
141     NewNamePhoto3=tiempo+_foto3.filename
142     _foto3.save('assets/static/img/'+NewNamePhoto3)
143
144 if _foto4.filename!='':
145     NewNamePhoto4=tiempo+_foto4.filename
146     _foto4.save('assets/static/img/'+NewNamePhoto4)
147
148 _titulo = request.form['titulo']
149 _descripcion = request.form['descripcion']
150 _tipo = request.form['tipo']
151
152
153 if _titulo == '' or _descripcion == '' or _tipo == '':
154     flash('Recuerda completar todos los campos')
155     return redirect(url_for('datos_wiki'))
156
157
158 sql = "INSERT INTO noticias (titulo, descripcion, foto1, foto2, foto3, foto4, tipo) VALUES (%s, %s, %s, %s, %s, %s, %s);"
159
160 datos = (_titulo, _descripcion, NewNamePhoto1, NewNamePhoto2, NewNamePhoto3, NewNamePhoto4, _tipo)
161 conn = mysql.connect()
162 cursor= conn.cursor()
163 cursor.execute(sql,datos)
164 conn.commit()
```

```
Archivo  Editar  Selección  Ver  Ir  Ejecutar  Terminal  Ayuda  app.py - Centaury.estructura - Visual Studio Code

assets > app.py > ...
198 def intr(Correo):
199     sql="SELECT i.id ,i.interes, w.foto ,w.nombre, c.tipo, COUNT(w.nombre) AS nombre2 FROM intereses AS i INNER JOIN wiki AS w ON i.interes=w.id INNER
200     JOIN categoria AS c ON w.tipo=c.id WHERE usuario = %s GROUP BY w.nombre;"
201     datos = (Correo)
202     conn= mysql.connect()
203     cursor=conn.cursor()
204     cursor.execute(sql, datos)
205
206     datos=cursor.fetchall()
207     print(datos)
208
209     conn.commit()
210
211 if session.get('Correo') is not None:
212     return render_template('interes.html', datos=datos, Correo = session['Correo'], Nombre = session['Nombre'])
213 else:
214     return render_template("index.html")
215
216 #Intereses
217
218 #Categoria
219
220 @app.route('/tipo/<categoria>')
221 def tipo(categoria):
222
223     sql = "SELECT w.id, w.foto ,w.nombre, c.tipo FROM wiki AS w INNER JOIN categoria AS c ON w.tipo=c.id WHERE c.tipo = %s;"
224
225     datos = (categoria )
226     conn = mysql.connect()
227     cursor= conn.cursor()
```



```
Archivo  Editar  Selección  Ver  Ir  Ejecutar  Terminal  Ayuda  app.py - Centaury_estructura - Visual Studio Code

assets > app.py > ...
198 def intr(Correo):
199     sql="SELECT i.id ,i.interes, w.foto ,w.nombre, c.tipo, COUNT(w.nombre) AS nombre2 FROM intereses AS i INNER JOIN wiki AS w ON i.interes=w.id INNER
200     JOIN categoria AS c ON w.tipo=c.id WHERE usuario = %s GROUP BY w.nombre;"
201     datos = (Correo)
202     conn= mysql.connect()
203     cursor=conn.cursor()
204     cursor.execute(sql, datos)
205
206     datos=cursor.fetchall()
207     print(datos)
208
209     conn.commit()
210
211     if session.get('Correo') is not None:
212         return render_template('interes.html', datos=datos, Correo = session['Correo'], Nombre = session['Nombre'])
213     else:
214         return render_template("index.html")
215
216 #Intereses
217
218 #Categoria
219
220 @app.route('/tipo/<categoria>')
221 def tipo(categoria):
222
223     sql = "SELECT w.id, w.foto ,w.nombre, c.tipo FROM wiki AS w INNER JOIN categoria AS c ON w.tipo=c.id WHERE c.tipo = %s ;"
224
225     datos = (categoria )
226     conn = mysql.connect()
227     cursor= conn.cursor()
228
229 Lin. 10, col. 16  Espacios: 4  UTF-8  CRLF  Python  3.9.13 64-bit (microsoft store)  Go Live  🔔
```

```
Archivo  Editar  Selección  Ver  Ir  Ejecutar  Terminal  Ayuda  app.py - Centaury_estructura - Visual Studio Code

assets > app.py > ...
230 cursor.execute(sql,datos)
231
232 datos=cursor.fetchall()
233
234 sql="SELECT * FROM categoria;"
235 conn= mysql.connect()
236 cursor=conn.cursor()
237 cursor.execute(sql)
238
239 tipos=cursor.fetchall()
240
241 print(datos, tipos)
242
243 conn.commit()
244
245 if session.get('Correo') is not None:
246     return render_template('categoria.html', datos=datos, tipos=tipos, Correo = session['Correo'], Nombre = session['Nombre'])
247 else:
248     return render_template("index.html")
249
250
251 #Categoria
252
253 #wiki
254
255 @app.route('/wiki/<int:id>/<nombre>')
256 def wiki(id, nombre):
257
258     conn= mysql.connect()
259     cursor=conn.cursor()
260     cursor.execute("SELECT w.id, w.foto, w.nombre, c.tipo, w.size, w.temperatura, w.descripcion FROM wiki AS w INNER JOIN categoria AS c ON w.tipo=c.id
261     WHERE w.id = %s AND w.nombre = %s", (id, nombre))
262
263 Lin. 10, col. 16  Espacios: 4  UTF-8  CRLF  Python  3.9.13 64-bit (microsoft store)  Go Live  🔔
```

```
Archivo  Editar  Selección  Ver  Ir  Ejecutar  Terminal  Ayuda  app.py - Centaury_estructura - Visual Studio Code

assets > app.py > ...
230 cursor.execute(sql,datos)
231
232 datos=cursor.fetchall()
233
234 sql="SELECT * FROM categoria;"
235 conn= mysql.connect()
236 cursor=conn.cursor()
237 cursor.execute(sql)
238
239 tipos=cursor.fetchall()
240
241 print(datos, tipos)
242
243 conn.commit()
244
245 if session.get('Correo') is not None:
246     return render_template('categoria.html', datos=datos, tipos=tipos, Correo = session['Correo'], Nombre = session['Nombre'])
247 else:
248     return render_template("index.html")
249
250
251 #Categoria
252
253 #Wiki
254
255 @app.route('/wiki/<int:id>/<nombre>')
256 def wiki(id, nombre):
257
258     conn= mysql.connect()
259     cursor=conn.cursor()
260     cursor.execute("SELECT w.id, w.foto, w.nombre, c.tipo, w.size, w.temperatura, w.descripcion FROM wiki AS w INNER JOIN categoria AS c ON w.tipo=c.id
261 WHERE w.id = %s AND w.nombre = %s", (id, nombre))
```

```
Archivo  Editar  Selección  Ver  Ir  Ejecutar  Terminal  Ayuda  app.py - Centaury_estructura - Visual Studio Code

assets > app.py > ...
295 conn.commit()
296
297 if session.get('Correo') is not None:
298     return render_template('wiki.html', datos=datos, tipos=tipos, Correo = session['Correo'], Nombre = session['Nombre'])
299 else:
300     return render_template("index.html")
301
302
303
304 @app.route("/datos_wiki")
305 def datos_wiki():
306     return render_template('datos.html')
307
308
309 @app.route('/datos', methods=['POST'])
310 def datos():
311
312     _foto = request.files['foto']
313
314     now = datetime.now()
315     tiempo=now.strftime('%Y%m%d%H%M%S')
316
317     if _foto.filename!='':
318         NewNamePhoto=tiempo+_foto.filename
319         _foto.save('assets/static/img/'+NewNamePhoto)
320
321     _nombre = request.form['nombre']
322     _tipo = request.form['tipo']
323     _size = request.form['size']
324     _temperatura = request.form['temperatura']
325     _descripcion = request.form['descripcion']
326
327
```

```
Archivo  Editar  Selección  Ver  Ir  Ejecutar  Terminal  Ayuda  app.py - Centaury.estructura - Visual Studio Code

assets > app.py > ...
328     if _foto == '' or _nombre == '' or _tipo == '' or _size == '' or _temperatura == '' or _descripcion == '':
329         flash('Recuerda completar todos los campos')
330         return redirect(url_for('datos_wiki'))
331
332
333     sql = "INSERT INTO wiki (foto, nombre, tipo, size, temperatura, descripcion) VALUES (%s, %s, %s, %s, %s, %s);"
334
335     datos = (NewNamePhoto, _nombre, _tipo, _size, _temperatura, _descripcion)
336     conn = mysql.connect()
337     cursor= conn.cursor()
338     cursor.execute(sql,datos)
339     conn.commit()
340
341     return redirect(url_for('datos_wiki'))
342
343 mWiki
344
345
346 @app.route('/quizz')
347 def quizz():
348
349     if session.get('Correo') is not None:
350         return render_template('quizz.html', Correo = session['Correo'], Nombre = session['Nombre'])
351     else:
352         return render_template("index.html")
353
354
355
356
357 @app.route("/solar")
358 def solar():
359     if session.get('Correo') is not None:
360         return render_template('solar.html', Correo = session['Correo'], Nombre = session['Nombre'])
```

```
Archivo  Editar  Selección  Ver  Ir  Ejecutar  Terminal  Ayuda  app.py - Centaury.estructura - Visual Studio Code

assets > app.py > ...
361     else:
362         return render_template("index.html")
363
364 @app.route('/registro')
365 def registro():
366     return render_template("registro.html")
367
368 @app.route('/store', methods=['POST'])
369 def store():
370
371     _Correo = request.form['Correo']
372     _Password = request.form['Password']
373     _Nombre = request.form['Nombre']
374
375     if _Correo == '' or _Password == '' or _Nombre == '':
376         flash('Recuerda completar todos los campos')
377         return redirect(url_for('registro'))
378
379     sql = "INSERT INTO users(Correo, Pass, Nombre) VALUES (%s,%s,%s);"
380
381     datos = (_Correo, _Password, _Nombre)
382     conn = mysql.connect()
383     cursor= conn.cursor()
384     cursor.execute(sql,datos)
385     conn.commit()
386     return redirect(url_for('login'))
387
388
389 if __name__ == '__main__':
390     app.run(debug=True)
```

5.LA DESCRIPCION DE MODULOS O PROCESOS DETALLANDO.

ENTRADA

-login: donde el usuario deberá rellenar lo huecos con sus datos, nombre, correo y contraseña.

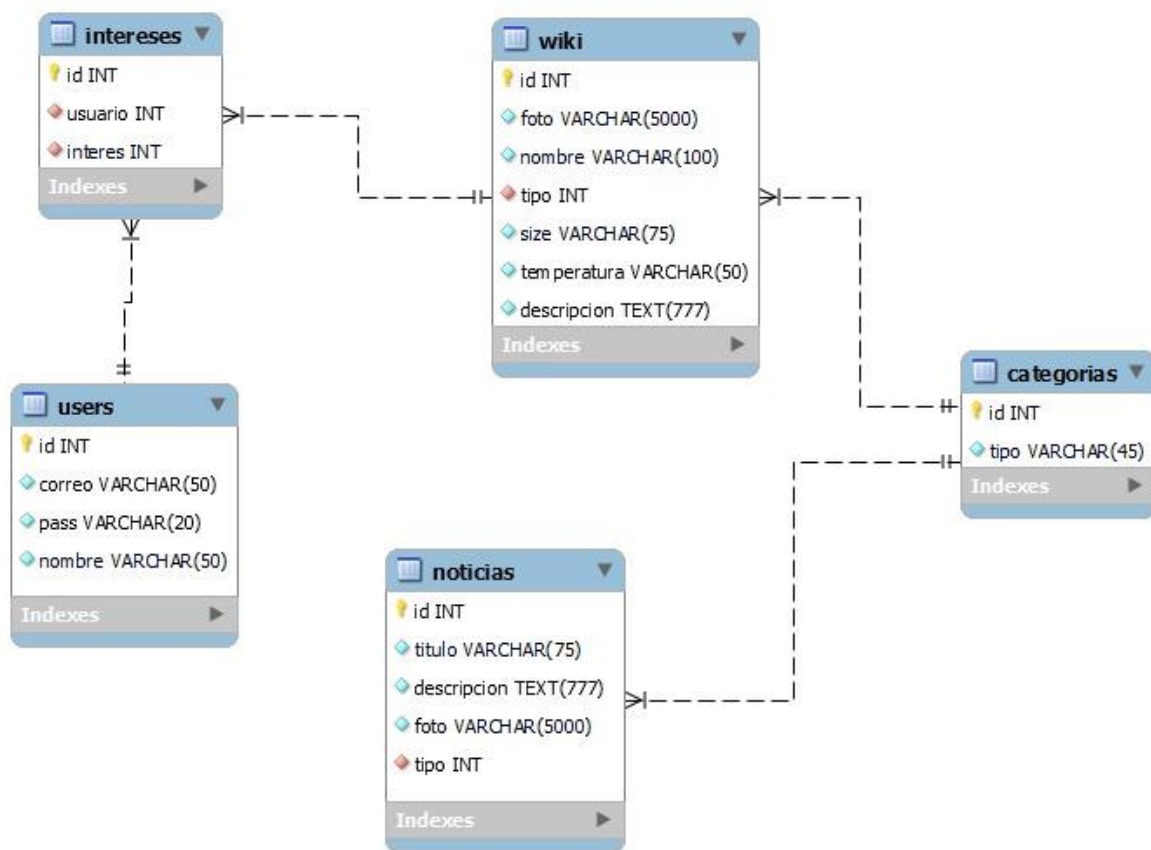
-Register: donde el usuario debe colocar los datos anteriormente usados para poder iniciar sesión en página web.

PROCESO-SALIDA.

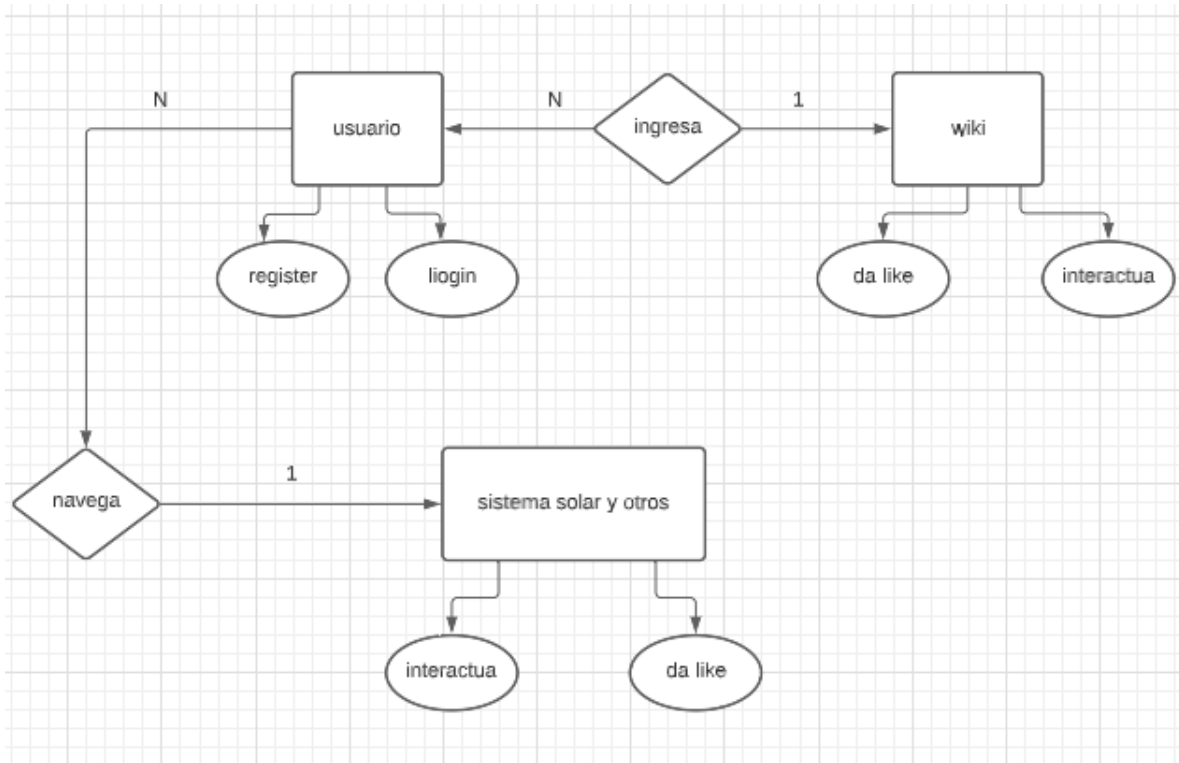
-El usuario ya puede navegar en las diferentes interfases del sistema web.

-el usuario si quiere puede cerrar la sesión de la página para más seguridad.

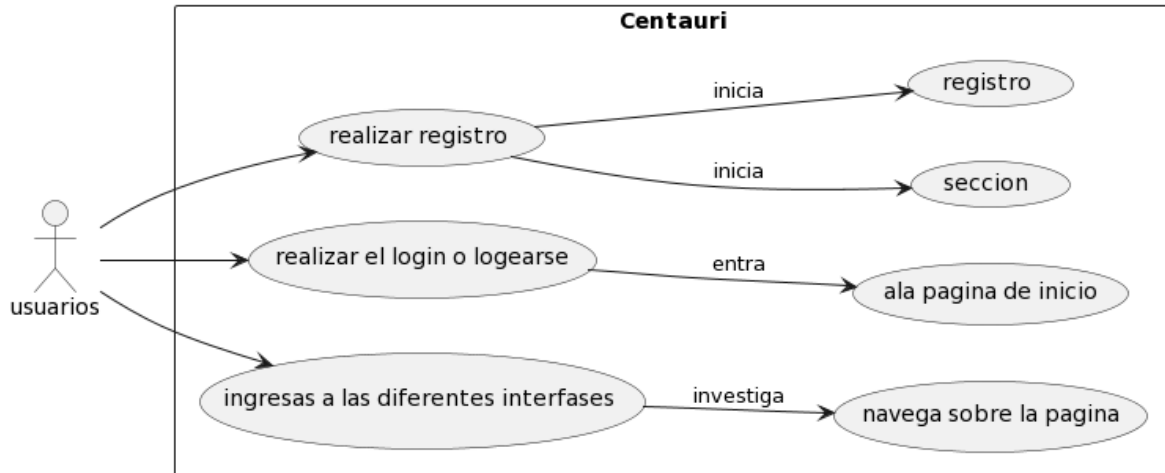
6. DIAGRAMA DE CLASES.



7. DIAGRAMA ENTIDAD RELACION.



8. DIAGRAMA CASO DE USO.



```
@startuml Centauri

left to right direction
actor "usuarios" as fc1

rectangle Centauri {
    usecase "realizar registro" as UC1
    usecase "realizar el login o logearse" as UC2
    usecase "ingresas a las diferentes interfases " as UC3

    (UC1) --> (seccion): inicia
    (UC1) --> (registro): inicia
    (UC2) --> (ala pagina de inicio): entra
    (UC3) --> (navega sobre la pagina): investiga
}

fc1 --> UC1
fc1 --> UC2
fc1 --> UC3

@enduml
```

9.DICCIONARIO DE DIAGRAMA CASO DE USO.

Nombre	Centauri
Autor	Usuarios
Autores implicados	Usuarios
Descripción	El sistema deberá permitir a los usuarios poder ver y navegar atreves de las diferentes interfases.
Precondiciones	El usuario debe registrarse e iniciar sesión.
Flujo normal de eventos	1.El usuario debe registrase. 2. El usuario debe logearse. 3.El usuario ya puede navegar en las diferentes interfases.
Flujos alternativos	El usuario olvida la contraseña. -Se puede cambiar la contraseña en el apartado olvido su contraseña.
Postcondiciones	Luego de solucionar el error se le pide una contraseña nueva
Excepciones	Ninguna

10.DICCIONARIO DE DATOS.

Usuarios			
Nombre campo	Tipo dato	Tamaño	Descripción
Id	INT		
Correo	Varchar	50	Correo del usuario
Pass	Varchar	20	Contraseña del usuario
Nombre	Varchar	50	Nombre de usuario

Intereses			
Nombre campo	Tipo dato	Tamaño	Descripción
Id	INT		
Usuarios	INT		Usuarios registrados
intereses	INT		Las cosas que le gustan al usuario

Wiki			
Nombre campo	Tipo dato	Tamaño	Descripción
Id	INT		
Foto	Varchar	5000	Imagen del planeta o estrella
Nombre	Varchar	100	Nombre del planeta
Tipo	INT		Tipo de al que pertenece
Size	Varchar	75	tamaño
Temperatura	Varchar	50	Temperatura del planeta
Descripción	TEXT	777	Descripción del planeta

Categoría			
Nombre campo	Tipo dato	Tamaño	Descripción
Id	INT		
Tipo	Varchar		Si es planeta o estrella u otro, etc.

Noticias			
Nombre campo	Tipo dato	Tamaño	Descripción
Id	INT		
Título	Varchar	75	Título del planeta u otro
Descripción	TEXT	777	Descripción sobre el planeta u otro
Foto	Varchar	5000	Imagen del planeta u otro

11. DEFINICION Y DESCRIPCION DE VARIABLES USADAS EN EL SISTEMA.

Ciclo for, If, Def, Try except, Return, Else, Flash.

-Estos fueron usados Para mostrar un mensaje en el HTML

Render template, Sección

-Estos fueron usados Para crear sesiones en la pagina

Os.path.join

-Estos fueron usados Para buscar las imágenes en las rutas de las carpetas

Request, Redirect, url_for, send_from_directory.

-Estos fueron usados Para mandar a llamar las imágenes de la carpeta que hemos creado.

12.INICIACION O IMPLEMENTACION DEL SISTEMA.

Paso1: ingresar al navegador de su preferencia.

Paso2: Buscar en el navegador la página web Centauri.

Paso3: Una vez haya entrado a la página, deberá registrarse con un correo, nombre y una contraseña.

Paso4: posteriormente realizara el login o logearse en la página con su nombre de usuario, correo y su contraseña.

Paso5: habiendo echo lo anterior ya podrá tener acceso a todo el contenido de la página web.