



Manual del Programador

Django Backend Manager v1.0

@DjangoBackend

20-12-2013

Autor: Angel Sullón Macalupú (@asullom)

Contenido

Descripción.....	3
Convenciones	5
1. Recursos o permisos.....	5
2. URLs.....	6
3. Menús.....	7
Requerimientos de software.....	8
Instalación y Ejecución	8
Usar otra base de datos como MySQL, PostgreSQL, etc.	9
Página de inicio	10
Formulario de Ingreso	11
IR ADMIN.....	12
Agregando nueva empresa para el usuario en sesión	13
Elección del módulo y de la sede	14
Agregar Sede	15
Cambiar de Asociación a una Sede	15
Agregar Empresa dentro de una Asociación	16
CREAR CUENTA.....	17
Usuarios.....	21
Contáctese con el autor	23

Descripción

El Backend es el módulo base con la parte administrativa lista para usar en aplicaciones web seguras escritas en Python y con la elegancia de Bootstrap. Integra componentes o librerías necesarios para comenzar y completar una aplicación web moderna y segura.

Sus principales características son:

Gestión de Usuarios del sistema. Los usuarios tienen acceso limitados a los recursos del sistema y los datos de las sedes los que le han sido otorgados por un súper usuario. Los súper usuarios tienen accesos ilimitados, pudiendo acceder a cualquier recurso del sistema. Si puede acceder a cualquier recurso, puede auto habilitarse para acceder a los datos de cualquier sede o empresa.

Gestión de Perfiles de usuario. Un perfil de usuario o rol indica las funciones y responsabilidades de un usuario dentro del sistema. Ejemplo: usuarios masters, gerentes, cajeros, etc.

Gestión de Recursos del sistema. Comúnmente un recurso es un archivo y la ruta de carpetas que lo contiene. En una aplicación web MVC viene a ser cada una de las acciones que se pueden ejecutar en los controladores, los controladores mismos o los módulos o apps que lo contiene. Ejemplo de recursos:

app_name/controllername/action_name/ Acción específica del Controlador del Módulo

app_name/controllername/ Todas las acciones del Controlador del Módulo

app_name/ Todos los Controladores y Acciones del Módulo

Importante: El controllername debe nombrarse de corrido sin subguión. Por ejemplo:
mod_backend/loclitytype/add_form/ Para mayor detalle ir a la sección de [Convenciones](#)

Gestión de Permisos de usuarios. Permite establecer a qué Recursos tiene acceso cada Perfil de usuario dentro del sistema.

Gestión de Módulos del sistema. Permite personalizar los Módulos del sistema asociando un conjunto de Perfiles de usuario.

Gestión de Soluciones del sistema. Una Solución indica el nivel del servicio a ofrecer a los clientes o usuarios del sistema. Ejemplo: Básico, Profesional, Empresarial, etc.

Gestión de Planes de Servicio (SaaS). Permite establecer a qué Módulos tiene acceso cada Solución del sistema, esto es con el fin de personalizar los Módulos que conforma el Servicio que se ofrece a los clientes.

Gestión de Menús. Cada menú está asociado a un Recurso y a un Módulo del sistema, esto con el fin de generar menús dinámicos que solo carguen los ítems a los que un Perfil de usuario tenga acceso dentro de un Módulo. Mayor detalle en la sección de [Convenciones](#)

Manual del Programador

Gestión de Asociaciones. Permite la administración de datos básicos de la asociación, así como el cambio de plan de servicio.

Gestión de Empresas. Permite la administración de datos básicos de la empresa. Y permite la creación, edición y eliminación de empresas cuyas sedes están vinculadas a una asociación específica, así como el cambio de plan de servicio.

Gestión de Sucursales. Permite la creación, edición y eliminación de sucursales de la empresa, así como el cambio de asociación.

Adicional a esto los componentes de seguridad, mensajería, carga de archivos, etc. del Backend es flexible para integrar en otros módulos de la aplicación. El frontend es compatible entre navegadores que combina HTML5, CSS3 y Javascript y se adecúa a diferentes dispositivos como celulares, tabletas y PCs.

Convenciones

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

1. Recursos o permisos.

Los recursos o permisos en el Backend 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 Recurso o permiso.

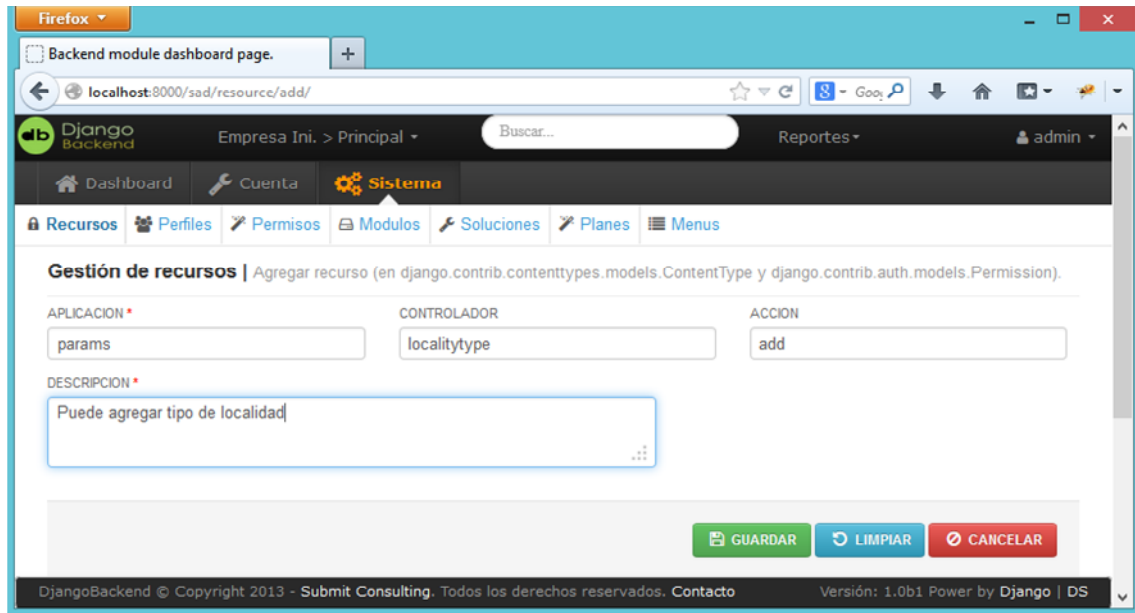
Un recurso o permiso se puede crear de dos formas: Directamente en el modelo o haciendo uso del Backend en el menú “Recursos”. Por ejemplo:

```
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 backend:



Usando la Gestión de recursos del Backend 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`

2. URLs.

Las urls deben crearse con las siguientes convenciones y características:

La url de las acciones para un CRUD de una tabla debe llevar el nombre indicado en el modelo o en la Gestión de recursos del Backend 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'^$', 'index', name='index'),
    url(r'^choice_headquart/$', 'choice_headquart', name='choice_headquart'),
)
```

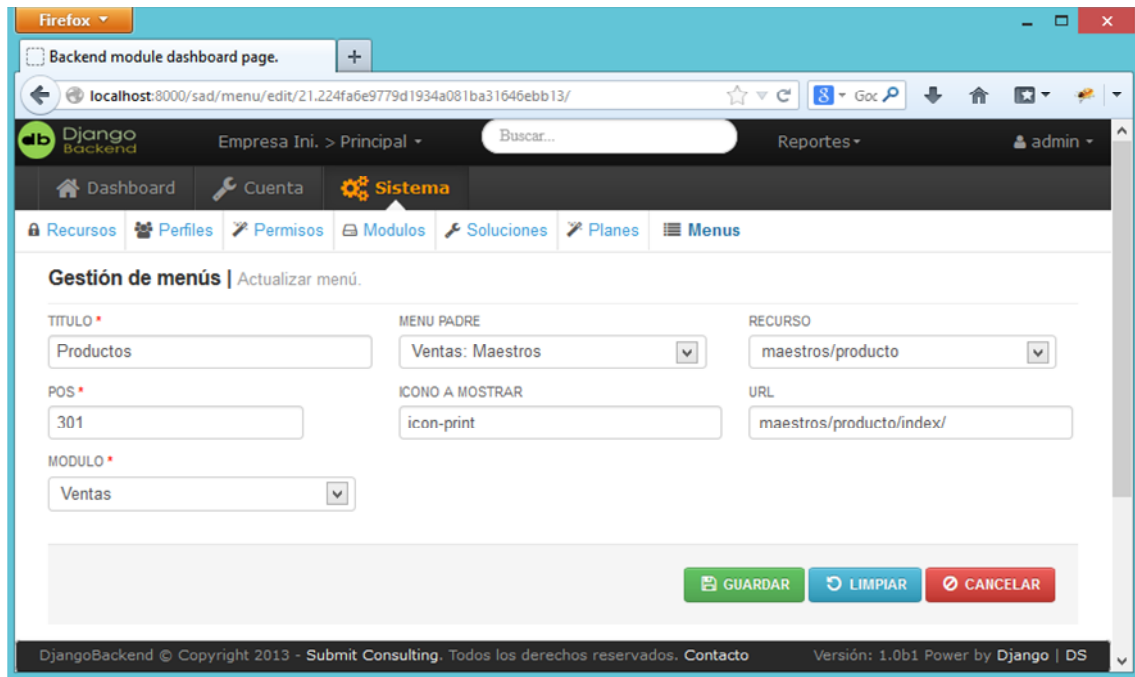
Manual del Programador

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

3. 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 bar o del sidebar.



Estos 3 son las únicas convenciones que debes cumplir en todo el proyecto. Estas convenciones se deben a los algoritmos de los componentes de seguridad y de renderización de los controladores o vistas.

Requerimientos de software

Se requiere de los siguientes programas:

Python 2.7.5

Django==1.5.1

PIL==1.1.7

+ Puede ver el archivo `\django_backend_project\requirements.txt`

Instalación y Ejecución

Una vez instalado los requerimientos, para instalar el Backend simplemente lo descargas de https://github.com/submitconsulting/django_backend_project

Para ejecutar ingresa a la carpeta `django_backend`, luego ejecuta el siguiente comando:

```
\django_backend_project\django_backend >python manage.py runserver
```


Usar otra base de datos como MySQL, PostgreSQL, etc.

Paso 1: Elimine la base de datos SQLite de
`\django_backend_project\django_backend\sdbm.db`

Paso 2: Creamos la base de datos y el super usuario.

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

```
\django_backend_project\django_backend >python manage.py syncdb
...
You just installed Django's auth system, which means you don't have any superuse
rs defined.

Would you like to create one now? (yes/no): yes
Username (leave blank to use 'yourpcname'): admin
Email address: admin@gmail.com
Password:12345
Password (again):12345
Superuser created successfully.

Installing custom SQL ...

Installing indexes ...

Installed 0 object(s) from 0 fixture(s)
```

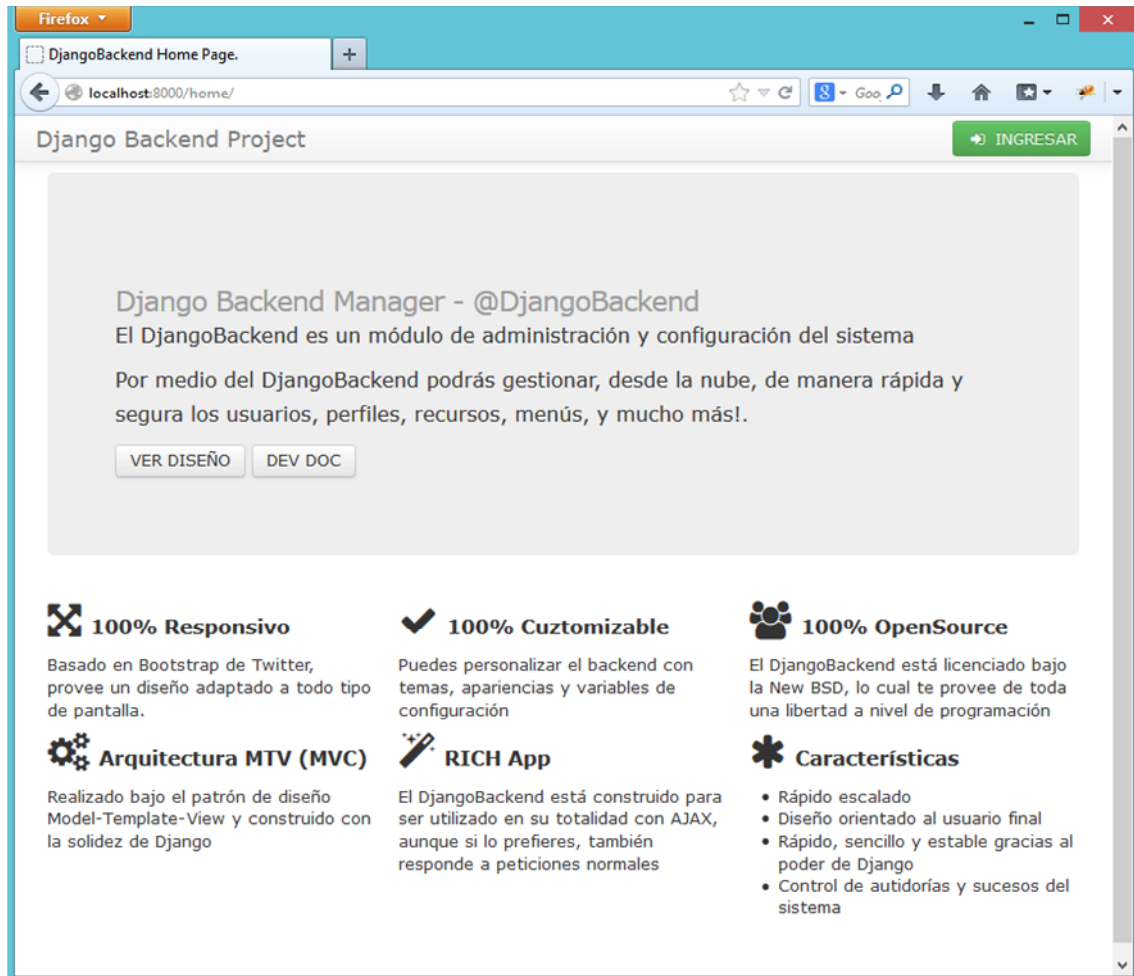
Paso 3: Usando el cualquier administrador de base de datos abrir el archivo
`\django_backend_project\django_backend\reset backend db.sql` y ejecutar todas las consultas. Tenga cuidado si usted ya tiene nuevos recursos, perfiles, módulos, planes y menús. Para MySQL es necesario truncar o limpiar la base de datos, luego abrir el archivo *reset backend db.sql* y ejecutar todas las consultas.

Paso 4: Ejecute con el siguiente comando:

```
\django_backend_project\django_backend >python manage.py runserver
```

Página de inicio

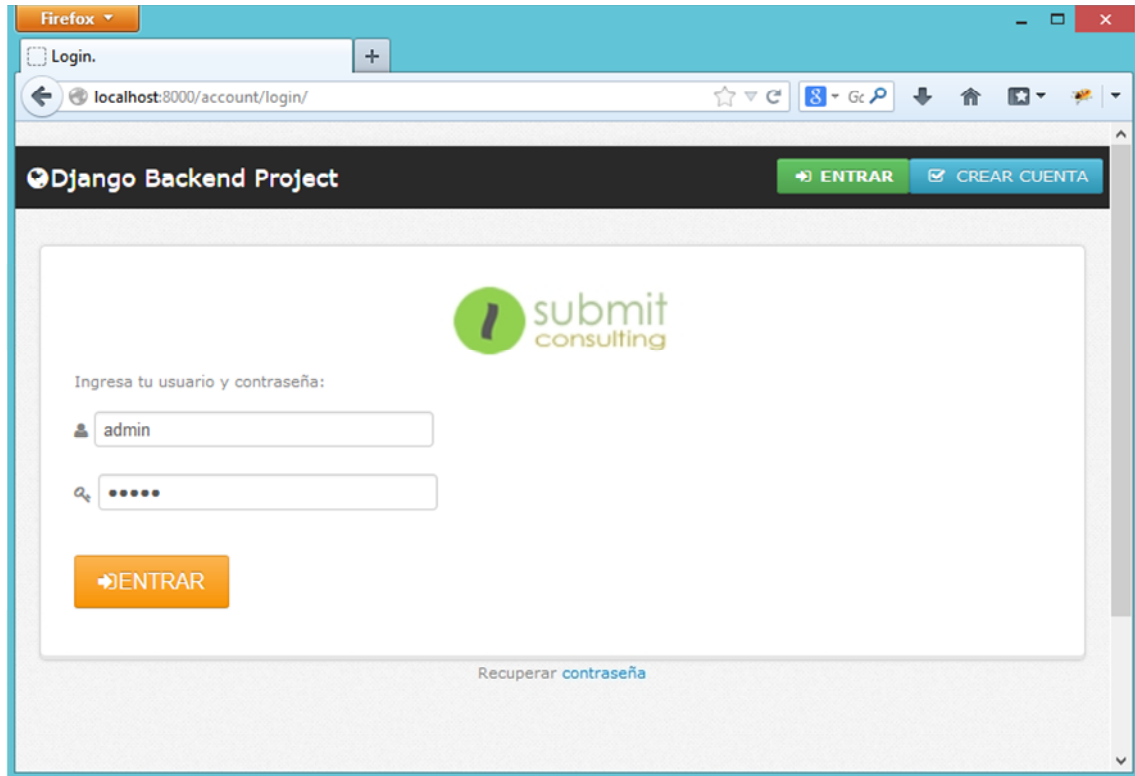
Para ingresar al sistema, digita en el browser de internet la siguiente dirección <http://localhost:8000/> y luego todo es intuitivo.



Formulario de Ingreso

Usuario: admin

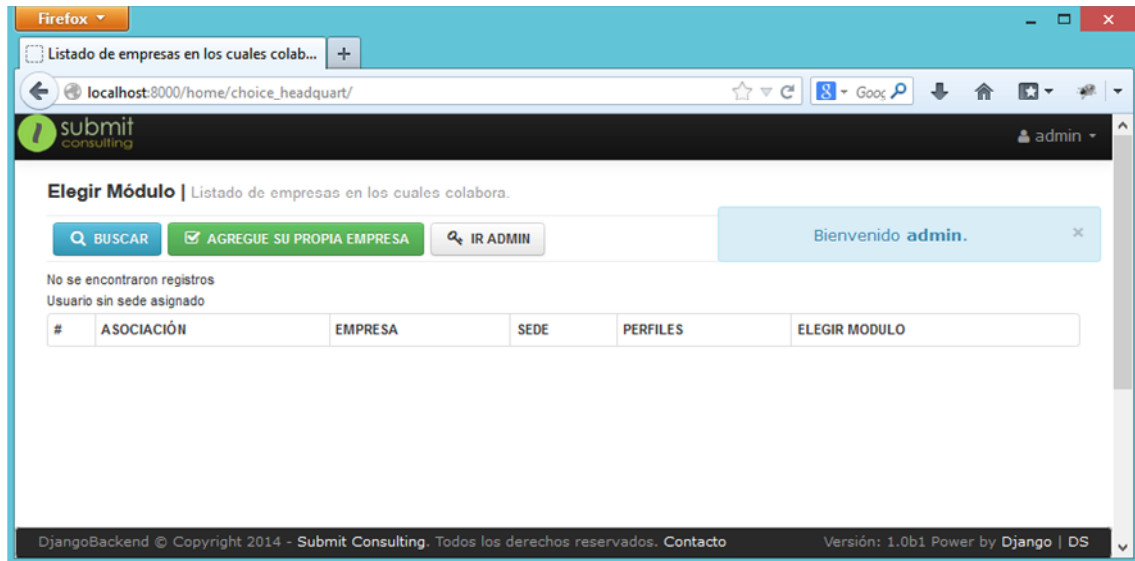
Contraseña: 12345



O ingrese el nuevo usuario y contraseña con la que sincronizó su base de datos.

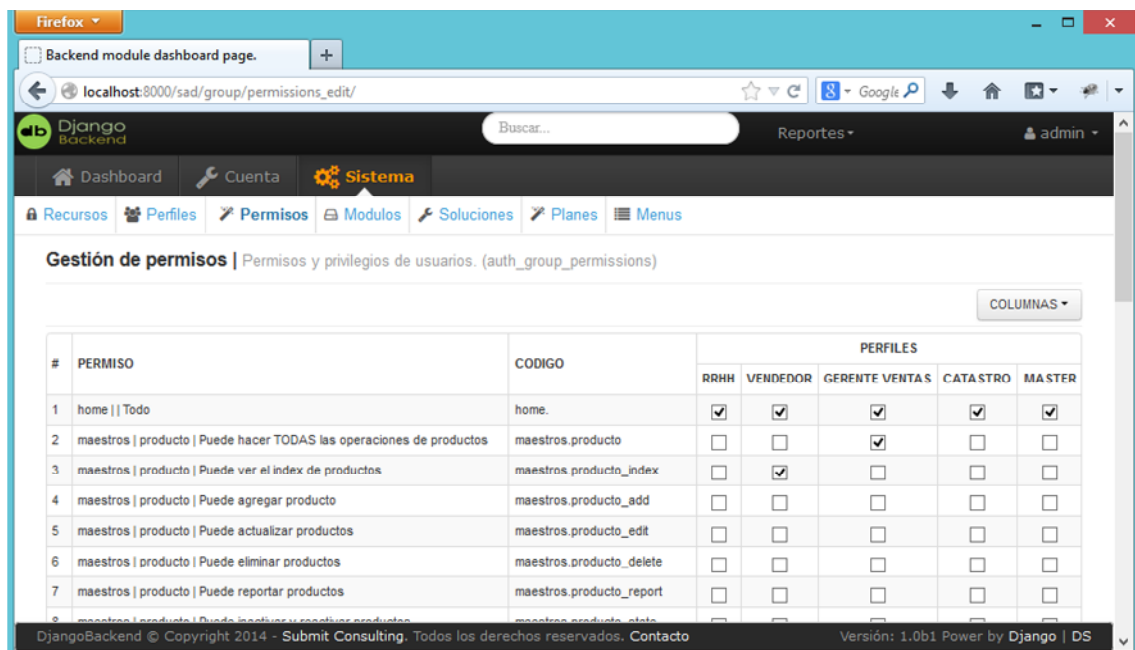
IR ADMIN

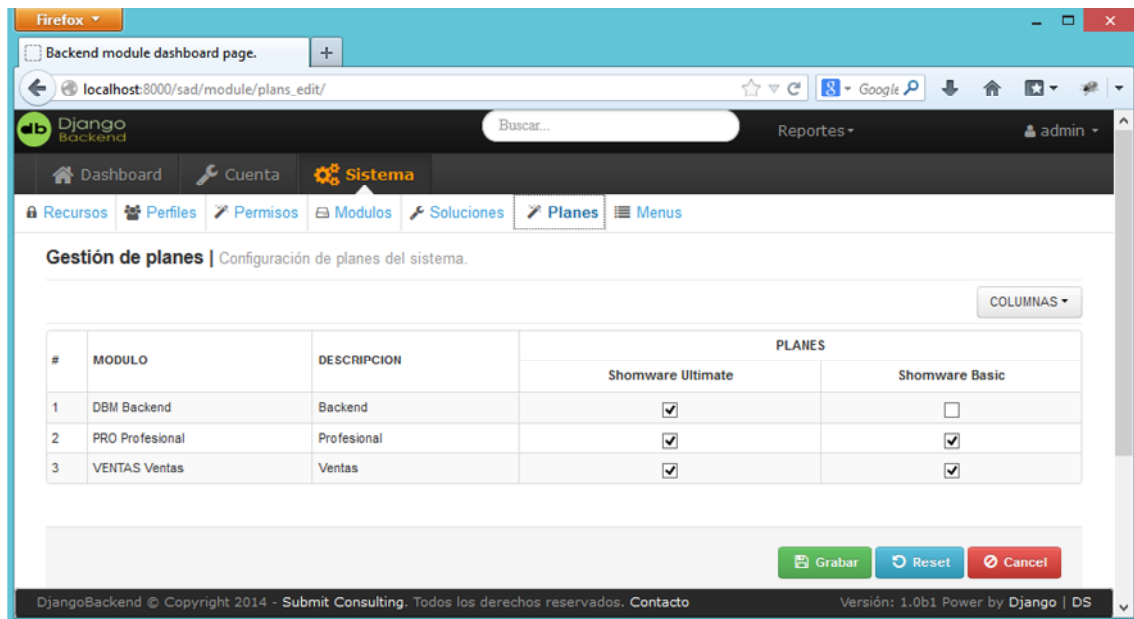
Inicialmente, en la “Página de inicio”, no lista ninguna empresa en el Listado de empresas en las cuales colabora, esto quiere decir que primero hay que revisar los planes de servicios brindados.



Esta es la “Página de Inicio”

Para ello clic en el botón “IR ADMIN” y revise:

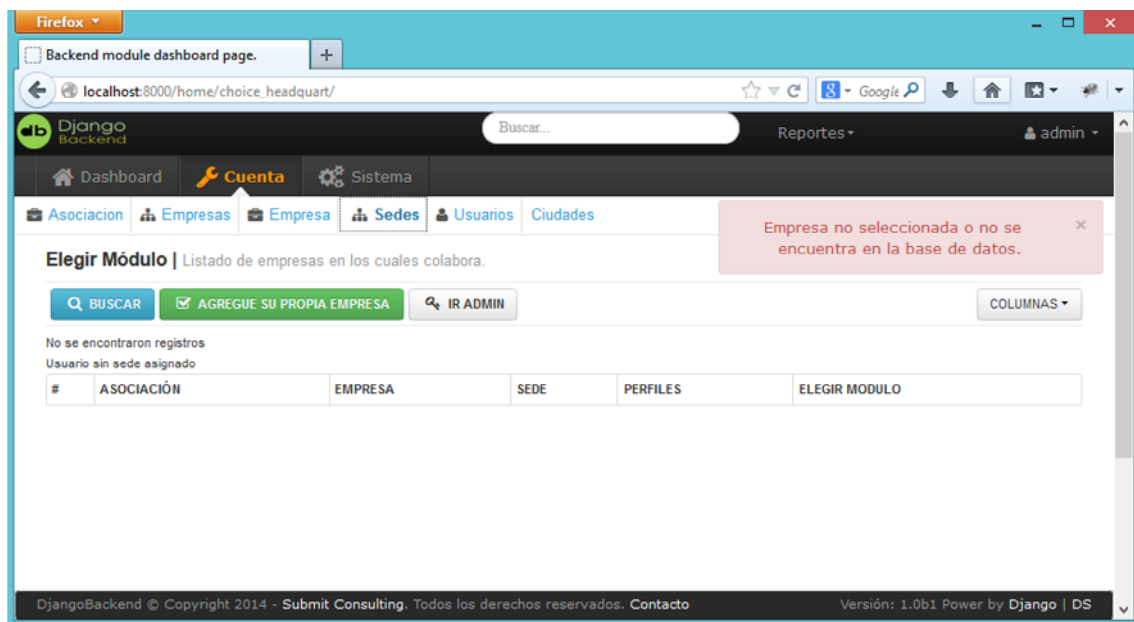




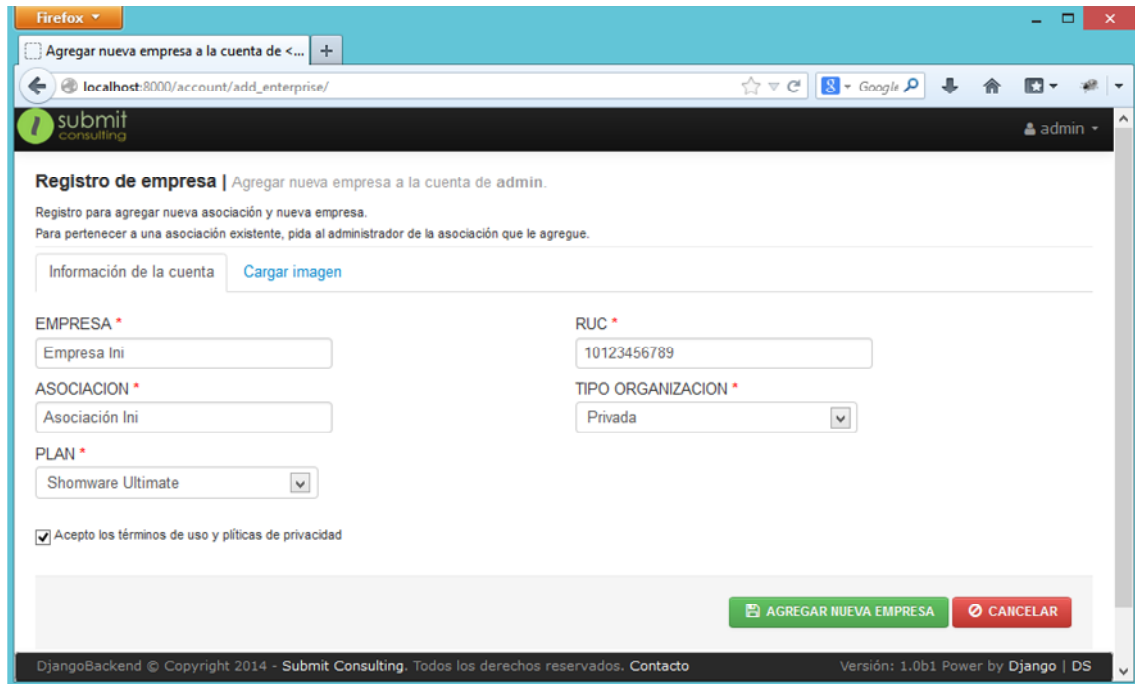
Haga los cambios según sus planes.

Agregando nueva empresa para el usuario en sesión

Ir a la “Página de inicio” o bien intente administrar la cuenta:



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



Firefox - Agregar nueva empresa a la cuenta de <...>

localhost:8000/account/add_enterprise/

submit consulting admin

Registro de empresa | Agregar nueva empresa a la cuenta de admin.

Registro para agregar nueva asociación y nueva empresa.
Para pertenecer a una asociación existente, pida al administrador de la asociación que le agregue.

Información de la cuenta [Cargar imagen](#)

EMPRESA *
Empresa Ini

RUC *
10123456789

ASOCIACION *
Asociación Ini

TIPO ORGANIZACION *
Privada

PLAN *
Shomware Ultimate

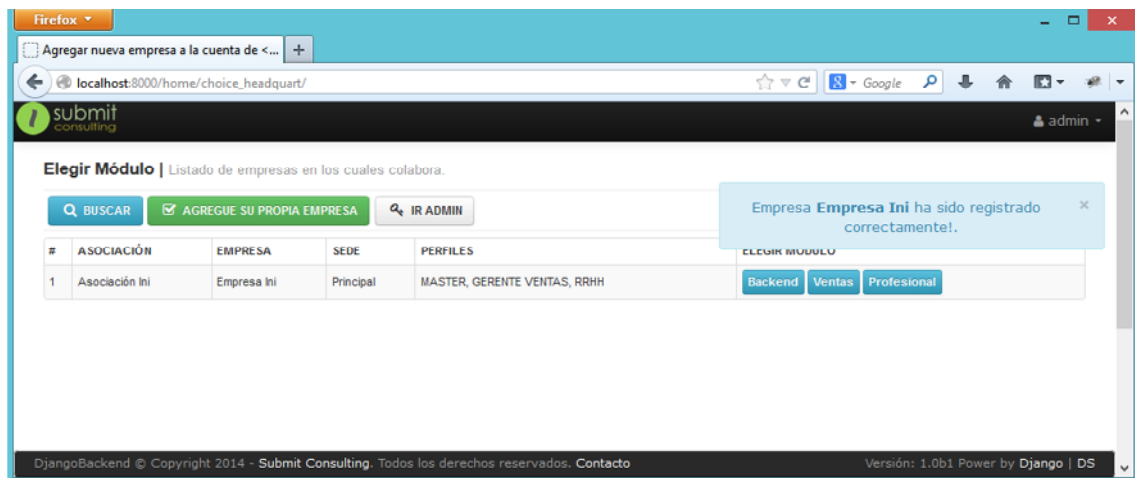
☒ Acepto los términos de uso y políticas de privacidad

[AGREGAR NUEVA EMPRESA](#) [CANCELAR](#)

DjangoBackend © Copyright 2014 - Submit Consulting. Todos los derechos reservados. Contacto Versión: 1.0b1 Power by Django | DS

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.



Firefox - Agregar nueva empresa a la cuenta de <...>

localhost:8000/home/choice_headquart/

submit consulting admin

Elegir Módulo | Listado de empresas en los cuales colabora.

[BUSCAR](#) [AGREGUE SU PROPIA EMPRESA](#) [IR ADMIN](#)

Empresa Empresa Ini ha sido registrado correctamente!

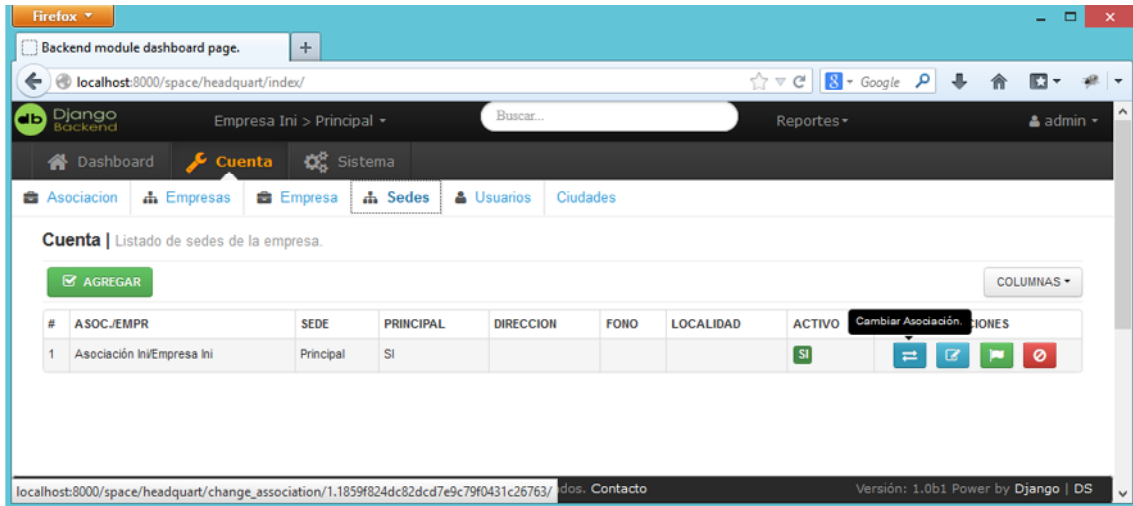
#	ASOCIACIÓN	EMPRESA	SEDE	PERFILES	ELEGIR MÓDULO
1	Asociación Ini	Empresa Ini	Principal	MASTER, GERENTE VENTAS, RRHH	Backend Ventas Profesional

DjangoBackend © Copyright 2014 - Submit Consulting. Todos los derechos reservados. Contacto Versión: 1.0b1 Power by Django | DS

Elija el módulo Backend.

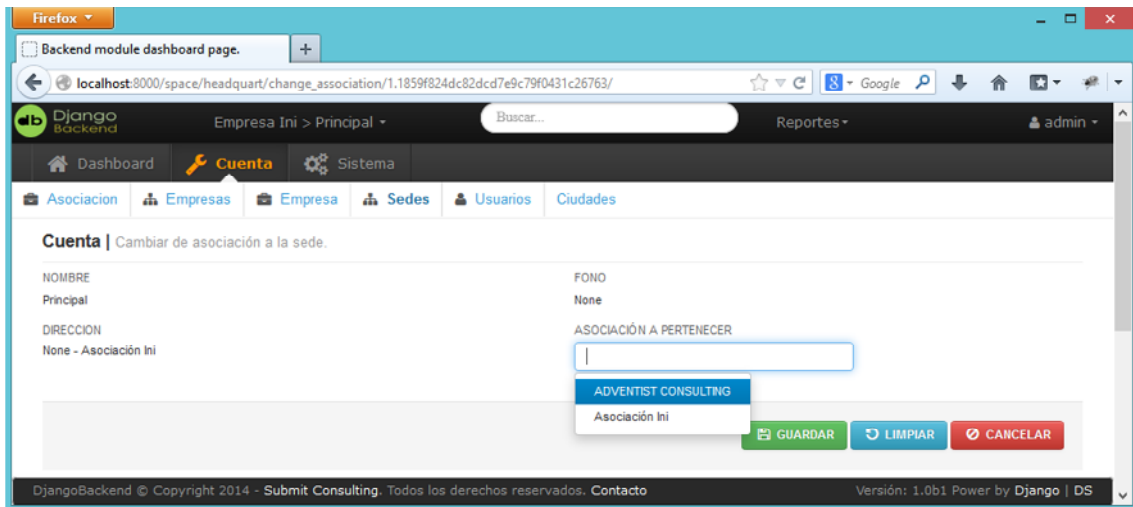
Agregar Sede

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

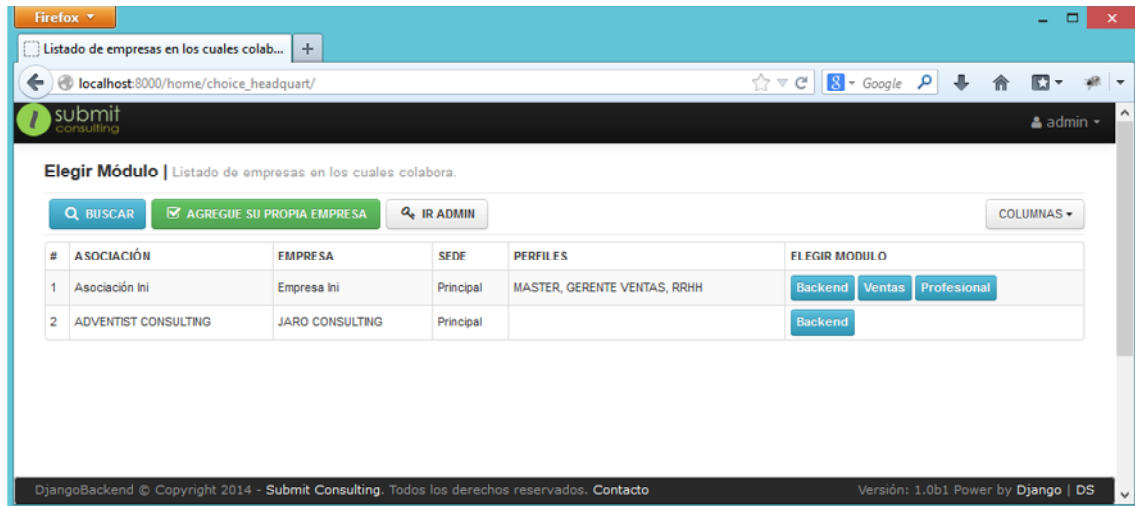
Haga clic en el botón “Cambiar Asociación” de la sede a cambiar:



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”



Firefox

Listado de empresas en los cuales colab...

localhost:8000/home/choice_headquart/

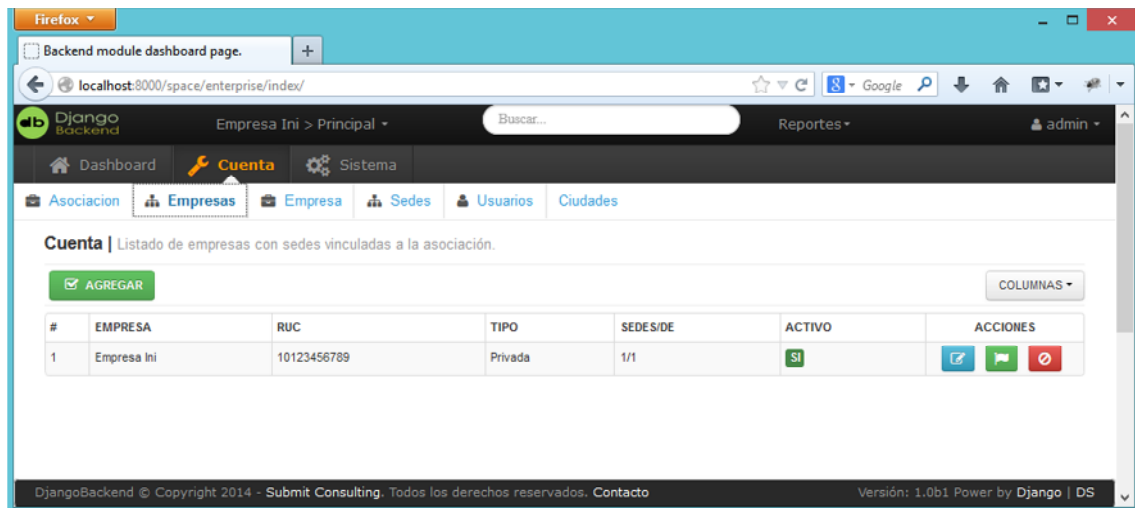
submit consulting admin

Elegir Módulo | Listado de empresas en los cuales colabora.

BUSCAR AGREGUE SU PROPIA EMPRESA IR ADMIN COLUMNAS

#	ASOCIACIÓN	EMPRESA	SEDE	PERFILES	ELIGIR MODULO
1	Asociación Ini	Empresa Ini	Principal	MASTER, GERENTE VENTAS, RRHH	Backend Ventas Profesional
2	ADVENTIST CONSULTING	JARO CONSULTING	Principal		Backend

DjangoBackend © Copyright 2014 - Submit Consulting. Todos los derechos reservados. Contacto Versión: 1.0b1 Power by Django | DS



Firefox

Backend module dashboard page.

localhost:8000/space/enterprise/index/

Django Backend Empresa Ini > Principal Buscar... Reportes admin

Dashboard Cuenta Sistema Asociacion Empresas Empresa Sedes Usuarios Ciudades

Cuenta | Listado de empresas con sedes vinculadas a la asociación.

AGREGAR COLUMNAS

#	EMPRESA	RUC	TIPO	SEDES/DE	ACTIVO	ACCIONES
1	Empresa Ini	10123456789	Privada	1/1	SI	

DjangoBackend © Copyright 2014 - Submit Consulting. Todos los derechos reservados. Contacto Versión: 1.0b1 Power by Django | DS

Firefox Backend module dashboard page.

localhost:8000/space/enterprise/add/

Django Backend Empresa Ini > Principal Buscar... Reportes admin

Dashboard Cuenta Sistema

Asociacion Empresas Empresa Sedes Usuarios Ciudades

Cuenta | Agregar empresa dentro de la asociación.
Registro para agregar nueva empresa a la asociación actual

RUC *

NOMBRE *

PLAN *

TIPO ORGANIZACION *
Privada

SEDE *
Principal

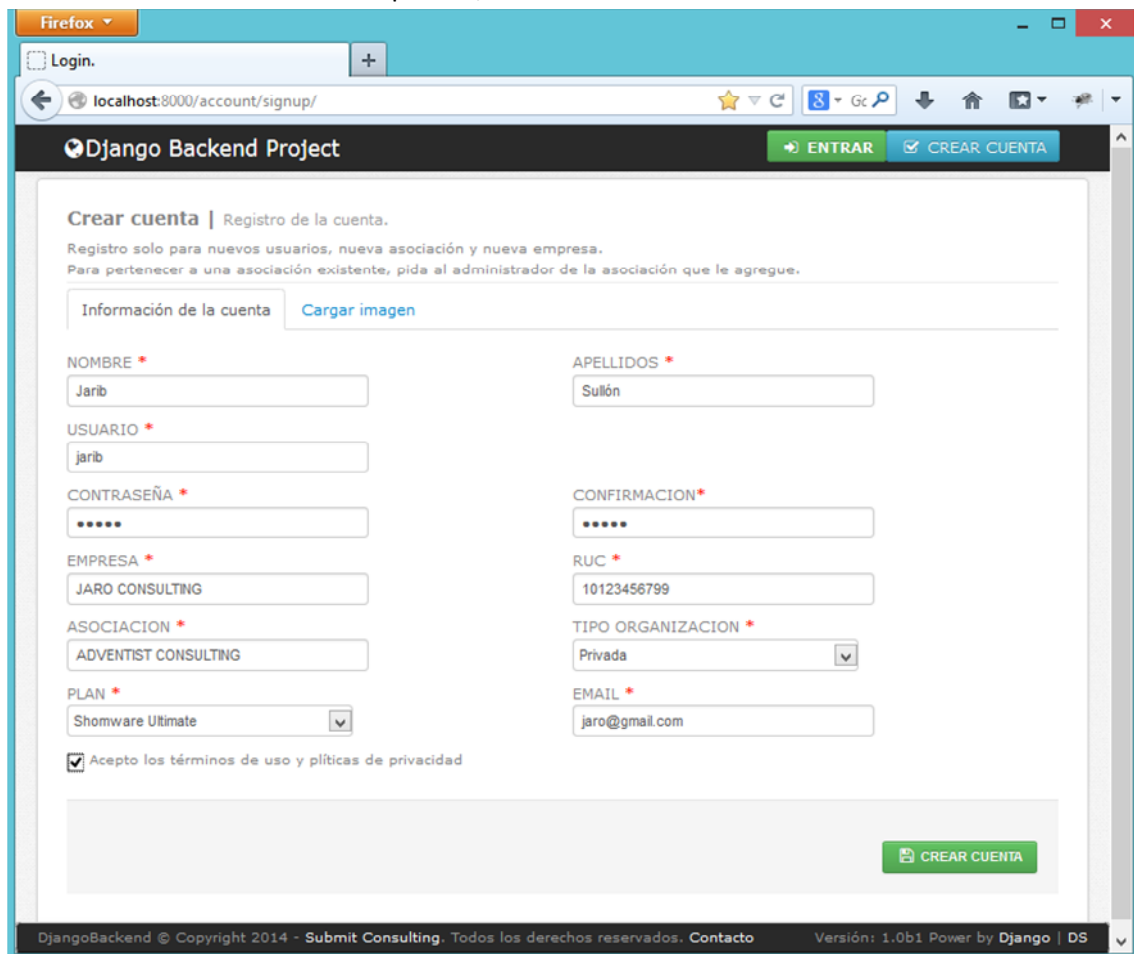
GUARDAR LIMPIAR CANCELAR

DjangoBackend © Copyright 2014 - Submit Consulting. Todos los derechos reservados. Contacto Versión: 1.0b1 Power by Django | DS

CREAR CUENTA

Manual del Programador

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



Firefox Login. localhost:8000/account/signup/

Django Backend Project ENTRAR CREAR CUENTA

Crear cuenta | Registro de la cuenta.
Registro solo para nuevos usuarios, nueva asociación y nueva empresa.
Para pertenecer a una asociación existente, pida al administrador de la asociación que le agregue.

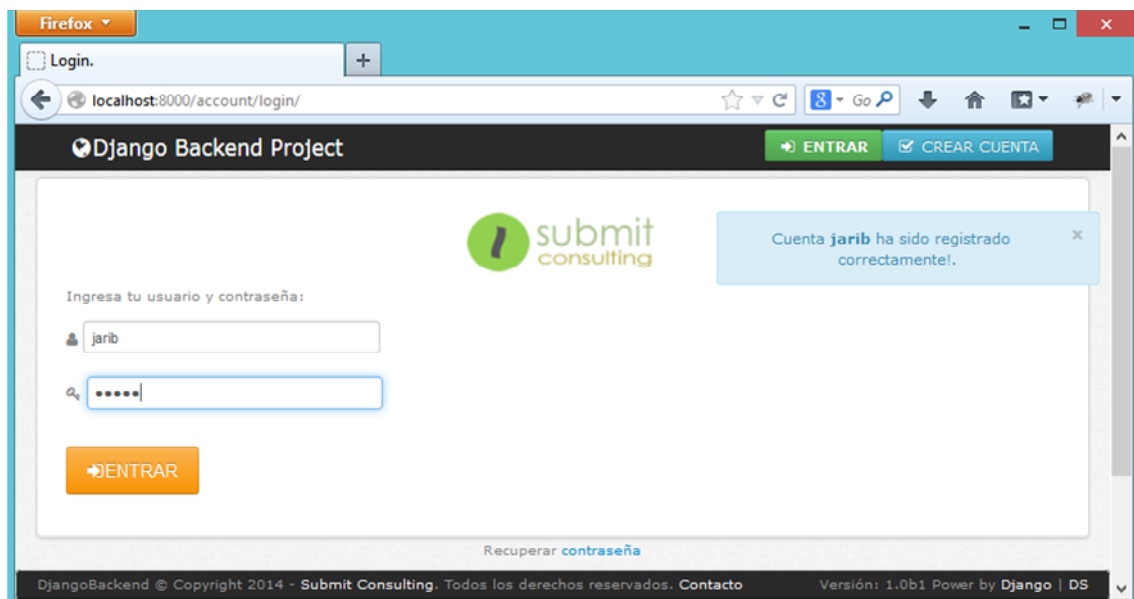
Información de la cuenta Cargar imagen

NOMBRE * jarib
APELLIDOS * Sullón
USUARIO * jarib
CONTRASEÑA *
CONFIRMACION *
EMPRESA * JARO CONSULTING
RUC * 10123456799
ASOCIACION * ADVENTIST CONSULTING
TIPO ORGANIZACION * Privada
PLAN * Showware Ultimate
EMAIL * jaro@gmail.com

☒ Acepto los términos de uso y políticas de privacidad


CREAR CUENTA

DjangoBackend © Copyright 2014 - Submit Consulting. Todos los derechos reservados. Contacto Versión: 1.0b1 Power by Django | DS



Firefox Login. localhost:8000/account/login/

Django Backend Project ENTRAR CREAR CUENTA

 submit consulting

Cuenta jarib ha sido registrado correctamente!

Ingresa tu usuario y contraseña:

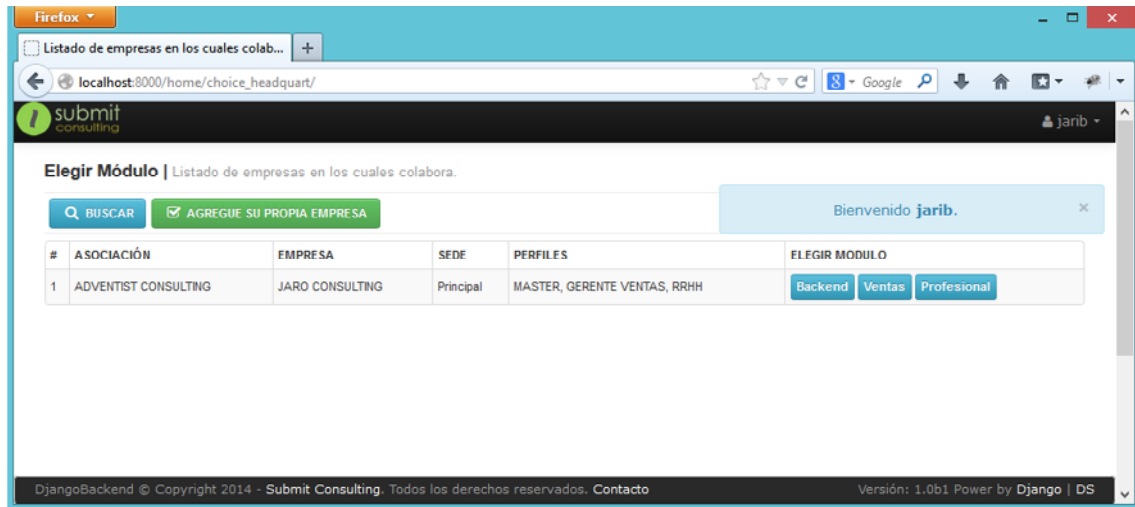
jarib
.....

ENTRAR

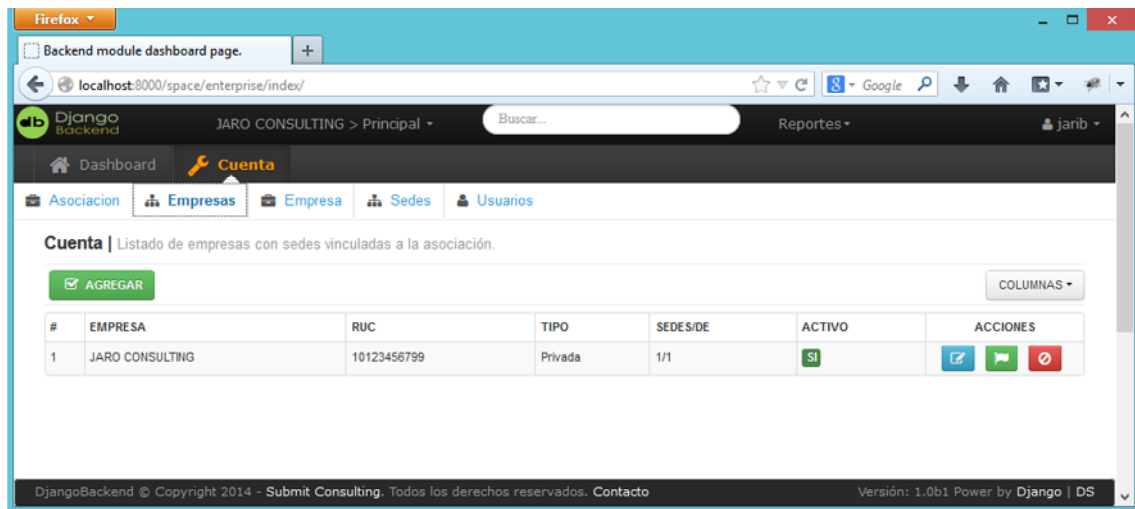
Recuperar contraseña

DjangoBackend © Copyright 2014 - Submit Consulting. Todos los derechos reservados. Contacto Versión: 1.0b1 Power by Django | DS

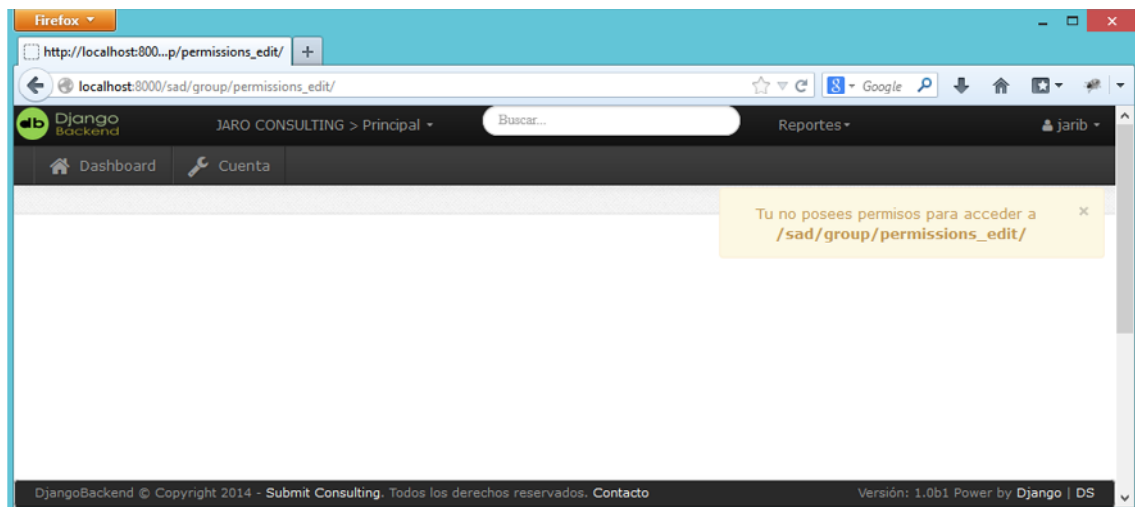
Manual del Programador



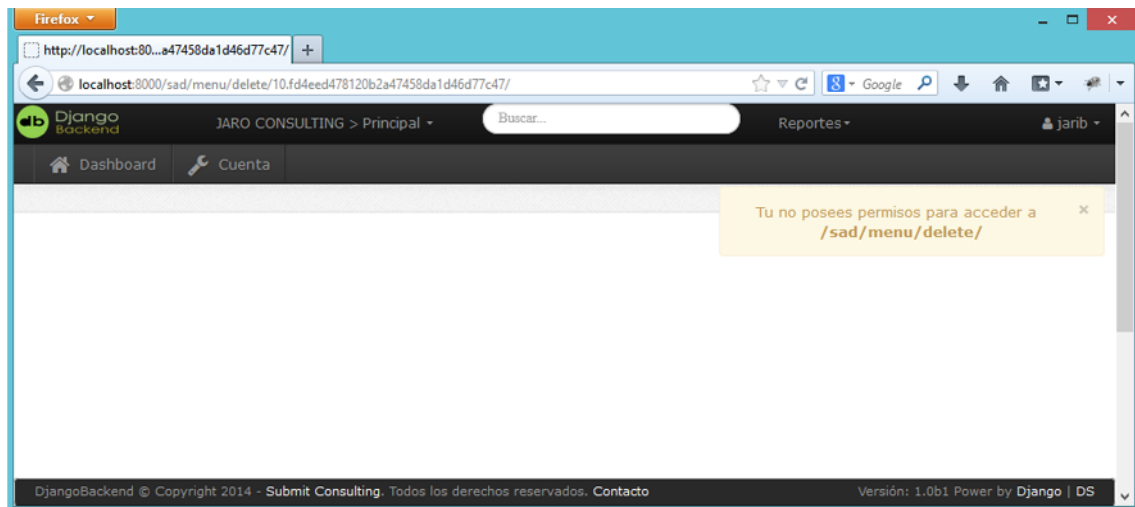
Este usuario tiene acceso al Backend, pero con accesos limitados.



Imagine que desea agregar permisos



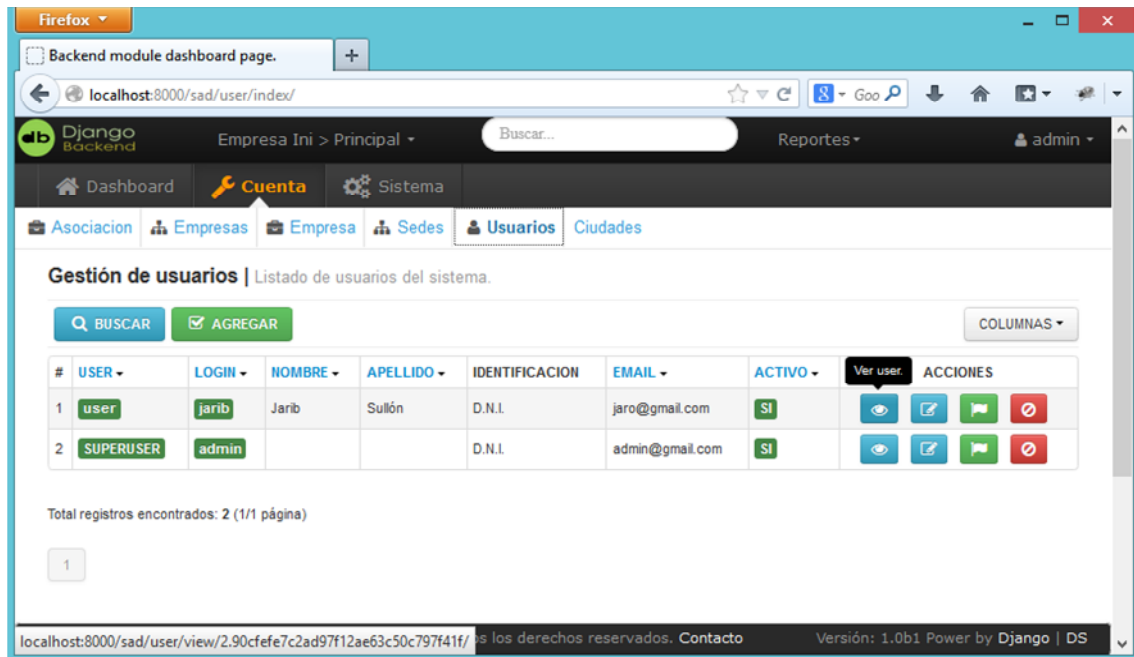
O eliminar un menú



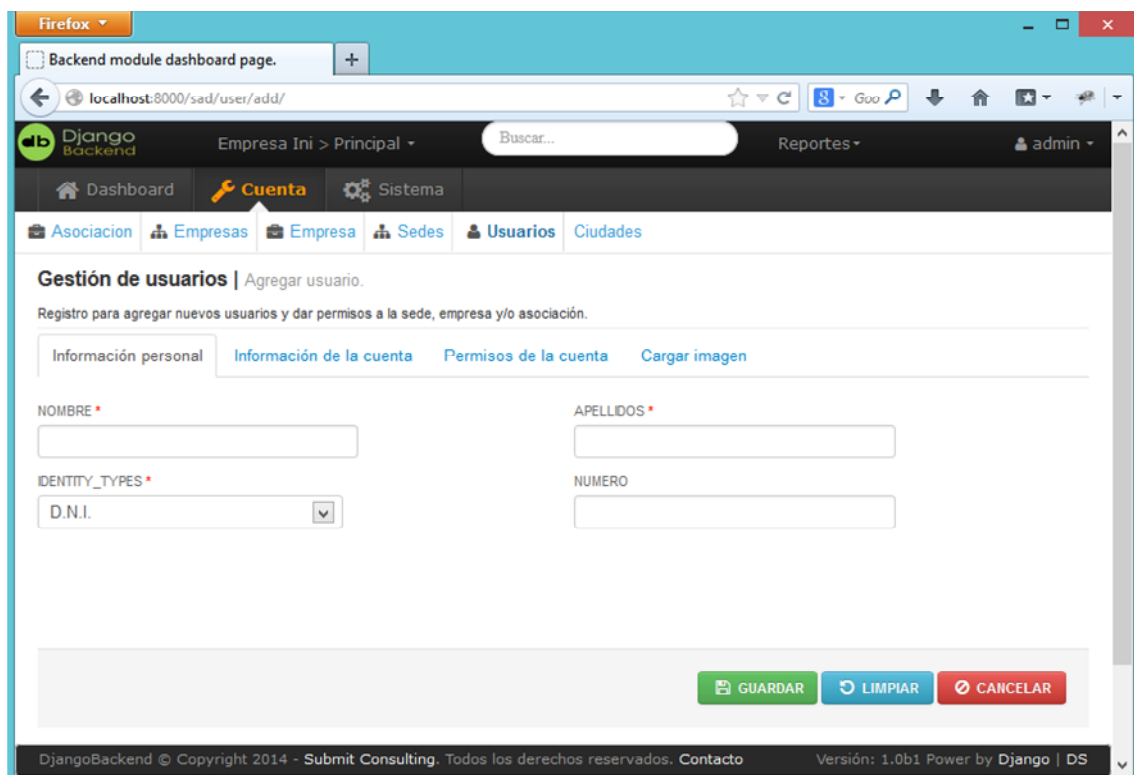
La llave de seguridad /10.fd4eed478120b2a47458da1d46d77c47/ solo le permitirá eliminar datos a las cuales el usuario tiene acceso, si se modifica la clave intentando acceder a otro registro, el sistema el enviará el mensaje de llave inválida.

Usuarios

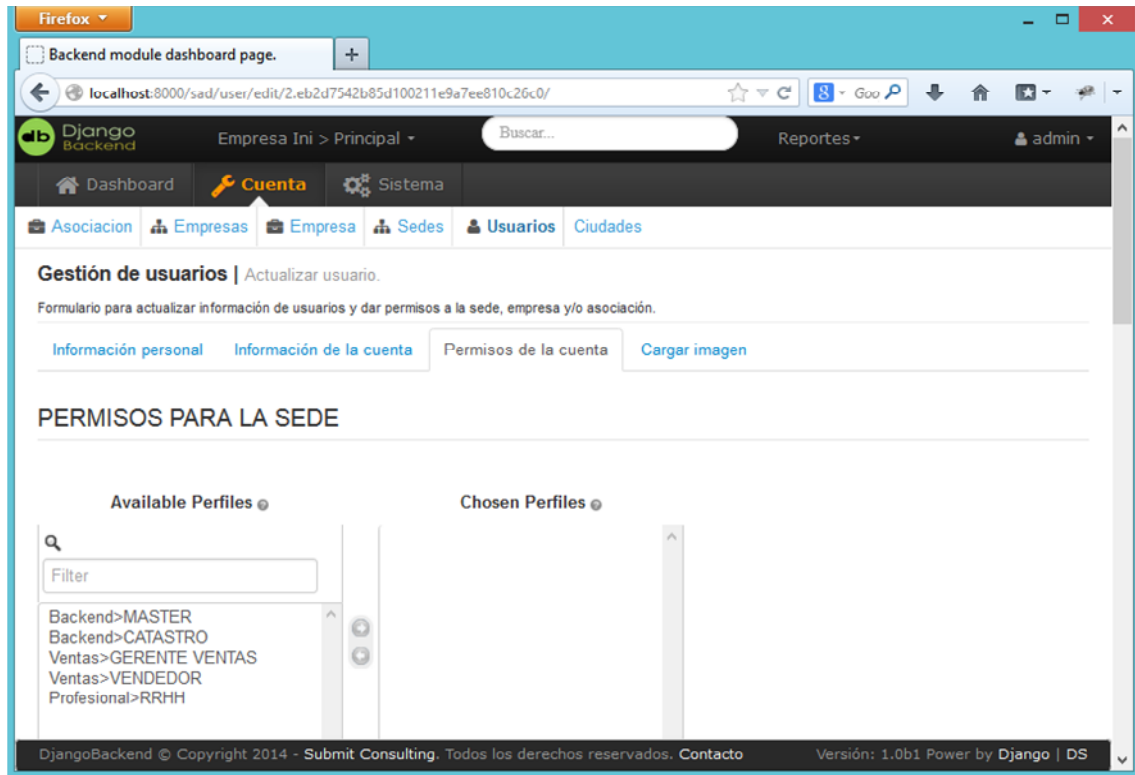
Para crear usuario, en el mismo menú “Cuenta” ir a menú ítem “Usuarios”:



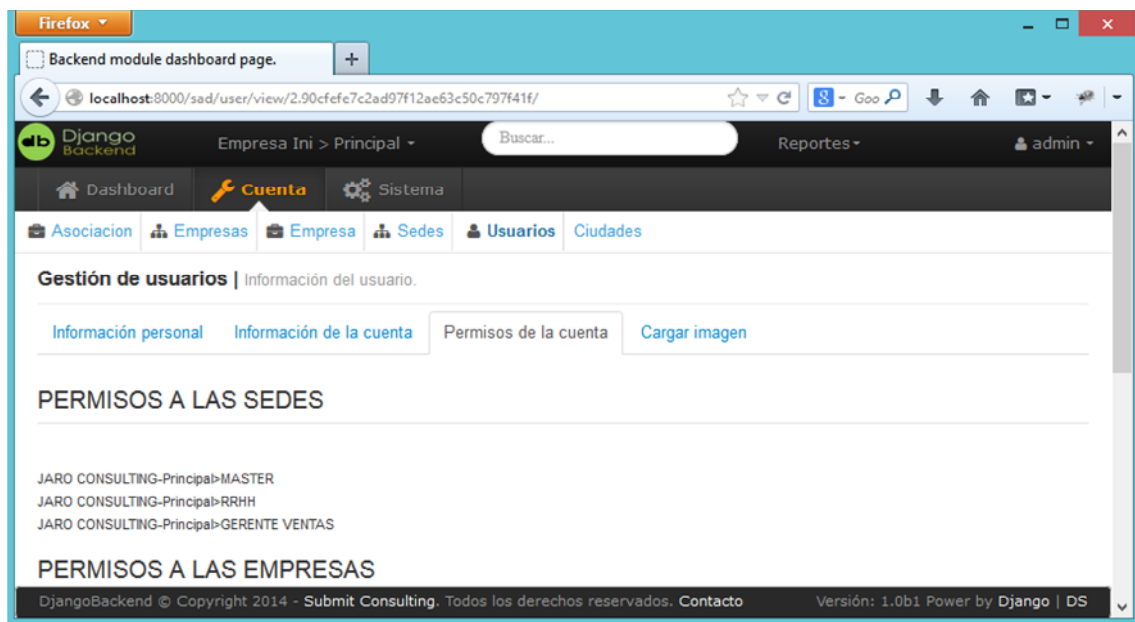
Si desea crear un nuevo usuario clic en el botón “AGREGAR”



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.



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

Celular dentro de PERÚ al RPC989597352/RPM*065067(990720536)

