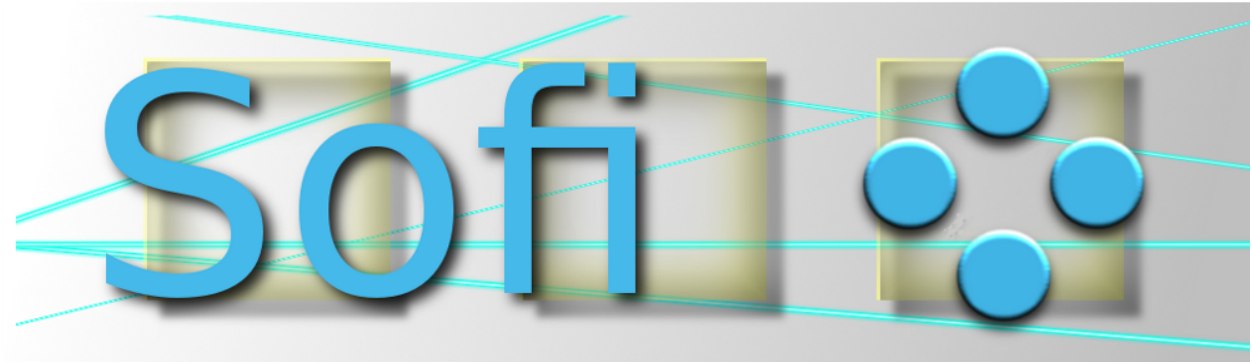

Sofi

Publicación 2.0.0

Leonardo J. Caballero G.

30 de July de 2014

1. ¿Qué es Sofi?	3
1.1. Funcionalidades	3
1.2. Referencias	4
2. Instalar Sofi 2.0	5
2.1. Dependencias previas	5
2.2. Descargar Sofi 2.0	5
3. Sofi con SQLite	7
3.1. Referencias	7
4. Apache2 con Sofi	9
4.1. Dependencias previas	9
4.2. VirtualHost y mod_python	9
4.3. Referencias	11
5. Sofi con PostgreSQL	13
5.1. Referencias	13
6. Sofi con MySQL	15
6.1. Referencias	15
7. Índices y tablas	17



Contenidos:

¿Qué es Sofi?

Sofi es una aplicación web para la gestión y organización de eventos, posee interfaz de administración para gestionar la información de (eventos, presentaciones, ponentes, suscripciones...) y es ideal como portal web de publicación de eventos tipo blog.

Sofi fue desarrollada con el lenguaje de programación Python utilizando Django como framework y liberada bajo la licencia GPLv2, por lo tanto usted tendrá derecho de utilizar, estudiar, modificar y distribuir libremente según los términos de la licencia.

1.1 Funcionalidades

Sofi ofrece diversas funcionalidades las cuales se describen a continuación:

1.1.1 Usuarios Administrativos

En la interfaz administrativa puede crear, modificar y eliminar la siguiente información:

- Usuarios
- Eventos
- Presentaciones
- Ponentes
- Suscripciones
- Certificados de asistencia
- Patrocinadores
- Organizadores
- Comentarios
- Notificaciones
- Configuración del Sitio

1.1.2 Usuarios no Administrativos

- Creación de usuarios y perfiles.

- Consultar información de eventos, presentaciones, ponentes, suscripciones, reporte de suscriptores, reporte de encuestas, dejar comentarios, obtener certificado de participación en evento, recibir información sobre eventos, sindicación de contenido rss y atom, entre otros.

1.2 Referencias

- [Wiki de Sofi](#).

Instalar Sofi 2.0

Autor(es) Leonardo J. Caballero G.

Correo(s) leonardocaballero@gmail.com

Lanzamiento 2.0.0

Actualizado el 30 de July de 2014

Para instalar Sofi requiere realizar los siguientes pasos:

2.1 Dependencias previas

Mínimamente requiere instalado las dependencias básicas para instalar aplicaciones Python, para hacer esto posible ejecute el siguiente comando:

```
# aptitude install build-essential python-dev python-pip python-setuptools libfreetype6-dev libpng-dev  
# pip install virtualenv  
# exit
```

2.2 Descargar Sofi 2.0

Para descargar Sofi 2.0 ejecute los siguientes comandos:

```
$ hg clone https://macagua@bitbucket.org/conectivo/sofi sofi-2.0.0  
$ cd sofi-2.0.0/  
$ hg checkout default
```

Para instalar Sofi requiere crear un entorno **virtual Python** dentro del cual realizara la instalación de dependencias Python con los siguientes comandos:

```
$ virtualenv --distribute --python=/usr/bin/python2.7 .  
$ source ./bin/activate
```

Primero, instale el resto de las dependencias usando la herramienta **pip** con los siguientes comandos:

```
$ cd sofi/  
$ pip install -r ./requirements/compilado.txt  
$ pip install -r ./requirements/sofi.txt
```

Segundo, crea la base de datos de Sofi 2.0 con el siguiente comando:

```
$ python manage.py syncdb
```

Nota: En un momento de la creación de la base de datos Django le realizará la siguiente pregunta:

You just installed Django's auth system, which means you don't have any superusers defined.

Would you like to create one now? (yes/no):

Usted le responde **yes**, seguidamente debe responder adecuadamente a las siguientes preguntas que Django le realizará.

Tercero, inicie el servidor de ejecución Django con el siguiente comando:

```
$ python manage.py runserver
```

Nota: Luego accedes a Sofi 2.0 en la siguiente dirección <http://localhost:8000/>

Opcionalmente puede acceder a la interfaz administrativa en la siguiente dirección <http://localhost:8000/admin/>

Cuarto, Si no va a instalar **paquetes Egg Python** adicionales a su aplicación Sofi, desactive el **entorno virtual Python** creado con el siguiente comando:

```
$ deactivate
```

Sofi con SQLite

Para entornos de pruebas y desarrollos se recomienda usar la base de datos [SQLite](#), para hacer esto posible ejecute el siguiente comando:

```
# aptitude install sqlite3 libsqlite3-dev
```

3.1 Referencias

- [Creación de entornos virtuales Python](#)
- [Instalación de paquetes Python con Distribute y pip](#)

Apache2 con Sofi

Autor(es) Leonardo J. Caballero G.

Correo(s) leonardocaballero@gmail.com

Lanzamiento 2.0.0

Actualizado el 30 de July de 2014

Para instalar Sofi requiere realizar los siguientes pasos:

4.1 Dependencias previas

Mínimamente requiere instalado las dependencias básicas para instalar aplicaciones Python, para hacer esto posible ejecute el siguiente comando:

```
# aptitude install apache2 libapache2-mod-python
# a2enmod python
# service apache2 reload
```

4.2 VirtualHost y mod_python

Para hacer disponible Sofi a través de un servidor Web, ejecute el siguiente comando:

```
# cp ./apache2_sofi.site /etc/apache2/sites-available/sofi.site
```

Entonces edite el archivo de configuración VirtualHost de Apache, con el siguiente comando:

```
# vim /etc/apache2/sites-available/sofi.site
```

Usted adaptar su archivo de configuración VirtualHost como se muestra a continuación:

```
<VirtualHost *:80>
```

```
    ServerAdmin TU-NOMBRE@CORREO
    ServerName TU-HOST-O-localhost
```

```
    <Location "/">
        SetHandler python-program
        PythonHandler myvirtualdjango
        SetEnv DJANGO_SETTINGS_MODULE sofi.settings
    </Location>
</VirtualHost>
```

```
SetEnv PYTHON_EGG_CACHE /var/tmp/egg
PythonDebug Off
PythonPath "['/<RUTA>/bin','/<RUTA>/lib/python2.7/site-packages','/<RUTA>/sofi','/<RUTA>/so
</Location>

Alias /admin_media/ /<RUTA>/lib/python2.7/site-packages/django/contrib/admin/media/
<Location /admin_media/>
    SetHandler none
    Options -Indexes
</Location>

Alias /site_media/ /<RUTA>/sofi/site_media/
<location /site_media/>
    SetHandler none
    Options -Indexes
</Location>

ErrorLog ${APACHE_LOG_DIR}/sofi.error.log

# Possible values include: debug, info, notice, warn, error, crit,
# alert, emerg.
LogLevel warn

CustomLog ${APACHE_LOG_DIR}/sofi.access.log combined

</VirtualHost>
```

Truco: Debe cambiar **<RUTA>** por la ruta adecuada en el contexto de la explicación.

Después debe habilitar la configuración VirtualHost creada como un sitio disponible para Apache:

```
# ln -s /etc/apache2/sites-available/sofi.site /etc/apache2/sites-enabled/sofi.site
# a2ensite sofi.site
```

Seguidamente agregue el archivo myvirtualdjango.py dentro del directorio bin de su virtualenv python (adecue la ruta a la ruta real de su instalación):

```
#myvirtualdjango.py

activate_this = '/<RUTA>/bin/activate_this.py'
execfile(activate_this, dict(__file__=activate_this))

from django.core.handlers.modpython import handler
```

Truco: Debe cambiar **<RUTA>** por la ruta adecuada en el contexto de la explicación.

En el archivo settings.py de tu aplicación debes cambiar la variable DEBUG de **True** a **False**

Otorgas permisos al servicio de Apache para que acceda al directorio de instalación de Sofi, con el siguiente comando:

```
# chown -R :www-data /<RUTA>/sofi
# chmod g+rw /<RUTA>/sofi/sofi.db
```

Truco: Debe cambiar **<RUTA>** por la ruta adecuada en el contexto de la explicación.

Advertencia: Si usted esta solo usando el servidor Web para ejecutar Sofi le recomiendo que deshabilite el sitio por defecto de Apache ya que esta configuración VirtualHost es para un solo sitio Web dentro del servidor Web Apache.

Para esto ejecute el siguiente comando: `a2dissite default`.

De no ser así necesita adaptar su configuración VirtualHost de Sofi para que conviva con los demás sitios Web.

Reinicie y recargue el servicio de Apache, con los siguientes comandos:

```
# service apache2 restart
# service apache2 reload
```

Nota: Luego accedes a Sofi en la siguiente dirección <http://TU-SITIO-WEB/>

Opcionalmente puede acceder a la interfaz administrativa en la siguiente dirección <http://TU-SITIO-WEB/admin/>

Truco: Cada vez que cambies algo en tu aplicación Sofi debe recargar las configuraciones del servidor Apache, con el comando: `service apache2 reload`

4.3 Referencias

- [Django, mod_python and virtualenv.](#)

Sofi con PostgreSQL

Autor(es) Leonardo J. Caballero G.

Correo(s) leonardocaballero@gmail.com

Lanzamiento 2.0.0

Actualizado el 30 de July de 2014

Para entornos de pruebas y producción es posible usar el servidor de base de datos PostgreSQL, para hacer esto posible ejecute el siguiente comando:

```
# aptitude install postgresql
# exit
```

Luego active su entorno virtual Python si no lo tiene activado para instalar el adaptador de base de datos para PostgreSQL en Python `psycopg2`, para esto ejecute el siguiente comando:

```
$ pip install -r ./requeriments/sofi-postgresql.txt
```

Seguidamente edita su archivo `settings.py`, y ajusta las configuraciones de conexión a la base de datos como se describe a continuación:

```
DATABASES = {
    'default': {
        'ENGINE': 'django.db.backends.postgresql_psycopg2', # Add 'postgresql_psycopg2', 'postgresql'
        'NAME': 'sofi'),                                     # Or path to database file if using sqlite3.
        'USER': 'TU-USUARIO',                               # Not used with sqlite3.
        'PASSWORD': 'TU-CONTRASENA',                       # Not used with sqlite3.
        'HOST': 'localhost',                               # Set to empty string for localhost. Not used with
        'PORT': '5432',                                     # Set to empty string for default. Not used with
    }
}
```

Y crea la base de datos como se describe acá:

5.1 Referencias

- Instalación de PostgreSQL en Debian GNU/Linux Wheezy

Sofi con MySQL

Autor(es) Leonardo J. Caballero G.

Correo(s) leonardocaballero@gmail.com

Lanzamiento 2.0.0

Actualizado el 30 de July de 2014

Opcionalmente, para entornos de pruebas y producción es posible usar el servidor de base de datos MySQL, para hacer esto posible ejecute el siguiente comando:

```
# aptitude install mysql-server mysql-client libmysqlclient15-dev
# exit
```

Luego active su entorno virtual Python si no lo tiene activado para instalar la interfaz Python a MySQL `MySQL-python`, para esto ejecute el siguiente comando:

```
$ pip install -r ./requeriments/sofi-mysql.txt
```

Seguidamente edita su archivo `settings.py`, y ajusta las configuraciones de conexión a la base de datos como se describe a continuación:

```
DATABASES = {
    'default': {
        'ENGINE': 'django.db.backends.mysql', # Add 'postgresql_psycopg2', 'postgresql', 'mysql', 's
        'NAME': 'sofi'),                      # Or path to database file if using sqlite3.
        'USER': 'TU-USUARIO',                # Not used with sqlite3.
        'PASSWORD': 'TU-CONTRASENA',         # Not used with sqlite3.
        'HOST': 'localhost',                 # Set to empty string for localhost. Not used with
        'PORT': '3306',                      # Set to empty string for default. Not used with sqlite.
    }
}
```

6.1 Referencias

- Instalación de un servidor MySQL en Debian Lenny
- Instalación y configuración de Python, Django y MySQL (WIndows)

Índices y tablas

- *genindex*
- *modindex*
- *search*