

Contenido

Introducción	2
¿Qué hay que hacer en la Parte III - Creación de un módulo en Odoo de la Tarea04 de SGE?	3
Estructura de un módulo en Odoo	3
¿Qué es importante a la hora de trabajar con módulos en Odoo?.....	4
¿Cómo empezamos creando un nuevo módulo en Odoo?.....	4
Pasos para instalar un módulo en Odoo	5
Realizando cambios en mi nuevo módulo.....	7
Creación del modelo y de las vistas.....	13

Unidad 04. Inserción, modificación y eliminación de datos en objetos

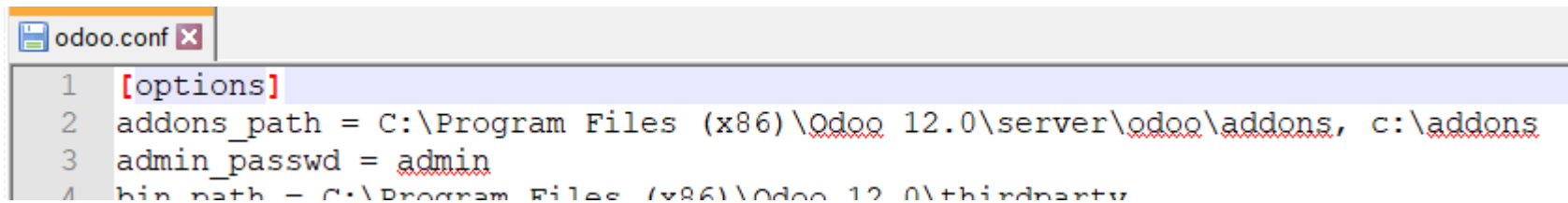
Introducción

Como comentábamos en apartados anteriores, existen aplicaciones ERP que proporcionan lenguajes propietarios, que no son lenguajes estándar y requieren que el programador se forme en ese lenguaje sólo y exclusivamente para manejar el ERP. Este no es el caso de Odoo, que utiliza el lenguaje de programación Python.

Los módulos de Odoo están escritos en lenguaje Python, y utilizan también lenguaje XML para configurar las vistas de la aplicación. Se guardan dentro de la carpeta `addons` que está en el directorio de instalación de la aplicación.

Como la ubicación de este directorio puede cambiar dependiendo de la versión del ERP y de nuestro sistema operativo, lo mejor es comprobar dónde está la carpeta `addons` en nuestra instalación. Para ello, buscamos dentro del archivo `odoo.conf` el parámetro `addons_path`, y la ruta que ahí se indique será donde Odoo busque los módulos por defecto.

Podemos usar una ruta distinta para nuestros módulos propios. Esto es interesante porque si reinstalamos Odoo no perderemos los módulos que hemos creado. Esto se hace editando el archivo `odoo.conf`, y dentro de la opción `addons_path`, metemos una nueva ruta, separada por coma. En el ejemplo siguiente además de la ruta por defecto, hemos metido la carpeta `c:\addons` donde meteremos nuestros módulos propios:



```
odoo.conf x
1 [options]
2 addons_path = C:\Program Files (x86)\Odoo 12.0\server\odoo\addons, c:\addons
3 admin_passwd = admin
4 bin_path = C:\Program Files (x86)\Odoo 12.0\thirdparty
```

¿Qué hay que hacer en la Parte III - Creación de un módulo en Odoo de la Tarea04 de SGE?

Hay que crear un módulo en Odoo, hay una plantilla con un módulo base que se proporciona en la tarea. Un módulo está compuesto por un modelo de datos y unas vistas de ese modelo.

- **Para los datos del módulo**, modificamos el archivo `__manifest__.py`.
- **Para el modelo**, modificamos el archivo `models.py` situado dentro de la carpeta `models`.
- **Para las vistas**, modificamos el archivo `views.xml` situado dentro de la carpeta `views`. Para las vistas utilizaremos lenguaje XML. Este lenguaje se da en el módulo Lenguaje de Marcas del ciclo de DAM, si no sabemos XML lo mejor es ir incorporando los cambios en la vista poco a poco y probando que el módulo funciona, también basándonos en la plantilla que se proporciona en el módulo base.

Estructura de un módulo en Odoo

La estructura de un módulo en Odoo está compuesta por los siguientes archivos:

- `__init__.py`. Es donde se inicia el módulo, es necesario para que la carpeta se trate como un paquete en Python. Aquí indicamos qué modelo queremos vincular a este módulo. El modelo viene a ser la base de datos. En definitiva, contiene los import de cada archivo del módulo que contenga código Python.
- `__manifest__.py`. Aquí establecemos los datos de configuración del módulo.
- carpeta `models`. Contiene el archivo `models.py` con el modelo de datos del módulo. En este archivo creamos la clase definiendo los campos que va a tener. Al crear la clase estamos creando el modelo (una tabla en la base de datos) y también estamos creando el controlador, porque cuando creamos una clase estamos definiendo el comportamiento que tiene.
- carpeta `views`. Contiene los archivos `.xml` que corresponden con las vistas. En este archivo definimos la vista de nuestro módulo, o del objeto que va a crear nuestro módulo. **Para crear este archivo necesitamos tener ciertos conocimientos de XML, o bien coger una vista de otro objeto y a partir de ella crear la del nuevo objeto.**

¿Qué es importante a la hora de trabajar con módulos en Odoo?

Hay que tener cuidado con la sintaxis de las órdenes dentro de los archivos. Si cometemos cualquier error de sintaxis el programa no nos avisará, simplemente no funcionará y podemos tardar mucho en localizar el error. Por eso para desarrollar nuestro módulo utilizaremos la técnica de integrar poco a poco los cambios e ir probando cada cambio realizado que no tenga errores.

Recomendaciones:

- Como la tarea en sí no es muy compleja, lo dificultoso de la misma está en respetar la sintaxis de la plantilla del módulo base.
- Respeta la sintaxis y sangría al programar en Python.
- Primero comprueba que la estructura básica del módulo es reconocida por Odoo en la lista de Aplicaciones. Después, ves incorporando la funcionalidad que se pide en la tarea, haz comprobaciones de funcionamiento en cada cambio importante. Así, si te sale algún error, sabrás dónde tienes el error o te será más fácil localizarlo. Empieza a trabajar con la vista vacía (archivo views.xml sin nada escrito).
- Haz las modificaciones en este orden:
 - Funciona la estructura básica (ver Pasos para instalar un módulo en Odoo).
 - Creamos el modelo (ver Realizando cambios en mi nuevo módulo).
 - Prueba del modelo. No toques la vista hasta que estés segur@ que has creado bien el modelo. Si hay error, revisamos el modelo, si no, continuamos con el siguiente paso. Puedes ver si se ha cargado bien el modelo en Ajustes/Técnico/Estructura de la base de datos/Modelos, ahí aparece el modelo creado con los diferentes campos.
 - Una vez probado el modelo, podemos seguir con los siguientes pasos.
 - En el ERP observarás que se crearon las vistas, acciones de ventana y elementos de menú. Ya tenemos nuestro módulo nuevo en funcionamiento.
 - Para la vista necesitamos tener ciertos conocimientos de XML, o bien coger una vista de otro objeto y a partir de ella crear la del nuevo objeto. Tienes enlaces en el foro sobre cómo crear vistas.

¿Cómo empezamos creando un nuevo módulo en Odoo?

Los módulos en Odoo los creamos a partir de una estructura base. Esta estructura base es [my_module.zip](#). Si la estructura en la que nos basamos no es la correcta o contiene errores algún archivo, Odoo no reconocerá el módulo.

Para modificar la estructura base del módulo lo haremos abriendo los archivos correspondientes y modificando lo que nos indique la tarea. Podemos utilizar un bloc de notas como por ejemplo Notepad++ o instalar un entorno de desarrollo. En principio no es necesario nada más que el bloc de notas, pero quien quiera o ya lo tenga instalado puede usar un IDE como por ejemplo Visual Studio Code. También se puede usar PyCharm de JetBrains que es un IDE para Python, ya que te va chivando todos estos errores. Puedes descargarla la versión Community desde su página oficial.

El archivo .zip con la estructura de un módulo en Odoo, está compuesto por una carpeta y los archivos vacíos de un módulo básico. Si descargamos el .zip y lo colocamos dentro de la carpeta addons, Odoo debe reconocer el módulo así:

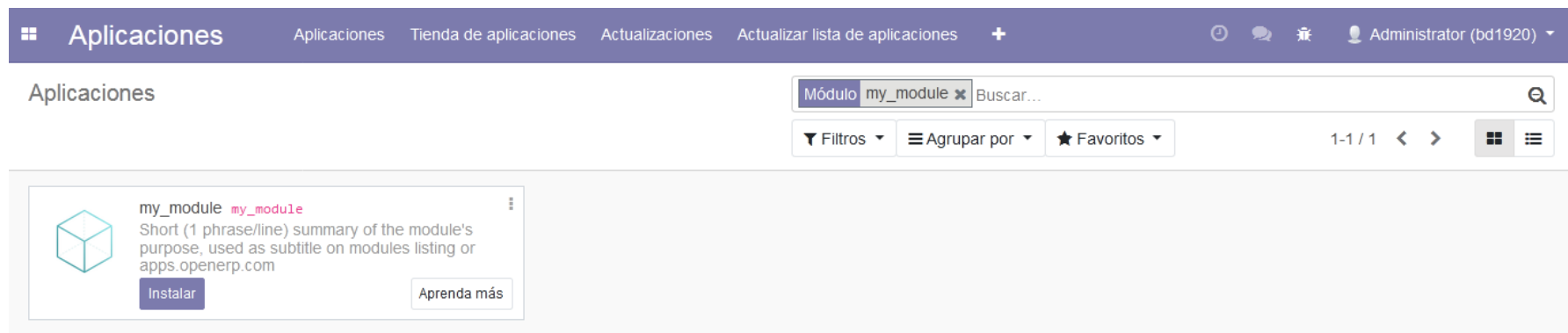


Imagen 01. Módulo cargado en Odoo

Pasos para instalar un módulo en Odoo

Los pasos para instalar un módulo básico son:

- Descargamos la estructura de un módulo vacío de la tarea.

- Descomprimos el .zip descargado dentro de la carpeta addons que aparece en `odoo.conf`. Cuidado, revisa bien cuál es el directorio addons en tu instalación.

- Dentro de Odoo, hacemos lo siguiente:

1. Activar modo desarrollador
2. **Actualiza lista de aplicaciones**, en el menú superior horizontal
3. Vamos a **Aplicaciones**, eliminamos filtro de aplicaciones, buscamos nuestro modulo introduciendo el texto `my_module`
4. El módulo debe aparecernos en la lista de aplicaciones (ver Imagen 01 anterior)
5. Instalamos el módulo para comprobar que la estructura básica no contiene errores:

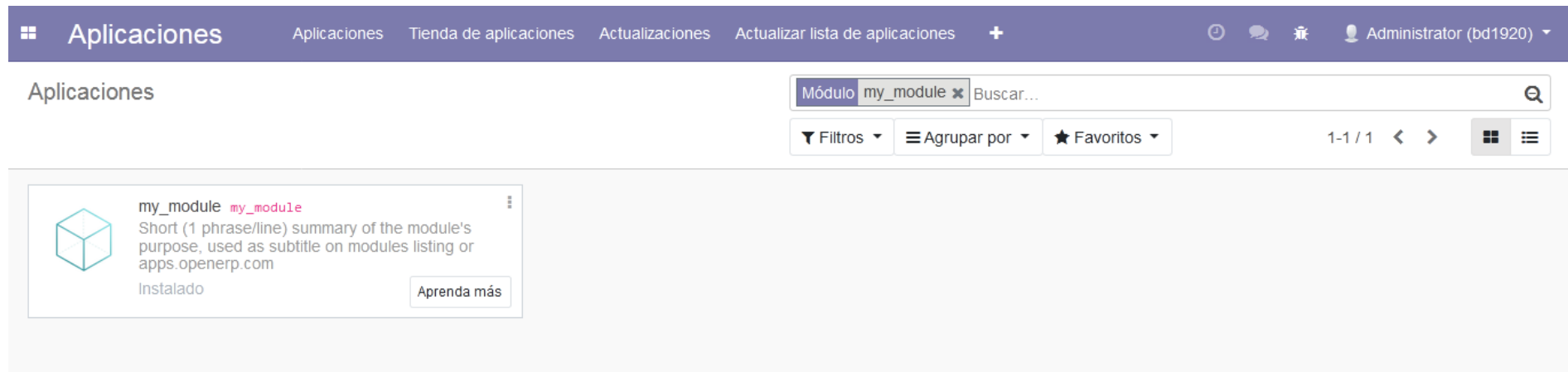


Imagen 02. Módulo instalado.

Si todos estos pasos te han funcionado, ya puedes empezar a programar tu nuevo módulo en Odoo.

Importante

En nuestros primeros pasos podemos instalar el módulo con la vista vacía para que no interfiera en la instalación (archivo **views.xml** sin nada).

Una vez que comprobemos que el módulo se instala sin dar ningún error, por tanto el modelo está bien (archivo **models.py**), es decir que no hay ningún problema de sintaxis en el archivo y que la estructura del módulo está correcta, entonces sí nos metemos a crear la vista.

De esta forma, si a partir de este punto tuviéramos problemas con la instalación del módulo, ya sabemos que sí o sí se trata de errores en la vista, y la solución se reduciría a revisar la sintaxis en el lenguaje XML de la vista (archivo **views.xml**).

Incluso si queremos, para probar el modelo, podemos crear un menú a mano a través de la interfaz de Odoo y así conseguimos comprobar que el modelo funciona y permite la introducción de datos. Usamos la opción **Convertirse en superusuario** para visualizar el menú creado.

Realizando cambios en mi nuevo módulo

Ahora que ya hemos probado que la estructura básica del módulo se instala, es el momento de realizar cambios y ver cómo estos cambios son aceptados/ mostrados en Odoo.

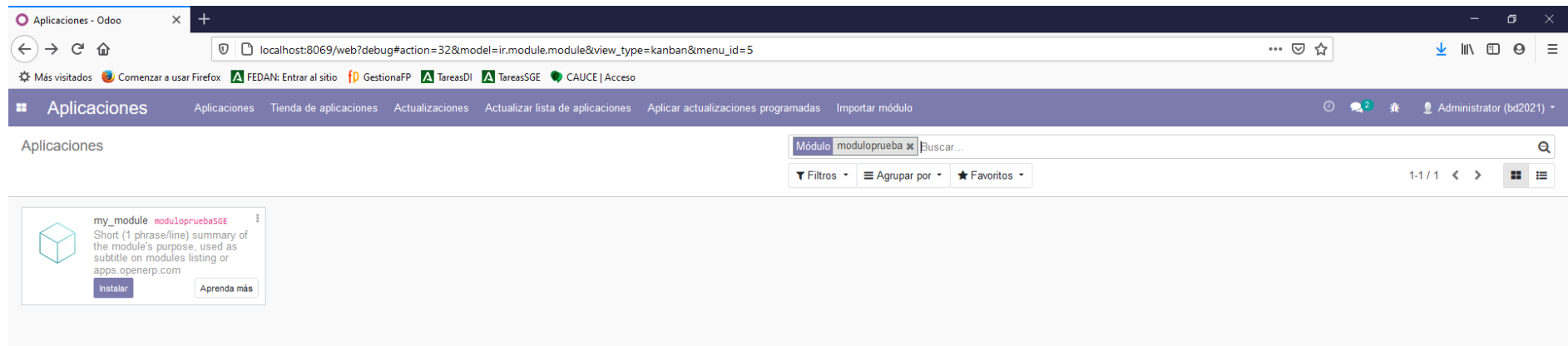
Realizamos cambios en los archivos dentro de la carpeta de nuestro módulo, por ejemplo podemos cambiar el nombre del módulo o el parámetro **summary** dentro del archivo **manifest**. Ahora tenemos que comprobar que esos cambios se reflejan en el módulo dentro de Odoo.

A veces ha ocurrido que Odoo no detecta los cambios producidos en el módulo. Por ejemplo, no coge los cambios que hayáis hecho en el archivo **manifest.py**, aunque lo hayáis eliminado con **Acción/Suprimir**. En esos casos, tal vez sea necesario reiniciar el servidor de Odoo, pero normalmente este paso de reiniciar el servidor no será necesario a menos que hagáis cambios en el archivo de configuración **odoo.conf**.

Comentaros que tenéis que eliminar el módulo en **Odoo (Acción/Suprimir)**, **acto seguido le dais a Actualizar lista de aplicaciones, entonces el módulo vuelve a salir en Aplicaciones**, pero en este caso ya sale con los datos últimos que haya en el manifest.

También tened cuidado si realmente estáis modificando el **manifest** de la carpeta del módulo situada en el directorio de addons, o estáis modificando otro **manifest** (de una copia del módulo que tengáis por ahí en el disco duro y por eso no os coge los cambios).

Para ello, lo mejor es que eliminemos el módulo antiguo. Buscamos el módulo en la lista de aplicaciones:



Elegimos la vista lista:

The screenshot shows the Odoo 'Aplicaciones' (Applications) interface. The top navigation bar includes links for 'Aplicaciones', 'Tienda de aplicaciones', 'Actualizaciones', 'Actualizar lista de aplicaciones', 'Aplicar actualizaciones programadas', and 'Importar módulo'. The main content area displays a table of applications. The table has columns for 'Nombre de módulo', 'Nombre técnico', 'Autor', 'Sitio web', 'Última versión', and 'Estado'. A single application is listed: 'my_module' with technical name 'modulopruebaSGE', author 'My Company', website 'http://www.yourcompany.com', version '12.0.1.0', and status 'No instalado'. Above the table, there is a search bar and filters.

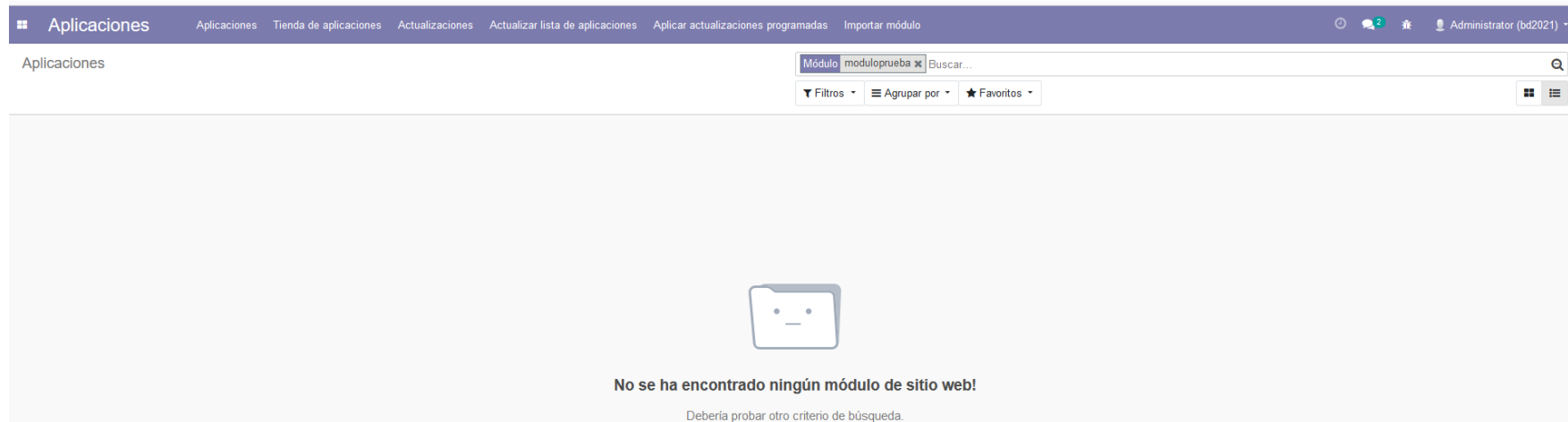
Nombre de módulo	Nombre técnico	Autor	Sitio web	Última versión	Estado
my_module	modulopruebaSGE	My Company	http://www.yourcompany.com	12.0.1.0	No instalado

Seleccionamos el módulo y lo suprimimos desde el menú Acción:

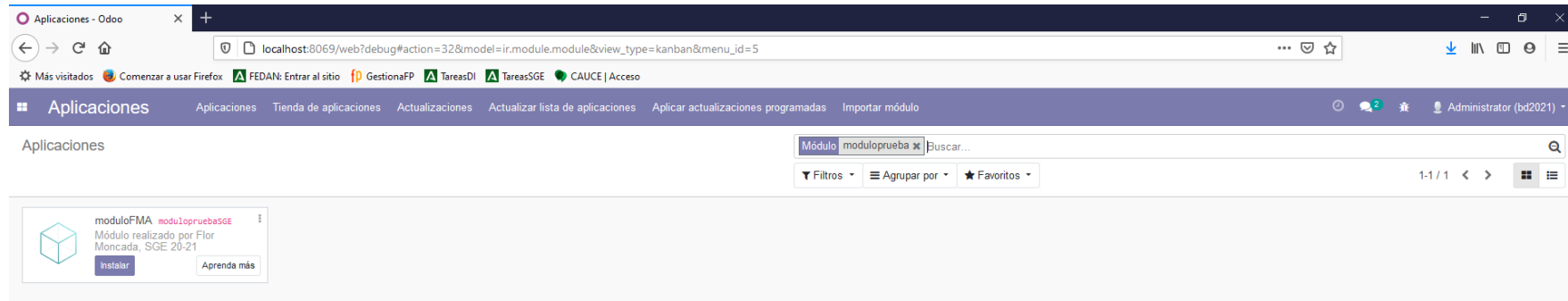
This screenshot is similar to the previous one, but the 'Acción' (Action) menu is open for the selected 'my_module' row. The menu options are 'Exportar', 'Suprimir' (Delete), and 'Instalar Módulos'. The 'Suprimir' option is highlighted.

Nombre de módulo	Nombre técnico	Autor	Sitio web	Última versión	Estado
my_module	modulopruebaSGE	My Company	http://www.yourcompany.com	12.0.1.0	No instalado

Eliminamos el módulo y ya no aparece en Odoo. Hemos eliminado el módulo (ya no aparece en Odoo aunque sigue estando dentro de la carpeta addons) y hemos eliminado también el modelo (ya no aparece en **Tecnico/Estructura de la base de datos/Modelo**).



Le damos a **Actualizar lista de Aplicaciones** del menú horizontal superior y comprobamos que Odoo ha aceptado las modificaciones realizadas en los archivos del módulo:



También podemos realizar estos pasos de eliminar el módulo para intentar solucionar un problema de instalación. Para ello, comentaremos las líneas dentro del archivo models.py que nos estén dando problemas (carácter # al inicio de la línea). Luego le damos a **Actualizar Lista de Aplicaciones** y volvemos a instalar el módulo. Seguidamente comprobamos si nuestro módulo ahora sí se instala correctamente. Lo dejamos así con la opción más sencilla y una vez que comprobemos que no da problemas de instalación, iremos incorporando los cambios en nuestro archivo models.py.

The screenshot shows the Odoo application manager interface. The browser address bar indicates the URL: `localhost:8069/web?debug#id=3318&action=32&model=ir.module.module&view_type=form&menu_id=5`. The top navigation bar includes links for 'Aplicaciones', 'Tienda de aplicaciones', 'Actualizaciones', 'Actualizar lista de aplicaciones', 'Aplicar actualizaciones programadas', and 'Importar módulo'. The user is logged in as 'Administrator (bd2021)'.

The main content area displays the details for the 'moduloFMA' application, developed by 'Por My Company'. The application is currently installed and can be updated. The 'Información' tab is active, showing the following details:

Información	Datos técnicos
Sitio web	http://www.yourcompany.com
Categoría	Sin categoría
Resumen	Módulo realizado por Fior Moncada, SGE 20-21
Nombre técnico	modulopruebaSGE
Licencia	LGPL versión 3
Última versión	12.0.0.1

Below the table, there is a section for the 'Long description of module's purpose'.

Creación del modelo y de las vistas

El modelo se corresponde con las tablas en la base de datos de PostgreSQL, van declaradas en el archivo models.py

Cada objeto en Odoo corresponde con una clase en Python. Vamos a crear dos objetos: Categoria y Artículo. Las clases las nombraremos utilizando el nombre del módulo seguido del nombre del objeto, por ejemplo para un módulo llamado Tienda las nombraremos así:

```
class Tienda_categoria(models.Model):
```

```
    _name = "tienda.categoria"
```

```
    # etiqueta que sale en la vista
```

```
    name = fields.Char('Nombre',required=True, help='Introduce el nombre de la categoría')
```

```
(...)
```

Creamos las nuevas clases en Python dentro del archivo models.py.

Para crear el modelo tendremos que saber trabajar con clases y objetos en Python, o bien coger un modelo de otro módulo y partir de él para crear el nuestro.

Para la creación de las vistas tenemos que tener en cuenta que el lenguaje utilizado es XML. **Para crear este archivo necesitamos tener ciertos conocimientos de XML, o bien coger una vista de otro objeto y a partir de ella crear la del nuevo objeto.**