

Práctica Primeros Módulos Odoo

Preparación

- Necesitamos un directorio donde guardar los addons (configurado en la opción `addons_path` del `odoo.conf`). Deberías de tenerlo de cuando se instalaron los módulos de OCA.
- Compartir el directorio por samba para poder acceder a él desde tu equipo.
- En nuestro equipo tener Visual Studio, Atom o cualquier editor de código que nos guste. Comprobar que puedo acceder al directorio de la máquina virtual (compartido por samba) donde están los addons.

Nuestro primer modulo: “Hola mundo”

1. Añade el módulo `MiMundo_tunombre` (di te llamas Serafín, pues `mimundo_serafin`) siguiendo las siguientes instrucciones.

Para este primer “Hola mundo”, crearemos una carpeta “Ejemplo01-HolaMundo” dentro del directorio que hayamos configurado para colocar nuestros módulos.

Esta carpeta contendrá dos ficheros:

a) Fichero “__init__.py”:

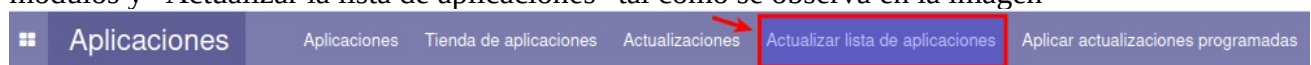
Este fichero estará vacío.

b) Fichero “__manifest__.py”:

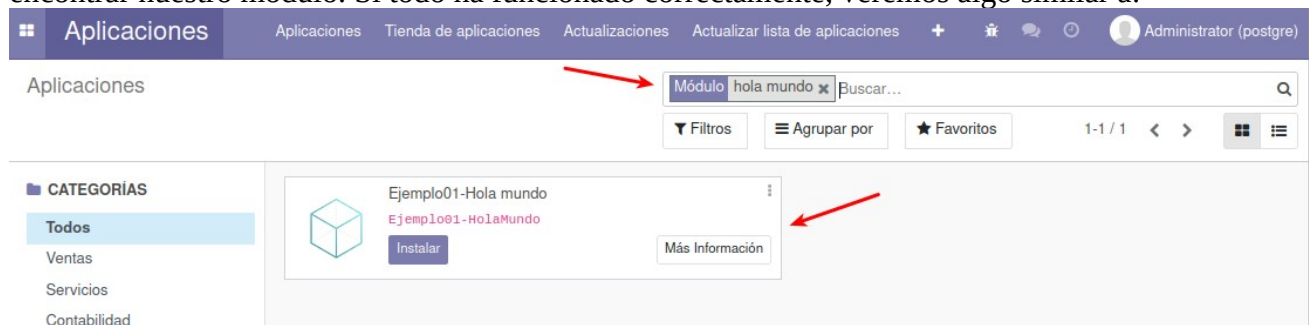
Este fichero contendrá el siguiente código:

```
# -*- coding: utf-8 -*-
{'name': 'Ejemplo01-Hola mundo'}
```

Una vez creada la estructura, con el “Modo Desarrollador” activado, podréis ir al listado de módulos y “Actualizar la lista de aplicaciones” tal como se observa en la imagen



Tras ello, eliminando los filtros de búsqueda por defecto y buscando “hola mundo”, podremos encontrar nuestro módulo. Si todo ha funcionado correctamente, veremos algo similar a:



2. Ahora podremos instalar nuestro módulo para probarlo (aunque este ejemplo no hace nada). Desinstálalo.

3. En la página 74 del libro Odoo Essentials tienes información de algunos de los campos que puedes rellenar en el fichero `__manifest__.py`. Añade la siguiente información:
 - a) Nombre del módulo
 - b) Resumen
 - c) Autor
 - d) Licencia (elige una con sentido)
 - e) Página web
 - f) Configura el flag aplicación para que salga el módulo cuando tienes el filtro Apps
 - g) Añade el módulo a la categoría Pruebas/Practica
 - h) Añade un icono
4. Crea un nuevo módulo que se llame `luna_tunombre`,
 - a) Configúralo de manera similar al módulo Mundo.
 - b) En las dependencias configura para que dependa del módulo Mundo.Con los dos módulos desinstalados instala el módulo Luna. Al depender de Mundo se debe de instalar Mundo automáticamente.

Comando scaffold

Los ficheros necesarios que aparecen en un módulo de Odoo siguen siempre la misma estructura

```
library_app/  
├── __init__.py  
├── __manifest__.py  
├── controllers  
│   ├── __init__.py  
│   └── controllers.py  
├── demo  
│   └── demo.xml  
├── models  
│   ├── __init__.py  
│   └── models.py  
├── security  
│   └── ir.model.access.csv  
└── views  
    ├── templates.xml  
    └── views.xml
```

Vamos a realizar una breve explicación de cada uno los ficheros generados:

- “**__manifest__.py**”: es el manifiesto del módulo. Incluye información como el título, descripción, así como ficheros a cargar.
- “**views**”: describe las vistas de nuestro módulo (formulario, árbol ,menús, etc.).
- “**demo**”: incluye datos “demo” para el ejemplo propuesto de modelo.
- “**security**”: Aquí se guardarán los ficheros de seguridad.
- “**controllers**”: aquí tendremos “métodos” para añadir control sobre los datos más allá de la declaración del modelo.

Con el comando `scaffold` creamos la estructura básica de ficheros para la implementación de un módulo. Por ejemplo, si quiero crear el módulo `lista_tareas` y el directorio donde tengo los addons es `/mnt/extra-addons` el comando sería:

```
odoo scaffold lista_tareas /mnt/extra-addons/
```

**Si estás utilizando docker composer deberás ejecutar el comando
utilizando docker exec**

```
docker exec contenedor comando
```

1. Crea un módulo llamado `hola_sol` utilizando el comando `scaffold`. Modifica algunos parámetros del fichero `manifest` e instálalo.