D:\dev\apps-docs\gui\assets\submitconsulting - copia.png

**Guía del programador (versión 1.0-17)**

Backend Manager for Django versión 1.0-17

**26-09-2014**

**Autor: Angel Sullón Macalupú (@asullom)**

# Capítulo 1. Introducción

## Alcance del documento

En este documento se describen los componentes de Backengo, las convenciones, las prácticas, técnicas y estrategias con la finalidad de que pueda entenderlo y reutilizarlo con facilidad.

IMPORTANTE: Los logos, diagramas o pantallas que se muestran en este documento están integrados dentro del proyecto SHOMWARE aunque la carpeta raíz se llame backengo. Ya se dará cuenta que el nombre de esta carpeta no tiene impacto en la configuración del proyecto. Así, reemplace a SHOMWARE por el nombre de su proyecto.

## Descripción General del proyecto Backengo

El Backengo es el módulo base de administración y configuración de una aplicación web SaaS escritas en Django y con la elegancia de Bootstrap.

Por medio del Backengo podrás gestionar las diferentes partes del sistema: usuarios, perfiles, recursos, permisos, módulos, planes SaaS, menús, asociaciones, empresas, sedes, logs, seguridad, internacionalización y mucho más!.

La figura 1 muestra los módulos del proyecto Backengo.

Figura 1 – Módulos del proyecto Backengo 1.0

Los módulo que pertenecen al Backengo son los círculos de color verde y esto son los módulos Backend y Home. Los otros módulos quedan fuera del alcance del Backengo.

## Descripción General del proyecto SHOMWARE

La figura 2 muestra los módulos del proyecto SHOMWARE.

Figura 2 – Módulos del proyecto SHOMWARE 1.0

Los módulo que pertenecen a SHOMEWARE son los círculos de color verde y esto son los módulos Backend, Home y Ventas. Los otros módulos quedan fuera del alcance del SHOMEWARE.

# Capítulo 2. Instalación y Ejecución

## Instalación de Python

* Backengo requiere Python 2.7.x. Descargue de <https://www.python.org/downloads> la última versión de la 2.7.x para su plataforma favorita e instale

Para instalar en windows, ejecute python-2.7.x.msi y siguiente, siguiente hasta terminar, verifique la variable del sistema Path que contenga c:\Python27;

* Instalación de las herramientas de descargas de paquetes easy\_install y pip
  + Instalación de easy\_install

Ir a <https://pypi.python.org/pypi/setuptools>

Clic sobre <https://bootstrap.pypa.io/ez_setup.py> y descargue (en una unidad con todos los permisos, por ejemplo la unidad D) el archivo ez\_setup.py

En la terminal, ejecute el siguiente comando:

d:\>python ez\_setup.py

* + Instalación de pip

C:\>easy\_install pip

* Si ya tiene una versión de python 2.7.x salte este paso.

## Descarga e instalación del proyecto

Descarga de Backengo desde GitHub de <https://github.com/submitconsulting/backengo>. Ingrese a la carpeta donde descomprimió o clonó el proyecto por ejemplo en D:\backengo-root\backengo

Cree un entorno virtual para el proyecto dentro de D:\backengo-root\

D:\backengo-root>virtualenv veb

Active el entorno virtual veb

D:\backengo-root> veb\Scripts\activate

Instale las dependencias según corresponda:

* Para desarrollo y pruebas

(veb)D:\backengo-root\backengo>pip install -r requirements.txt

* Para producción

(veb)D:\backengo-root\backengo>pip install -r production.txt

## Configurar idioma

El idioma por defecto es el español 'es-pe', todo el manual se hará en base al idioma predeterminado. Para cambiar solo basta actualizar en el archivo settings.py la variable:

LANGUAGE\_CODE = 'es-pe'

## Ejecución

Para ejecutar, colóquese en la carpeta del proyecto y ejecute el siguiente comando:

D:\backengo-root\backengo>python manage.py runserver

Luego en el browser de internet digite la dirección <http://localhost:8000/> y cargará la Página de inicio. Siga los botones que le indican en la página según lo que desea realizar. Todo es intuitivo, sin embargo se recomienda revisar esta guía para una mejor comprensión de lo que hace y cómo lo hace internamente. Puede complementar esta guía revisando el diseño del sistema de UML y la documentación técnica de los componentes.

También podrás editar y ejecutar backengo desde eclipse, gracias al plugin Pydev.

## Página de Inicio

La figura 3 muestra la página de inicio del sistema, la información de esta página deberá ser cambiada por la información de su nuevo proyecto, sólo deberá mantener el enlace del botón Iniciar sesión en alguna parte de esta página. Use la app home para estos fines.



Figura 3 – Página de inicio

A continuación de describe las acciones de los siguientes botones:

Diseño UML. Muestra el diseño del sistema en UML con lo diagramas básicos para el desarrollo distribuido. Las reglas de negocio indicadas en cada componente están debidamente implementadas, de modo que tal es la documentación tal es el código. Para no tener que volver a elaborar estos diagramas en Enterprise Architect puede solicitarlo por inbox a asullom@gmail.com.

Doc. en Django. Muestra la documentación de los componentes mediante django, requiere la instalación de las librerías docutils

Backend en GitHub. Muestra el proyecto en GitHub de donde podrá descargar esta aplicación y seguir las actualizaciones.

Principio lógico: La app home está separado totalmente de las otras apps, destinado para la página de su proyecto, por lo que puede ser cambiado sin afectar la funcionalidad.

## Formulario de Ingreso

La figura 4 muestra el formulario de ingreso al sistema, donde deberá identificarse mediante un usuario o correo electrónico y una contraseña. Inicialmente use la siguiente cuenta:

Usuario: admin

Contraseña: 12345

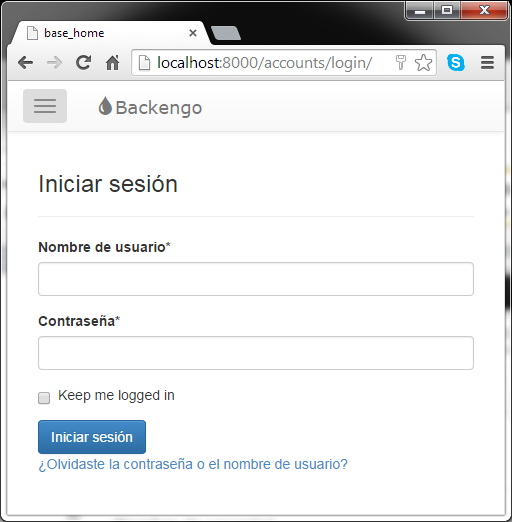


Figura 4 – Formulario de ingreso al sistema

Principio lógico: Un usuario no depende de ninguna sede, empresa o asociación en particular, sin embargo puede acceder a una o más empresas o asociaciones según los roles asignados a dicha asociación , empresa o sede.

## IR ADMIN

La figura 5, muestra la página de inicio de la app accounts donde se lista la sede con los perfiles y los módulos que el usuario tiene acceso. El superusuario admin tiene acceso a todas las empresas.

El backup inicial de datos del Backengo no tiene ninguna empresa agregada, Esto indica que el superusuario puede revisar o configurar los servicios SaaS que otorgará a sus clientes.

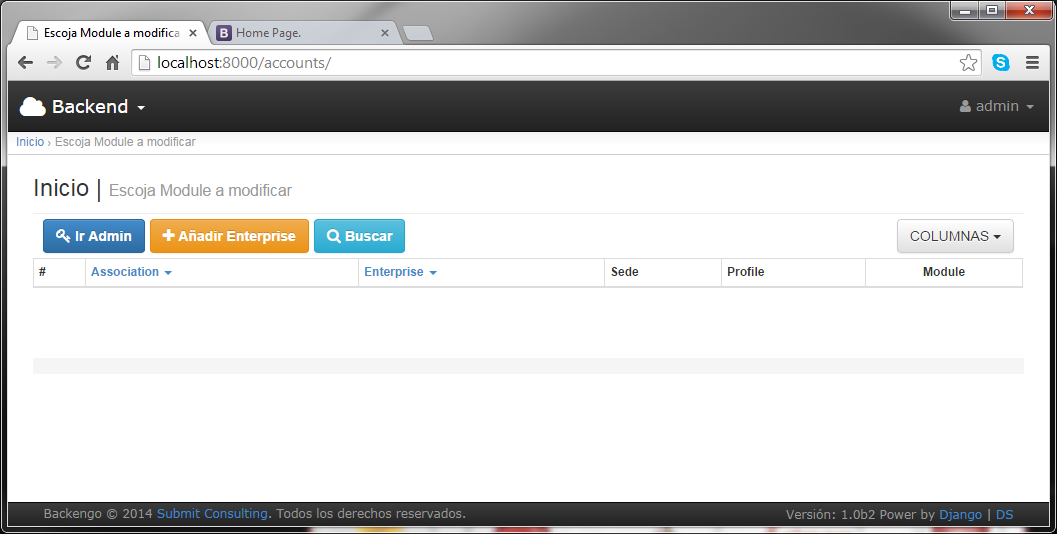
 “Elección de sede o empresa”

Figura 5 – Elección de la Sede

Para configurar el sistema, click sobre el botón IR ADMIN,

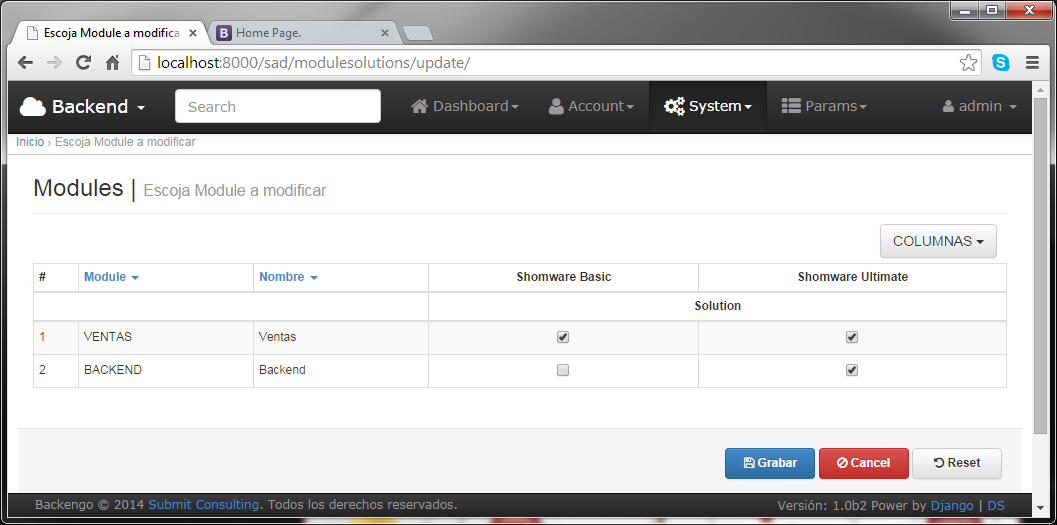


Figura 6 – Configración de los Planes de servicios

# Capítulo 4. Convenciones

Siguiendo las convenciones de Django, en Backengo se debe cumplir con las siguientes convenciones:

## Recursos (Permisos para Django).

Los recursos en el Backengo deben llevar las siguientes convenciones y características:

El recurso o permiso debe tener el nombre del **modelo** seguido de la **acción** separado por un “\_”. Para modelos o tablas con nombres compuestos deben nombrarse sin separación.

**Creación de un Permiso.**

Un recurso o permiso se puede crear de dos formas: Directamente en el modelo o haciendo uso del Backengo en el menú “Sistema”>“Permisos”. Por ejemplo: En el modelo

class LocalityType(models.Model):

"""

Tabla params\_localitytype para tipos de localidades.

P.e: Departamento, Provincia, Distrito, etc.

"""

name = models.CharField(max\_length=50)

class Meta:

permissions = (

("localitytype", "Puede hacer TODAS las oper. de tipos d localidades"),

#("localitytype\_index", "Puede ver el index de tipos de localidades"),

#("localitytype\_add", "Puede agregar tipo de localidad"),

#("localitytype\_edit", "Puede actualizar tipos de localidades"),

#("localitytype\_delete", "Puede eliminar tipos de localidades"),

#("localitytype\_report", "Puede reportar tipos de localidades"),

)

def \_\_unicode\_\_(self):

return self.name

o bien usando el Backengo como se muestra en la figura 7.

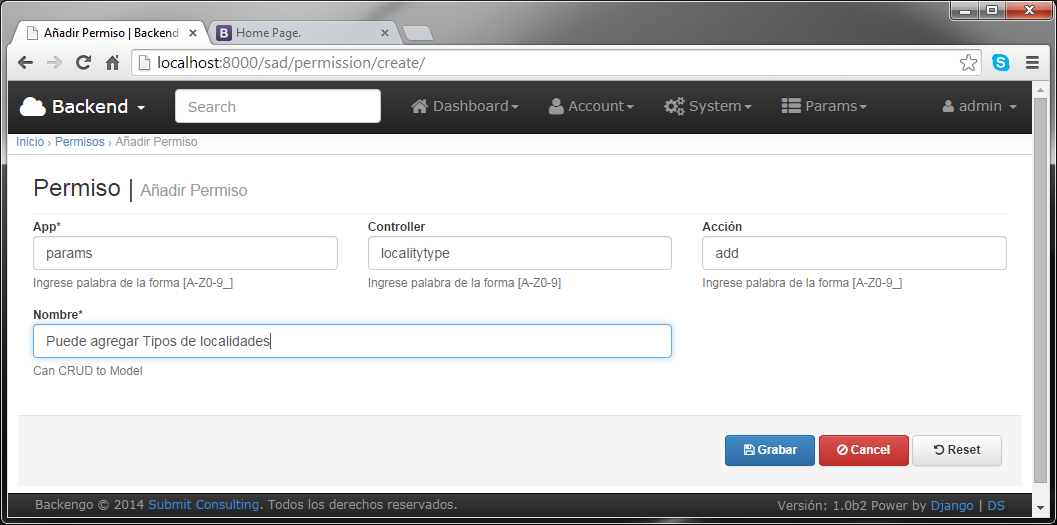


Figura 7 – Creación de recursos del sistema mediante el Backengo

Usando la Gestión de permisos del Backengo se puede definir los permisos o recursos para apps o modelos virtuales. El fundamento de esto, usted puede deducirlo de los permisos para django\_content\_type

## URLs (opcional).

Las urls deben crearse con las siguientes convenciones y características de django:

La url de las acciones para un CRUD de una tabla debe llevar el nombre indicado en el modelo o en la Gestión de permisos del Backengo separados por un “/”. Para modelos o tablas con nombres compuestos sus controladores deben nombrarse sin separación. Por ejemplo:

urlpatterns = patterns('apps.params.views',

#localitytype controllers

url(r'^localitytype/index/$', 'localitytype\_index', name='localitytype\_index'),

url(r'^localitytype/add\_form/$', 'localitytype\_add', name='localitytype\_add'),

url(r'^localitytype/edit/(?P<key>.\*)/$', 'localitytype\_edit', name='localitytype\_edit'),

url(r'^localitytype/delete/(?P<key>.\*)/$', 'localitytype\_delete', name='localitytype\_delete'),

url(r'^localitytype/report/$', 'localitytype\_report', name='localitytype\_report'),

#Fin localitytype controllers

)

Así una de las urls sería: <http://localhost:8000/params/localitytype/add_form/>

Importante: Una acción puede reducirse al nivel del controlador o hasta el nivel de la aplicación. Por ejemplo:

urlpatterns = patterns('apps.home.views',

url(r'^choice\_headquart/$', 'choice\_headquart', name='choice\_headquart'),

url(r'^$', 'index', name='index'),

)

Así las urls serían: <http://localhost:8000/home/choice_headquart/> y<http://localhost:8000/home/> respectivamente.

## Menús.

Todo menú debe crearse dentro de un módulo.

Los menús ítems o hijos deben estar asociado a un recurso, si este no está asociado a un recurso no se mostrará en el menú (ya sea menú bar o sidebar). La gura 8 muestra el llenado correcto del formulario para crear menú.

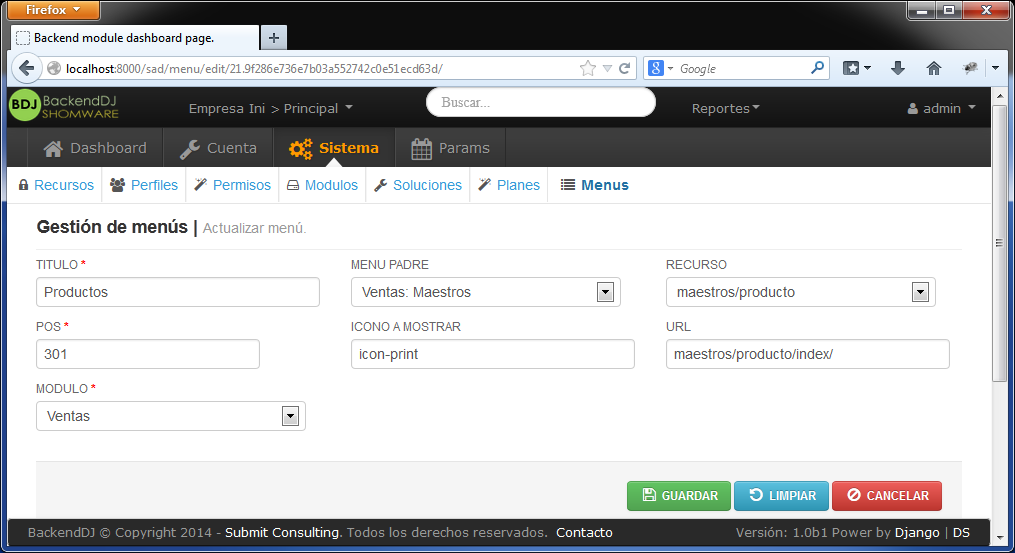


Figura 8 – Creación de menús

Estos 3 son las únicas convenciones que debes cumplir en todo el proyecto. Recuerde que estas convenciones ayudan a cumplir las convenciones de django.

# Capítulo 5. Mi nuevo proyecto con nueva base de datos.

## Creando su nuevo Proyecto utilizando Backengo

Backengo fue creado como el primer módulo de su nuevo proyecto con django. A continuación se indica una forma para cambiar el nombre backengo por el nombre de su nuevo proyecto por ejemplo shomware. Y sobre todo para tener nuevas llaves de seguridad generadas por django.

* Paso 1: Elija una unidad segura y defina una estructura universal de carpetas para el adecuado trabajo distribuido (en equipo).

D:\dev\apps>

Todas las PCs del equipo de trabajo debe tener esta misma estructura de carpetas y en la misma unidad.

* Paso 2: Pegue la carpeta Backengo en D:\dev\apps>
* Paso 3: Cambie de nombre a la carpeta que está dentro de Backengo\Backengo por el suyo Backengo\shomware
* Paso 4: Cambie de nombre a la carpeta raíz Backengo por el suyo shomware quedando shomware\shomware
* Paso 5: Dentro de la carpeta raíz shomware, elimine los archivos de configuración de eclipse y de git. Para que luego pueda editarlo desde eclipse y subir a su propio repositorio git.
* Paso 6: En otro lugar, cree su nuevo proyecto

(Esto es para obtener las nuevas llaves de seguridad de django y actualizarlos en D:\dev\apps\shomware\shomware\settings.py)

Ejecutando el siguiente comando:

D:\>django-admin.py startproject shomware

* Paso 7: Reemplace la palabra Backengo de los archivos D:\dev\apps\shomware\shomware\settings.py y D:\dev\apps\shomware\shomware\wsgi.py por shomware
* Paso 8: Actualice la variable SECRET\_KEY de settings.py
* Paso 9: Ejecute

D:\dev\apps\shomware>python manage.py runserver

## Volviendo a sincronizar la base de datos

**Paso 1:** Elimine la base de datos SQLite de \backengo\db.sqlite3

**Paso 2:** Migramos la base de datos.

Actualice la variable de configuración DATABASES del settings.py con la base de datos que desea usar. Luego ejecute el siguiente comando:

(ve17) D:\dev\apps\backengo-root\backengo>python manage.py migrate

…

Operations to perform:

Synchronize unmigrated apps: home, utils, accounts, mod\_backend, crispy\_forms

Apply all migrations: sessions, admin, space, sites, flatpages, sad, contentty

pes, params, auth

Synchronizing apps without migrations:

Creating tables...

Installing custom SQL...

Installing indexes...

Running migrations:

Applying contenttypes.0001\_initial... OK

Applying auth.0001\_initial... OK

Applying sad.0001\_initial... OK

Applying admin.0001\_initial... OK

Applying sites.0001\_initial... OK

Applying flatpages.0001\_initial... OK

Applying params.0001\_initial... OK

Applying params.0002\_remove\_person\_photo2... OK

Applying space.0001\_initial... OK

Applying sad.0002\_auto\_20140925\_0620... OK

Applying sessions.0001\_initial... OK

(ve17) D:\dev\apps\backengo-root\backengo>

**Paso 3:** Usando el cualquier administrador de base de datos truncar o eliminar todos los registros de todas las tablas, o ejecute el script:

delete from django\_content\_type;

delete from auth\_permission;

Tenga cuidado si usted ya tiene nuevos recursos, grupos, módulos, planes y menús.

Luego, restaurar la data inicial desde json, ejecutando el siguiente comando:

(ve17) D:\dev\apps\backengo-root\backengo>python manage.py loaddata ini\_data

Installed 95 object(s) from 1 fixture(s)

(ve17) D:\dev\apps\backengo-root\backengo>

Este backup fue creado ejecutando el siguiente comando:

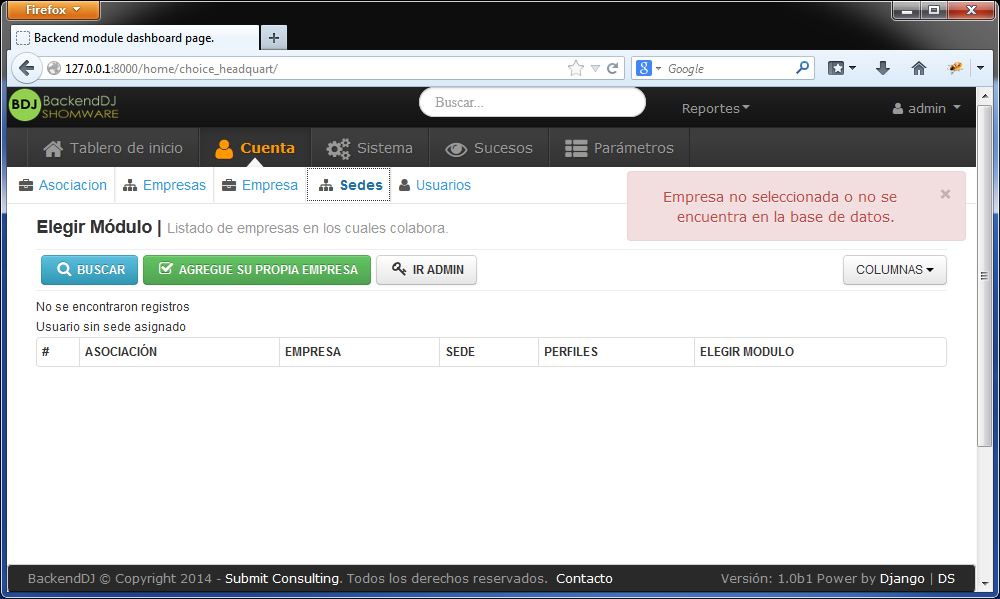
(ve17) D:\dev\apps\backengo-root\backengo>python manage.py dumpdata > fixtures/initial\_data.json

Use estos comandos para hacer copias de seguridad de su data

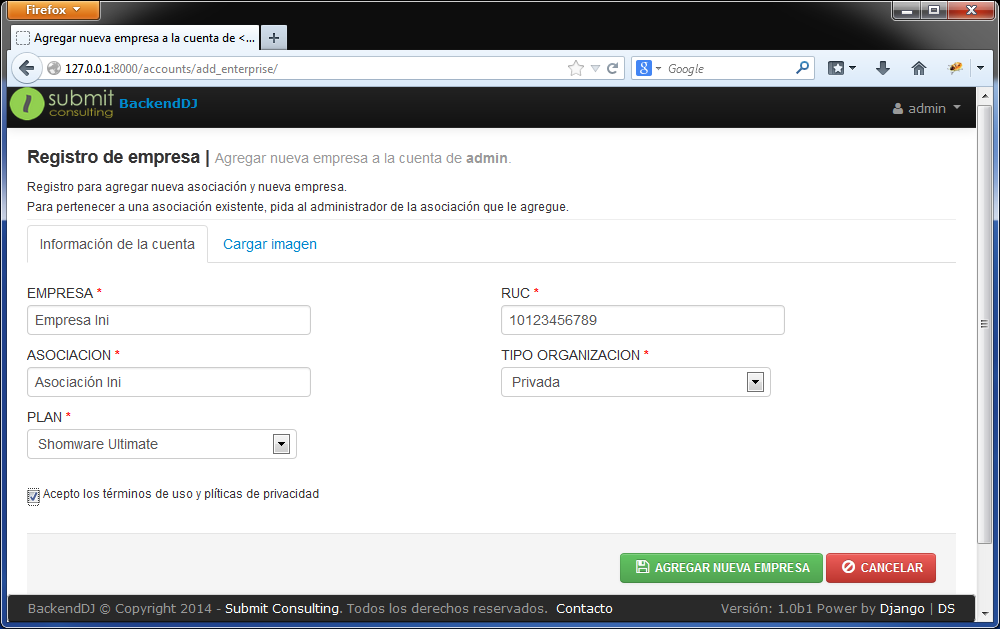
# Capítulo 6. Manual del usuario //TODO

Esta sección queda como TODO porque no está terminado, esperamos tu colaboración.

## Agregando nueva empresa para el usuario en sesión

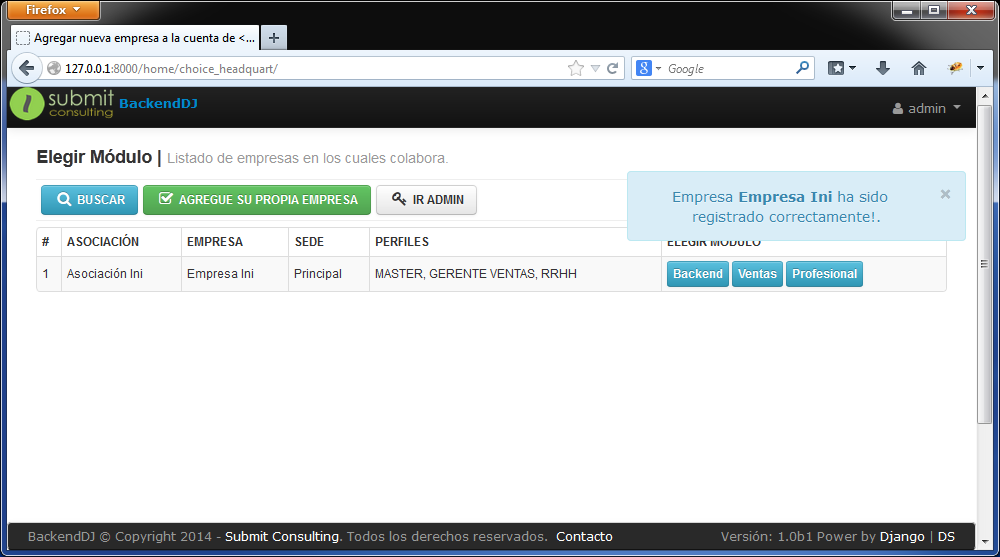


Para administrar la cuenta debemos crear una empresa, puede crearlo directamente haciendo clic en el botón “AGREGUE SU PROPIA EMPRESA”



## Elección del módulo y de la sede

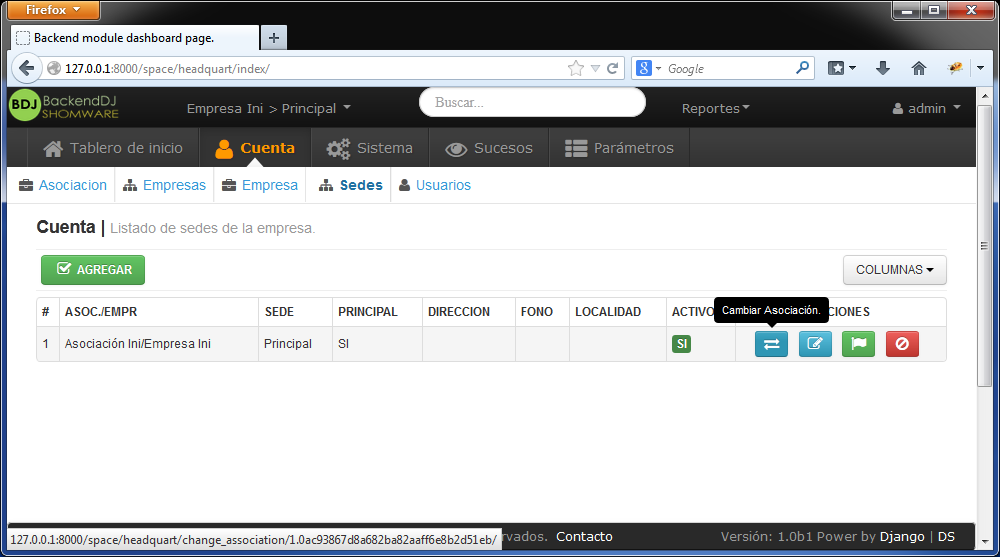
Vaya a la página de inicio y en el listado de empresas en las cuales colabora, elija el módulo de la sede a administrar.



Elija el módulo Backend.

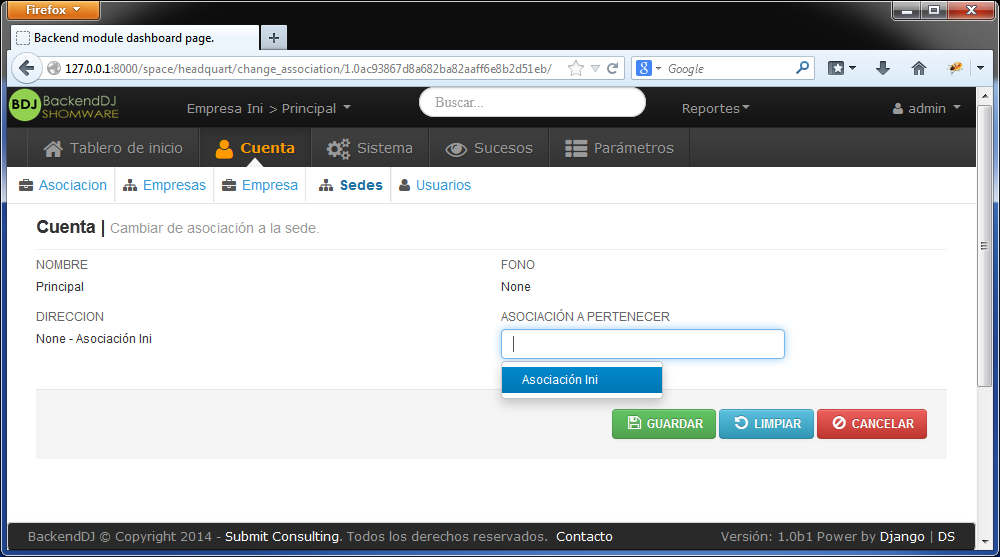
## Agregar Sede

Para agregar nueva sede, elija el módulo Backend, luego el menú “Cuenta”>”Sedes”, finalmente clic en el botón “AGREGAR”



## Cambiar de Asociación a una Sede

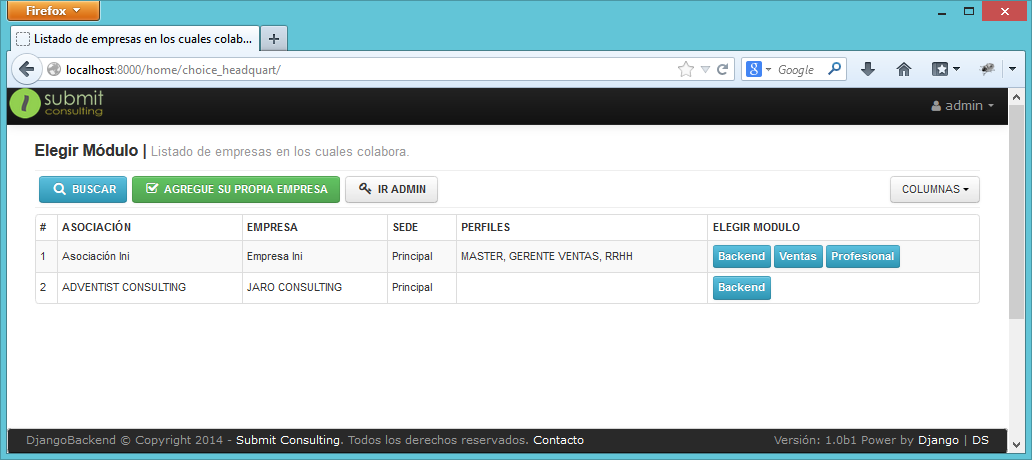
Haga clic en el botón “Cambiar Asociación” de la sede a cambiar:

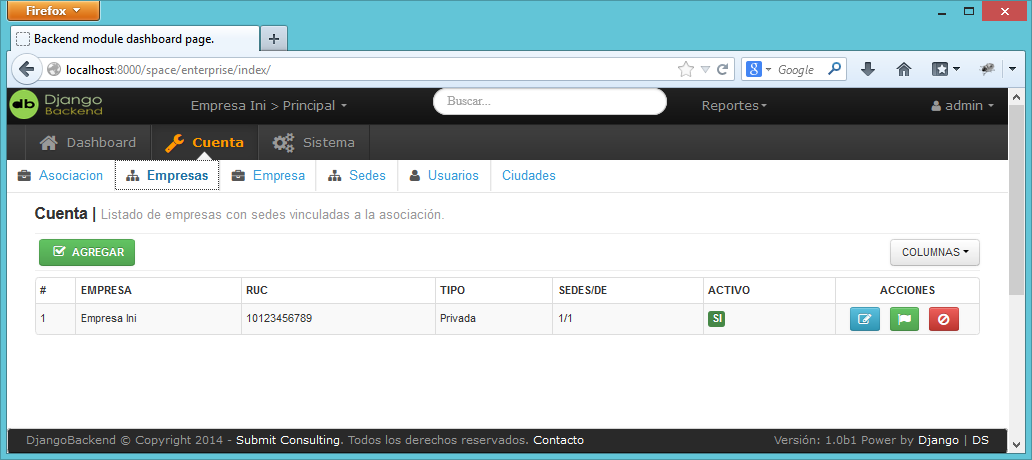


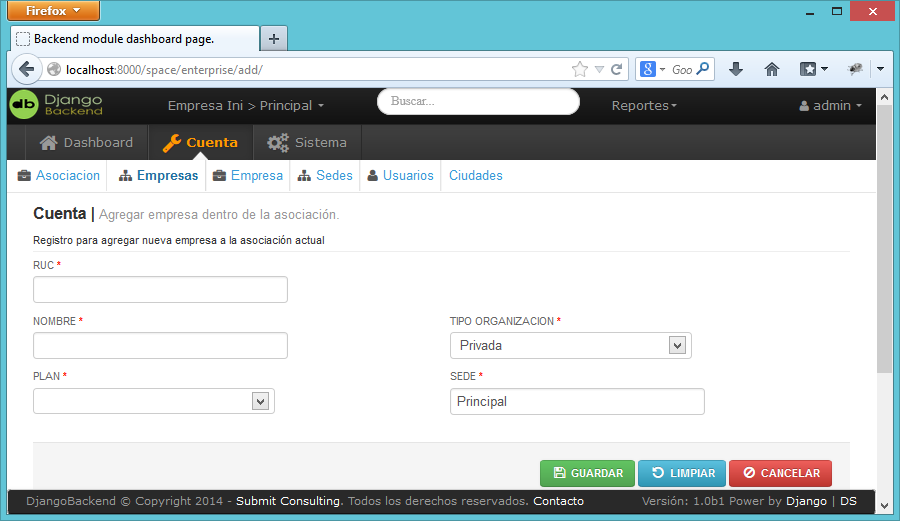
Busque la nueva asociación previamente registrado. Finalmente clic en “GUARDAR”

## Agregar Empresa dentro de una Asociación

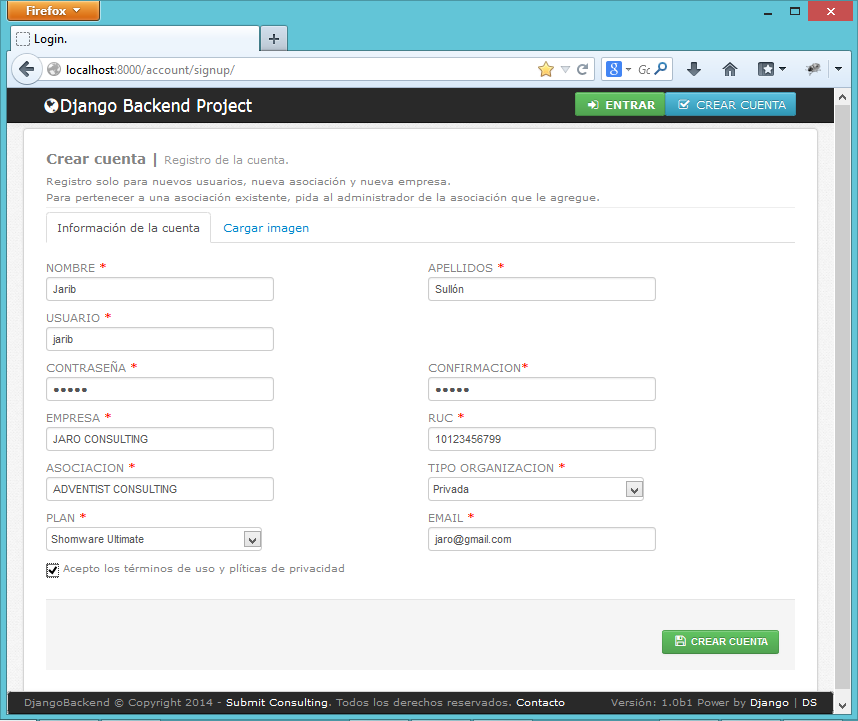
Para agregar nueva empresa dentro de una asociación específica, en la “Página de inicio” elija el módulo Backend de la SEDE a administrar, luego el menú “Cuenta”>”Empresas”, finalmente clic en el botón “AGREGAR”

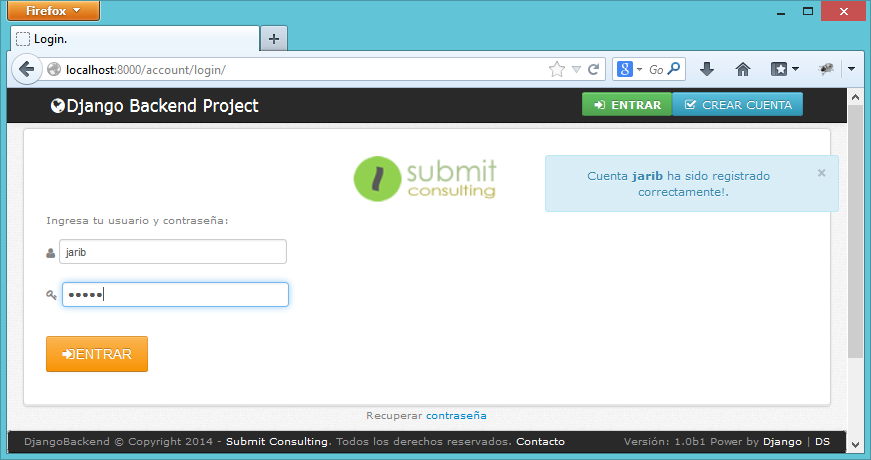


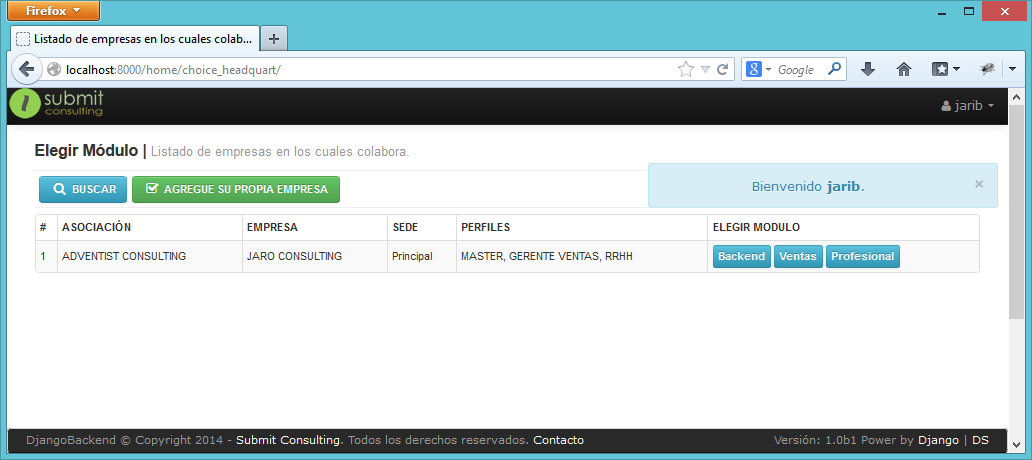




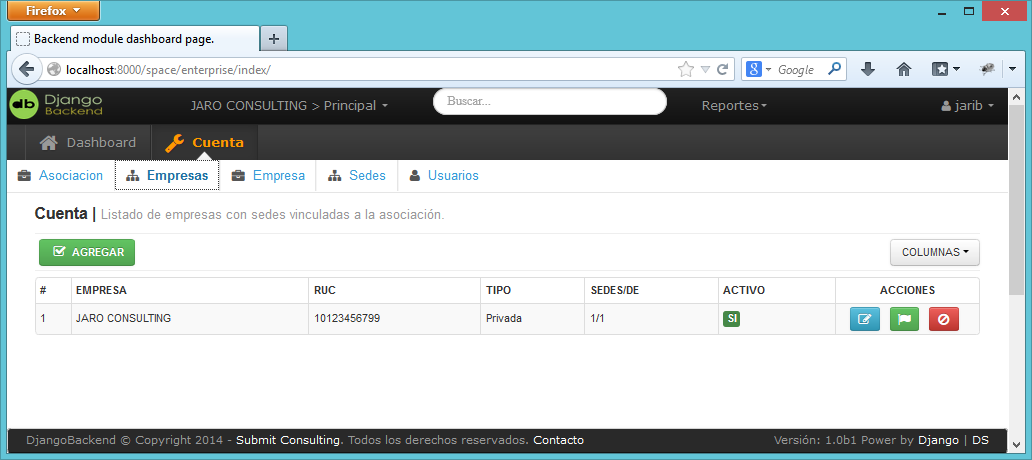
## CREAR CUENTA

Para crear una nueva cuenta de prueba, clic en el botón “CREAR CUENTA”:

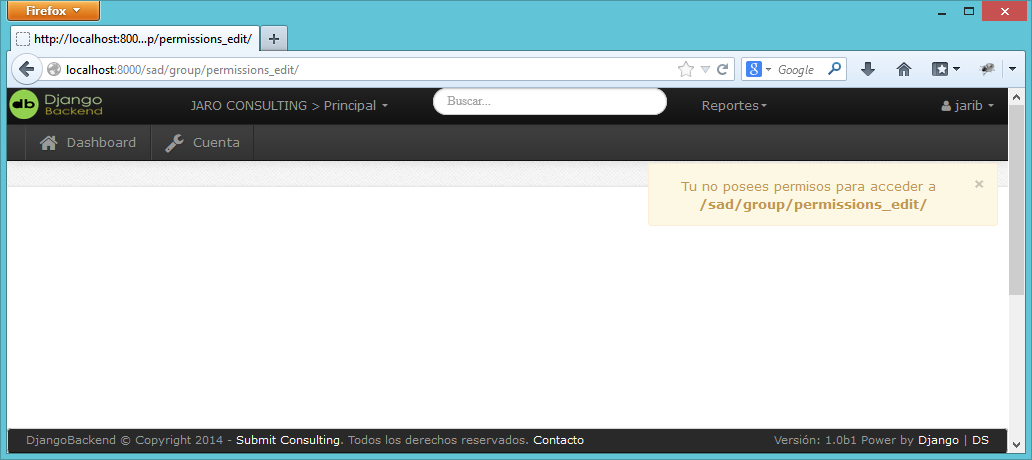




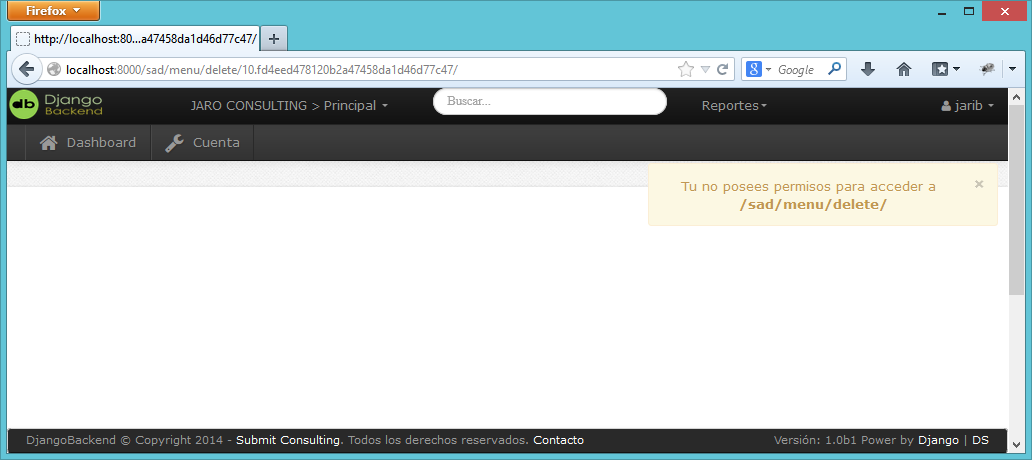
Este usuario tiene acceso al Backend, pero con accesos limitados.



Imagine que desea agregar permisos



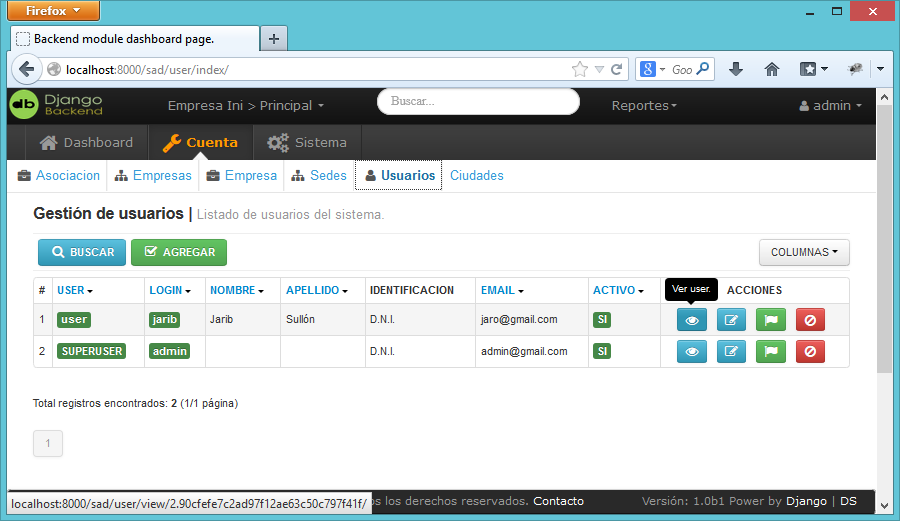
O eliminar un menú



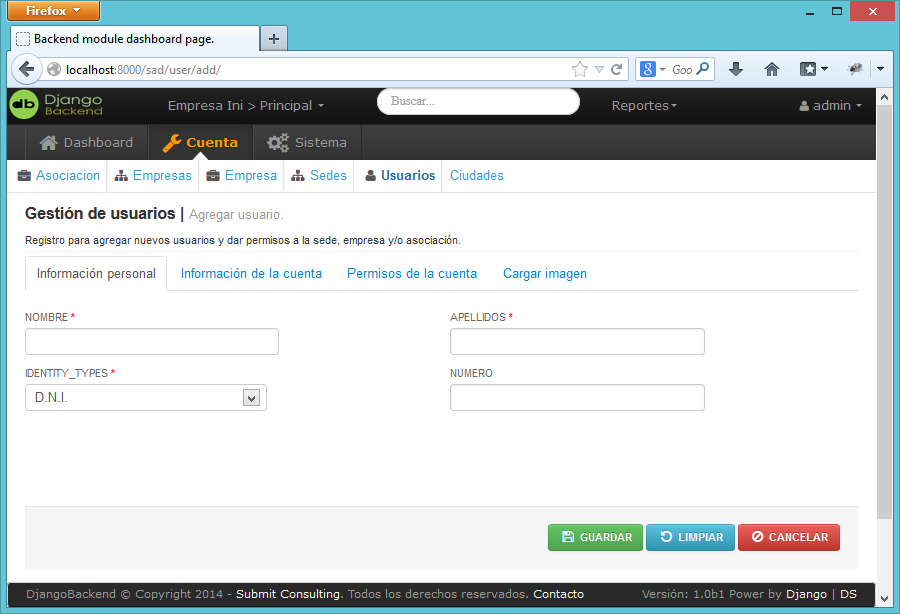
La llave de seguridad /10.fd4eed478120b2a47458da1d46d77c47/ solo le permitirá eliminar datos a las cuales el usuario tiene acceso, si se modifica la clave intentando acceder a otro registro, el sistema el enviará el mensaje de llave inválida.

## Usuarios

Para crear usuario, en el mismo menú “Cuenta” ir a menú ítem “Usuarios”:



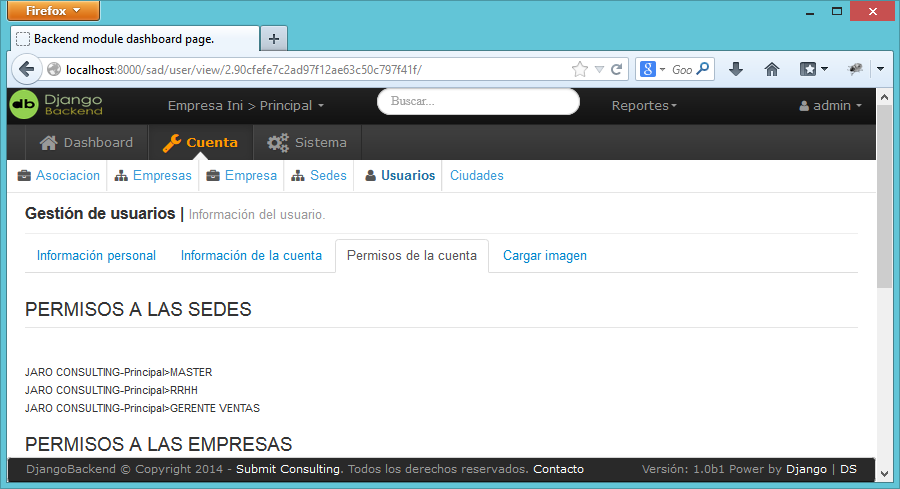
Si desea crear un nuevo usuario clic en el botón “AGREGAR”



Si el usuario ya existe y desea otorgar privilegios a la sede en cuestión, solo haga clic en el botón “Actualizar user”, no necesita crear otro usuario.



Como vemos el usuario “jarib” no tiene acceso a la sede “Empres ini > Principal”. Usted puede ver a qué sedes y con qué perfil un usuario tiene acceso: En el listado clic en el ícono del ojo.



# Capítulo 7: Acerca de…

## Contáctese con el autor

Para un entrenamiento con el equipo de su proyecto o para extender las facilidades de este Backend a sus necesidades, escribanos a:

E-mail: [asullom@gmail.com](mailto:asullom@gmail.com)   
Celular dentro de PERÚ al RPC989597352/RPM\*065067(990720536)

