



CONCEPCIÓN DEL URUGUAY

LICENCIATURA EN SISTEMAS DE INFORMACIÓN

Bases de Datos NSQL.

Docentes: Dr Bel Walter - Lic Ledesma Ernesto

Trabajo Número 3: RedisApi.

Alumnas:

Abrigo Andrea

Becker Laila

Opción #1 - Api Turismo

1) Deberá tener los siguientes grupos de interés:

- a. Cervecerías artesanales
- b. Universidades
- c. Farmacias
- d. Centro de atención de emergencias
- e. Supermercados

```
@app.route('/cargarGrupo')
def cargarGrupo():
    conexion = connect_db()
    conexion.geoadd('cervecerias', '-58.233851', '-32.480258',
'Drakkar', '-58.237062', '-32.484577', 'Bigua', '-58.238009',
'-32.480966', 'Tractor', '-58.235332', '-32.479629', '7 colinas',
'-58.232669', '-32.481815', 'Ambar')
    conexion.geoadd('universidades', '-58.2333326', '-32.479067',
'UADER FCyT', '-58.2320773', '-32.4806146', 'UCU', '-58.2593929',
'-32.4826601', 'UNER Salud', '-58.2379138', '-32.4838185', 'UNER
Rectorado', '-58.2343733', '-32.4925787', 'UTN')
    conexion.geoadd('farmacias', '-58.2327425', '-32.4849769',
'Alberdi', '-58.2327425', '-32.4849769', 'Argentina', '-58.2327425',
'-32.4849769', 'Popular', '-58.2327425', '-32.4849769', 'Central',
'-58.2327425', '-32.4849769', 'Don Bosco', '-58.2327425',
'-32.4849769', 'Ramirez')
    conexion.geoadd('emergencias', '-58.2322039', '-32.4830237',
'Vida', '-58.2356908', '-32.4831685', 'Alerta', '-58.2312003',
'-32.4832777', 'Clinica', '-58.2389064', '-32.479766', 'Cooperativa')
    conexion.geoadd('supermercados', '-58.2360203',
'-32.4839382', 'Supremo', '-58.2341428', '-32.4872144', 'GranRex',
'-58.2411165', '-32.4877664', 'Dia', '-58.240505', '-32.4828247',
'Estrella', '-58.240505', '-32.4828247', 'Guri', '-58.240505',
'-32.4828247', 'RT', '-58.240505', '-32.4828247', 'Impulso')
    return "hecho"
```

2) Cargar lugares a los grupos de interés:

```
@app.route('/nuevoLugar', methods=["POST"])
def nuevoLugar():
    if request.method == 'POST':
        nombre = request.form['nombre']
        titulo = request.form['titulo']
        lugar = request.form['lugar']
        latitud = request.form['latitud']
        longitud = request.form['longitud']

    conexion = connect_db()
    conexion.geoadadd(nombre, longitud, latitud, lugar)
    return render_template('/cargado.html', nombre=nombre,
lugar=lugar)
```

localhost:5000/cargarLugar

Agregar cerveceria

Ingresa el nombre

Ingresa la latitud

Ingresa la longitud

Agregar

3) Lugares de cada grupo que estén dentro de un radio de 5 km:

```
@app.route('/lugaresCerca', methods=["POST"])
def lugaresCerca():
    lug = []
    if request.method == 'POST':
        nombre = request.form['nombre']
        conexion = connect_db()
        ubicacion=obtener_ubicacion()
        coordenadas=ubicacion.split(",")
        latitud=coordenadas[0]
        longitud=coordenadas[1]
        aux =
str(conexion.georadius(nombre,longitud,latitud,'5','km'))
        aux=aux.replace("[", '')
        aux=aux.replace("]", '')
        aux=aux.replace("'", '')
        aux=aux.replace("b", '')
        lugares = aux.split(",")
        for lugar in lugares:
            lug.append(lugar.lstrip())
        titulo = nombre.capitalize()
        return render_template('/lugaresCerca.html',lugares=lug,
nombre=nombre, titulo=titulo)
```



- 4) Distancia entre la ubicación del usuario y uno de los puntos elegido por el usuario de los mostrados en el punto anterior.

```
app.route('/devolverDistancia', methods=["POST"])
def devolverDistancia():
    if request.method == 'POST':
        nombre = request.form['nombre']
        lugar = request.form['lugar']
        aux = []
        conexion = connect_db()
        aux = str(conexion.geodist(nombre,"here", lugar, 'm'))
        return render_template('/devolverDistancia.html', dato = aux, lugar
= lugar, nombre=nombre)
```

