

Curso de profundización de software

► VerifyDocs

MANUAL DE INSTALACIÓN

VerifyDocs

Link del repositorio en donde puede encontrar los comandos completos de instalación

[GitHub VerifyDocs](#)

Tabla de contenido

1.	<i>Instalar PostgreSQL.....</i>	4
2.	<i>Instalación supervisor</i>	4
3.	<i>Instalar node y bower</i>	4
4.	<i>Instalar gunicorn</i>	4
5.	<i>Instalar Nginx.....</i>	4
6.	<i>Creación del entorno virtual</i>	4
6.1	Instalación de virtualenvwrapper unado python 3.6 o superior	4
6.2	Crear entorno virtual.....	4
7.	<i>Clonar el repositorio privado desde GitHub.....</i>	4
7.1	Instalar las librerías necesarios para la instalación	5
8.	<i>Ajustar la variable de entorno necesaria</i>	5
8.1	Registrar la siguiente línea al final del archivo	5
9.	<i>Copiar y configurar el archivo de parámetros requeridos para la ejecución.</i>	5
9.1	Se presentan las siguientes variables para ajustarlas en el sistema.	5
10.	<i>Configuración de acceso a la base de datos</i>	6
10.1	PG_ENGINE: Esta variable representa el motor de base de datos que se pretende usar para el mantenimiento de los datos.	6
10.2	PG_DBHOST: Representa la dirección IP, o HOST en donde se encuentra ubicada la base de datos.....	7
10.3	PG_DBPORT: Representa el puerto por el cual se accede al motor de la base de datos. 7	
11.	<i>Configuración personal de la llave única de ejecución</i>	7
12.	<i>Creación y ejecución de las migraciones a la base de datos.</i>	7
13.	<i>Ejecutar bower</i>	7
14.	<i>Crear archivo en supervisor.....</i>	8
14.1	Crear carpeta de logs y archivos de ejecución.....	8
14.2	Copiar el texto en el archivo de configuración.....	8
14.3	Activar la configuración de supervisorctl y revisar el estado de ejecución de la tarea 8	

15.	<i>Configuración de Nginx</i>	9
15.1	Crear archivo de configuración del virtualhost en Nginx.....	9
15.2	Editar el archivo.....	9
15.3	Crear enlace en la carpeta <i>sites-enable</i>	10
15.4	Ejecutar pruebas de configuración de Nginx.....	10
16.	<i>Crear usuario superadministrador</i>	11
17.	<i>Configurar servicios de Google API</i>	11
17.1	Oauth 2.0	11
17.2	Registrar los datos de acceso en el verifyDocs.....	12
18.	<i>Servicio de aplicación para el almacenamiento de documentos</i>	13
19.	<i>Crear el usuario administrador de VerifyDocs</i>	13

Manual de instalación.

1. Instalar PostgreSQL

Guía de instalación y configuración de [postgresql](#), preferiblemente la versión mas reciente.

2. Instalación supervisor

Guía de instalación de [supervisor](#)

3. Instalar node y bower

Guía de instalación de [node](#) y [bower](#)

4. Instalar gunicorn

Guía de instalación de [gunicorn](#) en Ubuntu

5. Instalar Nginx

Guía de instalación de [nginx](#) en ubuntu

6. Creación del entorno virtual

6.1 Instalación de [virtualenvwrapper](#) usando python 3.6 o superior

6.2 Crear entorno virtual

```
$ mkvirtualenv verifydocs -p /usr/bin/python3.x  
$ workon verifydocs
```

7. Clonar el repositorio privado desde GitHub

```
$ git clone https://github.com/edwinVelasco/verifydocs.git verifydocs  
$ cd verifydocs
```

7.1 Instalar las librerías necesarios para la instalación

```
$ pip install -Ur requirements.txt
```

8. Ajustar la variable de entorno necesaria

Abrir el archivo que se ejecuta cuando se activa el entorno virtual.

```
sudo nano /home/<usuario>/virtualenvs/verifydocs/bin/postactivate
```

8.1 Registrar la siguiente línea al final del archivo

Editar al final del archivo la siguiente variable de entorno.

```
export DJANGO_SETTINGS_MODULE=verifydocs.settings
```

9. Copiar y configurar el archivo de parámetros requeridos para la ejecución.

Crear el archivo *parameters.py* de la instancia para ser configurado.

```
$ mv verifydocs/parameters-dist.py verifydocs/parameters.py  
$ nano verifydocs/parameters.py
```

9.1 Se presentan las siguientes variables para ajustarlas en el sistema.



```
# Sistema de base de datos
PG_ENGINE = ''

# Nombre de la base de datos
PG_DBNAME = ''

# Usuario con acceso a la base de datos
PG_DBUSER = ''

# Contraseña del usuario con acceso a la base de datos
PG_DBPASSWORD = ''

# host donde se ubica el servidor de base de datos, IP o dominio
PG_DBHOST = ''
# puerto donde se aloja la base de datos en caso de tenerlo
PG_DBPORT = ''

DJ_SECRET_KEY = ''

# True en caso de estar en ambiente de pruebas
DJ_DEBUG = False

# Dominio en donde se esta alojando el aplicativo ejemplo: ['verifydocs.ufps.edu.co'] o ['*'] para ejecuciones de prueba
DJ_ALLOWED_HOSTS = ['*']

# Lenguaje que se maneja en el aplicativo
DJ_LANGUAGE_CODE = 'es-co'

# Zona horaria
DJ_TIME_ZONE = 'America/Bogota'
DJ_USE_I18N = False
DJ_USE_L10N = True
DJ_USE_TZ = False

# Id del sitio configurado con Google
DJ_SITE_ID = 2

# Ruta absoluta de la raiz del proyecto ejemplo: '/home/<user>/verifydocs'
DJ_URL_PROJECT = ''

# URL del proyecto instalado, se usa para utilizarlo en la creación del texto que acompaña el código QR
WEB_CLIENT_URL = 'https://verifydocs.ufps.edu.co/'

EMAIL_HOST = 'smtp.gmail.com'
EMAIL_PORT = 587
# Correo electronico para el envío de notificación por smtp
EMAIL_HOST_USER = ''
# Contraseña de la cuenta de correo
EMAIL_HOST_PASSWORD = ''
EMAIL_USE_TLS = True
EMAIL_USE_SSL = False
EMAIL_BACKEND = 'django.core.mail.backends.smtp.EmailBackend'
```

10. Configuración de acceso a la base de datos

10.1 **PG_ENGINE:** Esta variable representa el motor de base de datos que se pretende usar para el mantenimiento de los datos.

```
PG_ENGINE: <django.db.backends.postgresql_psycopg2> Postgresql o <django.db.backends.mysql> Mysql
```

10.2 **PG_DBHOST:** Representa la dirección IP, o HOST en donde se encuentra ubicada la base de datos.

```
PG_DBHOST: 127.0.0.1, localhost, 192.689.3.119, https://dbpostgresq.edu.co
```

10.3 **PG_DBPORT:** Representa el puerto por el cual se accede al motor de la base de datos.

```
PG_DBPORT: El puerto por defecto para PostgreSQL es 5432 y para MySQL es 6379
```

11. Configuración personal de la llave única de ejecución

DJ_SECRET_KEY: Llave única por instancia instalada esta se debe generar en cada instancia instalada.

Ejecutar el siguiente comando para crear la llave única, retorna una cadena.

```
$ openssl rand -base64 32  
HuUk7o0LLxgdA0rw2J5qF+Et9kYW1+MwBTtMeFUHN8M=  
$ # copiar el texto generado a la variable DJ_SECRET_KEY
```

12. Creación y ejecución de las migraciones a la base de datos.

Se usa para crear el archivo de migraciones para posteriormente ser usado en la creación de tablas en la base de datos.

```
$ python manage.py makemigrations  
$ python manage.py migrate  
$ mkdir static & mkdir temp & mkdir media & mkdir media/tmp/  
$ python manage.py collectstatic
```

13. Ejecutar bower

Bower es utilizado para descargar las librerías de javascript necesarias

```
$ cd static/app/  
$ bower install --save blockchain/bc-qr-reader  
$ cd ../../
```

14. Crear archivo en supervisor

Este archivo se usa para permitir la ejecución del aplicativo con gunicorn.

14.1 Crear carpeta de logs y archivos de ejecución

```
$ sudo mkdir /var/log/verifydocs  
$ sudo nano /etc/supervisor/conf.d/verifydocs.conf
```

14.2 Copiar el texto en el archivo de configuración

```
[program:verifydocs_gunicorn]  
directory=/home/<usuario>/verifydocs  
command=/home/<usuario>/virtualenvs/verifydocs/bin/gunicorn --workers 2 --  
autostart=true  
autorestart=true  
stderr_logfile=/var/log/verifydocs/gunicorn.out.log  
stdout_logfile=/var/log/verifydocs/gunicorn.err.log  
user=<usuario>  
group=www-data  
environment=LANG=en_US.UTF-8,LC_ALL=en_US.UTF-8  
  
[group:gunicorn]  
programs:verifydocs_gunicorn
```

La configuración completa la puede encontrar en el repositorio.

14.3 Activar la configuración de supervisorctl y revisar el estado de ejecución de la tarea


```
$ sudo supervisorctl reread
$ sudo supervisorctl update
$ sudo supervisorctl status
```

15. Configuración de Nginx

Se debe tener en la carpeta `/home/<usuario>/ssl/` los archivos del certificado de seguridad SSL

15.1 Crear archivo de configuración del virtualhost en Nginx

```
$ sudo nano /etc/nginx/sites-available/verifydocs
```

15.2 Editar el archivo

► VerifyDocs



```
log_format main '$remote_addr - $remote_user [$time_local] "$request" '
                '$status $body_bytes_sent "$http_referer" '
                '"$http_user_agent" "$http_x_forwarded_for"';
access_log /var/log/nginx/access.log main;

server {
    listen 443 ssl;
    listen [::]:443 default_server ipv6only=on;

    server_name verifydocs.ufps.edu.co;
    include /etc/nginx/default.d/*.conf;

    ssl on;
    ssl_protocols TLSv1 TLSv1.1 TLSv1.2;
    ssl_certificate /home/<usuario>/ssl/ufps.edu.co.crt;
    ssl_certificate_key /home/<usuario>/ssl/ufps.edu.co.key;
    ssl_session_cache shared:SSL:10m;
    ssl_session_timeout 10m;

    location / {
        include proxy_params;
        proxy_pass https://unix:/home/<usuario>/verifydocs/verifydocs.sock;
    }

    location /static/ {
        root /home/<usuario>/verifydocs;
    }
    location /media/ {
        root /home/<usuario>/verifydocs;
    }

    error_page 404 /404.html;
        location = /40x.html {
    }

    error_page 500 502 503 504 /50x.html;
        location = /50x.html {
    }
}
```

15.3 Crear enlace en la carpeta *sites-enable*

```
$ sudo ln -s /etc/nginx/sites-available/verifydocs /etc/nginx/sites-enabled
```

15.4 Ejecutar pruebas de configuración de Nginx

```
$ nginx -t
```

Si todo parece estar correcto reiniciar nginx

```
$ sudo systemctl restart nginx
```

Crear usuario superadministrador

Ingresando a la ruta raíz del proyecto y ejecutar el siguiente comando, con el entorno virtual activo.

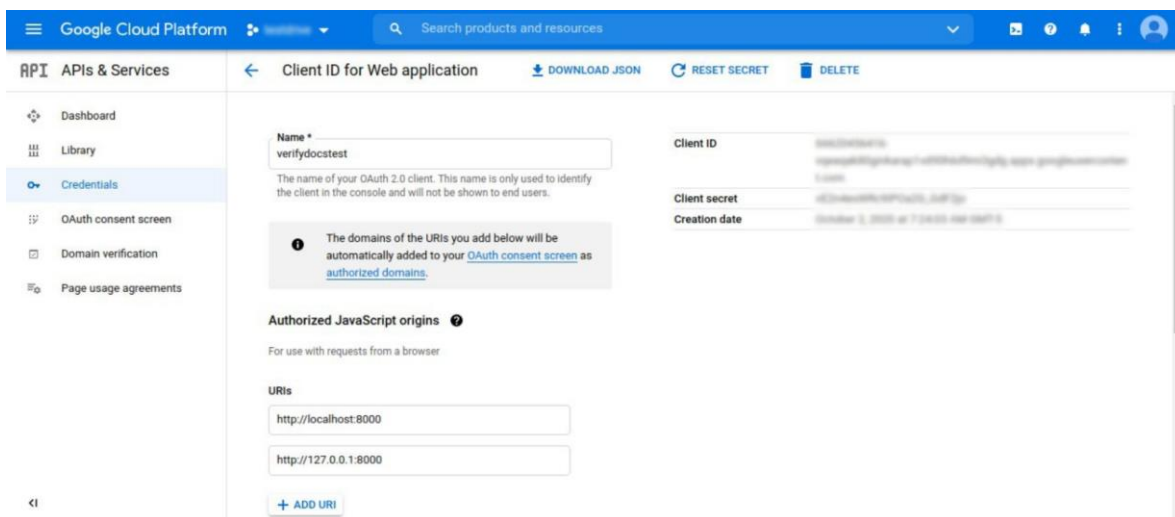
Registrar usuario, contraseña y correo electrónico de superadministrador.

```
$ python manage.py createsuperuser
```

16. Configurar servicios de Google API

16.1 Oauth 2.0

Para obtener credenciales de Google API Console use la siguiente [guía](#). tendrá un resultado como se muestra en la siguiente imagen:



16.2 Registrar los datos de acceso en el verifyDocs.

Ingresa a través del explorador web al proyecto, con la ruta admin e iniciar sesión.

Ejemplo: <https://verifydocs.ufps.edu.co/admin/>

Debe agregar un nuevo *site*, para esto presione el botón *add* que se encuentra delante del modelo *sites*.

SITES		
Sites	+ Add	Change
SOCIAL ACCOUNTS		
Social accounts	+ Add	Change
Social application tokens	+ Add	Change
Social applications	+ Add	Change

En los campos de *Domain name* y *Display name* el valor de <https://verifydocs.ufps.edu.co> y presionar el botón guardar.

Regresar a la lista de modelos y presionar el botón *add* del modelo *Social application* e ingresar los siguientes valores:

- Provider: Google
- Name: Google API
- Client id: *Client ID* de las credenciales proporcionadas por google.
- Secret key: *Client secret* de las credenciales proporcionadas por google.
- Sites: Seleccione el *site* creado en el paso anterior.

Add social application

Provider:	Google ▼
Name:	Google API
Client id:	xxxxxxxxxxxxxx <small>App ID, or consumer key</small>
Secret key:	xxxxxxxxxxxxxx <small>API secret, client secret, or consumer secret</small>
Key:	 <small>Key</small>
Sites:	<div>sites Disponibles ⓘ Filtro</div> <div>sites elegidos ⓘ http://127.0.0.1:8000/ +</div>

17. Servicio de aplicación para el almacenamiento de documentos

Para obtener el servicio de aplicación use la siguiente [guía](#) y descargue el archivo *json* del servicio.

Cargue el archivo en la siguiente ruta del servidor en donde se encuentra alojado el aplicativo y con el mismo nombre

```
$ verifydocs/tools/storage_key_file.json
```

18. Crear el usuario administrador de VerifyDocs

Regresar a la lista de modelos y presionar el botón *add* del modelo Correos permitidos e ingresar los siguientes valores:

Curso de profundización de software

► VerifyDocs

Django administration

Site administration

ACCOUNTS	
Email addresses	+ Add Change
APP	
Correos permitidos	+ Add Change
Dependencias	+ Add Change
AUTH TOKEN	
Tokens	+ Add Change

Registrar el correo del administrador.

Add Correo permitido

Correo:	<input type="text" value="correoadmin@ufps.edu.co"/>	El correo electrónico debe ser un correo institucional
Rol:	<input type="text" value="Administrador"/>	Rol administrador
<input checked="" type="checkbox"/> Activo		
Última modificación:	-	
Creado:	-	
Contraseña:	No registrar contraseña	
<div>Save and add another Save and continue editing SAVE</div>		

Guardar

Ingresar a VerifyDocs para iniciar sesión por primera vez con las credenciales de la cuenta institucional.