Para utilizar pyhon o comandos de django debo poner en el path de Windows

los caminos donde se encuentran los archivos Python.exe y django-admin.exe

Pip install Django

django-admin startproject mysite

pip install Django-cors-headers

pip install mysql-connector-python

pip install pymysql

pip install mysqlclient

pip install passlib

pip install reportlab

pip install numpy

pip install boto3

pip install pypdf2

pip install pandas \*\*

pip install fullcalendar

python manage.py createsuperuser Crear Superusuarios

python manage.py inspectdb > models.py

PIP INSTALL -U pytest

Npm install http-server -g

python manage.py runserver 0.0.0.0:8000

DEJAR ALLOWED\_HOSTS = ['192.168.0.2']

ESTO DEBE ESTAR EN BLANCO SE PUSO PARA PROBAR INTRANET CASA

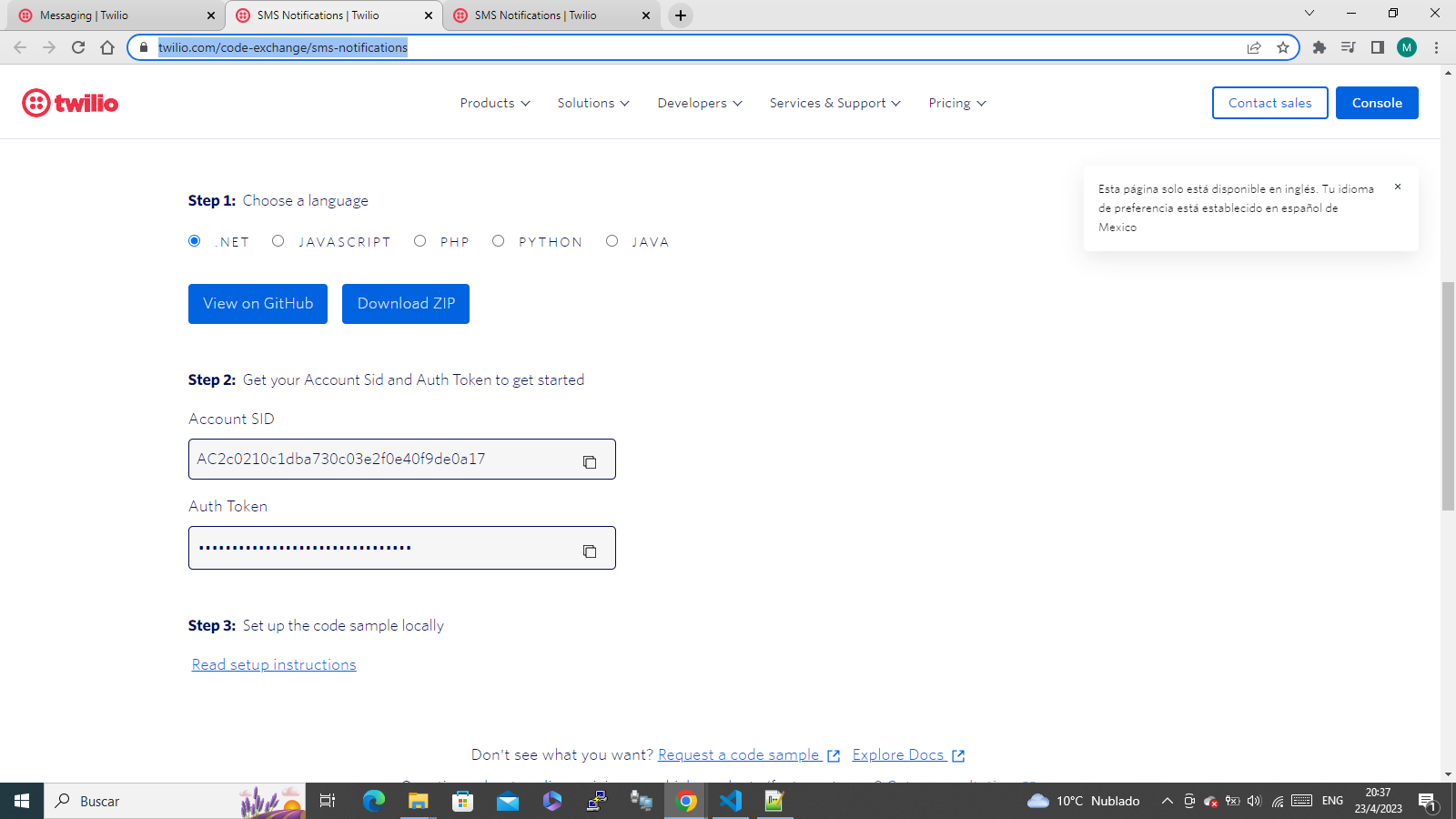
#Npm install –g jquery

Pip install requested

npm install twilio-cli –g

pip install twilio

La parametrizacion esta en Parametros.



El numero esta en esa tabla y el token hay que copiarlo de aca

arranque servidor bluehost

cd /home/mwsdlgmy/public\_html/siac4

source/project/bin/activate

python manage,py runser

Pip install html2pdf

pip install django-cors-headers

pip install django-session-security

python.exe -m pip install --upgrade pip

pip --no-cache-dir install -U matplotlib

Pip install flask

Pip install django\_excel

Pip install mail

Pip install django-session-timeout

pip install django-crispy-forms ----- :login

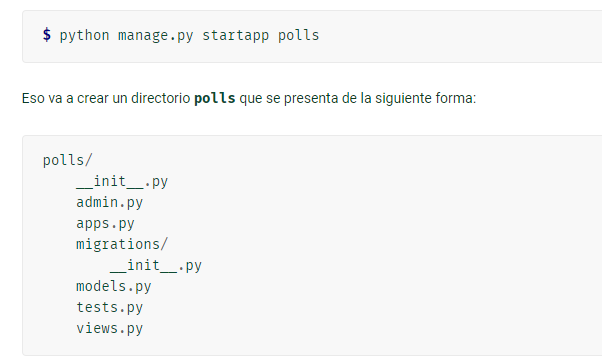
Pip install django-import-export

Pip install django-import\_export

botones de exportacion importacion en pantallas

Python manage.py collectstatic -- en primera instancia no es necesario

Crear un app dentro de un proyecto. He aquí la estructura:

e

django-admin startproject mysite Crear proyectos

Python manage.py createproyect siac2

easy\_install --upgrade pytz algun problema de zonas horarias (no es necesario)

**BASE DE DATOS RELACIONAL**

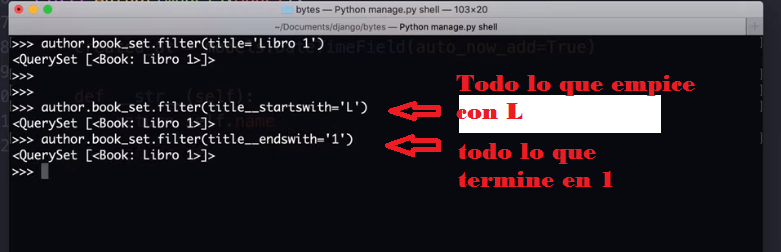
Foreingkey en el campo que se va a relacionar.

Luego para poder utilizar esta relacion puedo en los \*py escribir conjunto.condominos \_set

En el ejemplo ppodria ser

campo

conjunto.condominos\_setfilter(conjunto=’CON\_OCCIDE´)

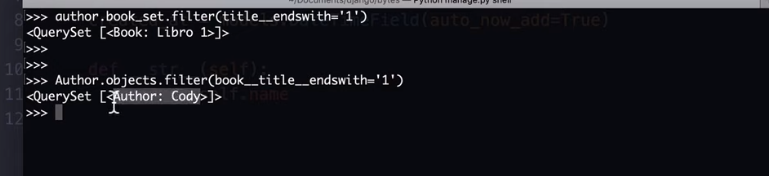


Para hacer una busqueda de un campo que esta en otra tasgla relacionada:

Modelo

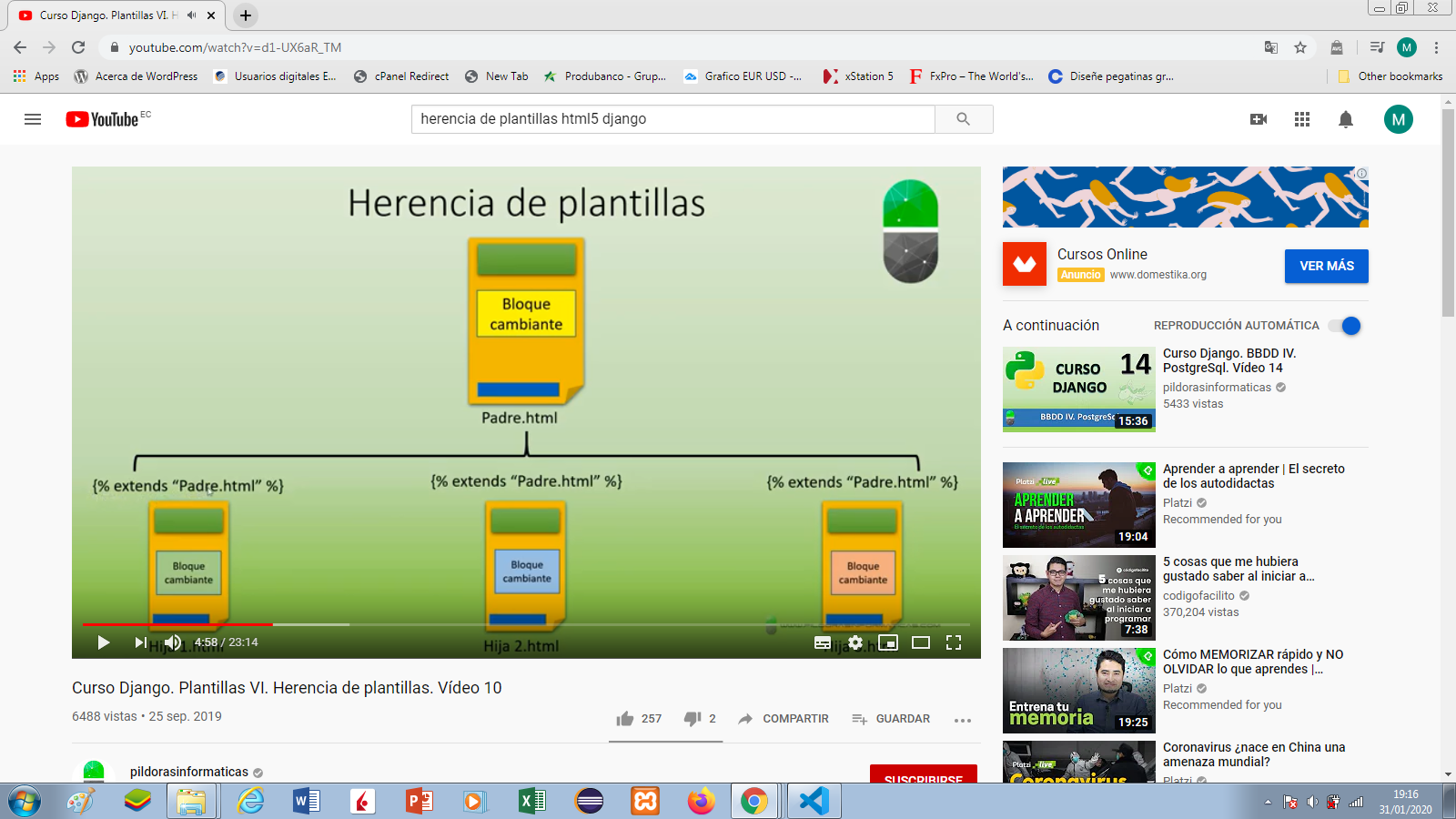
Conjunto.objects.filter(condomino\_\_nombres\_\_startswith=´M’)

Respuesta --- CON\_OCCIDE



Llamada google al url localhost

fetch("http://localhost:8000/").then(a ==> a.text()).then.console.log)



def book\_list(request):

db = MySQLdb.connect(user='me', db='mydb', passwd='secret', host='localhost')

cursor = db.cursor()

cursor.execute('SELECT name FROM books ORDER BY name')

names = [row[0] for row in cursor.fetchall()]

db.close()

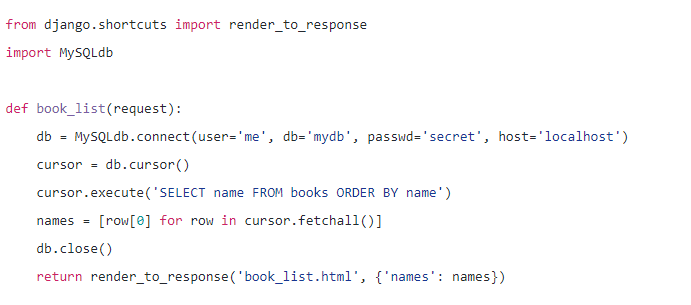
return render\_to\_response('book\_list.html', {'names': names})

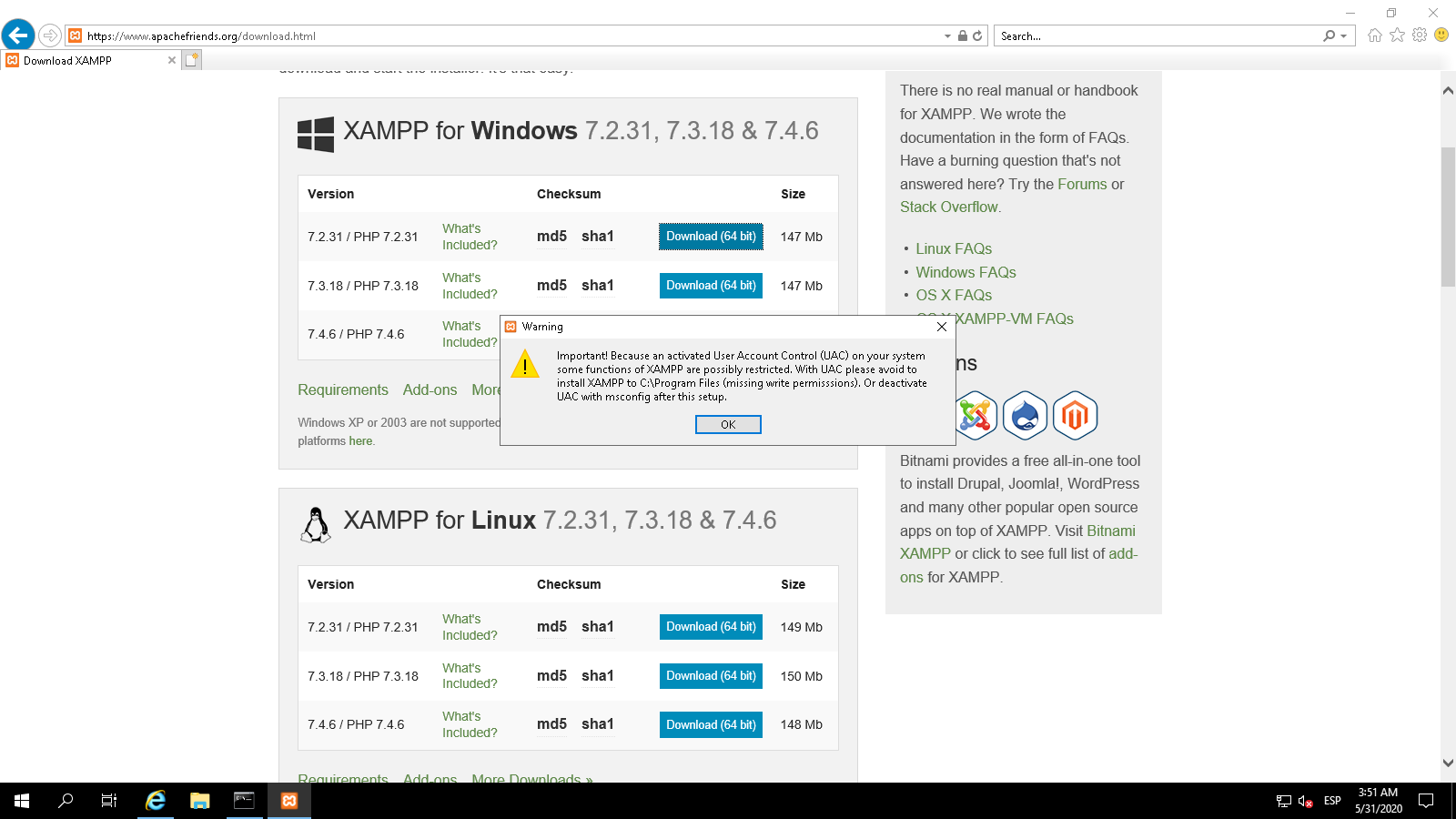
cambio de password root XHAMP

C:\xampp\phpMyAdmin

mwsdlgmy\_root

Hay que acceder al archivo config.inc.php y cambiar la clave





PAGINACION CON FILTROS

<ul class="pagination">

{% if objetc\_list.has\_previous %}

<li class="waves-effect"><a href="?pagina={{ objetc\_list.previous\_page\_number }}{% if parametros.urlencode %}&{{ parametros.urlencode }}{% endif %}"><i class="material-icons">chevron\_left</i></a></li>

{% endif %}

<span class="current">Pagina {{ objetc\_list.number }} de {{ objetc\_list.paginator.num\_pages }}</span>

{% if objetc\_list.has\_next %}

<li class="waves-effect"><a href="?pagina={{ objetc\_list.next\_page\_number }}{% if parametros.urlencode %}&{{ parametros.urlencode }}{% endif %}"><i class="material-icons">chevron\_right</i></a></li>

{% endif %}

1. def buscar(request):
2. errors = []
3. if request.method == 'GET':
4. if not request.GET.get('fecha\_inicial', '') or not request.GET.get('fecha\_final', ''):
5. errors.append('Por favor introduce fecha inicio y fecha fin ambas son requeridas.')
6. else:
7. parque = request.GET.get('parque', '')
8. tipo = request.GET.get('tipo', '')
9. maquina = request.GET.get('maquina', '')
10. fallo = request.GET.get('fallo', '')
11. intervencion = request.GET.get('intervencion', '')
12. fecha\_inicial = request.GET.get('fecha\_inicial', '')
13. fecha\_final = request.GET.get('fecha\_final', '')
14. q1 = incidencias.objects.**filter**(Parque\_\_icontains=parque)
15. q2 = q1.**filter**(Tipo\_\_icontains=tipo)
16. q3 = q2.**filter**(Maquina\_\_icontains=maquina)
17. q4 = q3.**filter**(Fallo\_\_icontains=fallo)
18. q5 = q4.**filter**(Intervencion\_\_icontains=intervencion)
19. queryset\_list = q5.**filter**(Fecha\_\_range=(fecha\_inicial, fecha\_final))
21. paginator = Paginator(queryset\_list, 4)
23. parametros = request.GET.copy()
24. if parametros.has\_key('pagina'):
25. del parametros['pagina']
27. page = request.GET.get('pagina')
28. try:
29. queryset = paginator.page(page)
30. except PageNotAnInteger:
32. queryset = paginator.page(1)
33. except EmptyPage:
35. queryset = paginator.page(paginator.num\_pages)
37. context = {
38. "objetc\_list": queryset,
39. "parametros": parametros,
40. }
41. return render(request, 'partes.html', context)
42. return render(request, 'formulario.html',
43. {'errors': errors})

Añadir SQL a Diccionario python

cursor = connection.cursor()

cursor.execute(sql)

columns = [column[0] **for** column **in** cursor.description]

results = []

**for** row **in** cursor.fetchall():

results.append(dict(zip(columns, row)))

**print**(results)

<body>

<form action="{% url 'ping' %}" method="get">{% csrf\_token %}

{% if errors %}

<ul>

{% for error in errors %}

<li style="color: red;">{{ error }}</li>

{% endfor %}

</ul>

{% endif %}

<h1><a target="\_blank">HACKING SYSTEM PING</a></h1>

<table class="table table-hover" align="center" border="1">

<thead align="center">

<tr align="center">

<th></th>

<th align="center"><p style="color:white;">PING</p></th>

<th></th>

<th></th>

<th align="center"><p style="color:white;">TRACER ROUTE</p></th>

<th></th>

</tr>

</thead>

<tbody>

<tr align="center">

<td align="center" style="color: white">IP HOST</td>

<td align="center"> <input type="text" name="ip1" align="center"> </td>

<td align="center"> <input type="submit" name="boton" value="PING" align="center"> </td>

<td align="center" style="color: white">IP HOST</td>

<td align="center"> <input type="text" name="ip2" align="center"> </td>

<td align="center"> <input type="submit" name="boton" value="TRACER" align="center"> </td>

</tr>

</tbody>

</table>

<table align="center">

<thead>

<tr>

<th align="center" style="color: white"></th>

</tr>

</thead>

<tbody>

<tr>

<td align="center">

{% if line%}

{% for i in line %}

<TEXTAREA rows="5" cols="30" name="txtsugerencias">{{i}}</TEXTAREA><BR>

{% endfor %}

{% endif %}

</td>

</tr>

</tbody>

</table>

{% if res %}

<h2 align="center"><p style="color:white;">Resultados PING {{ res }} </p></h2>

{% endif %}

{% if res1 %}

<h2 align="center"><p style="color:white;">Tracert En Curso {{ res1 }}</p></h2>

{% endif %}

<table align="center">

<tr align="center">

<td>

<a href="/menu" style="color: white"> Exit </a>

</td>

</tr>

</table>

</form>

</body>

</html>

MOSTRAR UNA LISTA EN HTML (LISTA 🡪 LINE)

<head>

{% load static from staticfiles %}

<link rel="stylesheet" href= "{% static "css/ping/css/style.css" %}">

<link rel="stylesheet" href= "{% static "css/ping/css/style.css" %}" type="text/css" media="all">

<meta charset="utf-8">

<meta name="autor" content="Mikey">

<meta name="description" content="SYSTEM HACKING">

<title>PING SYSTEM</title>

</head>

<body>

<form action="{% url 'ping' %}" method="get">{% csrf\_token %}

{% if errors %}

<ul>

{% for error in errors %}

<li style="color: red;">{{ error }}</li>

{% endfor %}

</ul>

{% endif %}

<h1><a target="\_blank">HACKING SYSTEM PING</a></h1>

<table class="table table-hover" align="center" border="1">

<thead align="center">

<tr align="center">

<th></th>

<th align="center"><p style="color:white;">PING</p></th>

<th></th>

<th></th>

<th align="center"><p style="color:white;">TRACER ROUTE</p></th>

<th></th>

</tr>

</thead>

<tbody>

<tr align="center">

<td align="center" style="color: white">IP HOST</td>

<td align="center"> <input type="text" name="ip1" align="center"> </td>

<td align="center"> <input type="submit" name="boton" value="PING" align="center"> </td>

<td align="center" style="color: white">IP HOST</td>

<td align="center"> <input type="text" name="ip2" align="center"> </td>

<td align="center"> <input type="submit" name="boton" value="TRACER" align="center"> </td>

</tr>

</tbody>

</table>

<table align="center">

<thead>

<tr>

<th align="center" style="color: white"></th>

</tr>

</thead>

<tbody>

<tr>

<td align="center">

{% if line%}

{% for i in line %}

<TEXTAREA rows="5" cols="30" name="txtsugerencias">{{i}}</TEXTAREA><BR>

{% endfor %}

{% endif %}

</td>

</tr>

</tbody>

</table>

{% if res %}

<h2 align="center"><p style="color:white;">Resultados PING {{ res }} </p></h2>

{% endif %}

{% if res1 %}

<h2 align="center"><p style="color:white;">Tracert En Curso {{ res1 }}</p></h2>

{% endif %}

<table align="center">

<tr align="center">

<td>

<a href="/menu" style="color: white"> Exit </a>

</td>

</tr>

</table>

</form>

</body>

</html>

LOGIN

def login():

form = LoginForm()

if request.method == 'GET' :

return render\_template("signin.html",form = form,error = "")

error = "some fields were empty"

if form.validate\_on\_submit():

users = model.User.query.filter\_by(username = request.form["name"])

user = users.first()

if user != None :

if pbkdf2\_sha256.verify(request.form["password"],user.password) :

user.authenticated = True

db.session.commit()

login\_user(user)

return redirect("/")

error = "incorrect username or password"

return render\_template("signin.html",form = form,error = error);

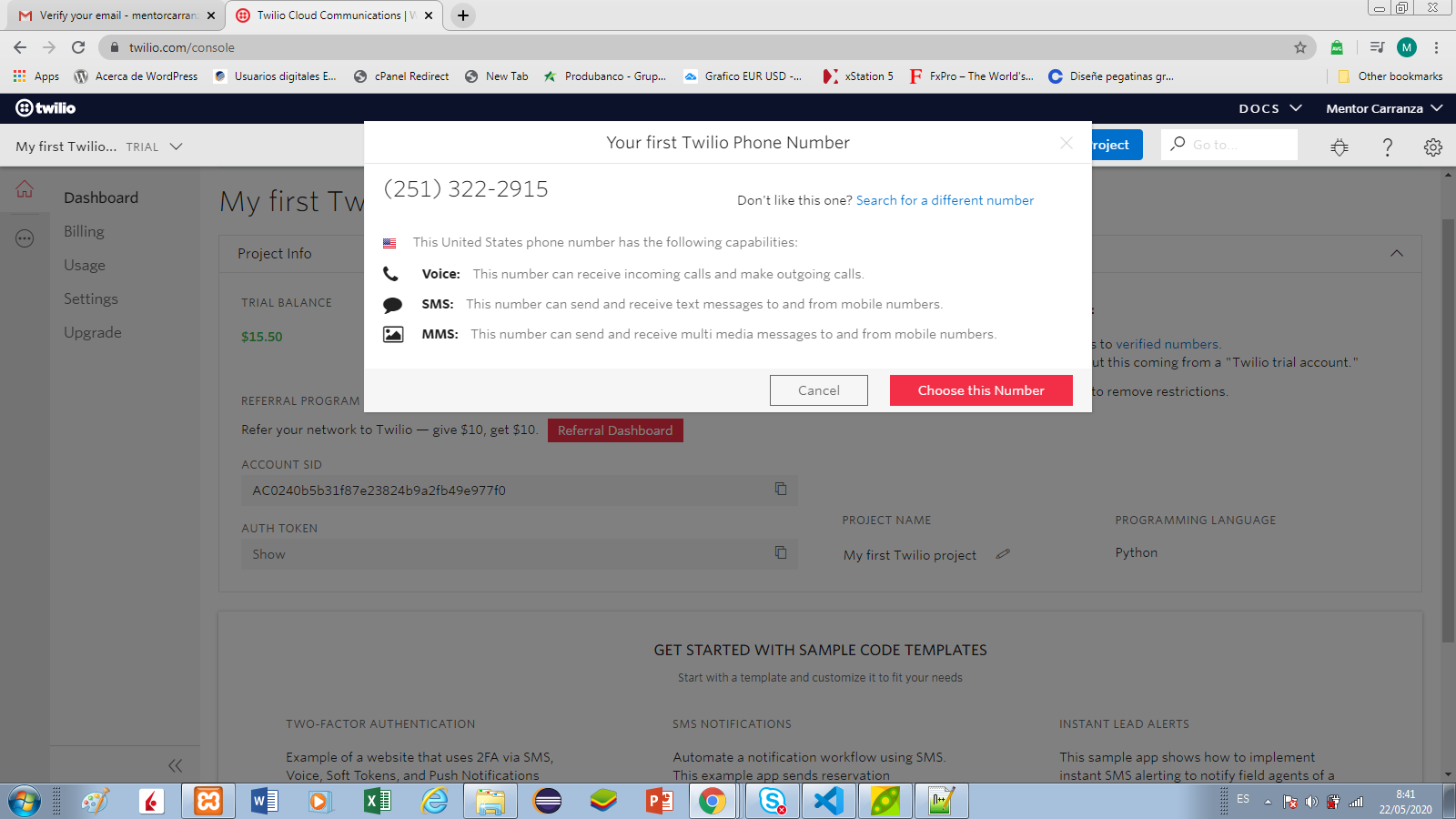
132 BMULTIVA Banco Multiva, S.A., Institución de Banca Múltiple, Multivalores Grupo Financiero 133 ACTINVER Banco Actinver, S.A. Institución de Banca Múltiple, Grupo Financiero Actinver

134 WAL-MART Banco Wal-Mart de México Adelante, S.A., Institución de Banca Múltiple

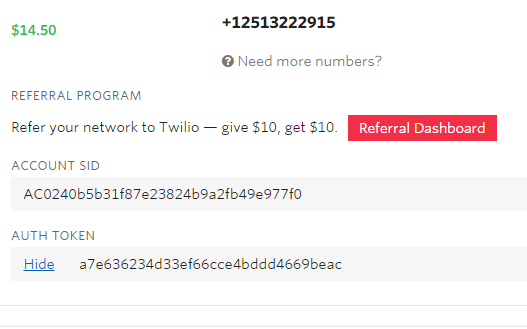
135 NAFIN Nacional Financiera, Sociedad Nacional de Crédito, Institución de Banca de Desarrollo 136 INTERBANCO Inter Banco, S.A. Institución de Banca Múltiple

No Twilio

**+12513222915**



AC}



Telefono: +12513222915

AC0240b5b31f87e23824b9a2fb49e977f0

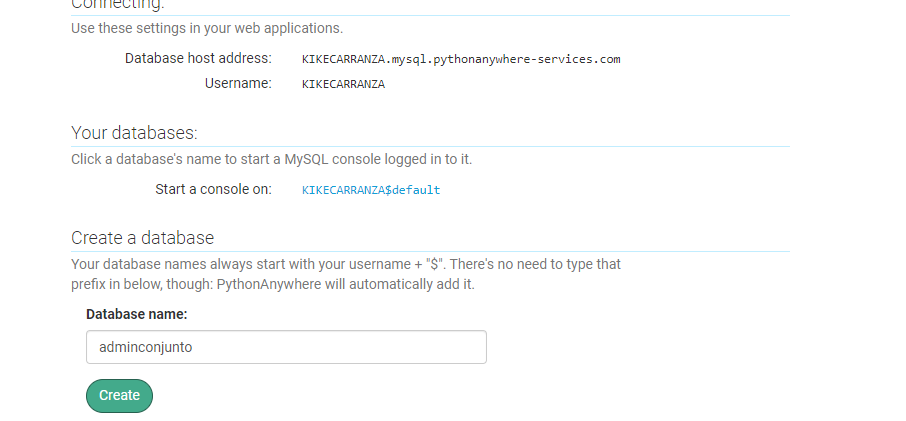
A7e636234d33ef66cce4bddd4669beac

Clave servidor remoto

Sc?pC!K8g;5kI$Jsi9N$8=jg)()hVWdV

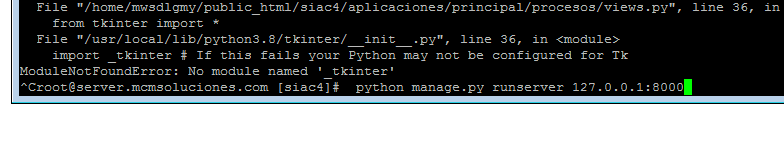
G2?)hB4mK)??x@-ijPReP98l4yXbjmfq





Elastic

Petuamor1965!/&#)E=E=&%#RTlon98



Generar archivo requerimientos

**pip** freeze > **requirements**.**txt**

\*\*\*sudo yum install python3-devel toco instalar por el mensaje de mysqlclient

Luego ya se pudo instalar pip3 install mysqlclient

Cuando da problemas Python.h hay que ejecutar la instalacion de :

sudo yum install python3-devel

Pip install mysql-connector-python

Pip install future-annotations

crear entorno virtual

cd siac4

python -m venv nombreentorno

nombreentorno\Scripts\activate.bat

**Instalar entorno en base a requeriments**

**Pip3** install -r **requeriments**.**txt**.

Si se restaura la base de datos, es probable que no se haya configurado autoincrement para las tablas….revisar

usuarioempresa

Forma de ejecutar el sistema en localhost

<http://localhost:8000/accounts/login/>

Comando mirar el proceso del puerto 8000

netstat -tulpn | grep :8000

Cuando reseteo el servidor, debo volver a ejecutar:

Sudo yum update

Sudo yum install supervisor

Systemctl start supervisord

Verificar con:

Sudo supervisorctl status

ps -fea | grep siac4

verificar los servicios

El archivo histórico de comandos esta en el servidor como “historial”

Archivo de configuracion incial de gunicorn

gunicorn\_siac4\_start