

TEMA 5: PART 2

Aquesta tasca aten a el repte de la creació de nous components a l'Odoo. Aplicant solució a la capacitat de el propi odoo, de creacio de moduls custimitzats, es a dir, de adaptats a les nostres necessitats.

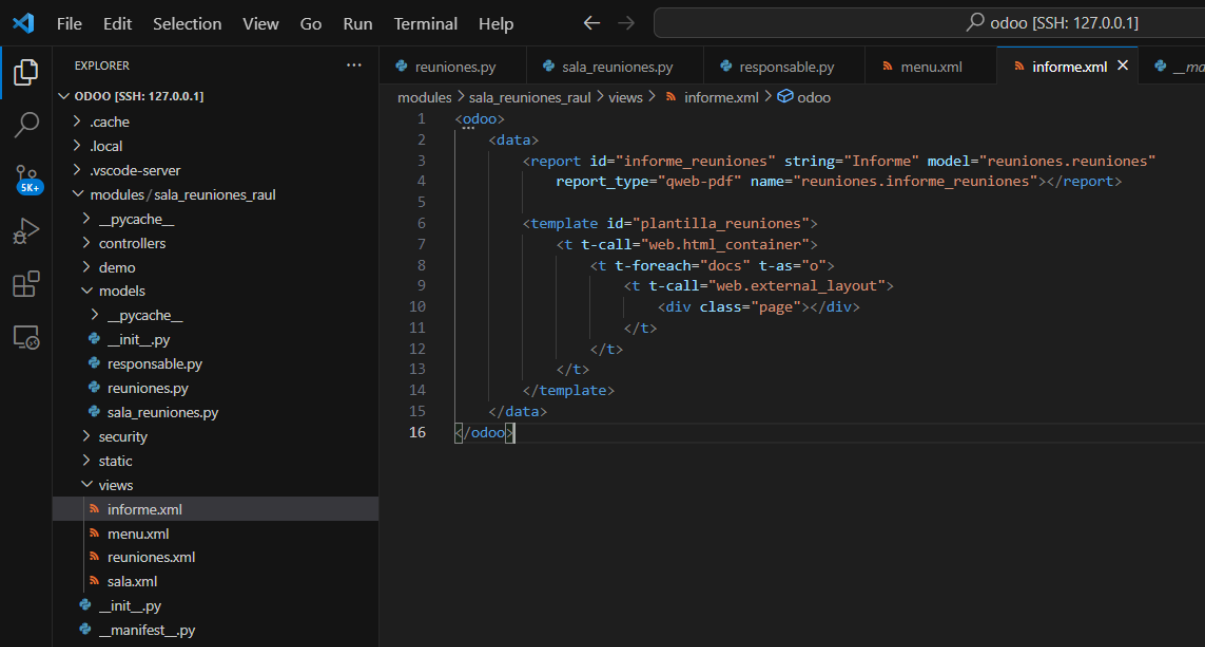
per a realitzar aquestes tasques, abanda de haber arribat a a tindre odoo configurat, modul custom creat, amb diversos models, vistes tree, form, accions de finestra etc, deurem tindre asolits conceptes, de com i perque estem colocalant cada nom.

Tambè hi podem recorrer a moduls ja creats com a exemple, veure el foro de la pàgina de Odoo, i utilitzar bones practiques per a el naming.

Atenint a aquest requeriment:

“Necesitamos crear un informe para el modelo de Reunión: Para cada reunión, tiene que mostrar el nombre de la reunión y su responsable, la fecha de inicio y de final, y enumerar los asistentes a dicha reunión. ”

-construim l'informe amb els camps corresponents.



The screenshot shows a code editor with the following XML content in the 'informe.xml' file:

```
1 <odoo>
2   <data>
3     <report id="informe_reuniones" string="Informe" model="reuniones.reuniones"
4       report_type="qweb-pdf" name="reuniones.informe_reuniones"></report>
5
6     <template id="plantilla_reuniones">
7       <t t-call="web.html_container">
8         <t t-foreach="docs" t-as="o">
9           <t t-call="web.external_layout">
10             <div class="page"></div>
11           </t>
12         </t>
13       </t>
14     </template>
15   </data>
16 </odoo>
```

-Finalment prenent una forma final que mostra els camps de reunions en aquest element tipus page. Tambè es important saber que el name, i el

identificador de la template han de correspondre amb el mateix nom.

```
modules > sala_reuniones_raul > views > informe.xml > odoo > data
1 <?xml version="1.0" encoding="UTF-8"?>
2 <odoo>
3 <data>
4 <report id="informe_reuniones" string="Informe" model="sala_reuniones_raul.reuniones" report_type="qweb-pdf" name="plantilla_reuniones"></report>
5 <template id="plantilla_reuniones">
6 <t t-call="web.html_container">
7 <t t-foreach="docs" t-as="o">
8 <t t-call="web.external_layout">
9 <div class="page">
10 <p t-field="o.name">nombre: </p>
11 <p t-field="o.fechaInicio">fecha de inicio: </p>
12 <p t-field="o.duracion">duracion: </p>
13 <p t-field="o.asientos">asientos: </p>
14 <p t-field="o.sala">sala: </p>
15 <p t-field="o.responsable">responsable: </p>
16 <p t-field="o.asistentes">asistentes: </p>
17 </div>
18 </t>
19 </t>
20 </t>
21 </template>AA
22 </data>
23 </odoo>
```

-Per a mostrar el informe, hi deurem de tindre la colocació de les vistes dintre de data, ordenades de forma, que no es xafen.

```
File Edit Selection View Go Run Terminal Help
EXPLORER
ODOO [SSH: 127.0.0.1]
> .cache
> .local
> .vscode-server
5K+
modules / sala_reuniones_raul
> __pycache__
> controllers
> demo
models
> __pycache__
> __init__.py
> responsable.py
> reuniones.py
> sala_reuniones.py
> security
> static
views
  informe.xml
  menu.xml
  reuniones.xml
  sala.xml

reuniones.py sala_reuniones.py responsable.py
modules > sala_reuniones_raul > __manifest__.py
1 # -*- coding: utf-8 -*-
2
3 'name': "salaReunionesRaul",
4 'summary': ""
5 | act1"",
6 'description': ""
7 | Testing description
8 "",
9 'author': "Raul Sales",
10 'website': "https://www.yourcompany.com",
11 'category': 'Uncategorized',
12 'version': '0.1',
13 'depends': ['base'],
14 'data': [
15 | 'views/sala.xml',
16 | 'views/reuniones.xml',
17 | 'views/menu.xml',
18 | 'views/informe.xml'
19 ],
20 'demo': [
21 | 'demo/demo.xml',
22 ],
23
```

ara que ja hi tenim la estructura de el informe, anem a complir la tasca de enumerar seients:

```
@api.depends('asientos','asistentes')
def _calc_asientos_ocup(self):
    for record in self:
        if record.asientos:
            record.asientosOcupados = (len(record.asistentes) / record.asientos)*100
        else :
            record.asientosOcupados = 0
```

-atentint a:Añadir una función en la clase para avisar sobre valores no válidos, como por ejemplo un número negativo de asientos o más participantes que no asientos.

a el model de reunions, crearem esta funció, el cual s'encarrega de validar que el valor es matjor que 0, per a indicar que hi han seints i asistentes:

```
@api.constrains('asientos','asistentes')
def _validador_asi(self):
    for record in self:
        if record.asientos < 0:
            raise ValidationError("el numero de asientos no puede ser negativo.")
        if len(record.asistentes) > record.asientos:
            raise ValidationError("el numero de asistentes no puede ser mayor los asientos")
```

atenint a: Añade una función que comprueba que el responsable no está presente en los asistentes de su propia reunión. (función constraint).

-Validem que un responsable no siga asistent.

```
@api.constrains('responsable','asistentes')
def _validador_responsable(self):
    for record in self:
        if record.responsable and record.responsable in record.asistentes:
            raise ValidationError("no puede ser responsable un asistente")
```

-Quedaría així el model de reunions, amb els camps relacionats i les seues funcions.

```

reuniones.py X sala_reuniones.py _init_.py responsable.py controllers.py menu.xml sala.xml reunion
modules > sala_reuniones_raul > models > reuniones.py
1 from odoo import models, fields, api
2 from odoo.exceptions import ValidationError
3
4 class Reuniones(models.Model):
5     _name = "sala_reuniones_raul.reuniones"
6     #valor per defecte
7     nombre = fields.Char(string="name", required=True)
8     fechaInicio = fields.Date(string="date", required=True, default=fields.Date.today())
9     duracion = fields.Integer(string="duracion", required=True)
10    asientos = fields.Integer(string="asientos", required=True)
11    sala = fields.Many2one("sala_reuniones_raul.sala_reuniones", string="sala", ondelete='cascade')
12    responsable = fields.Many2one("res.partner", string="responsable")
13    asistentes = fields.Many2many("res.partner", string="asistentes")
14    asientosOcupados = fields.Float(string="percentatge", compute="_calc_asientos_ocup", store=True)
15
16    @api.depends('asientos', 'asistentes')
17    def _calc_asientos_ocup(self):
18        for record in self:
19            if record.asientos:
20                record.asientosOcupados = (len(record.asistentes) / record.asientos)*100
21            else:
22                record.asientosOcupados = 0
23
24
25    @api.constrains('asientos', 'asistentes')
26    def _validador_asi(self):
27        for record in self:
28            if record.asientos < 0:
29                raise ValidationError("el numero de asientos no puede ser negativo.")
30            if len(record.asistentes) > record.asientos:
31                raise ValidationError("el numero de asistentes no puede ser mayor los asientos")
32
33    @api.constrains('responsable', 'asistentes')
34    def _validador_responsable(self):
35        for record in self:
36            if record.responsable and record.responsable in record.asistentes:
37                raise ValidationError("no puede ser responsable un asistente")
38

```

-Atenint a:

Utilizando la herencia, modificaremos el modelo res.partner existente, para añadir un nuevo campo booleano para saber si este es Personal responsable o no y añadiremos también un campo many2many que se corresponda con la relación reunion-partner para ver las reuniones de las cuales este contacto es la persona responsable.

Creació de model que hereda de res.partner amb dos camps relacionats que utilitzarem per a saber dades de els responsables.

```

reuniones.py sala_reuniones.py _init_.py responsable.py X controllers.py menu.xml sala.xml reunion
modules > sala_reuniones_raul > models > responsable.py
1 from odoo import models, fields, api
2
3 class Responsable(models.Model):
4     _inherit = 'res.partner'
5
6     metodo = fields.Boolean(string="Es responsable ?")
7     reuniones_responsable = fields.Many2many("sala_reuniones_raul.reuniones", string="Reuniones como responsable")

```

-Atenint a:

Utilizando también la herencia de vistas, muestra estos campos en el formulario de res.partner y además deberías hacer que aparezca un nuevo elemento de menú que dependa de Gestión de reuniones para hacer modificaciones en los Contactos sin tener que salir del módulo.

-En aquest cas, si hi mirem com es el .xml de els contactes, hi podem veure que el que volem afegir es una nova page, amb la informació de els camps de el model. Dada (Es pot donar format amb widgets a la vista per donar format, pero ens dona el handicap de no poder veure la tabla amb les dades)

```
Ajustes  Opciones generales  Usuarios y compañías  Traducciones  Herramientas de ludificación  Técnico

Vistas / res.partner.form

152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191

</div>
</div>
</group>
<field name="email" widget="email"/>
<field name="phone" widget="phone"/>
<field name="mobile" widget="phone"/>
<field name="company_id" invisible="1"/>
</group>
</group>
<field name="comment" placeholder="Internal notes..." nolabel="1" colspan="2"/>
</group>
<field name="lang" invisible="True"/>
<field name="user_id" invisible="True"/>
</sheet>
</form>
</field>
</page>
<page name="sales_purchases" string="Sales & Purchase">
  <group name="container_row_2">
    <group string="Sales" name="sale" priority="1">
      <field name="user_id" widget="many2one_avatar_user" domain="['!share', '=', False]"/>
    </group>
    <group string="Purchase" name="purchase" priority="2">
    </group>
    <group name="misc" string="Misc">
      <field name="company_registry" attrs="['invisible': [('parent_id', '=', False)]]"/>
      <field name="ref" string="Reference"/>
      <field name="company_id" groups="base.group_multi_company" options="{'no_create': True}" attrs="['readonly': [('parent_id', '=', False)]]"/>
      <field name="industry_id" attrs="['invisible': [('is_company', '=', False)]]" options="{'no_create': True}"/>
    </group>
  </group>
</page>
<page name="internal_notes" string="Internal Notes">
  <field name="comment" placeholder="Internal notes...">
</page>
</notebook>
</sheet>
</form>
```

-Com es veu, hi volem generar una nova page, mostrant el camp que indica si es asistent, o no i les reunions. A part de els metodes que hem creat, també funcionalitats sobre les reunions de les que es responsable utilitzant domains.

```

modules > sala_reuniones_raul > views > responsable.xml > odoo > data > record
1  <?xml version="1.0" encoding="UTF-8"?>
2  <odoo>
3      <data>
4          <record id="action_res_partner" model="ir.actions.act_window">
5              <field name="name">Responsable</field>
6              <field name="res_model">res.partner</field>
7              <field name="view_mode">form</field>
8          </record>
9          <record id="view_partner_form_inherit" model="ir.ui.view">
10             <field name="name">res.partner.form.inherit</field>
11             <field name="model">res.partner</field>
12             <field name="inherit_id" ref="base.view_partner_form" />
13             <field name="arch" type="xml">
14                 <xpath expr="//page[@name='sales_purchases']" position="after">
15                     <page string="Responsable">
16                         <group>
17                             <field name="metodo" />
18                             <field name="reuniones_responsable" widget="many2many_tags" />
19                         </group>
20                     </page>
21                 </xpath>
22             </field>
23         </record>
24     </data>
25 </odoo>

```



-aquest seria el element de menu que depèn de gestionar reuniones.

```

modules > sala_reuniones_raul > views > menu.xml > ...
1  <?xml version="1.0" encoding="UTF-8"?>
2  <odoo>
3      <data>
4          <menuitem id="main_menu" name="Gestión de las salas de reuniones"/>
5          <menuitem id="submenu_salas" name="Salas" parent="main_menu" action="sala_reuniones_raul.action_sala_reuniones"/>
6          <menuitem id="submenu_reuniones" name="Reuniones" parent="main_menu" action="action_reuniones"/>
7          <menuitem id="menu_reuniones_contactos" name="Contactos" parent="main_menu" action="action_res_partner"/>
8      </data>
9  </odoo>
10

```

-visual i finalment es veuria així:

Nuevo  

Acción Nuevo

p. ej. Brandon Freeman

Nombre de la empresa...

Calle... Puesto de trabajo ? Por ejemplo, Director de Ventas

Calle 2...

Estado C.P. Ciudad Teléfono ?

País Móvil ?

NIF ? Por ejemplo, ESA00000000 Correo electrónico ?

Sitio web ? e.j. https://www.odoo.com

Título ? Por ejemplo, Señor

Etiquetas ? e.g. "B2B", "VIP", "Consulting", ...

Contactos y direcciones Venta y compra Responsable Facturación / Contabilidad Notas internas

Es responsable ? ? ☐

Reuniones como responsable ?

na...	date	dura...	asien...	sala	respons...	asistentes	percent...	Acti...
Agregar línea								

atentint a:

Debemos restringir los datos de entrada al campo responsable con un dominio en campos relacionales: Al seleccionar un responsable para una reunión, solo aquellos marcados como responsables (elementos de la tabla res.partner con el campo responsable configurado como verdadero deberán estar visibles)

-Utilitzant el domain, amb el operador que hi volem, podem aconseguir que sols es que hi siguen apareguen. D'aquesta forma ahorrant-mos metodes engorrosos.

```

1  from odoo import models, fields, api
2  from odoo.exceptions import ValidationError
3
4  class Reuniones(models.Model):
5      _name = "sala_reuniones_raul.reuniones"
6      #valor per defecte
7      name = fields.Char(string="name",required=True)
8      fechaInicio = fields.Date(string="date",required=True,default=fields.Date.today())
9      duracion = fields.Integer(string="duracion", required=True)
10     asientos = fields.Integer(string="asientos", required=True)
11     sala = fields.Many2one("sala_reuniones_raul.sala_reuniones",string="sala", ondelete='cascade')
12     responsable = fields.Many2one("res.partner",string="responsable", domain=[("metodo", "=", True)])
13     asistentes = fields.Many2many("res.partner", string="asistentes")
14     asientosOcupados = fields.Float(string="percentatge", compute="_calc_asientos_ocup", store=True)
15     activos = fields.Boolean()
16

```

atenint a: Definir un wizard para el modelo de reunión: Crear un modelo de asistente con una relación many2one con el modelo de reunión y una relación many2many con el modelo de partner.

-

```
from odoo import models, fields, api

class Wizard(models.TransientModel):
    _name = 'sala_reuniones_raul.wizard'

    reuniones=fields.Many2one('sala_reuniones_raul.reuniones', default=_perdefecte, string="Reunion", required="True")
    partner = fields.Many2many('res.partner', string="Asistentes")
```

atenint a: Definir una vista de formulario para el asistente (wizard). Añadir la acción para lanzarla en el contexto de la reunión, tendremos que definir un valor predeterminado para el campo de reunión en el wizard (debeis recuperar el valor de la reunión actual).

```
<odoo>
<data>
<record id="view_meeting_wizard_form" model="ir.ui.view">
  <field name="name">sala_reuniones_raul.wizard.form</field>
  <field name="model">sala_reuniones_raul.wizard</field>
  <field name="arch" type="xml">
    <form string="Reuniones ">
      <group>
        <field name="reuniones"/>
        <field name="partner"/>
      </group>
      <footer>
        <button name="launch" string="Register Attendees" type="object" class="btn-primary"/>
        <button string="Cancel" class="btn-secondary" special="cancel"/>
      </footer>
    </form>
  </field>
</record>
```

- i per a recuperar el valor actual utilitzarem aquesta funcio 👍

```
modules > sala_reuniones_raul > models > wizard.py
1  from odoo import models, fields, api
2
3  class Wizard(models.TransientModel):
4      _name = 'sala_reuniones_raul.wizard'
5
6      #funcio que pasa el modul actiu
7      def _perdefecte(self):
8          return self.env['sala_reuniones_raul.reuniones'].browse(self._context.get('active_ids'))
9
10     #crida a la funcio per defecte per a afegir valor per defecte
11     reuniones=fields.Many2one('sala_reuniones_raul.reuniones', default=_perdefecte, string="Reunion", required="True")
12     partner = fields.Many2many('res.partner', string="Asistentes")
13
```

-atenint a: Registrar los asistentes a la reunión con un botón en el wizard e implementar también el poder cancelar.


```

1 <odoo>
2 <data>
3 <record id="view_meeting_wizard_form" model="ir.ui.view">
4 <field name="name">sala_reuniones_raul.wizard.form</field>
5 <field name="model">sala_reuniones_raul.wizard</field>
6 <field name="arch" type="xml">
7 <form string="Reuniones ">
8 <group>
9 <field name="reuniones"/>
10 <field name="partner"/>
11 </group>
12 <footer>
13 <button name="launch" string="Register Attendees" type="object" class="btn-primary"/>
14 <button string="Cancel" class="btn-secondary" special="cancel"/>
15 </footer>
16 </form>
17 </field>
18 </record>

```

```

modules > sala_reuniones_raul > models > wizard.py
1 from odoo import models, fields, api
2
3 class Wizard(models.TransientModel):
4     _name = 'sala_reuniones_raul.wizard'
5
6     #funcio que pasa el modul actiu
7     def _perdefecte(self):
8         return self.env['sala_reuniones_raul.reuniones'].browse(self._context.get('active_ids'))
9
10    #crida a la funcio per defecte per a afegir valor per defecte
11    reuniones=fields.Many2one('sala_reuniones_raul.reuniones', default=_perdefecte, string="Reunion", required=True")
12    partner = fields.Many2many('res.partner', string="Asistentes")
13
14    # special=calcel de el xml es una funcio especial que cancela (funcio nativa)
15    def launch(self):
16        reuniones = self.reuniones
17        reuniones.write({"asistentes" : [(6, 0, self.partner.ids)]})
18        return {"type": "ir.actions.act_window"}
19
20

```

-atentint a: Para lanzar el wizard, tenéis que hacer que aparezca en el modelo de reunión desde el desplegable acción del propio odoo.

per a complir , acudirem a l'odoo a una vista de per a vore el identificador de la acció, de forma que a el boto d'accions assignem l'acció que volem fer:

```

1 <odoo>
2 <data>
3 <record id="view_meeting_wizard_form" model="ir.ui.view">
4 <field name="name">sala_reuniones_raul.wizard.form</field>
5 <field name="model">sala_reuniones_raul.wizard</field>
6 <field name="arch" type="xml">
7 <form string="Reuniones ">
8 <group>
9 <field name="reuniones"/>
10 <field name="partner"/>
11 </group>
12 <footer>
13 <button name="launch" string="Register Attendees" type="object" class="btn-primary"/>
14 <button string="Cancel" class="btn-secondary" special="cancel"/>
15 </footer>
16 </form>
17 </field>
18 </record>
19 <record id="action_open_meeting_wizard" model="ir.actions.act_window">
20 <field name="name">Open Meeting Wizard</field>
21 <field name="type">ir.actions.act_window</field>
22 <field name="res_model">sala_reuniones_raul.wizard</field>
23 <field name="view_mode">form</field>
24 <field name="target">new</field>
25 <field name="binding_model_id" ref="model_sala_reuniones_raul_reuniones"/>
26 </record>
27 </data>
28 </odoo>

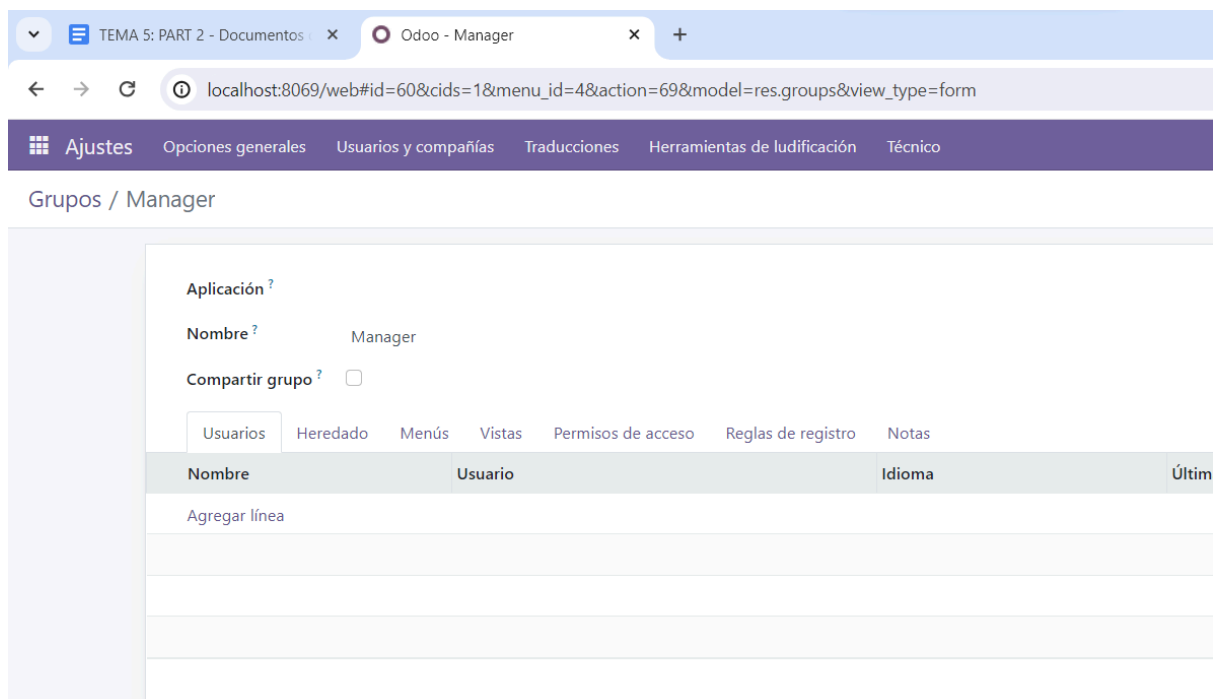
```

-I finalment atent a :Añadir control de acceso a través de archivos de datos al módulo. Deberéis crear un grupo Manager con acceso completo a todos los modelos del módulo de las reuniones y el resto de los usuarios solo puedan leer los datos, esto será aplicable a los menús y al wizard creados en el módulo.

- si utilitzem aquest security.xml, estem creant un grup anomenat manager, ara posteriorment hi vorem com després de afegir-lo al manifest ens apareix el grup:

```
modules > sala_reuniones_raul > security > security.xml > ...
1  <odoo>
2  <data noupdate="1">
3      <record id="sala_reuniones_raul.manager" model="res.groups">
4          <!--Nom de el grup-->
5          <field name="name">Manager</field>
6          <!-- <field name="category_id" ref="..." /> -->
7          <!-- <field name="implied_ids" eval="..." /> -->
8          <!-- <field name="users" eval="..." /> -->
9      </record>
10 </data>
11 </odoo>
```

-Com es veu, apareix el grup creat:



-i aquest i seria el fitxer de seguretat csv, el cual a la primera linea ens explica com es el ordre de colocacio per a indicar permis.

En aquest cas anem a donar permis a l'usuari base, i manager per a actuar sobre reunions i sales amb permis total, i per a el usuari base, permis de lectura.

```
modules > sala_reuniones_raul > security > ir.model.access.csv
1 id,name,model_id:id,group_id:id,perm_read,perm_write,perm_create,perm_unlink
2 base_sala_reuniones_raul_sala_reuniones_raul,sala_reuniones_raul.reuniones,model_sala_reuniones_raul_reuniones,base.group_user,1,0,0,0
3 base_sala_reuniones_raul_sala_reuniones_raul,sala_reuniones_raul.sala_reuniones,model_sala_reuniones_raul_sala_reuniones,base.group_user,1,0,0,0
4 manager_sala_reuniones_raul_manager,sala_reuniones_raul.reuniones,model_sala_reuniones_raul_reuniones,sala_reuniones_raul.manager,1,1,1,1
5 manager_sala_reuniones_raul_manager,sala_reuniones_raul.sala_reuniones,model_sala_reuniones_raul_sala_reuniones,sala_reuniones_raul.manager,1,1,1,1
```

-Aquest és el fitxer de seguretat, modificat amb els requeriments que hi volem complir, agent afegit el wizard també amb els permisos necessaris

```
modules > sala_reuniones_raul > security > ir.model.access.csv
1 id,name,model_id:id,group_id:id,perm_read,perm_write,perm_create,perm_unlink
2 base_sala_reuniones_raul_sala_reuniones_raul,sala_reuniones_raul.reuniones,model_sala_reuniones_raul_reuniones,base.group_user,1,0,0,0
3 base_sala_reuniones_raul_sala_reuniones_raul,sala_reuniones_raul.sala_reuniones,model_sala_reuniones_raul_sala_reuniones,base.group_user,1,0,0,0
4 manager_sala_reuniones_raul_manager,sala_reuniones_raul.reuniones,model_sala_reuniones_raul_reuniones,manager,1,1,1,1
5 manager_sala_reuniones_raul_manager,sala_reuniones_raul.sala_reuniones,model_sala_reuniones_raul_sala_reuniones,manager,1,1,1,1
6 base_wizard,sala_reuniones_raul.wizard,model_sala_reuniones_raul_wizard,base.group_user,1,0,0,0
7 manager_wizard,sala_reuniones_raul.wizard,model_sala_reuniones_raul_wizard,manager,1,1,1,1
```

-Afegim les restriccions a les accions de menu, sols el grup manager pot veure aquesta configuració.

```
modules > sala_reuniones_raul > views > menu.xml
1 <?xml version="1.0" encoding="UTF-8"?>
2 <odoo>
3 <data>
4 <menuitem id="main menu" name="Gestión de las salas de reuniones"/>
5 <menuitem id="submenu_salas" name="Salas" parent="main menu" groups="manager,base.group_user" action="sala_reuniones_raul.action_sala_reuniones"/>
6 <menuitem id="submenu_reuniones" name="Reuniones" parent="main menu" groups="manager" action="action_reuniones"/>
7 <menuitem id="submenu_contactos" name="Contactos" parent="main menu" groups="manager" action="action_res_partner"/>
8 <!-- <menuitem id="submenu_responsable" name="Responsables" parent="main menu" action="action_responsable"/> -->
9 </data>
10 </odoo>
11
```

- assignem a el nostre usuari a el grup, per a que hi puga veure el desplegable i finalment els usuari amb grup manager podrà realitzar tot tipus d'accions sobre les sales, reunions, i responsables. Pero si hi afegim a el grup com en les sales a el usuari base, sols pot llegir-ho.

