



Actividad 5.2: Proyecto: Realiza una Aplicación desde cero en Odoo 15.2alpha1 (Community Edition)

Grupo: DA2D1E
Alumnos: Ronny Collaguazo y Alejandro Díez

Índice

Enunciado.....	3
1. Conexión a Odoo:.....	5
2. Creación del módulo Charlas Claras, sincronización y edición desde Atom:	8
3. Instalación del módulo y creación de los modelos:.....	20
4. Creación de las vistas, acciones de ventana y elementos de menú del módulo Charlas Claras	27
4.1 Vistas.....	27
4.2 Acciones de ventana	41
4.3 Elementos de menú	43
5. Informes en pdf o web	47
6. Crear 2 grupos de usuarios: administradores y usuarios	51
7. Ejemplo de funcionamiento.....	57

Enunciado

Esta tarea consiste en realizar un módulo desde cero. Para ello debemos de disponer del escenario planteado en la práctica anterior. El proyecto trata de realizar una aplicación para una empresa, para gestionar una base de datos escogida por el alumno o similar. El proyecto debe contener:

- Una nueva aplicación. Con sus ficheros y archivos básicos.
- Al menos 2 nuevos modelos asociados a la aplicación.
- Debe existir una relación entre los modelos.
- Debemos disponer en el modelo de campos básicos:
 - Boolean
 - Char
 - Float
 - Integer
 - Otros
- Debemos disponer en el modelo de campos avanzados
 - Monetary
 - Date
 - Otros
- Debemos disponer en el modelo campos de relacionales
 - Many2one ó/y
 - Many2many ó/y
 - One2many
- Debemos disponer de al menos un campo calculado
- Al menos un campo debe ser seleccionado entre las opciones que aparezcan en una lista desplegable (opcional)
- Dentro de los atributos de los campos debemos tener en cuenta que se deben incluir:
 - String
 - default
 - **required**
 - readonly
 - compute
 - otros
- Debemos crear las vistas para los nuevos modelos. Las obligatorias son:
 - Vista árbol
 - Vista kanban
 - Vista formulario
- Realizar una vista de búsqueda para lo que sea necesario (opcional)
- **Es obligatorio que las vistas tengan una buena disposición, para que sean visualmente atractivas.**
- Modificar, ampliar o crear una vista gracias a la herencia. (opcional)
- Debemos de disponer de los menús principales y de submenús correspondientes en cada modelo.
- Al menos un sub-menú por modulo.
- Las acciones de ventana correspondientes.
- Debemos poder crear e imprimir un informe, sea pdf o visto en web.
- Debemos proporcionar un módulo cuya visión sea lo más profesional posible. Incluir:
 - Icono para aplicación.
 - Descripción adecuada.
 - **Nomenclatura correcta siguiendo buenas prácticas.**
- Crea dos grupos de usuarios.
 - Administradores (propietarios) - Lo que sea
 - Usuarios (clientes) - Lo que sea
- Debemos ver en algún punto (sea en vista, en menús, o donde se desee por el alumno) la diferencia entre ambos usuarios.

Se revisará por parte del profesor el progreso y finalización el martes 22 de febrero.

El Proyecto no debe de eliminarse. Y se entregará tanto el informe en pdf del proyecto realizado como el propio proyecto en un .zip o .rar.

Todo el proceso de creación de la nueva aplicación se realizará desarrollando los ficheros correspondientes. Se debe dejar constancia con imágenes de todo el proceso. Se debe entrar un documento pdf con el nombre GE_Proyecto_nombre_apellidos

1. Conexión a Odoo:

Primero iniciamos la máquina virtual con Odoo Server

User: alejandro

Password: 1234

Para averiguar la ip de la MV utilizamos el comando:

`ifconfig`

```
alejandro@alejandros-server:~$ ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.1.53 netmask 255.255.255.0 broadcast 192.168.1.255
        inet6 fe80::a00:27ff:fe47:6d5 prefixlen 64 scopeid 0x20<link>
            ether 08:00:27:47:06:d5 txqueuelen 1000 (Ethernet)
            RX packets 16038 bytes 14381410 (14.3 MB)
            RX errors 0 dropped 45 overruns 0 frame 0
            TX packets 11071 bytes 9871973 (9.8 MB)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
        loop txqueuelen 1000 (Local Loopback)
        RX packets 2593 bytes 1203777 (1.2 MB)
        RX errors 0 dropped 0 overruns 0 frame 0
        TX packets 2593 bytes 1203777 (1.2 MB)
        TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

Nos conectamos desde el cmd de Windows con el comando:

`ssh alejandro@192.168.1.53`

```
C:\Users\W10Alex>ssh alejandro@192.168.1.53
alejandro@192.168.1.53's password:
Welcome to Ubuntu 20.04.3 LTS (GNU/Linux 5.4.0-91-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

 System information as of mié 02 feb 2022 17:13:35 UTC

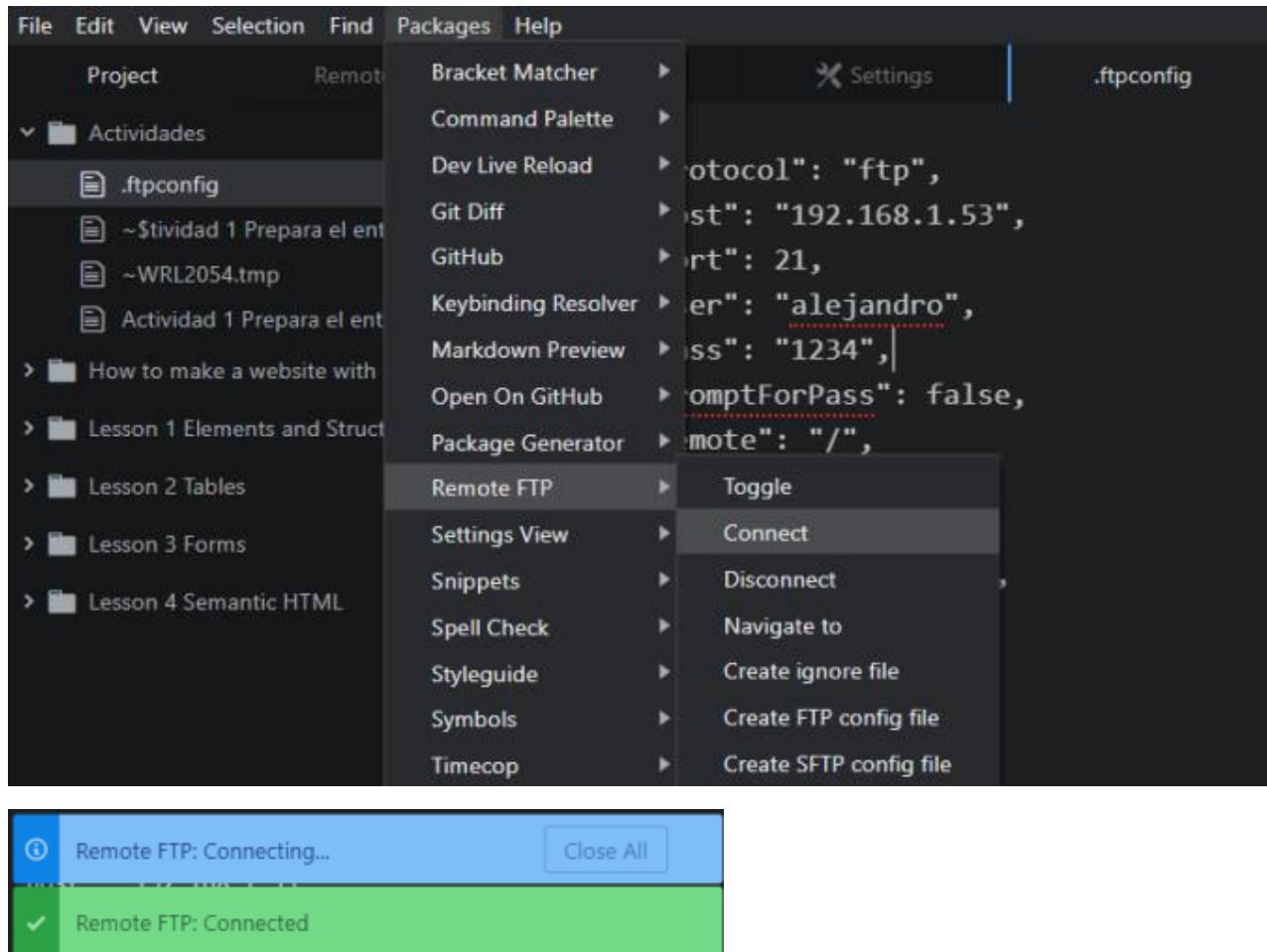
 System load:  0.45           Processes:          134
 Usage of /:   35.3% of 18.57GB  Users logged in:     1
 Memory usage: 11%           IPv4 address for enp0s3: 192.168.1.53
 Swap usage:   0%

 * Super-optimized for small spaces - read how we shrank the memory
   footprint of MicroK8s to make it the smallest full K8s around.
```

Nota: La IP va cambiando a **192.168.1.54/.55/.etc** al restaurar instantáneas del Ubuntu Server para intentar que funcionen los comandos de conexión a Odoo a través de la conexión ssh o por otras causas.

Abrimos Atom y realizamos la conexión FTP que habíamos configurado en la práctica anterior cambiando la IP cuando sea necesario:

Atom/Packages/Remote FTP/Connect

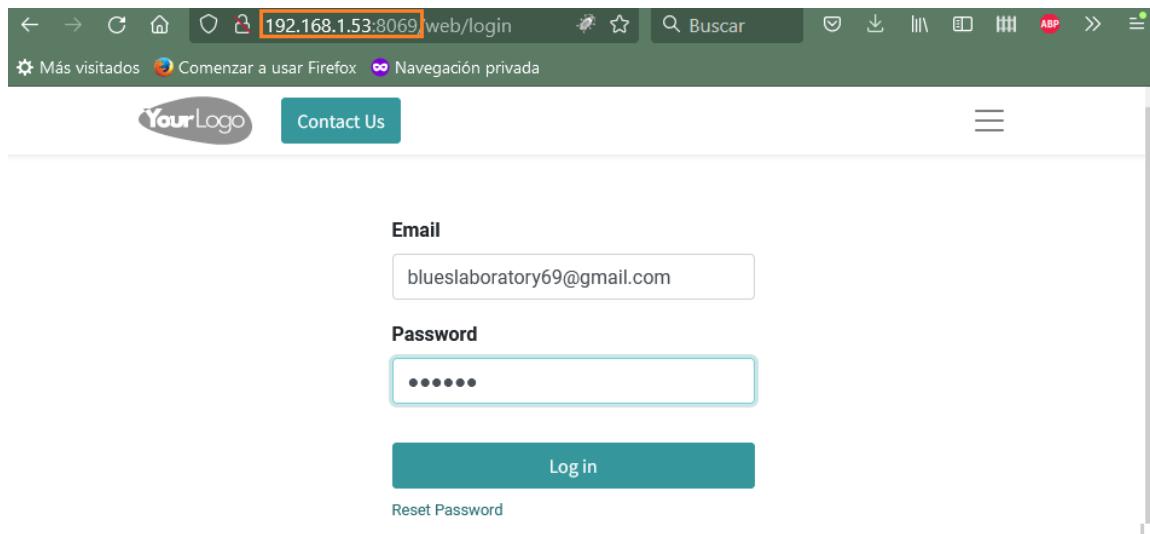


En ocasiones tendremos que Iniciar Odoo desde la máquina virtual, ya que no reconoce algunos comandos:

```
sudo su odoo -s /bin/bash
cd /odoo/odoo-server
./odoo-bin -c /etc/odoo-server.conf
```

```
alejandro@alejandros-server:~$ sudo su odoo -s /bin/bash
[sudo] password for alejandro:
odoo@alejandros-server:/home/alejandro$ cd /odoo/odoo-server
odoo@alejandros-server:/odoo/odoo-server$ ./odoo-bin -c /etc/odoo-server.conf
```

Odoo te marca por defecto el puerto 8069, lo utilizamos para conectarnos a nuestro Odoo:



Email: blueslaboratory69@gmail.com

Password: 123456

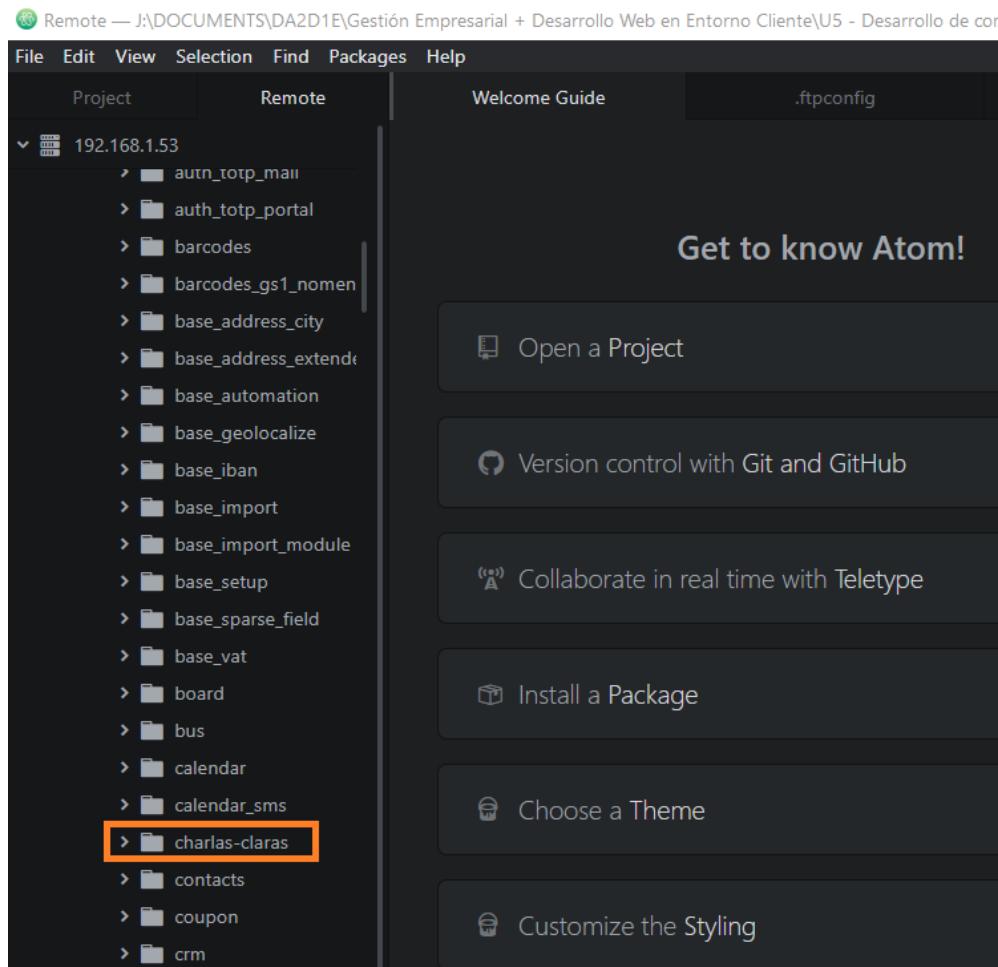
2. Creación del módulo Charlas Claras, sincronización y edición desde Atom:

Generamos el módulo de forma automática a través del comando scaffold, que genera los archivos y carpetas mínimas necesarias para que el módulo sea reconocido.

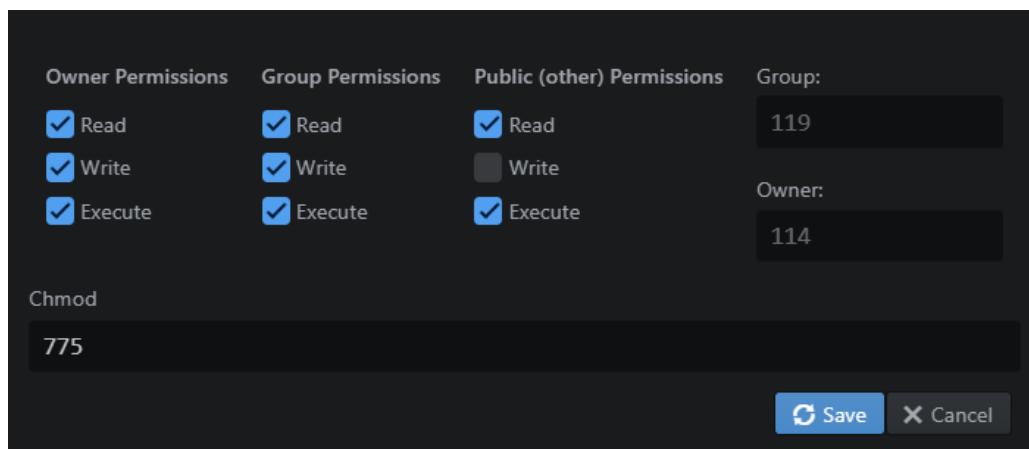
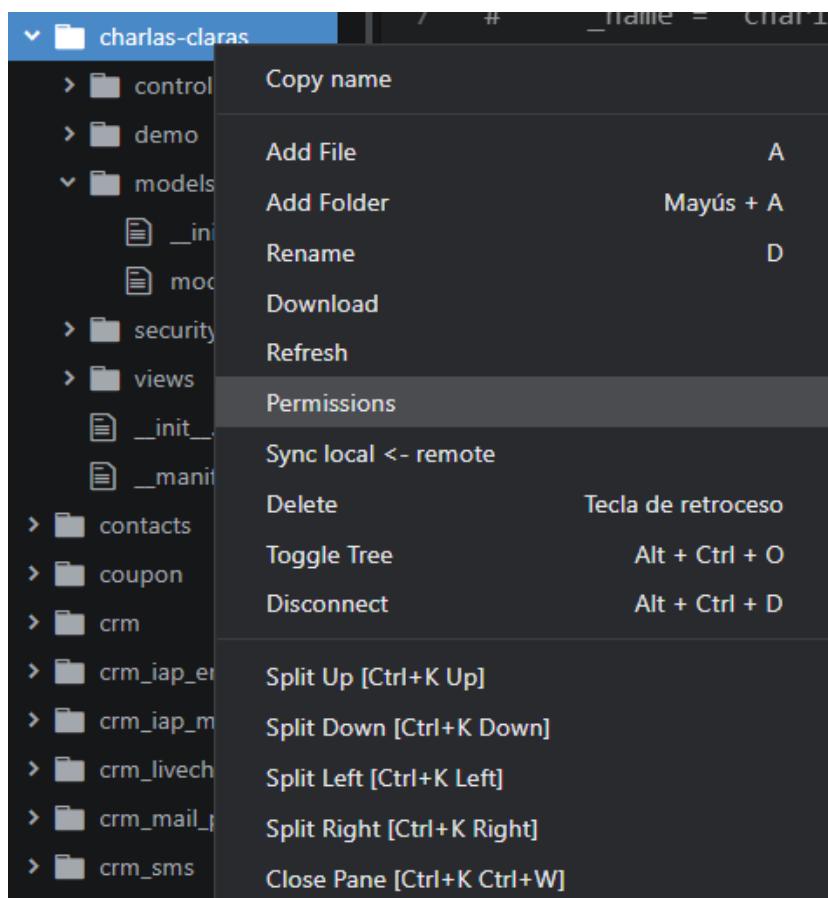
```
./odoo-bin scaffold charlas-claras addons/
```

```
odoo@alejandros-server:/odoo/odoo-server$ ./odoo-bin scaffold charlas-claras addons/
odoo@alejandros-server:/odoo/odoo-server$
```

Y ya nos debería aparecer en el Atom en la carpeta de addons de la pestaña Remote el módulo: **charlas-claras**



Para poder editar tanto en remoto como en local los modelos y demás archivos de **charlas-claras** desde el IDE Atom, hay que modificar los permisos:

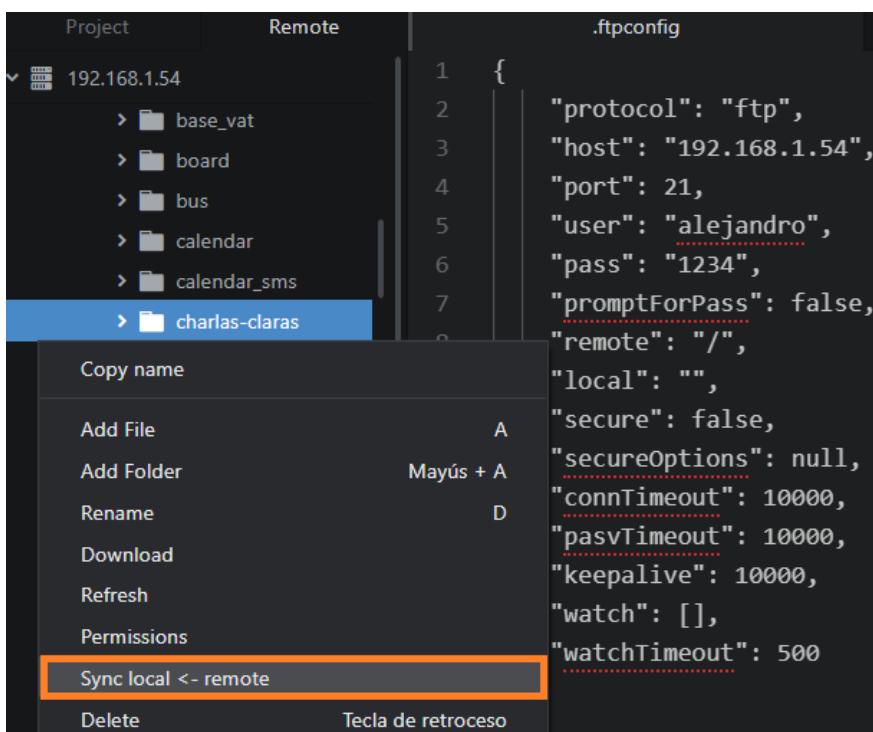


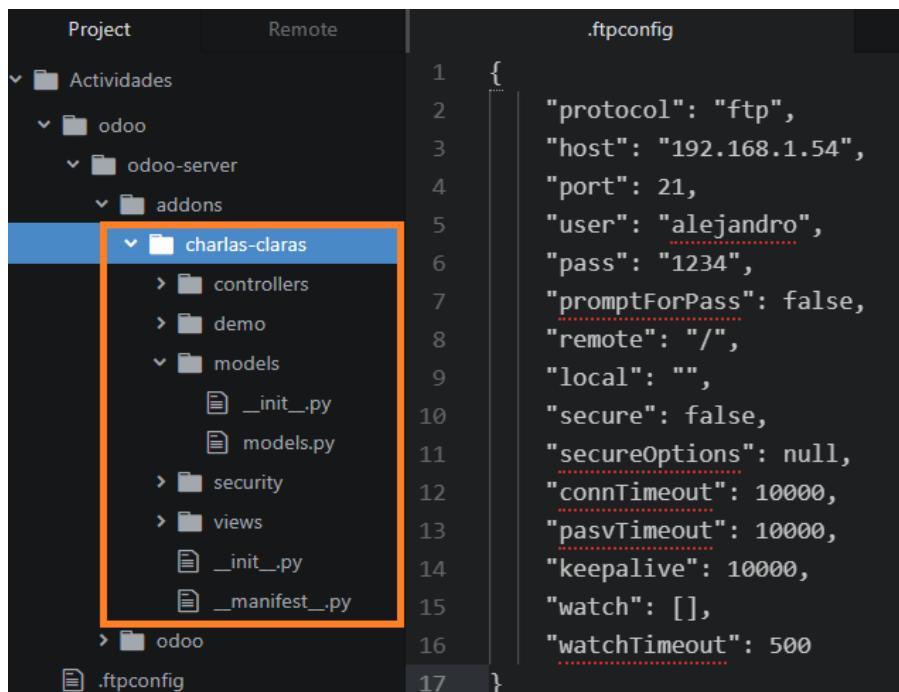
Para ello, hay que ejecutar el siguiente comando:

```
alejandro@alejandros-server:/$ sudo chmod -R 777 odoo/
```

```
alejandro@alejandros-server:/$ sudo chmod -R 777 odoo/
[sudo] password for alejandro:
alejandro@alejandros-server:/$ ls -l
total 2097228
lrwxrwxrwx  1 root root      7 ago 24 08:41 bin -> /usr/bin
drwxr-xr-x  4 root root    4096 dic 14 18:00 boot
drwxr-xr-x  2 root root    4096 dic 14 16:47 cdrom
drwxr-xr-x 20 root root   4120 feb  8 19:30 dev
drwxr-xr-x 108 root root  4096 feb  8 19:06 etc
drwxr-xr-x  3 root root  4096 dic 14 17:34 home
lrwxrwxrwx  1 root root      7 ago 24 08:41 lib -> /usr/lib
lrwxrwxrwx  1 root root      9 ago 24 08:41 lib32 -> /usr/lib32
lrwxrwxrwx  1 root root      9 ago 24 08:41 lib64 -> /usr/lib64
lrwxrwxrwx  1 root root     10 ago 24 08:41 libx32 -> /usr/libx32
drwx-----  2 root root  16384 dic 14 16:44 lost+found
drwxr-xr-x  2 root root   4096 ago 24 08:42 media
drwxr-xr-x  2 root root   4096 ago 24 08:42 mnt
drwxrwxrwx  3 root root  4096 dic 15 14:27 odoo
drwxr-xr-x  3 root root   4096 dic 14 19:36 opt
dr-xr-xr-x 182 root root      0 feb  8 19:29 proc
drwx-----  8 root root  4096 feb  8 19:05 root
drwxr-xr-x 30 root root   880 feb  8 19:30 run
lrwxrwxrwx  1 root root      8 ago 24 08:41 sbin -> /usr/sbin
drwxr-xr-x  7 root root  4096 dic 15 11:15 snap
drwxr-xr-x  3 root root  4096 feb  8 19:04 srv
-rw-----  1 root root 2147483648 dic 14 16:57 swap.img
dr-xr-xr-x 13 root root      0 feb  8 19:29 sys
drwxrwxrwt 11 root root  4096 feb  8 19:30 tmp
drwxr-xr-x 15 root root  4096 ago 24 08:46 usr
drwxr-xr-x 13 root root  4096 ago 24 08:47 var
alejandro@alejandros-server:/$
```

Sincronizamos el módulo **charlas-claras** de remoto->local, para tener una copia. Si tenemos algún problema desde remoto o local y rompemos algo, sincronizamos del lado que corresponda o hacemos copias de archivos o restauramos instantáneas de la máquina virtual.





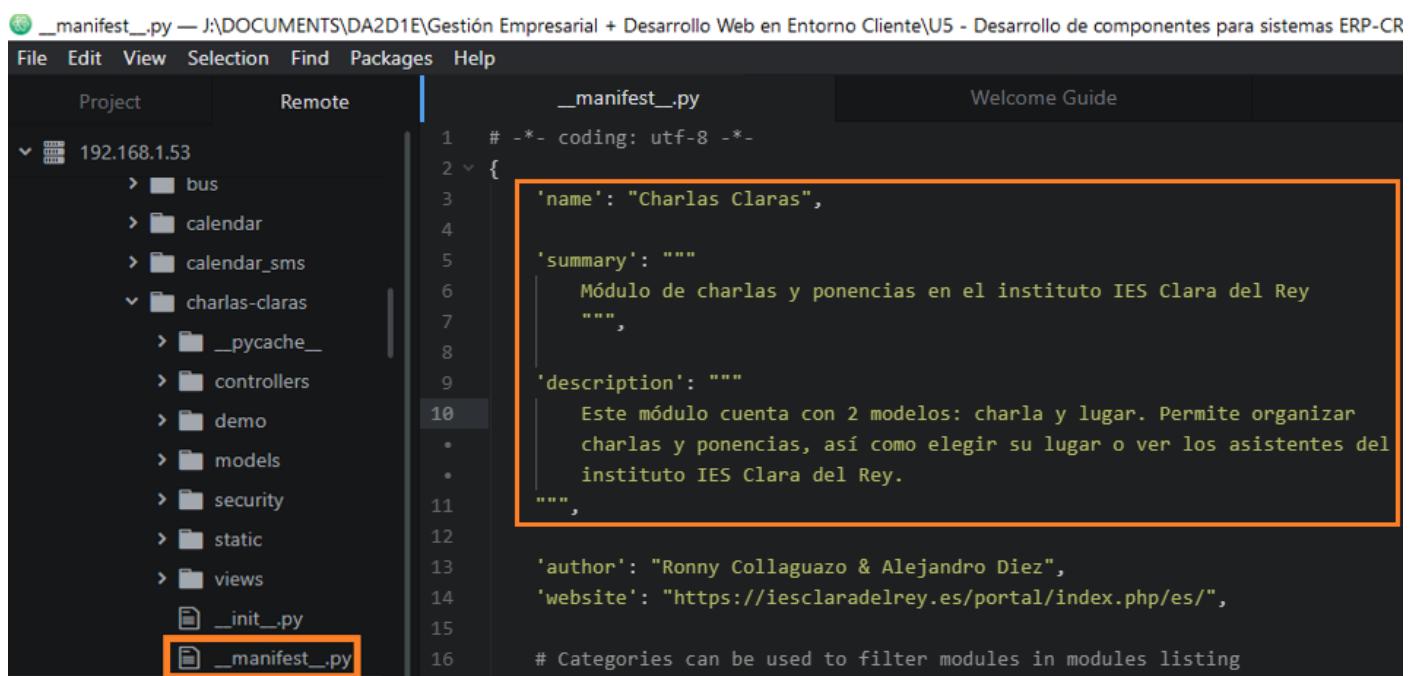
```

Project Remote .ftpconfig
Actividades
odoo
  odoo-server
    addons
      charlas-claras
        controllers
        demo
        models
          __init__.py
          models.py
        security
        views
          __init__.py
          __manifest__.py
      odoo
      .ftpconfig
1  {
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17 }

```

The screenshot shows a code editor interface with a sidebar for 'Project' and 'Remote' connections. In the project tree, there's a folder 'Actividades' containing an 'odoo' folder, which in turn has an 'odoo-server' folder with an 'addons' folder. Inside 'addons', there's a folder 'charlas-claras' which is highlighted with an orange box. This folder contains several subfolders: 'controllers', 'demo', 'models' (which contains files like '__init__.py', 'models.py', and '__manifest__.py'), 'security', and 'views'. Below 'models' is another 'odoo' folder and a '.ftpconfig' file. The code editor itself displays the content of the '.ftpconfig' file, which is a JSON object defining an FTP configuration for connecting to a host at 192.168.1.54 on port 21.

Añadimos un resumen y una descripción de lo que hace nuestro módulo en **__manifest__.py**



```

File Edit View Selection Find Packages Help
Project Remote __manifest__.py Welcome Guide
192.168.1.53
  bus
  calendar
  calendar_sms
  charlas-claras
    __pycache__
    controllers
    demo
    models
    security
    static
    views
      __init__.py
      __manifest__.py
1  # -*- coding: utf-8 -*-
2  {
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17 }

  'name': "Charlas Claras",
  'summary': """
    Módulo de charlas y ponencias en el instituto IES Clara del Rey
    """,
  'description': """
    Este módulo cuenta con 2 modelos: charla y lugar. Permite organizar
    charlas y ponencias, así como elegir su lugar o ver los asistentes del
    instituto IES Clara del Rey.
    """,
  'author': "Ronny Collaguazo & Alejandro Díez",
  'website': "https://iesclaradelrey.es/portal/index.php/es/",
  '# Categories can be used to filter modules in modules listing

```

The screenshot shows a code editor interface with a sidebar for 'File', 'Edit', 'View', 'Selection', 'Find', 'Packages', and 'Help'. The project tree on the left shows a connection to '192.168.1.53' with various Odoo modules like 'bus', 'calendar', etc. The main editor area shows the content of the __manifest__.py file for the 'charlas-claras' module. The 'summary' and 'description' fields are highlighted with an orange box. The 'summary' field contains a brief description of the module's purpose, and the 'description' field provides more detailed information about the features, such as having two models: 'charla' and 'lugar', and allowing users to organize events and view attendees.

Activamos el modo desarrollador desde **Settings/Developer Tools**:

Developer Tools

- Activate the developer mode**
- Activate the developer mode (with assets)
- Activate the developer mode (with tests assets)

Nuestro módulo **charlas-claras**, en caso de heredar, va a heredar de **Marketing/Eventos**, para ello instalamos la aplicación **Events**:

The screenshot shows the Odoo Apps page. On the left, there's a sidebar with categories like Sales, Services, Accounting, etc., and a section for Marketing which is highlighted with an orange box. In the main area, there are several modules listed: Email Marketing, Marketing Automation, SMS Marketing, and Events. The 'Events' module is selected and has an orange box around it. It shows the icon, name, category (website_event), status (Installed), and a 'Learn More' button.

Y la abrimos:

The screenshot shows a browser window with the URL `192.168.1.54:8069/web#cid=1&menu_id=127&action=1858&model=event.event&view_type=kanban`. The page title is 'Events'. The main content area displays a kanban view of event records.

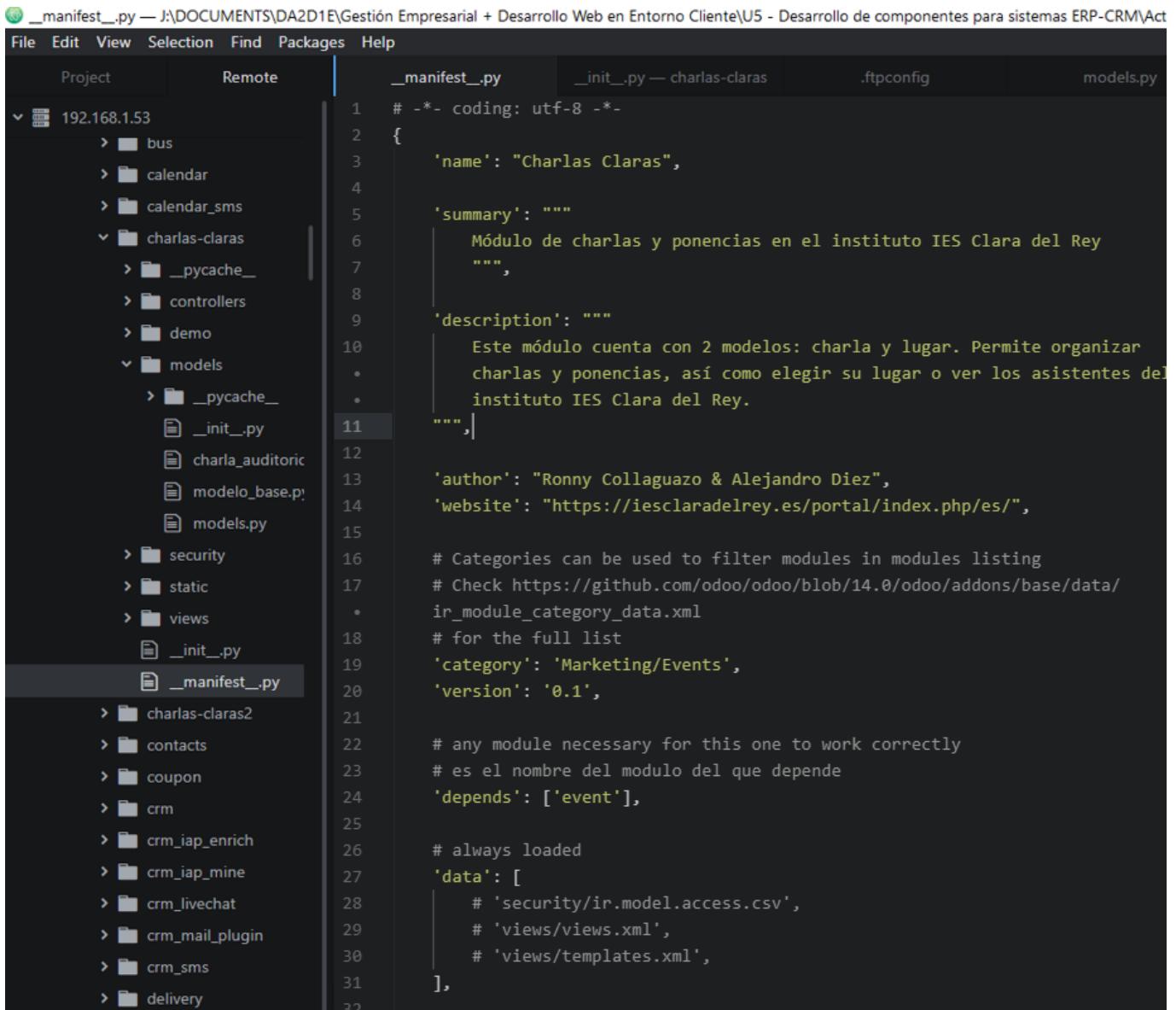
En la captura anterior podemos ver que nuestro módulo, en caso de heredar de algún módulo, heredará de **event**, lo incluimos en el **__manifest__.py** por si acaso:

```

__init__.py
__manifest__.py
contacts
coupon
crm
crm_iap_enrich
crm_iap_mine
crm_livechat
16      # Categories can be used to filter modules in modules listing
17      # Check https://github.com/odoo/odoo/blob/14.0/odoo/addons/base/
18      # data/ir_module_category_data.xml
19      # for the full list
20      'category': 'Uncategorized',
21      'version': '0.1',
22
23      # any module necessary for this one to work correctly
24      'depends': ['event'],

```

La versión actual del `__manifest__.py` quedaría de la siguiente manera:



```

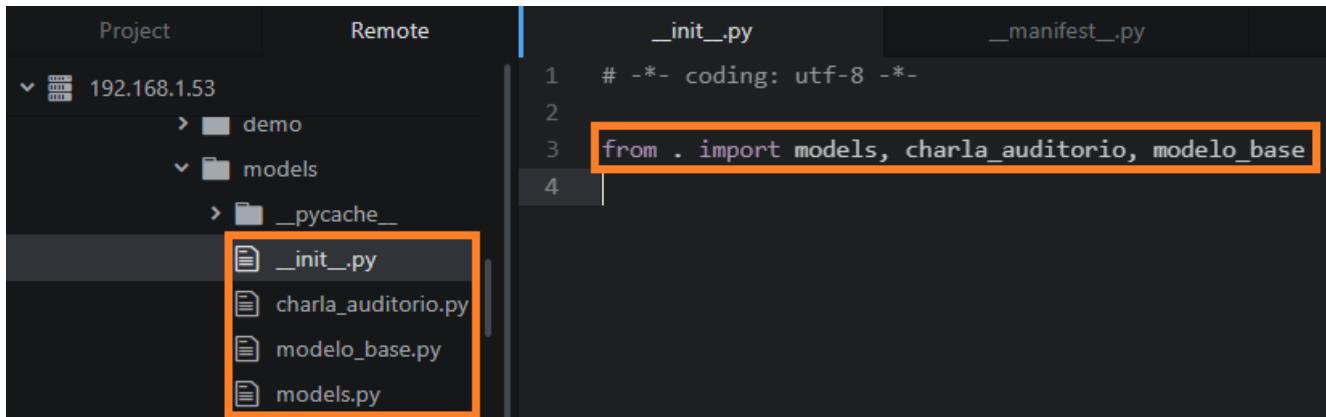
__manifest__.py — J:\DOCUMENTS\DA2D1E\Gestión Empresarial + Desarrollo Web en Entorno Cliente\U5 - Desarrollo de componentes para sistemas ERP-CRM\Act
File Edit View Selection Find Packages Help

Project Remote
192.168.1.53
    bus
    calendar
    calendar_sms
    charlas-claras
        __pycache__
        controllers
        demo
        models
            __pycache__
            __init__.py
            charla_auditorio
            modelo_base.py
            models.py
        security
        static
        views
            __init__.py
            __manifest__.py
    charlas-claras2
    contacts
    coupon
    crm
    crm_iap_enrich
    crm_iap_mine
    crm_livechat
    crm_mail_plugin
    crm_sms
    delivery

__manifest__.py
1 # -*- coding: utf-8 -*-
2 {
3     'name': "Charlas Claras",
4
5     'summary': """
6         Módulo de charlas y ponencias en el instituto IES Clara del Rey
7         """,
8
9     'description': """
10         Este módulo cuenta con 2 modelos: charla y lugar. Permite organizar
11         charlas y ponencias, así como elegir su lugar o ver los asistentes del
12         instituto IES Clara del Rey.
13         """,
14
15     'author': "Ronny Collaguazo & Alejandro Díez",
16     'website': "https://iesclaradelrey.es/portal/index.php/es/",
17
18     # Categories can be used to filter modules in modules listing
19     # Check https://github.com/odoo/odoo/blob/14.0/odoo/addons/base/data/
20     # ir_module_category_data.xml
21     # for the full list
22     'category': 'Marketing/Events',
23     'version': '0.1',
24
25     # any module necessary for this one to work correctly
26     # es el nombre del modulo del que depende
27     'depends': ['event'],
28
29     # always loaded
30     'data': [
31         # 'security/ir.model.access.csv',
32         # 'views/views.xml',
33         # 'views/templates.xml',
34     ],
35
36 }

```

Para que las características de los modelos se muestren en la pestaña de módulos de nuestro Odoo, tendremos que importar él/los modelos desde `__init__.py`



```

__init__.py
1 # -*- coding: utf-8 -*-
2
3 from . import models, charla_auditorio, modelo_base
4

```

Paramos el servicio de Odoo desde la terminal pulsando CTRL+C y lo volvemos a arrancar con el comando:

```
./odoo-bin -c /etc/odoo-server.conf
```

Teniendo el modo desarrollador activado, actualizamos la lista de Apps:

A screenshot of the Odoo web interface. At the top, the URL bar shows '192.168.1.55:8069/web?debug=1#cid=18'. Below the URL bar is a purple header with tabs for 'Apps', 'App Store', 'Updates', and 'Update Apps List'. The 'Update Apps List' tab is highlighted with a red box and has a small orange arrow pointing towards it. On the right side of the header, there are several status icons and a user profile for 'Mitchell Admin (DBlue)'.

Y procedemos a buscar nuestro módulo desde la caja de búsqueda quitando todos los filtros predeterminados para que nos aparezca:

A screenshot of the Odoo Apps page. The search bar at the top contains the text 'Module charlas'. Below the search bar, there are filters, group by options, and a page number '1-1 / 1'. On the left, there's a sidebar with 'CATEGORIES' and 'All' selected. In the main area, a module named 'charlas-claras' is listed. This module has an icon, the name 'Charlas Claras', the technical name 'charlas-claras', and an 'Install' button. A small menu next to the module listing includes 'Module Info' and 'Learn More', both of which are highlighted with a red box.

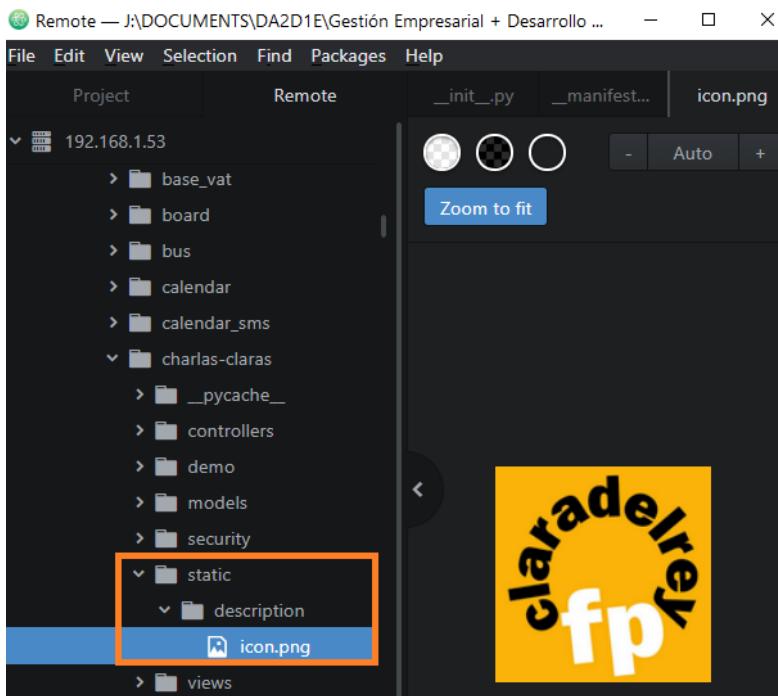
Si pulsamos en los 3 puntos del módulo y nos vamos a "Module info" podemos ver lo que hemos rellenado en `__manifest__.py`

A screenshot of the Odoo 'Module Info' page for the 'charlas-claras' module. The title is 'Charlas Claras' and it's 'By Ronny Collaguazo & Alejandro Diez'. There is an 'Install' button. Below the title, there are tabs for 'Information' and 'Technical Data'. Under 'Information', there are fields for 'Website' (https://iesclarade...), 'Category' (Events), and 'Summary' (Módulo de charlas y ponencias en el instituto IES Clara del Rey). Under 'Technical Data', there are fields for 'Technical Name' (charlas-claras), 'License' (LGPL Version 3), and 'Latest Version' (15.2.0.1). To the right of the summary, there is a logo for 'IES Clara del Rey'.

Este módulo cuenta con 2 modelos: *charla* y *lugar*. Permite organizar charlas y ponencias, así como elegir su lugar o ver los asistentes del instituto IES Clara del Rey.

Para que aparezca nuestro icono como se puede ver en la captura anterior, hay que guardar un archivo llamado icon de tipo .png en la siguiente ruta:

/charlas-claras/static/description/icon.png



Y volver a ejecutar el comando:

```
alejandro@alejandros-server:/$ sudo chmod -R 777 odoo/
```

Para que haya permisos de ejecución de la imagen, ya que al sincronizar local->remoto desde Atom se pierden los permisos:

The terminal window shows the following command sequence:

```
alejandro@alejandros-server:~$ cd /
alejandro@alejandros-server:/$ sudo chmod -R 777 odoo/
alejandro@alejandros-server:/$ ls -l
```

The output of the 'ls -l' command lists numerous files and directories in the root directory, including 'bin', 'boot', 'cdrom', 'dev', 'etc', 'home', 'lib', 'lib32', 'lib64', 'lost+found', 'media', 'mnt', 'sbin', 'snap', 'sys', 'tmp', 'usr', and 'var'. The file 'odoo' is highlighted with a red box in the terminal window.

Si queremos que nos aparezca en el cajón de aplicaciones y que sea instalable, hay que añadirle a `__manifest__.py`

```
'application': True,
'installable': True,
```

Para que aparezca la primera en el cajón de aplicaciones, agregamos:

```
'sequence': -100,
```

Y para que aparezca en la pestaña de Marketing, modificamos la 'category' a:

```
'category': 'Marketing/Events',
```

La versión actual del `__manifest__.py` quedaría de la siguiente manera:

```

__manifest__.py — charlas-claras — J:\DOCUMENTS\DA2D1E\Gestión Empresarial + Desarrollo Web en Entorno Cliente\U5 - Desarrollo de componentes\charlas-claras\__manifest__.py
File Edit View Selection Find Packages Help

Project Remote __init__.py __manifest__.py icon.png
1 # -*- coding: utf-8 -*-
2 {
3     'name': "Charlas Claras",
4
5     'summary': """
6         Módulo de charlas y ponencias en el instituto IES Clara del Rey
7         """,
8
9     'description': """
10        Este módulo cuenta con 2 modelos: charla y lugar. Permite organizar
11        charlas y ponencias, así como elegir su lugar o ver los asistentes del
12        instituto IES Clara del Rey.
13        """,
14
15     'author': "Ronny Collaguazo & Alejandro Diez",
16     'website': "https://iesclaradelrey.es/portal/index.php/es/",
17
18     # Categories can be used to filter modules in modules listing
19     # Check https://github.com/odoo/odoo/blob/14.0/odoo/addons/base/data/
20     # ir_module_category_data.xml
21     # for the full list
22     'category': 'Marketing/Events',
23     'version': '0.1',
24
25     # any module necessary for this one to work correctly
26     # es el nombre del modulo del que depende
27     'depends': ['event'],
28
29     # always loaded
30     'data': [
31         # 'security/ir.model.access.csv',
32         # 'views/views.xml',
33         # 'views/templates.xml',
34     ],
35
36     # mostrarla en el cajón de aplicaciones:
37     'application': True,
38     # mostrarla la 1ra en el cajón de aplicaciones:
39     'sequence': -100,
40     # para permitir la instalación del modulo:
41     'installable': True,
42 }
```

Reiniciamos los servicios y actualizamos la lista de aplicaciones:

The screenshot shows the Odoo Apps page. On the left, there's a sidebar with categories like Sales, Services, Accounting, etc., and Marketing is highlighted. The main area lists various apps with their icons, names, and status (e.g., Install, Learn More). One app, 'Charlas Claras', is highlighted with an orange border.

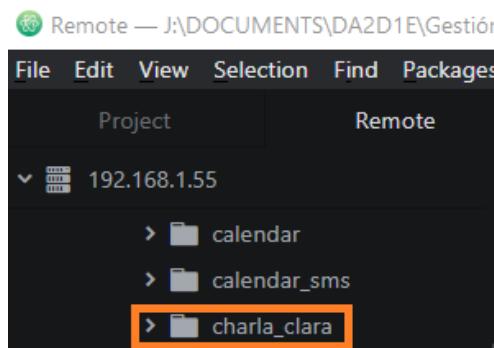
Icon	Name	Description	Status
	Charlas Claras	charlas-claras	Install
	Invoicing	account	Install
	MRP II	mrp_workorder	Upgrade
	Sales	sale_management	Install
	CRM	crm	Install
	Website	website	Installed

Y si entramos en la categoría de Marketing:

The screenshot shows the Odoo Apps page with the Marketing category selected in the sidebar. The main area lists specific marketing apps: Email Marketing, Marketing Automation, Surveys, and SMS Marketing, each with its own icon, name, description, and status (Install, Upgrade, Learn More).

Icon	Name	Description	Status
	Email Marketing	mass_mailing	Install
	Marketing Automation	marketing_automation	Upgrade
	Surveys	survey	Install
	SMS Marketing	mass_mailing_sms	Install

Ahora queremos cambiarle el nombre a `charlas-claras` por el de `charla_clara` para que esté de acuerdo con la nomenclatura y buena praxis que utiliza Odoo 15 como se puede ver en el resto de módulos. Para ello paramos los servicios interrumpiendo la conexión y editamos los nombres en Atom:



Reiniciamos servicios y vemos que nos aparece el nombre del otro módulo también, para eliminarlo, vamos a “Module Info”-> “Discard”-> “Action”-> “Delete”:

Icon	Name	Action
Icon	Charlas Claras charla_clara	Install
Icon	Charlas Claras charlas-claras	Install

Actions

- Save
- Discard

3 / 67

Actions

- Action
- Delete

En la máquina virtual de mi compañero se pueden ver todos los iconos del cajón de aplicaciones como se muestra en la siguiente captura:

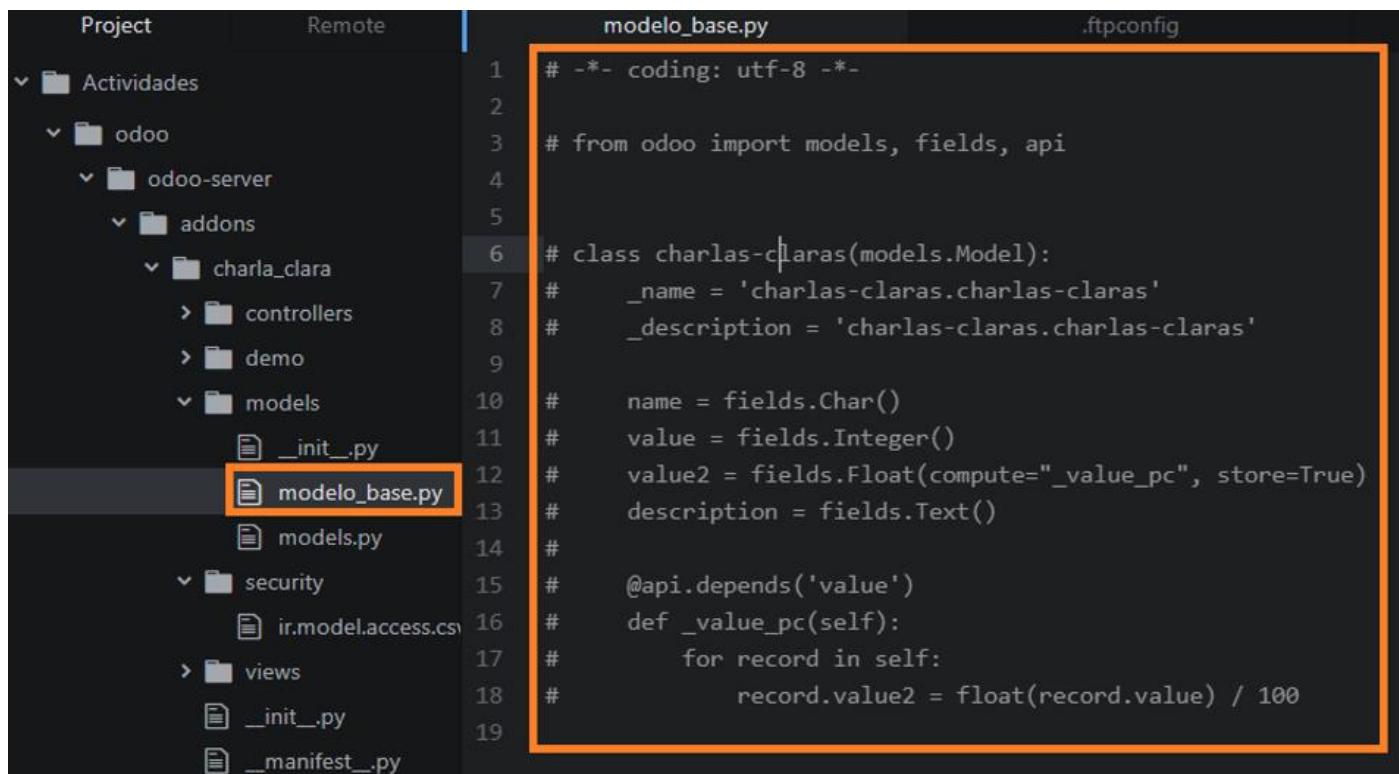
The screenshot shows the Odoo application menu titled "Aplicaciones". On the left, there's a sidebar with "CATEGORÍAS" and a list of modules: Todos (selected), Ventas, Servicios, Contabilidad, Inventario, Fabricación, Sitio web, and Marketing. The main area displays six application cards:

- Charlas Claras**: Organiza de manera eficiente las charlas más importantes y claras para tus empleados. Status: Instalado. Buttons: Aprenda más.
- Ventas**: De presupuestos a facturas. Status: Instalado. Buttons: Aprenda más.
- Facturación**: Facturas y pagos. Status: Instalado. Buttons: Aprenda más.
- CRM**: Seguimiento de clientes potenciales y oportunidades próximas. Status: Instalar. Buttons: Aprenda más.
- MRP II**: Órdenes de trabajo, planificación, informes de. Status: Instalado. Buttons: Aprenda más.
- Sitio web**: Constructor de sitio web empresarial. Status: Instalado. Buttons: Aprenda más.

At the top right, there are notifications (1 message, 4 messages), user info (Your Company, Mitchell Admin), and system icons.

3. Instalación del módulo y creación de los modelos:

Lo primero que haremos será una copia de **models.py**, que es el esqueleto de los modelos que vamos a utilizar que se llamará **modelo_base.py** para trabajar más cómodamente con models.py u otros modelos:



```

Project          Remote
Actividades
  odoo
    odoo-server
      addons
        charla_clara
        controllers
        demo
        models
          __init__.py
          modelo_base.py
          models.py
        security
          ir.model.access.csv
      views
        __init__.py
        __manifest__.py
  .ftpconfig

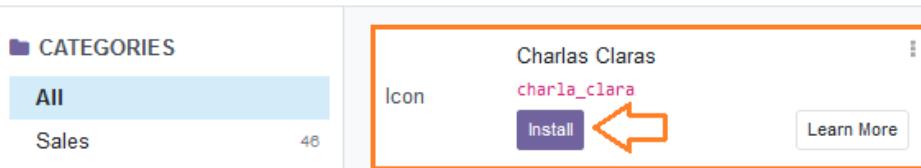
model_base.py
1  # -*- coding: utf-8 -*-
2
3  # from odoo import models, fields, api
4
5
6  # class charlas-claras(models.Model):
7  #     _name = 'charlas-claras.charlas-claras'
8  #     _description = 'charlas-claras.charlas-claras'
9
10 #     name = fields.Char()
11 #     value = fields.Integer()
12 #     value2 = fields.Float(compute="_value_pc", store=True)
13 #     description = fields.Text()
14 #
15 #     @api.depends('value')
16 #     def _value_pc(self):
17 #         for record in self:
18 #             record.value2 = float(record.value) / 100
19

```

Instalamos el módulo y lo iremos actualizando según añadamos: modelos, vistas, etc...; a veces será necesario desinstalarlo para actualizarlo.



Apps



Pasamos ahora a definir cómo queremos que sean nuestros modelos en un diseño preliminar:

Modelo	Campos	Tipo	Para qué sirve	Otros atributos
<code>charla_charla.py</code>	name	Char	Título de la charla	string, required
	fecha	Date	Fecha en la que tiene lugar	string, required
	lugar	Many2one	Elegir uno de los lugares que hemos creado gracias al modelo de la clase CharlaLugar	string, required, ondelete
	tipo	Selection	Seleccionamos el tipo de charla: "pública", "privada" o "secreta"	string, required, default
	currency_id	Many2one	Seleccionamos una moneda de las disponibles en Odoo	string
	precio	Monetary	El precio de entrada	string, required
	ponente	Many2one	Elegimos un contacto de los existentes dentro de Odoo	string, required
	duracion	Float	La duración en minutos con decimales si es necesario	string, required
	sinopsis	Text	Breve descripción de la charla o una intro...	string
	asistentes_confirmados	integer	Asistentes que han confirmado su asistencia a la charla	string, required
	ganancias_totales	Compute	Llama a una función que ejecuta el cálculo: <code>precio*asistentes_confirmados</code>	compute, readonly

Modelo	Campos	Tipo	Para qué sirve	Otros atributos
<code>charla_lugar.py</code>	name	Char	Nombre del lugar donde se va a celebrar la charla	string, required
	aforo	Integer	Capacidad máxima del lugar	string
	imagen	Binary	Foto del lugar	string
	parking_dis	Boolean	Si el lugar dispone de parking	string

Podemos ya diseñar nuestros modelos e implementarlos en nuestra aplicación. Para ello será necesario desinstalar la aplicación Charlas Claras y una vez terminados los modelos volver a instalarla:

charla_charla.py

```

charla_charla.py — J:\DOCUMENTS\DA2D1E\Gestión Empresarial + Desarrollo Web en Entorno Cliente\U5 - Desarrollo de componentes para sistemas ERP-C
File Edit View Selection Find Packages Help
Project Remote charla_charla.py .ftpconfig __init__.py te
1 # -*- coding: utf-8 -*-
2
3 from odoo import models, fields, api
4
5 class CharlaCharla(models.Model):
6     # _inherit = 'event.event'
7     _name = 'charla_clara.charla'
8     _description = 'Charla Clara'
9
10     name = fields.Char(string='Título', required=True)
11     fecha = fields.Date(string='Fecha', required=True)
12     lugar = fields.Many2one("charla_clara.lugar", string="Lugar", ondelete="set
13     null", required=True)
14
15     tipo = fields.Selection(
16         string="Tipo de Charla",
17         selection=[
18             ('P', "Pública"),
19             ('CI', "Privada, con invitación"),
20             ('S', "¿Qué charla? *le guiña un ojo*")],
21             required=True,
22             default='P')
23
24     currency_id = fields.Many2one('res.currency', string='Moneda')
25     precio = fields.Monetary(string='Precio', required=True)
26
27     ponente = fields.Many2one("res.users", string="Ponente", required=True)
28
29     duracion = fields.Float(string='Duración (minutos)', required=True)
30     sinopsis = fields.Text(string="Sinopsis")
31     asistentes_confirmados = fields.Integer(string='Asistentes confirmados',
32     required=True)
33
34     ganancias_totales = fields.Float(compute='_compute_ganancias_totales',
35     readonly=True)
36     @api.depends('precio', 'asistentes_confirmados')
37     def _compute_ganancias_totales(self):
38         for record in self:
39             record.ganancias_totales =
40             record.precio*record.asistentes_confirmados

```

charla_lugar.py

The screenshot shows a code editor interface with a dark theme. The top menu bar includes File, Edit, View, Selection, Find, Packages, and Help. Below the menu is a navigation bar with Project, Remote, charla_charla.py, charla_lugar.py (which is highlighted with a red box), and __init__.py. On the left, a tree view shows a project structure under 192.168.1.55, including base_vat, board, bus, calendar, calendar_sms, charla_clara (which is expanded to show __pycache__, controllers, demo, models, security, static, views, __init__.py, and __manifest__.py), and modelo_base.py. The main pane displays the code for charla_lugar.py:

```

1 # -*- coding: utf-8 -*-
2
3 from odoo import models, fields, api
4
5
6 class CharlaLugar(models.Model):
7     _inherit = 'event.event'
8     _name = 'charla_clara.lugar'
9     _description = 'Lugar Charla'
10
11     name = fields.Char(string='Lugar', required=True)
12
13     # codigo_lugar, sirve para guardar un código.
14     ...
15
16     codigo_lugar = fields.Selection(
17         string="Lugar de la Ponencia",
18         selection=[
19             ('ENSA', "Edificio Norte - Salón de Actos"),
20             ('EN1A', "Edificio Norte - Planta 1 - Sala A"),
21             ('ES2H', "Edificio Sur - Planta 2 - Sala H"),
22             ('POLI', "Polideportivo")],
23             default="POLI",
24             required=True
25     )
26
27     ...
28
29     aforo = fields.Integer(string='Aforo')
30     imagen = fields.Binary(string='Imagen')
31     parking_dis = fields.Boolean(string='Dispone de parking')

```

Para que nuestro módulo funcione, hay que importarlos desde `__init__.py`

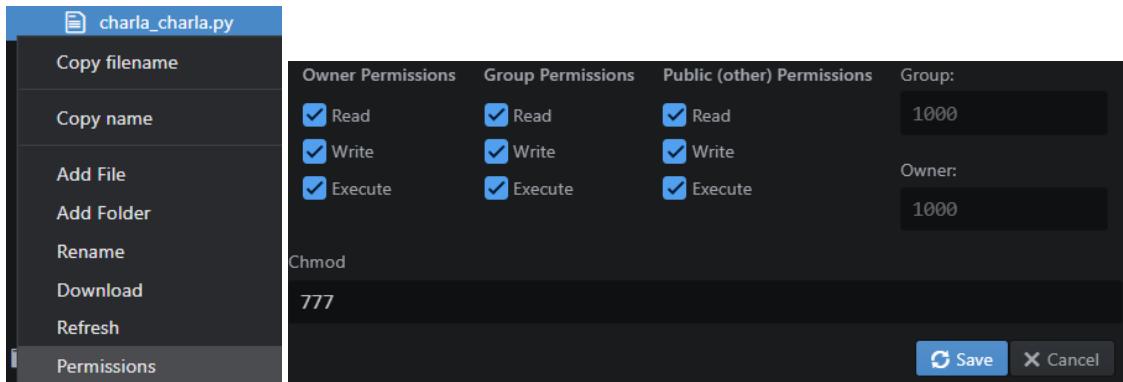
The screenshot shows a code editor interface with a dark theme. The top menu bar includes File, Edit, View, Selection, Find, Packages, and Help. Below the menu is a navigation bar with Project, Remote, __init__.py — charla_clara\models..., charla_charla.py, and charla_lugar.py. On the left, a tree view shows a project structure under 192.168.1.55, including charla_clara (expanded to show __pycache__, controllers, demo, models, security, static, views, __init__.py, and __manifest__.py), and modelo_base.py. The main pane displays the code for __init__.py:

```

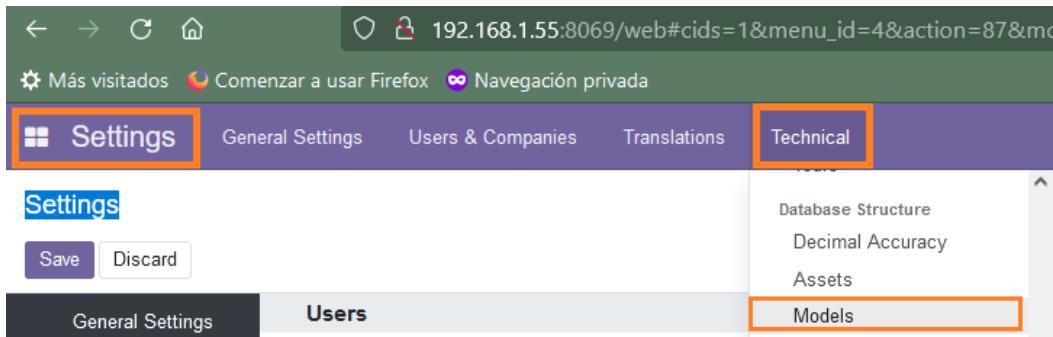
1 # -*- coding: utf-8 -*-
2
3 from . import models, modelo_base, charla_charla, charla_lugar
4

```

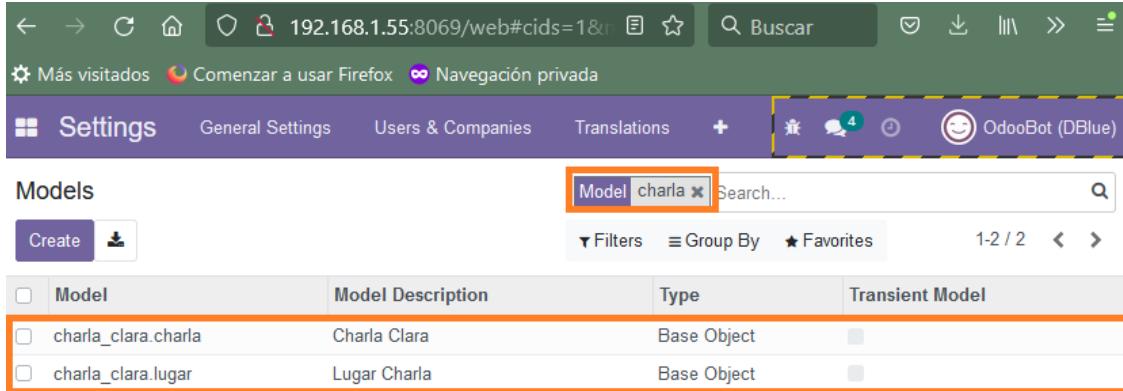
Comprobamos que los permisos de ambos archivos son 777 y si no lo son se los damos:



Procedemos aquí a reinstalar nuestro módulo Charlas Claras. Y a reconectarnos al servidor de Odoo para comprobar que nuestros modelos se han cargado correctamente. Y vamos a *Settings/Technical/Models*:



Y buscamos en el cajón *charla* para que aparezcan ambos modelos:



Modelo: charla_clara.charla

Settings General Settings Users & Companies Translations Technical OdooBot (DBlue)

Models / Charla Clara

Edit Create Print Action 1 / 2

Model Description	Charla Clara	Type	Base Object
Model	charla_clara.charla	In Apps	charla_clara
Order	id		
Transient Model			
Mail Thread			
Mail Activity			
Mail Blacklist			

Fields Access Rights Record Rules Notes Views Website Forms

Field Name	Field Label	Field Type	Required	Readonly	Indexed	Type
__last_update	Last Modified on	datetime	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Base Field
asistentes_confirmados	Asistentes confirmados	integer	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Base Field
create_date	Created on	datetime	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Base Field
create_uid	Created by	many2one	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Base Field
currency_id	Moneda	many2one	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Base Field
display_name	Display Name	char	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Base Field
duracion	Duracion (minutos)	float	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Base Field
fecha	Fecha	date	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Base Field
ganancias_totales	Ganancias Totales	float	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Base Field
id	ID	integer	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Base Field
lugar	Lugar	many2one	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Base Field
name	Título	char	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Base Field
ponente	Ponente	many2one	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Base Field
precio	Precio	monetary	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Base Field
sinopsis	Sinopsis	text	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Base Field
tipo	Tipo de Charla	selection	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Base Field
write_date	Last Updated on	datetime	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Base Field
write_uid	Last Updated by	many2one	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Base Field
Add a line						

Create a Menu

Modelo: charla_clara.lugar

Settings General Settings Users & Companies Translations Technical OdooBot (DBlue) 2 / 2

Models / Lugar Charla

Edit Create Print Action

Model Description	Lugar Charla	Type	Base Object
Model	charla_clara.lugar	In Apps	charla_clara
Order	id		
Transient Model			
Mail Thread			
Mail Activity			
Mail Blacklist			

Fields Access Rights Record Rules Notes Views Website Forms

Field Name	Field Label	Field Type	Required	Readonly	Indexed	Type
__last_update	Last Modified on	datetime	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Base Field
aforo	Aforo	integer	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Base Field
create_date	Created on	datetime	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Base Field
create_uid	Created by	many2one	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Base Field
display_name	Display Name	char	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Base Field
id	ID	integer	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Base Field
imagen	Imagen	binary	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Base Field
name	Lugar	char	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Base Field
parking_dis	Dispone de parking	boolean	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Base Field
write_date	Last Updated on	datetime	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Base Field
write_uid	Last Updated by	many2one	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Base Field
Add a line						

Create a Menu

4. Creación de las vistas, acciones de ventana y elementos de menú del módulo Charlas Claras

4.1 Vistas

Las vistas que crearemos para nuestros modelos son: árbol, formulario, kanban, búsqueda y heredada de búsqueda. Se encuentran en el archivo de la ruta: *charla_clara/views/views.xml*

En *__init__.py* debe de encontrarse importada la carpeta views:

The screenshot shows a file explorer window with the following details:

- File Explorer:** Shows the project structure under "192.168.1.55".
 - charla_clara:** Contains __pycache__, controllers, demo, models, security, static, and views.
 - models:** Contains __pycache__, __init__.py, charla_charla.py, charla_lugar.py, modelo_base.py, and models.py.
 - views:** Contains templates.xml, views.xml, __init__.py, and __manifest__.py.
- Code Editor:** Displays the content of *__init__.py*. The line `from . import views` is highlighted with an orange box.
- Code Editor:** Displays the content of *views.xml*. The entire file is highlighted with an orange box.

A la hora de crear las vistas seguiremos las buenas prácticas para que tengan una nomenclatura acorde correcta.

Vista árbol: Modelo charla_clara.charla

views.xml — J:\DOCUMENTS\DA2D1E\Gestión Empresarial + Desarrollo Web en Entorno Cliente\U5 - Desarrollo de componentes para sistemas ERP-C

```

<odoo>
    <data>
        <!-- explicit list view definition -->
        <!--
        <record model="ir.ui.view" id="charla_clara.list">
            <field name="name">charla_clara list</field>
            <field name="model">charla_clara.charla_clara</field>
            <field name="arch" type="xml">
                <tree>
                    <field name="name"/>
                    <field name="value"/>
                    <field name="value2"/>
                </tree>
            </field>
        </record>
        -->
        <!-- VISTA ÁRBOL - MODELO: CHARLA -->
        <record model="ir.ui.view" id="charla_clara.charla_tree">
            <field name="name">charla_clara.charla.tree</field>
            <field name="model">charla_clara.charla</field>
            <field name="arch" type="xml">
                <tree string="Lista de charlas">
                    <field name="name"/>
                    <field name="fecha"/>
                    <field name="lugar"/>
                    <field name="tipo"/>
                    <field name="precio"/>
                    <field name="ponente"/>
                    <field name="sinopsis"/>
                    <field name="duracion" optional="show"/>
                    <field name="asistentes_confirmados" optional="show"/>
                    <field name="ganancias_totales" optional="show"/>
                </tree>
            </field>
        </record>
    </data>
</odoo>

```

Vista formulario: Modelo charla_clara.charla

```
<!-- VISTA FORMULARIO - MODELO: CHARLA -->
<record model="ir.ui.view" id="charla_clara.charla_form">
    <field name="name">charla_clara.charla.form</field>
    <field name="model">charla_clara.charla</field>
    <field name="arch" type="xml">
        <form string="Formulario Charla">
            <sheet>
                <h1 style="padding-left: 20px; padding-top: 20px; font-size: 32px; color: #1e272e; font-weight: bold; text-decoration: underline;">
                    <field style="text-decoration: underline;" name="name"/>
                </h1>
                <group>
                    <group>
                        <field name="fecha"/>
                        <field name="lugar"/>
                        <field name="duracion"/>
                        <field name="ponente"/>
                        <field name="tipo"/>
                    </group>
                    <group>
                        <field name="precio"/>
                        <field name="currency_id"/>
                        <field name="asistentes_confirmados"/>
                        <field name="ganancias_totales"/>
                    </group>
                    <group>
                        <field name="sinopsis"/>
                    </group>
                </sheet>
            </form>
        </field>
    </record>
```

Vista kanban: Modelo charla_clara.charla

```

<!-- VISTA KANBAN - MODELO: CHARLA -->
<record model="ir.ui.view" id="charla_clara.charla_kanban">
    <field name="name">charla_clara.charla.kanban</field>
    <field name="model">charla_clara.charla</field>
    <field name="arch" type="xml">
        <kanban string="Vista Kanban de Charlas">
            <field name="name"/>
            <field name="fecha"/>
            <field name="lugar"/>
            <field name="ponente"/>
            <field name="tipo"/>
            <field name="ganancias_totales"/>
            <templates>
                <t t-name="kanban-box">
                    <div class="oe_kanban_global_click" style="padding: 20px;
                        display: flex; flex-direction: column; gap: 10px;">
                        <strong class="o_kanban_record_title" style="font-size: 20px;
                            font-weight: bold; text-decoration: underline;"><field
                            name="name"/></strong>
                        <ul style="font-size: 14px; list-style-type: none; padding: 0;
                            display: flex; flex-direction: column; gap: 10px;">
                            <li>
                                <span style="font-weight: bold;">- Fecha: </span><field
                                    name="fecha"/>
                            </li>
                            <li>
                                <span style="font-weight: bold;">- Lugar: </span><field
                                    name="lugar"/>
                            </li>
                            <li>
                                <span style="font-weight: bold;">- Ponente: </span><field
                                    name="ponente"/>
                            </li>
                            <li>
                                <span style="font-weight: bold;">- Tipo: </span><field
                                    name="tipo"/>
                            </li>
                            <li>
                                <span style="font-weight: bold;">- Ganancia Totales: </
                                    span><field name="ganancias_totales"/>
                            </li>
                        </ul>
                    </div>
                </t>
            </templates>
        </kanban>
    </field>
</record>

```

Vista búsqueda: Modelo charla_clara.charla

```
<!-- VISTA BÚSQUEDA (FILTROS) - MODELO: CHARLA -->
<record model="ir.ui.view" id="charla_clara.charla_search">
    <field name="name">charla_clara.charla.search</field>
    <field name="model">charla_clara.charla</field>
    <field name="arch" type="xml">
        <search string="Charlas">
            <field name="name" string="Charla"/>
            <filter string="Precio ≥ 100$" name="precio_mayor_igual_100"
                help="Precio ≥ 100$" domain="[(‘precio’, ‘>=’, 100)]"/>
            <filter string="Precio ≤ 100$" name="precio_menor_igual_100"
                help="Precio ≤ 100$" domain="[(‘precio’, ‘<=’, 100)]"/>
            <separator/>
            <filter string="Duración ≥ 120mins"
                name="duracion_mayor_igual_120" help="Duración ≥ 120"
                domain="[(‘duracion’, ‘>=’, 120.0)]"/>
            <filter string="Duración ≤ 120mins"
                name="duracion_menor_igual_120" help="Duración ≤ 120"
                domain="[(‘duracion’, ‘<=’, 120.0)]"/>
            <separator/>
            <filter string="Charlas del Admin" name="charlas_del_admin"
                help="Charlas del Admin" domain="[(‘ponente’, ‘=’, ‘Admin
                Blue’)]"/>
            <separator/>
        </search>
    </field>
</record>
```

Vista heredada de búsqueda: Modelo charla_clara.charla

```
<!-- VISTA HEREDADA BÚSQUEDA (AGRUPAR POR) - MODELO: CHARLA -->
<record model="ir.ui.view" id="charla_clara.charla_search_inherit">
    <field name="name">charla_clara.charla.search.inherit</field>
    <field name="model">charla_clara.charla</field>
    <field name="inherit_id" ref="charla_clara.charla_search"/>
    <field name="arch" type="xml">
        <search string="Charlas">
            <group expand="0" string="Group By">
                <filter string="Tipo" name="tipo" context="{'group_by':
                    'tipo'}"/>
                <filter string="Lugar" name="lugar" context="{'group_by':
                    'lugar'}"/>
                <filter string="Ponente" name="ponente"
                    context="{'group_by': 'ponente'}"/>
                <filter string="Moneda" name="moneda" context="{'group_by':
                    'currency_id'}"/>
                <filter string="Fecha" name="fecha" domain="[]"
                    context="{'group_by': 'fecha'}"/>
            </group>
        </search>
    </field>
</record>
```

Podemos ver nuestras vistas creadas en Odoo en *Settings/Technical/Views*:

The screenshot shows the Odoo interface with the 'Technical' tab selected. In the main area, there is a table titled 'Views' listing four entries: 'charla_clara.charla.form' (Form), 'charla_clara.charla.kanban' (Kanban), 'charla_clara.charla.search' (Search), and 'charla_clara.charla.tree' (Tree). To the right of the table is a sidebar with sections for 'User-defined Defaults', 'IAP', 'IAP Accounts', 'User Interface', 'Menu Items', and 'Views'. The 'Views' section is highlighted with an orange arrow. Below it are 'Customized Views' and 'User-defined Filters'. A second screenshot below shows a search result for 'charla' with the same four view entries listed.

	View Name	Website	View Type	Model	Key	External ID	Inherited View
charla_clara.charla.form			Form	charla_clara.charla		charla_clara.charla_form	
charla_clara.charla.kanban			Kanban	charla_clara.charla		charla_clara.charla_kanban	
charla_clara.charla.search			Search	charla_clara.charla		charla_clara.charla_search	
charla_clara.charla.search.inherit			Search	charla_clara.charla		charla_clara.charla_search_inherit	charla_clara.charla.search
charla_clara.charla.tree			Tree	charla_clara.charla		charla_clara.charla_tree	

Vista formulario: Modelo charla_clara.charla

The screenshot shows the configuration for the 'charla_clara.charla.form' view. It includes fields for View Name, View Type, Website, Key, Model, Sequence, and Active status. On the right, there are sections for Child Field (Inherited View), View inheritance mode (Base view), Model Data (charla_clara.charla.form), and External ID (charla_clara.charla_form). At the bottom, tabs for Architecture, Access Rights, and Inherited Views are visible, along with an 'Edit Translations' button. Below the configuration is the XML code for the form view.

View Name	charla_clara.charla.form	Child Field	
View Type	Form	Inherited View	
Website		View inheritance mode	Base view
Key		Model Data	charla_clara.charla.form
Model	charla_clara.charla	External ID	charla_clara.charla_form
Sequence	16		
Active	<input checked="" type="checkbox"/>		

Architecture Access Rights Inherited Views Edit Translations

```
<?xml version="1.0"?>
<form string="Formulario Charla">
    <sheet>
        <h1 style="padding-left: 20px; padding-top: 20px; font-size: 32px; color: #1e272e; font-weight: bold; text-decoration: underline;">
            <field style="text-decoration: underline;" name="name"/>
        </h1>
        <group>
            <group>
                <field name="fecha"/>
                <field name="lugar"/>
                <field name="duracion"/>
                <field name="ponente"/>
                <field name="tipo"/>
            </group>
            <group>
                <field name="precio"/>
                <field name="currency_id"/>
                <field name="asistentes_confirmados"/>
                <field name="ganancias_totales"/>
            </group>
            <group>
                <field name="sinopsis"/>
            </group>
        </sheet>
    </form>
```

Vista kanban: Modelo charla_clara.charla

The screenshot shows the Odoo Settings interface for managing views. The top navigation bar includes 'Settings', 'General Settings', 'Users & Companies', 'Translations', and 'Technical'. On the right, there are icons for OdooBot (DBlue) and notifications (4). The main content area is titled 'Views / charla_clara.charla.kanban'. Below this, there are tabs for 'Edit' (selected), 'Create', 'Action', and '2 / 6' with navigation arrows. The main form contains the following fields:

View Name	charla_clara.charla.kanban	Child Field	
View Type	Kanban	Inherited View	
Website		View inheritance mode	Base view
Key		Model Data	charla_clara.charla.kanban
Model	charla_clara.charla	External ID	charla_clara.charla_kanban
Sequence	16		
Active	<input checked="" type="checkbox"/>		

Below the form, there are three tabs: 'Architecture' (selected), 'Access Rights', and 'Inherited Views'. To the right of the tabs is a link 'Edit Translations'. The XML code for the Kanban view is displayed in a large text area:

```

<?xml version="1.0"?>
<kanban string="Vista Kanban de Charlas">
    <field name="name"/>
    <field name="fecha"/>
    <field name="lugar"/>
    <field name="ponente"/>
    <field name="tipo"/>
    <field name="ganancias_totales"/>
    <templates>
        <t t-name="Kanban-box">
            <div class="oe_kanban_global_click" style="padding: 20px; display: flex; flex-direction: column; gap: 10px;">
                <strong class="o_kanban_record_title" style="font-size: 20px; font-weight: bold;text-decoration: underline;"><field name="name"/></strong>
                <ul style="font-size: 14px; list-style-type: none; padding: 0; display: flex; flex-direction: column; gap: 10px;">
                    <li>
                        <span style="font-weight: bold;">- Fecha: </span><field name="fecha"/>
                    </li>
                    <li>
                        <span style="font-weight: bold;">- Lugar: </span><field name="lugar"/>
                    </li>
                    <li>
                        <span style="font-weight: bold;">- Ponente: </span><field name="ponente"/>
                    </li>
                    <li>
                        <span style="font-weight: bold;">- Tipo: </span><field name="tipo"/>
                    </li>
                    <li>
                        <span style="font-weight: bold;">- Ganancia Totales: </span><field name="ganancias_totales"/>
                    </li>
                </ul>
            </div>
        </t>
    </templates>
</kanban>

```

Vista árbol: Modelo charla_clara.charla

The screenshot shows the Odoo interface for managing views. At the top, there's a purple header bar with tabs for 'Settings' (selected), 'General Settings', 'Users & Companies', 'Translations', and 'Technical'. On the right of the header are icons for a user, notifications (4), and OdooBot (DBlue). Below the header, the page title is 'Views / charla_clara.charla.tree'. There are buttons for 'Edit' (highlighted in blue) and 'Create', along with an 'Action' button. A navigation bar shows '3 / 6' and arrows for navigating through the views.

View Name	charla_clara.charla.tree	Child Field	
View Type	Tree	Inherited View	
Website		View inheritance mode	Base view
Key		Model Data	charla_clara.charla.tree
Model	charla_clara.charla	External ID	charla_clara.charla_tree
Sequence	16		
Active	<input checked="" type="checkbox"/>		

Below the table, there are three tabs: 'Architecture' (selected), 'Access Rights', and 'Inherited Views'. To the right, there's a link 'Edit Translations'.

```
<?xml version="1.0"?>
<tree string="Lista de charlas">
    <field name="name"/>
    <field name="fecha"/>
    <field name="lugar"/>
    <field name="tipo"/>
    <field name="precio"/>
    <field name="ponente"/>
    <field name="sinopsis"/>
    <field name="duracion" optional="show"/>
    <field name="asistentes_confirmados" optional="show"/>
    <field name="ganancias_totales" optional="show"/>
</tree>
```

Vista búsqueda: Modelo charla_clara.charla

The screenshot shows the Odoo Settings interface for a search view named 'charla_clara.charla.search'. The view type is 'Search' and it inherits from 'charla_clara.charla.search'. The XML code for the search view is displayed below:

```
<?xml version="1.0"?>
<search string="Charlas">
    <field name="name" string="Charla"/>
    <filter string="Precio ≥ 100" name="precio_mayor_igual_100" help="Precio ≥ 100" domain="[['precio', '>=', 100]]"/>
    <filter string="Precio ≤ 100" name="precio_menor_igual_100" help="Precio ≤ 100" domain="[['precio', '<=', 100]]"/>
    <separator/>
    <filter string="Duración ≥ 120mins" name="duracion_mayor_igual_120" help="Duración ≥ 120" domain="[['duracion', '>=', 120.0]]"/>
    <filter string="Duración ≤ 120mins" name="duracion_menor_igual_120" help="Duración ≤ 120" domain="[['duracion', '<=', 120.0]]"/>
    <separator/>
    <filter string="Charlas del Admin" name="charlas_del_admin" help="Charlas del Admin" domain="[['ponente', '=', 'Admin Blue']]"/>
    <separator/>
</search>
```

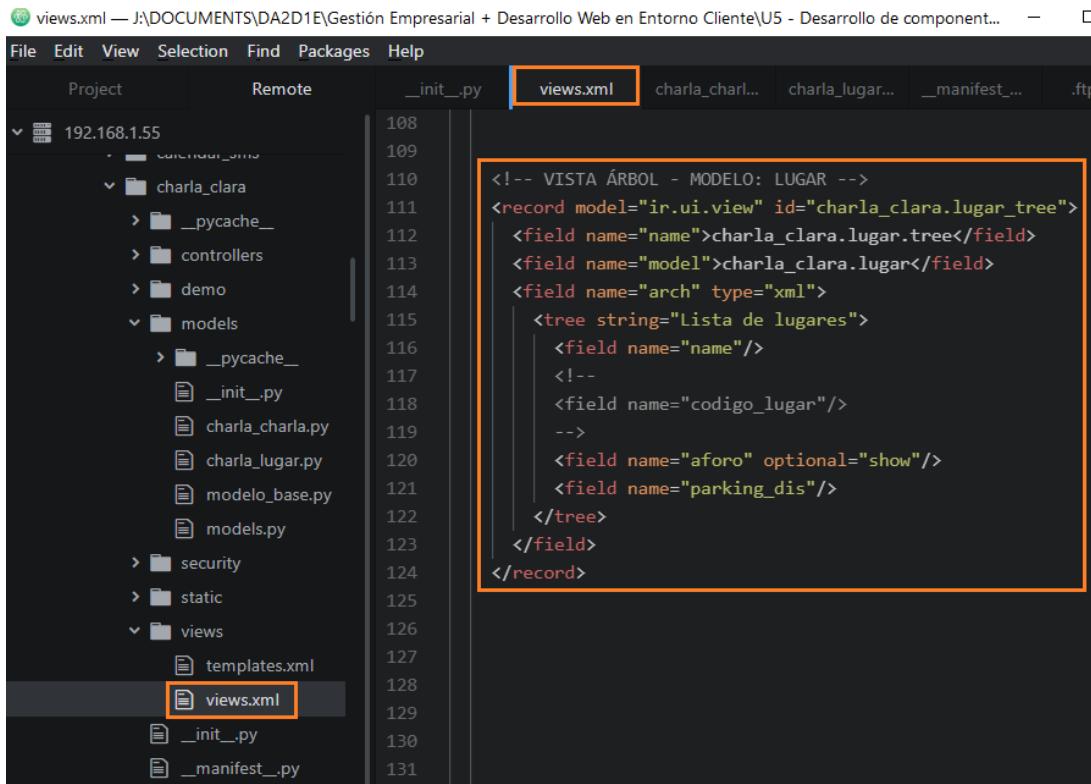
Vista heredada de búsqueda: Modelo charla_clara.charla

The screenshot shows the Odoo Settings interface for an inherited search view named 'charla_clara.charla.search.inherit'. The view type is 'Search' and it inherits from 'charla_clara.charla.search'. The XML code for the search view is displayed below:

```
<?xml version="1.0"?>
<search string="Charlas">
    <group expand="0" string="Group By">
        <filter string="Tipo" name="tipo" context="{'group_by': 'tipo'}"/>
        <filter string="Lugar" name="lugar" context="{'group_by': 'lugar'}"/>
        <filter string="Ponente" name="ponente" context="{'group_by': 'ponente'}"/>
        <filter string="Moneda" name="moneda" context="{'group_by': 'currency_id'}"/>
        <filter string="Fecha" name="fecha" domain="[]" context="{'group_by': 'fecha'}"/>
    </group>
</search>
```

Creamos ahora las vistas para el modelo charla_clara.lugar:

Vista árbol: Modelo charla_clara.lugar



```

<!-- VISTA ÁRBOL - MODELO: LUGAR -->
<record model="ir.ui.view" id="charla_clara.lugar_tree">
    <field name="name">charla_clara.lugar.tree</field>
    <field name="model">charla_clara.lugar</field>
    <field name="arch" type="xml">
        <tree string="Lista de lugares">
            <field name="name"/>
            <!--
            <field name="codigo_lugar"/>
            -->
            <field name="aforo" optional="show"/>
            <field name="parking_dis"/>
        </tree>
    </field>
</record>

```

Vista formulario: Modelo charla_clara.lugar



```

<!-- VISTA FORMULARIO - MODELO: LUGAR -->
<record model="ir.ui.view" id="charla_clara.lugar_form">
    <field name="name">charla_clara.lugar.form</field>
    <field name="model">charla_clara.lugar</field>
    <field name="arch" type="xml">
        <form string="Formulario Lugar">
            <sheet>
                <h1 style="padding-left: 20px; padding-top: 20px; font-size: 38px; color: #1e272e; font-weight: bold; text-decoration: underline;">
                    <field style="text-decoration: underline;" name="name"/>
                </h1>
                <group>
                    <group>
                        <span><field style="width: auto; margin-right: 30px; box-shadow: 0px 0px 5px 1px #1e272e; margin-left: 20px;" name="imagen" widget="image"/></span>
                    </group>
                    <group>
                        <!--
                        <field name="codigo_lugar"/>
                        -->
                        <field name="aforo"/>
                        <field name="parking_dis"/>
                    </group>
                </group>
            </sheet>
        </form>
    </field>
</record>

```

Vista kanban: Modelo charla_clara.lugar

```
<!-- VISTA KANBAN - MODELO: LUGAR -->
<record model="ir.ui.view" id="charla_clara.lugar_kanban">
    <field name="name">charla_clara.lugar.kanban</field>
    <field name="model">charla_clara.lugar</field>
    <field name="arch" type="xml">
        <kanban string="Vista Kanban de Lugares">
            <field name="name"/>
            <field name="imagen"/>
            <!--
            <field name="codigo_lugar"/>
            -->
            <field name="aforo"/>
            <field name="parking_dis"/>
        <templates>
            <t t-name="kanban-box">
                <div class="oe_kanban_global_click" style="padding: 20px;
display: flex; flex-direction: column; gap: 10px;">
                    <strong class="o_kanban_record_title" style="font-size: 20px;
font-weight: bold;text-decoration: underline;"><field
name="name"/></strong>
                    <ul style="font-size: 14px; list-style-type: none; padding: 0;
display: flex; flex-direction: column; gap: 10px;" >
                        <li>
                            <field style="max-width: 200px;box-shadow: 0px 0px 5px 1px
#1e272e;" name="imagen" widget="image"/>
                        </li>
                        <li>
                            <span style="font-weight: bold;">- Código Lugar: </span>
                            <!--
                            <field name="codigo_lugar"/>
                            -->
                        </li>
                        <li>
                            <span style="font-weight: bold;">- Aforo: </span><field
name="aforo"/>
                        </li>
                        <li>
                            <span style="font-weight: bold;">- Dispone de parking: </
span><field name="parking_dis"/>
                        </li>
                    </ul>
                </div>
            </t>
        </templates>
    </kanban>
</field>
</record>
```

Podemos ver nuestras vistas creadas en Odoo en *Settings/Technical/Views*:

The screenshot shows two instances of the Odoo interface. The top instance is a general view of the 'Views' list under the 'Technical' tab. The bottom instance is a detailed view of a specific record ('charla_clara.charla.form') within the same list.

Top Instance (General View):

- Header:** Settings, General Settings, Users & Companies, Translations, Technical.
- Section:** Views
- Buttons:** Create, Filter, Search...
- Table Headers:** View Name, Website, View Type, Model
- Data:**
 - charla_clara.charla.form (Form, charla_clara.charla)
 - charla_clara.charla.kanban (Kanban, charla_clara.charla)
 - charla_clara.charla.tree (Tree, charla_clara.charla)

Bottom Instance (Detailed View):

- Header:** Settings, General Settings, Users & Companies, Translations, Technical.
- Section:** Views
- Buttons:** View **charla**, Filter, Group By, Favorites, Search...
- Table Headers:** View Name, Website, View Type, Model, Key, External ID, Inherited View
- Data:**
 - charla_clara.charla.form (Form, charla_clara.charla, charla_clara.charla_form)
 - charla_clara.charla.kanban (Kanban, charla_clara.charla, charla_clara.charla_kanban)
 - charla_clara.charla.tree (Tree, charla_clara.charla, charla_clara.charla_tree)
 - charla_clara.lugar.form (Form, charla_clara.lugar, charla_clara.lugar_form) - This row is highlighted with an orange border.**
 - charla_clara.lugar.kanban (Kanban, charla_clara.lugar, charla_clara.lugar_kanban)
 - charla_clara.lugar.tree (Tree, charla_clara.lugar, charla_clara.lugar_tree)

Vista formulario: Modelo **charla_clara.lugar**

This screenshot shows the detailed configuration of the 'charla_clara.lugar.form' view. It includes a table of settings and a large text area displaying the XML code for the view's structure.

Table Settings:

View Name	charla_clara.lugar.form	Child Field	
View Type	Form	Inherited View	
Website		View inheritance mode	Base view
Key		Model Data	charla_clara.lugar.form
Model	charla_clara.lugar	External ID	charla_clara.lugar_form
Sequence	16		
Active	<input checked="" type="checkbox"/>		

Buttons: Edit, Create, Action, 4 / 6, Back, Forward

Sections: Architecture, Access Rights, Inherited Views

XML Code:

```

<?xml version="1.0"?>
<form string="Formulario Lugar">
    <sheet>
        <field style="padding-left: 20px; padding-top: 20px; font-size: 38px; color: #1e272e; font-weight: bold; text-decoration: underline;" name="name"/>
        <group>
            <group>
                <span><field style="width: auto; margin-right: 30px; box-shadow: 0px 0px 5px 1px #1e272e; margin-left: 20px;" name="imagen" widget="image"/></span>
            </group>
            <group>
                <!-->
                <field name="codigo_lugar"/>
                <!-->
                <field name="aforo"/>
                <field name="parking_dis"/>
            </group>
        </group>
    </sheet>
</form>

```

Vista kanban: Modelo charla_clara.lugar

The screenshot shows the Odoo configuration interface for a Kanban view named 'charla_clara.lugar.kanban'. The top navigation bar includes 'Settings', 'General Settings', 'Users & Companies', 'Translations', 'Technical', and a notification bar for OdooBot (DBlue). Below the navigation is a toolbar with 'Edit', 'Create', 'Action', and navigation icons.

View Name	charla_clara.lugar.kanban	Child Field	
View Type	Kanban	Inherited View	
Website		View inheritance mode	Base view
Key		Model Data	charla_clara.lugar.kanban
Model	charla_clara.lugar	View Architecture	
Sequence	16	This field is the same as 'arch' field without translations	
Active	<input checked="" type="checkbox"/>	<ul style="list-style-type: none"> Field: arch_base Object: ir.ui.view Type: text Widget: aoe Modifiers: {'required':true} 	
Architecture Access Rights Inherited Views		Edit Translations	

Below the table, the XML code for the Kanban view is displayed:

```

<?xml version="1.0"?>
<kanban string="Vista Kanban de Lugares">
    <field name="name"/>
    <field name="Imagen"/>
    <!--
        <field name="codigo_lugar"/>
    -->
    <field name="aforo"/>
    <field name="parking_dis"/>
    <templates>
        <t t-name="kanban-box">
            <div class="oe_kanban_global_click" style="padding: 20px; display: flex; flex-direction: column; gap: 10px;">
                <strong class="o_kanban_record_title" style="font-size: 20px; font-weight: bold;text-decoration: underline;"><field name="name"/></strong>
                <ul style="font-size: 14px; list-style-type: none; padding: 0; display: flex; flex-direction: column; gap: 10px;">
                    <li>
                        <field style="max-width: 200px;box-shadow: 0px 0px 5px 1px #1e272e;" name="Imagen" widget="image"/>
                    </li>
                    <li>
                        <span style="font-weight: bold;">- Código Lugar: </span>
                    <!--
                        <field name="codigo_lugar"/>
                    -->
                    </li>
                    <li>
                        <span style="font-weight: bold;">- Aforo: </span><field name="aforo"/>
                    </li>
                    <li>
                        <span style="font-weight: bold;">- Dispone de parking: </span><field name="parking_dis"/>
                    </li>
                </ul>
            </div>
        </t>
    </templates>
</kanban>

```

Vista árbol: Modelo charla_clara.lugar

Views / charla_clara.lugar.tree

Edit Create Action 6 / 6 < >

View Name	charla_clara.lugar.tree	Child Field	
View Type	Tree	Inherited View	
Website		View inheritance mode	Base view
Key		Model Data	charla_clara.lugar.tree
Model	charla_clara.lugar	External ID	charla_clara.lugar_tree
Sequence	16		
Active	<input checked="" type="checkbox"/>		

Architecture Access Rights Inherited Views Edit Translations

```
<?xml version="1.0"?>
<tree string="Lista de lugares">
    <field name="name"/>
    <!--
        <field name="codigo_lugar"/>
    -->
    <field name="aforo" optional="show"/>
    <field name="parking_dis"/>
</tree>
```

4.2 Acciones de ventana

The screenshot shows the Odoo XML editor interface. On the left, the file tree displays a project structure under '192.168.1.55'. The 'views' folder contains 'templates.xml' and 'views.xml'. The 'views.xml' file is selected and shown in the main editor area. The code defines two window actions: 'charla_action_window' for the 'charla_clara' model and 'lugar_action_window' for the 'charla_clara.lugar' model. Both actions use 'ir.actions.act_window' and specify 'tree,kanban,form' view modes. The 'help' field contains a message encouraging users to create their first record.

```

<!-- ACCIONES DE VENTANA -->
<!-- La nomenclatura que menciona el profesor se cumple luego internamente.
En id ponemos el nombre del modelo
-->

<!-- ACCION DE VENTANA: MODELO CHARLA -->
<record id="charla_action_window" model="ir.actions.act_window">
    <field name="name">Charlas</field>
    <field name="type">ir.actions.act_window</field>
    <field name="res_model">charla_clara.charla</field>
    <field name="view_mode">tree,kanban,form</field>
    <field name="help" type="html">
        <p class="o_view_nocontent_smiling_face">
            | ;Crea tu primera charla!
        </p>
    </field>
</record>

<!-- ACCION DE VENTANA: MODELO LUGAR -->
<record id="lugar_action_window" model="ir.actions.act_window">
    <field name="name">Lugares</field>
    <field name="type">ir.actions.act_window</field>
    <field name="res_model">charla_clara.lugar</field>
    <field name="view_mode">tree,kanban,form</field>
    <field name="help" type="html">
        <p class="o_view_nocontent_smiling_face">
            | ;Crea tu primer lugar!
        </p>
    </field>
</record>

```

Podemos ver nuestras acciones de ventana creadas en Odoo en *Settings/Technical/Window Actions*:

The screenshot shows the Odoo technical interface with the 'Technical' tab selected. In the sidebar, 'Window Actions' is highlighted. The main area shows a list of window actions for the 'charla_clara' model. Two actions are listed: 'Charlas' and 'Lugares'. The 'Charlas' action has an 'Action Name' of 'charla' and a 'Destination Model' of 'charla_clara.charla'. The 'Lugares' action has an 'Action Name' of 'lugar' and a 'Destination Model' of 'charla_clara.lugar'.

Action Name	Destination Model	View Ref.	Domain Value	Context Value
Charlas	charla_clara.charla			
Lugares	charla_clara.lugar			

Action Name: Charlas:

The screenshot shows the Odoo interface for managing action settings. The top navigation bar includes 'Settings', 'General Settings', 'Users & Companies', 'Translations', and 'Technical'. A notification bar on the right indicates 4 messages and shows 'OdooBot (DBlue)'. The main content area is titled 'Window Actions / Charlas'.

Action Details:

Action Name	Charlas	Action Usage	
External ID	charla_clara.charla_action_window	Action Type	ir.actions.act_window
Object	charla_clara.charla	Target Window	Current Window

General Settings:

- View Mode: tree,kanban,form
- Domain Value: 0
- Context Value: 80
- Limit: 80
- Filter:

Help:
¡Crea tu primera charla!

Views:

View Type	View
Add a line	

Action Name: Lugares:

The screenshot shows the Odoo interface for managing action settings. The top navigation bar includes 'Settings', 'General Settings', 'Users & Companies', 'Translations', and 'Technical'. A notification bar on the right indicates 4 messages and shows 'OdooBot (DBlue)'. The main content area is titled 'Window Actions / Lugares'.

Action Details:

Action Name	Lugares	Action Usage	
External ID	charla_clara.lugar_action_window	Action Type	ir.actions.act_window
Object	charla_clara.lugar	Target Window	Current Window

General Settings:

Views:

View Type	View
Add a line	

Help:
¡Crea tu primer lugar!

Views:

View Type	View
Add a line	

4.3 Elementos de menú

```

<!-- MENÚS Y SUBMENÚS -->
<menuitem id="charla_clara.menu_root"
           name="Charlas Claras"
           sequence="10"/>

<!-- Submenu del subtítulo charlas -->
<menuitem id="charla_clara.charlas"
           name="Charlas"
           parent="charla_clara.menu_root"
           action="charla_action_window"
           sequence="10"/>

<!-- Submenú del subtítulo Charlas -->
<menuitem id="charla_clara.gestionar_charlas"
           name="Gestionar Charlas"
           parent="charla_clara.charlas"
           action="charla_action_window"
           sequence="10"/>

<!-- Subtítulo lugares -->
<menuitem id="charla_clara.lugares"
           name="Lugares"
           parent="charla_clara.menu_root"
           action="lugar_action_window"
           sequence="10"/>

<!-- Submenú del subtítulo Lugares -->
<menuitem id="charla_clara.gestionar_lugares"
           name="Gestionar Lugares"
           parent="charla_clara.lugares"
           action="lugar_action_window"
           sequence="10"/>

```

Aquí el atributo `action` está relleno con el valor del atributo `id` de la acción de ventana correspondiente.

Podemos ver nuestros elementos de menú creados en Odoo en *Settings/Technical/Menu items*:

Module	Menu Item	Action
charla_clara	charlas	charla_action_window
charla_clara	gestionar_charlas	charla_action_window
charla_clara	lugares	lugar_action_window
charla_clara	gestionar_lugares	lugar_action_window

Buscamos "charla":

The screenshot shows the Odoo Settings interface with the 'Menu Items' tab selected. A search bar at the top contains the text 'charla'. Below the search bar, there are filters for 'Filters', 'Group By', and 'Favorites'. The results list shows three menu items under the 'Menu' category:

- Charlas Claras
- Charlas Claras/Charlas
- Charlas Claras/Charlas/Gestionar Charlas

Menu: Charlas Claras:

The screenshot shows the 'Menu Items / Charlas Claras' screen. The menu item details are as follows:

Menu	Charlas Claras	Full Path	Charlas Claras
Parent Menu		Action	
Sequence	10	Web Icon File	
		Web Icon Image	

Below the table, there are tabs for 'Access Rights' and 'Submenus'. A 'Group Name' field is present with the placeholder 'Add a line'.

Menu: Charlas Claras/Charlas:

The screenshot shows the 'Menu Items / Charlas Claras/Charlas' screen. The menu item details are as follows:

Menu	Charlas	Full Path	Charlas Claras/Charlas
Parent Menu	Charlas Claras	Action	Charlas
Sequence	10	Web Icon File	
	<th>Web Icon Image</th> <td></td>	Web Icon Image	

Below the table, there are tabs for 'Access Rights' and 'Submenus'. A 'Group Name' field is present with the placeholder 'Add a line'.

Menu: Charlas Claras/Charlas/Gestionar Charlas:

The screenshot shows the Odoo Settings interface under the General Settings tab. The top navigation bar includes 'Settings', 'General Settings', 'Users & Companies', 'Translations', and 'Technical'. On the right, there are icons for OdooBot (DBlue), a message bubble with '4', and a gear. The main content area displays the 'Menu Items / Charlas Claras/Charlas/Gestionar Charlas' screen. It shows a table with one row for the menu item. The table columns are: Menu (Gestionar Charlas), Parent Menu (Charlas Claras/Charlas), Sequence (10), Full Path (Charlas Claras/Charlas/Gestionar Charlas), Action (Action), Web Icon File, and Web Icon Image. Below the table, tabs for 'Access Rights' and 'Submenus' are visible, along with a note 'Add a line'.

Menu	Gestionar Charlas	Full Path	Charlas Claras/Charlas/Gestionar Charlas
Parent Menu	Charlas Claras/Charlas	Action	Charlas
Sequence	10	Web Icon File	
		Web Icon Image	

Access Rights Submenus

Add a line

Buscamos "lugar":

The screenshot shows the Odoo Settings interface with the 'Menu Items' tab selected. A search bar at the top contains the text 'lugar'. Below the search bar, there are filters for 'Filters', 'Group By', and 'Favorites'. The main list shows two menu items: 'Charlas Claras/Lugares' and 'Charlas Claras/Lugares/Gestionar Lugares', both of which are highlighted with a red box.

Menu: Charlas Claras/Lugares

The screenshot shows the details for the first menu item, 'Lugares'. The table includes columns for 'Menu', 'Parent Menu', 'Sequence', 'Full Path', 'Action', 'Web Icon File', and 'Web Icon Image'. The 'Full Path' column shows 'Charlas Claras/Lugares'. The 'Action' column shows 'Lugares'. The 'Sequence' column shows '10'. The 'Access Rights' and 'Submenus' tabs are visible below the table.

Menu	Lugares	Full Path	Charlas Claras/Lugares
Parent Menu	Charlas Claras	Action	Lugares
Sequence	10	Web Icon File	
		Web Icon Image	

Menu: Charlas Claras/Lugares/Gestionar Lugares

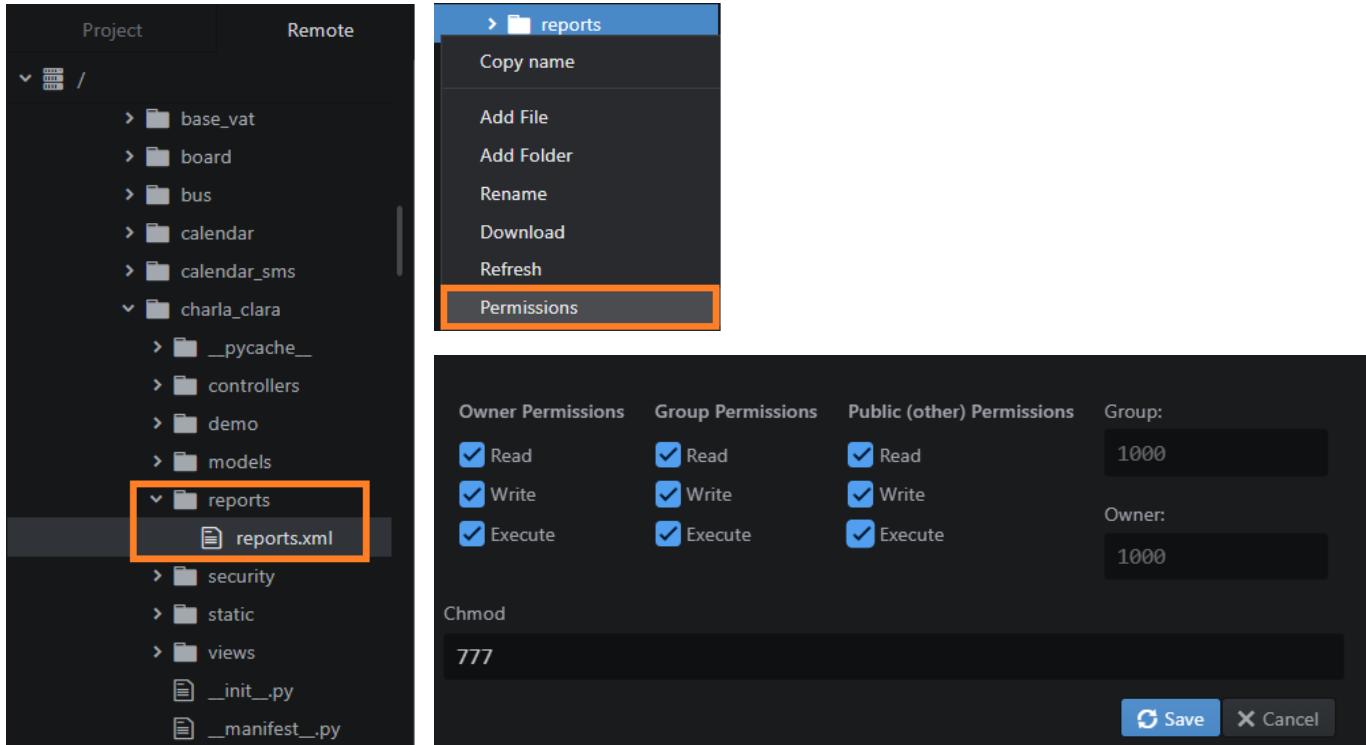
The screenshot shows the details for the second menu item, 'Gestionar Lugares'. The table includes columns for 'Menu', 'Parent Menu', 'Sequence', 'Full Path', 'Action', 'Web Icon File', and 'Web Icon Image'. The 'Full Path' column shows 'Charlas Claras/Lugares/Gestionar Lugares'. The 'Action' column shows 'Lugares'. The 'Sequence' column shows '10'. The 'Access Rights' and 'Submenus' tabs are visible below the table.

Menu	Gestionar Lugares	Full Path	Charlas Claras/Lugares/Gestionar Lugares
Parent Menu	Charlas Claras/Lugares	Action	Lugares
Sequence	10	Web Icon File	
		Web Icon Image	

5. Informes en pdf o web

Lo primero que haremos será crear la siguiente ruta: *charla_clara/reports*
 Y dentro de la carpeta reports crearemos un archivo llamado **reports.xml**

Nos aseguramos de que tanto la carpeta como el archivo cuentan con todos los permisos: **777**



Y utilizamos el siguiente comando para asegurarnos:

```
alejandro@alejandros-server:/$ sudo chmod -R 777 odoo/
```

Importamos la carpeta reports en *__init__.py*

```

__init__.py — J:\DOCUMENTS\DA2D1E\Gestión Empresarial + Desarrollo Web en Entorno Cliente
File Edit View Selection Find Packages Help
Project Remote
192.168.1.55
  charla_clara
    __pycache__
    controllers
    demo
    models
  reports
    reports.xml
    security
    static
    views
  __init__.py
  __manifest__.py

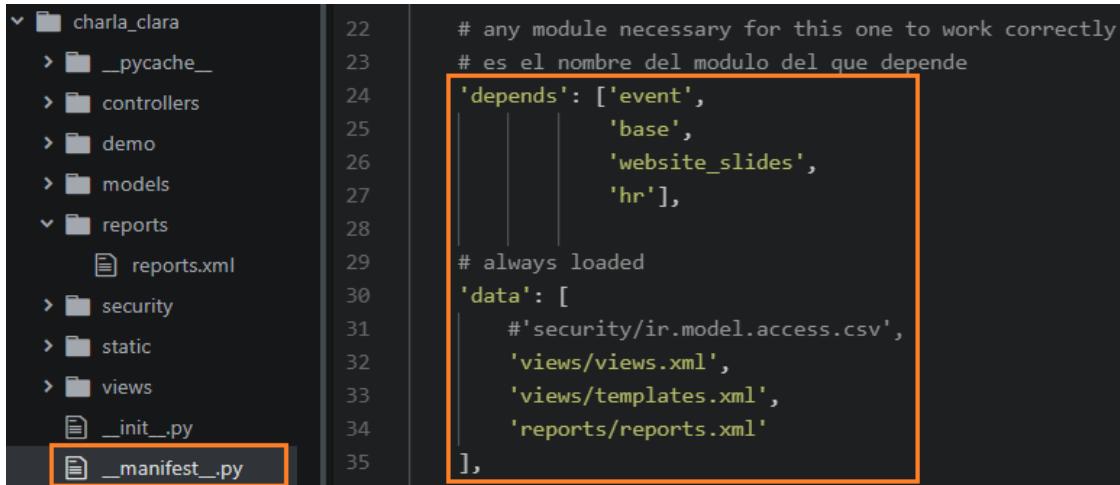
```

```

__init__.py
1 # -*- coding: utf-8 -*-
2
3 from . import controllers
4 from . import models
5 from . import views
6 from . import reports
7

```

Modificamos en el archivo **__manifest__.py** las dependencias y el data que precarga y agregamos/modificamos:

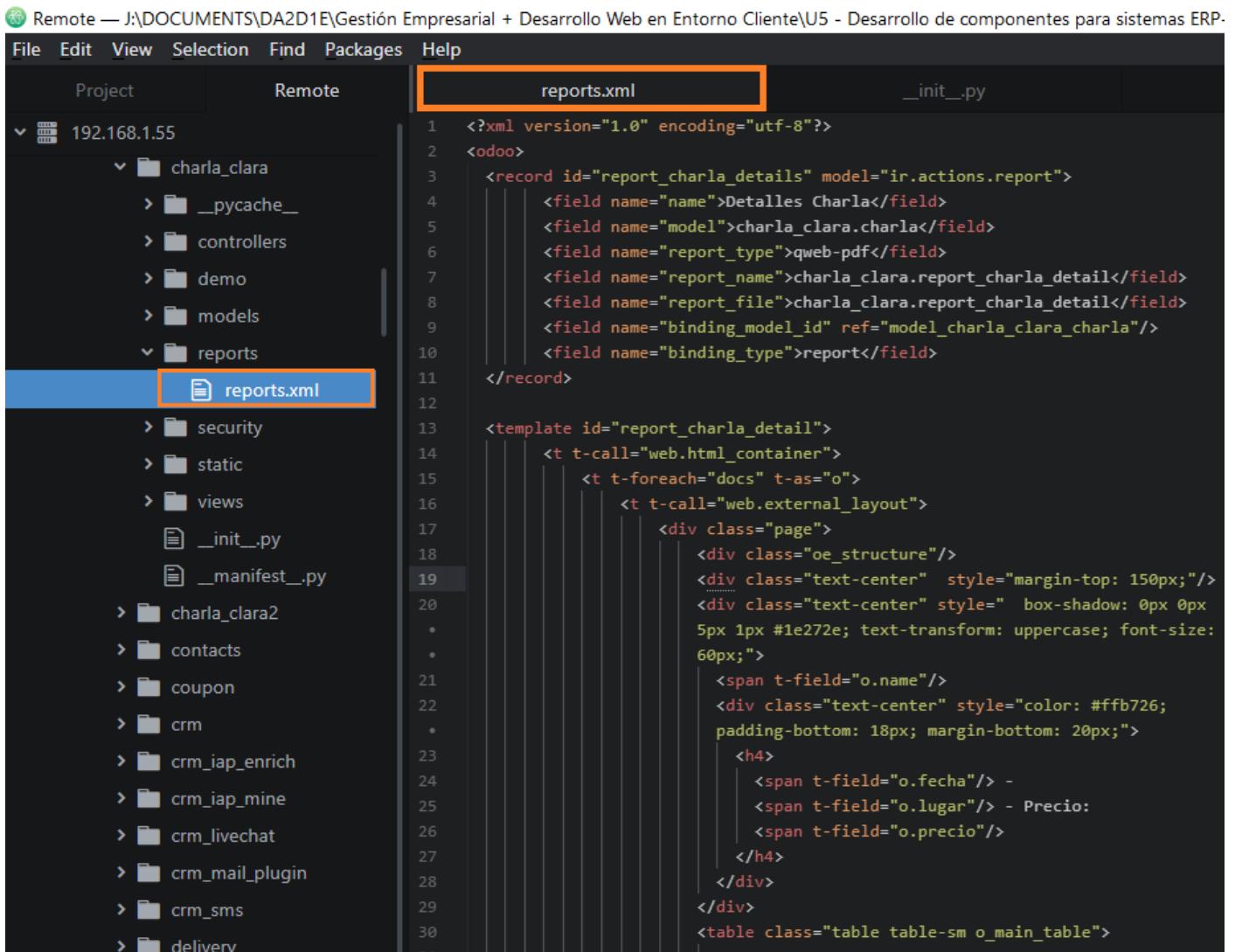


```

22     # any module necessary for this one to work correctly
23     # es el nombre del modulo del que depende
24     'depends': [
25         'event',
26         'base',
27         'website_slides',
28         'hr'],
29
30     # always loaded
31     'data': [
32         #'security/ir.model.access.csv',
33         'views/views.xml',
34         'views/templates.xml',
35         'reports/reports.xml'
36     ],

```

Y ya podemos editar el archivo **reports.xml**



```

<?xml version="1.0" encoding="utf-8"?>
<odoo>
    <record id="report_charla_details" model="ir.actions.report">
        <field name="name">Detalles Charla</field>
        <field name="model">charla_clara.charla</field>
        <field name="report_type">qweb-pdf</field>
        <field name="report_name">charla_clara.report_charla_detail</field>
        <field name="report_file">charla_clara.report_charla_detail</field>
        <field name="binding_model_id" ref="model_charla_clara_charla"/>
        <field name="binding_type">report</field>
    </record>
    <template id="report_charla_detail">
        <t t-call="web.html_container">
            <t t-foreach="docs" t-as="o">
                <t t-call="web.external_layout">
                    <div class="page">
                        <div class="oe_structure"/>
                        <div class="text-center" style="margin-top: 150px;">
                            <div class="text-center" style=" box-shadow: 0px 0px 5px 1px #1e272e; text-transform: uppercase; font-size: 60px;">
                                <span t-field="o.name"/>
                                <div class="text-center" style="color: #ffb726; padding-bottom: 18px; margin-bottom: 20px;">
                                    <h4>
                                        <span t-field="o.fecha"/> -
                                        <span t-field="o.lugar"/> - Precio:
                                        <span t-field="o.precio"/>
                                    </h4>
                                </div>
                            </div>
                        <table class="table table-sm o_main_table">

```

```

32      <table class="table table-sm o_main_table">
33          <!-- In case we want to repeat the header, remove
34              "display: table-row-group" -->
35          <thead style="display: table-row-group">
36              <tr>
37                  <th name="th_description" class="text-left">Tipo de charla</th>
38                  <th name="th_quantity" class="text-left">Ponente</th>
39                  <th name="th_priceunit" class="text-left">Duración(h)</th>
40                  <th name="th_taxes" class="text-left">Asistentes Confirmados</th>
41                  <th name="th_subtotal" class="text-left">Ganancias Totales</th>
42          </tr>
43      </thead>
44
45      <tbody class="sale_tbody">
46          <t>
47              <tr>
48                  <t>
49                      <td name="td_quantity" class="text-left">
50                          <span t-field="o.tipo"/>
51                      </td>
52                      <td name="td_priceunit" class="text-left">
53                          <span t-field="o.ponente"/>
54                      </td>
55                      <td name="td_priceunift" class="text-left">
56                          <span t-field="o.duracion"/>
57                      </td>
58                      <td name="td_priceunifgt" class="text-left">
59                          <span t-field="o.asistentes_confirmados"/>
60                      </td>
61                      <td name="td_priceunit" class="text-left">
62                          <span t-field="o.ganancias_totales"/>
63                          <span t-field="o.currency_id"/>
64                      </td>
65                  </t>
66              </tr>
67          </t>
68      </tbody>
69
70      </table>
71
72          <h5 class="text-left" style="text-decoration:underline;">
73              | Sinopsis
74          </h5>
75          <div class="text-left">
76              | | <span t-field="o.sinopsis"/>
77          </div>
78
79          </div>
80      </t>
81  </t>
82      </template>
83  </odoo>

```

Podemos ver nuestro informe creado en Odoo en *Settings/Technical/Reports*:

The screenshot shows the Odoo web interface with the URL 192.168.1.55:8069/web#cid=1&menu_id=1. The top navigation bar includes links for General Settings, Users & Companies, Translations, Gamification Tools, Technical, and Reports. The Reports section is highlighted with an orange box. A search bar at the top right contains the text 'Buscar'. Below the search bar, there are buttons for 'Actions', 'Actions', and 'Reports'. An orange arrow points from the 'Reports' button to the 'Reports' link in the breadcrumb navigation.

Reports

Report clara

Name	Model Name	Action Type	Template Name	Report Type	Save as Attachment Prefix
Detalles Charla	charla_clara.charla	ir.actions.report	charla_clara.report_charla_detail	PDF	

Reports / Detalles Charla

Detalles Charla

Name	Model Name	Report Type	Template Name	Printed Report Name
Detalles Charla	charla_clara.charla	PDF	charla_clara.report_charla_detail	

Advanced Properties

Group Name

Add a line

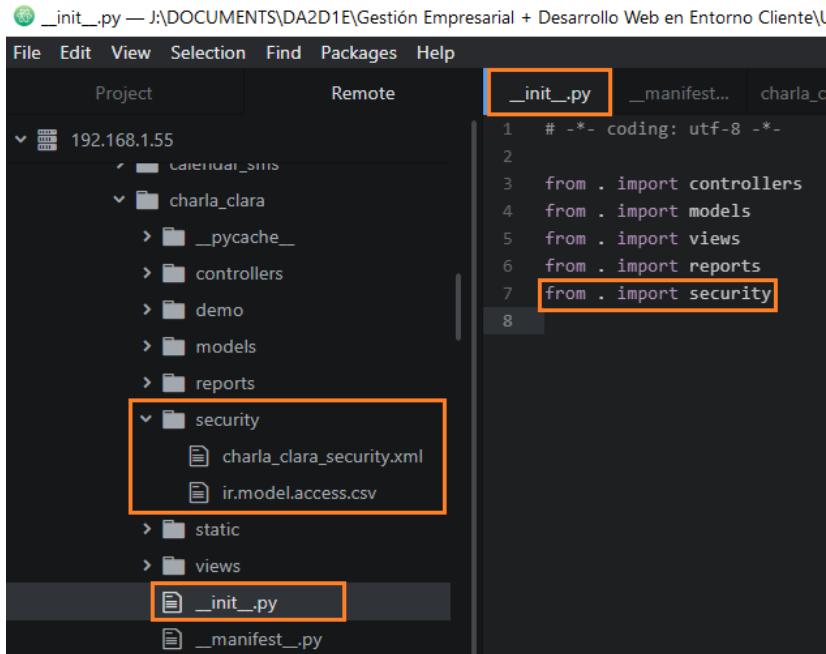
6. Crear 2 grupos de usuarios: administradores y usuarios

Lo primero que haremos será crear la ruta `charla_clara/security` y dentro de esta los archivos `charla_clara_security.xml` e `ir.model.access.csv`

Y concederles todos los permisos a través de la interfaz gráfica de Atom como hemos visto anteriormente o con el comando:

```
alejandro@alejandros-server:/$ sudo chmod -R 777 odoo/
```

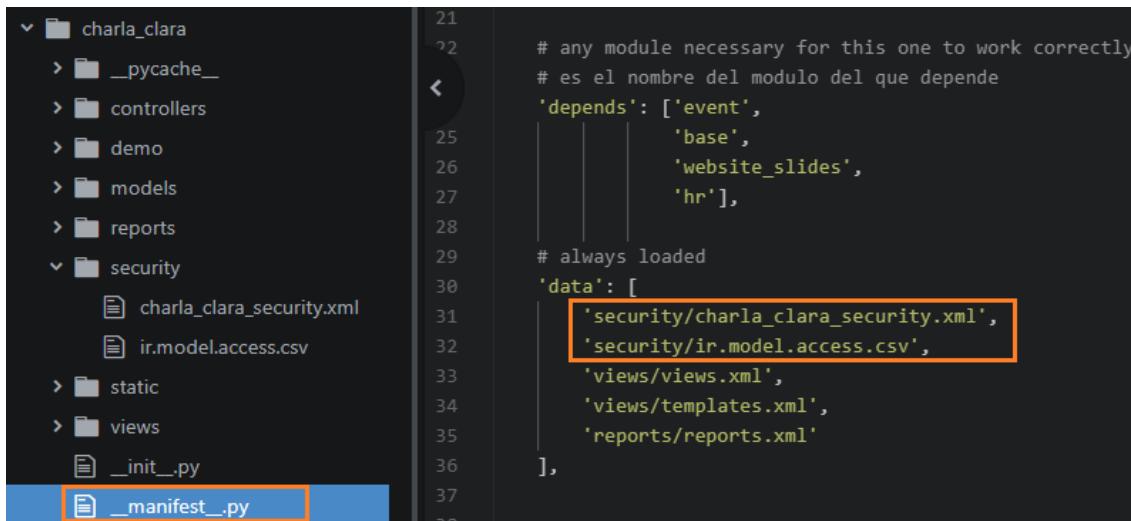
Desinstalamos nuestro módulo Charlas Claras si lo teníamos instalado y modificamos `__init__.py` para importar la carpeta security:



```
File Edit View Selection Find Packages Help
Project Remote
192.168.1.55
    calendar_slides
    charla_clara
        __pycache__
        controllers
        demo
        models
        reports
        security
            charla_clara_security.xml
            ir.model.access.csv
        static
        views
    __init__.py
    __manifest__.py
```

```
__init__.py
1 # -*- coding: utf-8 -*-
2
3 from . import controllers
4 from . import models
5 from . import views
6 from . import reports
7 from . import security
8
```

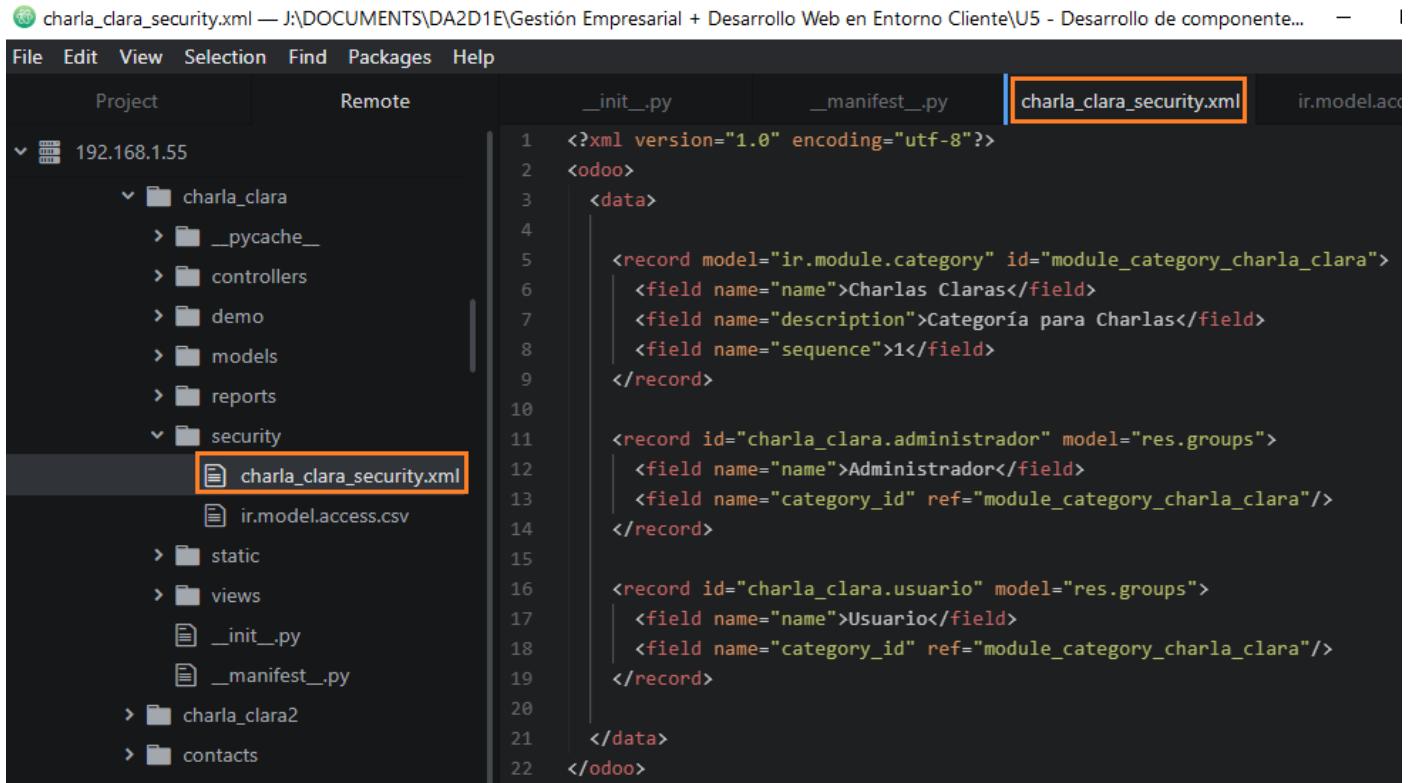
Modificamos también el archivo `__manifest__.py`, la parte de data, para cargar los archivos de la carpeta security:



```
charla_clara
    __pycache__
    controllers
    demo
    models
    reports
    security
        charla_clara_security.xml
        ir.model.access.csv
    static
    views
    __init__.py
    __manifest__.py
```

```
__manifest__.py
21     # any module necessary for this one to work correctly
22     # es el nombre del modulo del que depende
23     'depends': [
24         'event',
25             'base',
26             'website_slides',
27             'hr'],
28
29     # always loaded
30     'data': [
31         'security/charla_clara_security.xml',
32             'security/ir.model.access.csv',
33                 'views/views.xml',
34                 'views/templates.xml',
35                 'reports/reports.xml'
36     ],
37
38
```

El archivo **charla_clara_security.xml** queda de la siguiente manera:

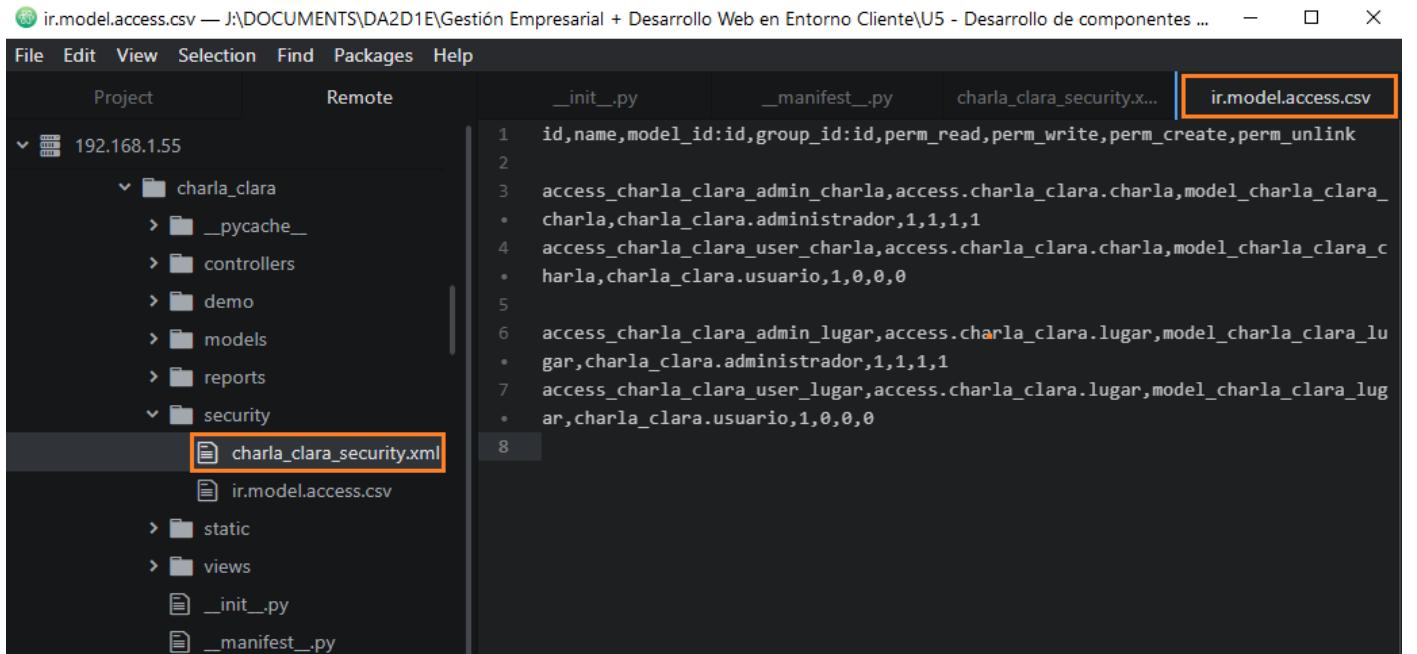


```

<?xml version="1.0" encoding="utf-8"?>
<odoo>
    <data>
        <record model="ir.module.category" id="module_category_charla_clara">
            <field name="name">Charlas Claras</field>
            <field name="description">Categoría para Charlas</field>
            <field name="sequence">1</field>
        </record>
        <record id="charla_clara.administrador" model="res.groups">
            <field name="name">Administrador</field>
            <field name="category_id" ref="module_category_charla_clara"/>
        </record>
        <record id="charla_clara.usuario" model="res.groups">
            <field name="name">Usuario</field>
            <field name="category_id" ref="module_category_charla_clara"/>
        </record>
    </data>
</odoo>

```

Y el archivo **ir.model.access.csv** queda de la siguiente manera:



```

id,name,model_id:id,group_id:id,perm_read,perm_write,perm_create,perm_unlink
access_charla_clara_admin_charla,access.charla_clara.charla,model_charla_clara_
charla,charla_clara.administrador,1,1,1,1
access_charla_clara_user_charla,access.charla_clara.charla,model_charla_clara_c
harla,charla_clara.usuario,1,0,0,0
access_charla_clara_admin_lugar,access.charla_clara.lugar,model_charla_clara_lu
gar,charla_clara.administrador,1,1,1,1
access_charla_clara_user_lugar,access.charla_clara.lugar,model_charla_clara_lug
ar,charla_clara.usuario,1,0,0,0

```

Aquí ya podemos reinstalar el módulo de Charlas Claras.

Para ver los usuarios que hemos creado vamos a *Settings/User & Companies/Users*:

Name	Login	Language	Latest authentication	Status
Admin Blue	bluelaboratory69@gmail.com	English (US)	02/18/2022 18:14:09	Confirmed
Joel Willis	portal	English (US)		Never Connected
Marc Demo	demo	English (US)		Never Connected

Vamos a convertir a **Admin Blue** en **administrador de Charlas Claras**, para ello entramos a su perfil pinchando en su nombre, le damos a editar y seleccionamos Administrador en el apartado Charlas Claras:

<input checked="" type="radio"/> Internal User
<input type="radio"/> Portal
<input type="radio"/> Public

Website	Marketing
eLearning	Events
Website	Administrator
Human Resources	Administration
Employees	Administrator
Charlas Claras	Administrador
Administrador	Usuario

Ahora vamos a **Access Rights** para ver si los permisos se han dado correctamente:

Users / Admin Blue

Name: **Admin Blue**

Email Address: **blueslaboratory69@gmail.com**

Profile picture: A cartoon character holding a flask.

Action: **Access Rights** (highlighted with an orange box)

Other links: Groups (19), Record Rules (46), Employee(s) (1).

En la siguiente captura podemos ver que los permisos se han dado correctamente y que nuestro usuario cuenta con todos los permisos:

<input type="checkbox"/>	Name	Model	Group	Read Access	Write Access	Create Access	Delete Access
<input type="checkbox"/>	access.charla_clara.charla	Charla Clara	Charlas Claras / Administrador	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
<input type="checkbox"/>	access.charla_clara.lugar	Lugar Charla	Charlas Claras / Administrador	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Repetimos un proceso similar con el usuario **Marc Demo** convirtiéndolo en **usuario de Charlas Claras**:

Name: Marc Demo

Email Address: demo

Related Partner: YourCompany, Marc Demo

User Type:

- Internal User
- Portal
- Public

Website	Marketing
eLearning	Events
Website	User
Human Resources	Administration
Employees	Administration

Charlas Claras

Administrador	<input type="checkbox"/>	Usuario	<input checked="" type="checkbox"/>
---------------	--------------------------	---------	-------------------------------------

Access Rights: charla

Name	Model	Group	Read Access	Write Access	Create Access	Delete Access
access.charla_clara.cha...	Charla Clara	Charlas Claras / Usua...	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
access.charla_clara.lugar	Lugar Charla	Charlas Claras / Usua...	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Para ver los grupos que hemos creado vamos a *Settings/User & Companies/ Groups*

Groups

Name	Role
Charlas Claras / Administrador	Administrador
Charlas Claras / Usuario	Usuario

Entramos en el grupo **Charlas Claras/Administrador**, donde encontraremos a nuestro usuario *Admin Blue*:

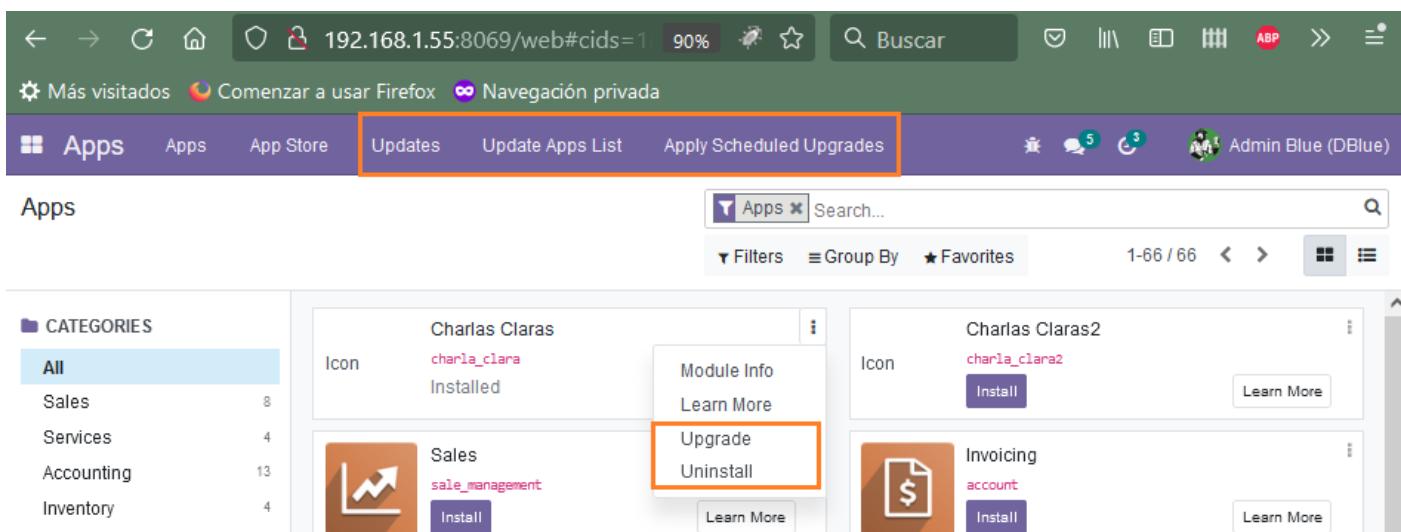
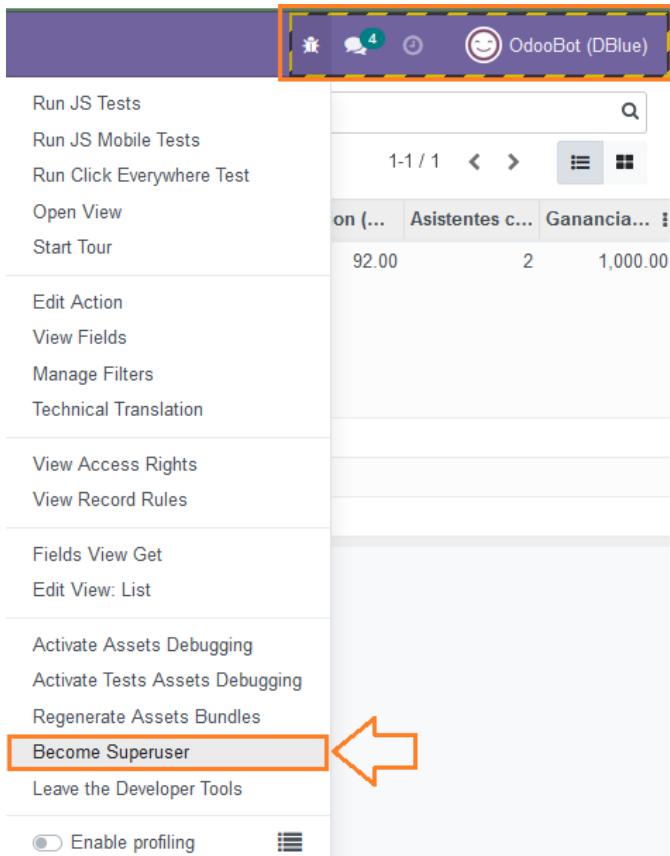
Name	Login	Language	Latest authentication	Status
Admin Blue	blueslaboratory69@gmail.com	English (US)	02/18/2022 18:14:09	Confirmed

Y entramos en el grupo **Charlas Claras/Usuario**, donde encontraremos a nuestro usuario *Marc Demo*:

Name	Login	Language	Latest authentication	Status
Marc Demo	demo	English (US)		Never Connected

7. Ejemplo de funcionamiento

Hay que mencionar que para que se vieran los elementos de menú durante la práctica, las vistas, etc. e ir probando mientras la hacíamos, cuando todavía no se encontraban creados los permisos del punto anterior, en ocasiones es necesario activar el modo superuser, hacer *Updates*, *Update Apps List*, *Apply Scheduled Upgrades*; *Upgrade* o incluso desinstalar y reinstalar el módulo; en especial después de hacer cambios en `__manifest__.py` o `__init__.py`:



Vamos a comprobar el funcionamiento del módulo *Charlas Claras* creado desde el usuario **Admin Blue** teniendo permisos de administrador sobre él ahora que hemos finalizado el proyecto. En la siguiente captura vemos que nuestro módulo *Charlas Claras*, se encuentra instalado y que está visible desde el panel de aplicaciones:

The screenshot shows a web-based application interface. At the top, there's a navigation bar with icons for back, forward, home, search, and other system functions. The URL bar shows '192.168.1.55:8069/web#cid=1'. Below the bar, there's a toolbar with 'Más visitados', 'Comenzar a usar Firefox', and 'Navegación privada'. The main area has a purple header with 'Apps' selected. A sidebar on the left lists 'Discuss', 'Charlas Claras' (which is highlighted with an orange box and has an arrow pointing to it), 'Website', 'eLearning', 'Events', 'Employees', and 'Link Tracker'. The main content area shows a grid of application cards. One card for 'Charlas Claras' is highlighted with an orange box and has an arrow pointing to it. It displays the name 'Charlas Claras', the icon 'charla_clara', and the status 'Installed'. Another card for 'Charlas Claras2' is partially visible. At the bottom right of the main area, there are 'Learn More' and 'Install' buttons.

Entramos en nuestro módulo. Para crear una charla, es necesario disponer de un lugar, campo requerido por el modelo **charla_clara.charla**, por lo que lo primero que haremos será **crear un Lugar** pinchando en el submenú de "Lugares" -> "Gestionar Lugares" y posteriormente en "Create":

The screenshot shows a sub-module of 'Lugares'. The top navigation bar includes 'Charlas Claras', 'Charlas', 'Lugares' (which is highlighted with an orange box and has an arrow pointing to it), and 'Admin Blue (DBlue)'. Below this, there's a sub-navigation with 'Lugares' and 'Create' (also highlighted with an orange box and has an arrow pointing to it). The main content area features a large image of a smiling document icon and the text '¡Crea tu primer lugar!'. At the bottom, there's a call-to-action button labeled 'Gestionar Lugares'.

Rellenamos lo que vendría a ser la **vista Form** de Lugares y creamos el lugar Marte dándole a Save:

The screenshot shows the 'Lugares / New' form. The top navigation bar includes 'Charlas Claras', 'Charlas', 'Lugares', and 'Admin Blue (DBlue)'. Below the navigation, there are 'Save' and 'Discard' buttons, with an orange arrow pointing to the 'Save' button. The main form area has a title 'Marte'. On the left is a large image of the planet Mars. To the right, there are fields for 'Aforo' (with the value '2000') and 'Dispone de parking' (with a checked checkbox).

Para ver nuestro lugar desde la **vista Lista** y la **vista Kanban**, volvemos a Lugares:

Lugares: Vista Lista:

The screenshot shows a software interface for managing locations. At the top, there's a navigation bar with tabs for 'Charlas Claras' and 'Lugares'. The 'Lugares' tab is highlighted with an orange border. Below the navigation is a search bar labeled 'Search...'. To the right of the search bar are buttons for 'Filters', 'Group By', 'Favorites', and page navigation ('1-1 / 1'). On the far right, there are icons for sorting and filtering, with the 'List' icon highlighted by an orange box. The main content area displays a single location entry in a table format. The columns are 'Lugar' (checkbox), 'Aforo' (text input '2,000'), and 'Dispone de parking' (checkbox). The entry 'Marte' is selected, indicated by a checked checkbox in the first column. The entire row for 'Marte' is also highlighted with an orange box.

Lugares: Vista Kanban:

The screenshot shows the same software interface, but the 'Kanban' view is selected. The 'Kanban' icon is highlighted with an orange box. In the Kanban view, the location 'Marte' is represented as a card. The card has a title 'Marte' and contains a large image of the planet Mars. Below the image, there is descriptive text: '- Código Lugar:', '- Aforo: 2,000', and '- Dispone de parking: True'. The entire card is highlighted with an orange box.

Podemos ahora **crear una charla**. Pinchando en el submenú de "Charlas" -> "Gestionar Charlas" y posteriormente en "Create":

The screenshot shows the Odoo web interface for managing presentations ('Charlas'). The top navigation bar has tabs for 'Charlas Claras', 'Charlas', and 'Lugares'. A sub-menu 'Gestionar Charlas' is open, with its 'Create' option highlighted by an orange arrow. Below the menu, there's a search bar and some filtering options. The main content area features a smiling notepad icon and the text '¡Crea tu primera charla!'.

Para ver el funcionamiento de los campos required, la intentamos crear vacía, veremos que no nos deja:

This screenshot shows the 'Charlas / New' form. The 'Fecha' field is empty and highlighted in red. A validation error message box titled 'Invalid fields:' lists four required fields: 'Título', 'Fecha', 'Lugar', and 'Ponente'. The other fields shown are 'Precio' (0.00), 'Moneda' (USD), 'Asistentes confirmados' (0), 'Ganancias Totales' (0.00), and 'Tipo de Charla' (Pública). The 'Save' button is highlighted with an orange arrow.

Rellenamos ahora lo que vendría a ser la **vista Form** de Charlas y creamos nuestra charla dándole a Save:

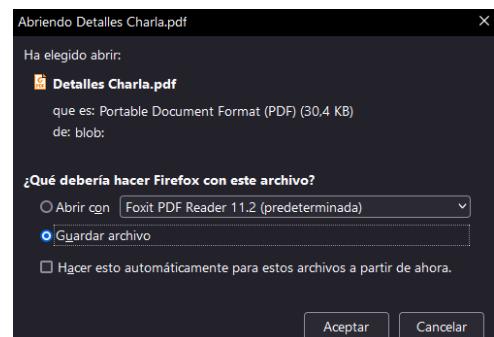
This screenshot shows the 'Charlas / New' form with all required fields filled: 'Fecha' (02/22/2022), 'Lugar' (Marte), 'Duracion (minutos)' (90), 'Ponente' (Admin Blue), and 'Tipo de Charla' (¿Qué charla? *le guiña un ojo*). The 'Save' button is highlighted with an orange arrow. At the bottom of the form, there is a note in a red box: 'La charla de las charlas es la charla final de Odoo, dónde explicaré cómo Odoo me convirtió en lo que soy hoy en día.'

Comprobamos ahora el funcionamiento de crear un informe:

Odoo: La Charla de las Charlas

Fecha	02/22/2022	Precio	\$ 10.00
Lugar	Marte	Moneda	USD
Duracion (minutos)	90.00	Asistentes confirmados	1,000
Ponente	Admin Blue	Ganancias Totales	10,000.00
Tipo de Charla	¿Qué charla? *le guiña un ojo*		

La charla de las charlas es la charla final de Odoo, dónde explicaré cómo Odoo me convirtió en lo que soy hoy en día.



Your logo

ODOO: LA CHARLA DE LAS CHARLAS

02/22/2022 - MARTE - PRECIO: \$ 10.00

Tipo de charla	Ponente	Duración(h)	Asistentes Confirmados	Ganancias Totales
¿Qué charla? *le guiña un ojo*	Admin Blue	90.0	1,000	10,000.0 USD

Sinopsis

La charla de las charlas es la charla final de Odoo, dónde explicaré cómo Odoo me convirtió en lo que soy hoy en día.

Para ver nuestra charla desde la **vista Lista**, la **vista Kanban** y demás, volvemos a Charlas:

Charlas: Vista Lista:

Título	Fecha	Lugar	Tipo de Charla	Precio	Ponente	Sinopsis	Duración (minutos)	Asistentes confirmados	Ganancia	Total
Odoo: La Charla de las Charlas	02/22/2022	Marte	¿Qué charla? *le guiña un ojo*	10.00	Admin Blue	La charla de las charlas es la charla final de Odoo, dónde explicaré cómo Odoo me convirtió en lo que soy hoy en día.	90.00	1,000	10,000	

Charlas: Vista Kanban:

Odoo: La Charla de las Charlas

- Fecha: 02/22/2022
- Lugar: Marte
- Ponente: Admin Blue
- Tipo: ¿Qué charla? *le guiña un ojo*
- Ganancia Totales: 10,000.00

Charlas: Vista Búsqueda - Filtros:

Título	Fecha	Lugar	Tipo de Ch...	Precio	Ponente
Odoo: La C...	02/22/2022	Marte	¿Qué charl...	10.00	Admir...

Precio ≥ 100\$
Precio ≤ 100\$
Duración ≥ 120mins
Duración ≤ 120mins
Charlas del Admin
Add Custom Filter

Charlas: Vista Heredada Búsqueda – Agrupar Por:

Título	Fecha	Lugar	Tipo de Ch...	Precio	Ponente	Sinopsis
Odoo: La C...	02/22/2022	Marte	¿Qué charl...	10.00	Admin Blue	La c...

Group By

1-
Tipo
Lugar
Ponente
Moneda
Fecha
Add Custom Group

Ahora comprobamos el funcionamiento de nuestro módulo como **usuario**, para ello vamos a *Settings/User & Companies/Users:*

Name	Login	Language	Latest authentication	Status
Admin Blue	blueslaboratory69@gmail.com	English (US)	02/18/2022 18:14:09	Confirmed
Joel Willis	portal	English (US)		Never Connected
Marc Demo	demo	English (US)		Never Connected

Vamos a convertir a *Admin Blue* en **usuario** de **Charlas Claras**, para ello entramos a su perfil pinchando en su nombre, le damos a editar y seleccionamos Usuario en el apartado Charlas Claras:

Website	Marketing
eLearning	Manager
Website	Events
Human Resources	
Employees	Administrator
Administration	
Administrador	Usuario

Refrescamos la página y volvemos a nuestro módulo para comprobar que el usuario sólo tiene permisos de lectura tanto en Lugares como en Charlas:

The screenshot shows two Odoo module pages. The top page is 'Lugares' (Places) with a table showing one record: 'Lugar' (Place) Marte, 'Aforo' (Capacity) 2,000, and 'Dispone de parking' (Has parking). The bottom page is 'Charlas' (Speeches) with a table showing one record: 'Lugar' (Place) Marte, 'Aforo' (Capacity) 2,000, and 'Dispone de parking' (Has parking). Both pages have an orange box around the 'Gestionar' (Manage) button in the top navigation bar.

Si entramos en Charla, vemos que el permiso de lectura permite imprimir un informe en pdf:

The screenshot shows the 'Odoo: La Charla de las Charlas' details page. It displays the following information:

- Fecha: 02/22/2022
- Lugar: Marte
- Ponente: Admin Blue
- Tipo: ¿Qué charla? *le guiña un ojo*
- Ganancia Totales: 10,000.00

At the top, there is a 'Print' button and a 'Detalles Charla' button, both highlighted with orange boxes. Below the table, there is a '1 / 1' page indicator and navigation arrows.

Fecha	02/22/2022 <th>Precio</th> <td>\$ 10.00</td>	Precio	\$ 10.00
Lugar	Marte	Moneda	USD
Duración (minutos)	90.00	Asistentes confirmados	1,000
Ponente	Admin Blue	Ganancias Totales	10,000.00
Tipo de Charla	¿Qué charla? *le guiña un ojo*		

La charla de las charlas es la charla final de Odoo, dónde explicaré cómo Odoo me convirtió en lo que soy hoy en día.

A screenshot of a PDF file download dialog box titled 'Abriendo Detalles Charla.pdf'. The box contains the following information:

- Ha elegido abrir:
- Detalles Charla.pdf**
- que es: Portable Document Format (PDF) (30.4 KB)
- de: blob:

Below this, there is a question: '¿Qué debería hacer Firefox con este archivo?' with two options:

- Abrir con Foxit PDF Reader 11.2 (predeterminada)
- Guardar archivo

At the bottom, there is a checkbox: 'Hacer esto automáticamente para estos archivos a partir de ahora.' and two buttons: 'Aceptar' and 'Cancelar'.

FIN