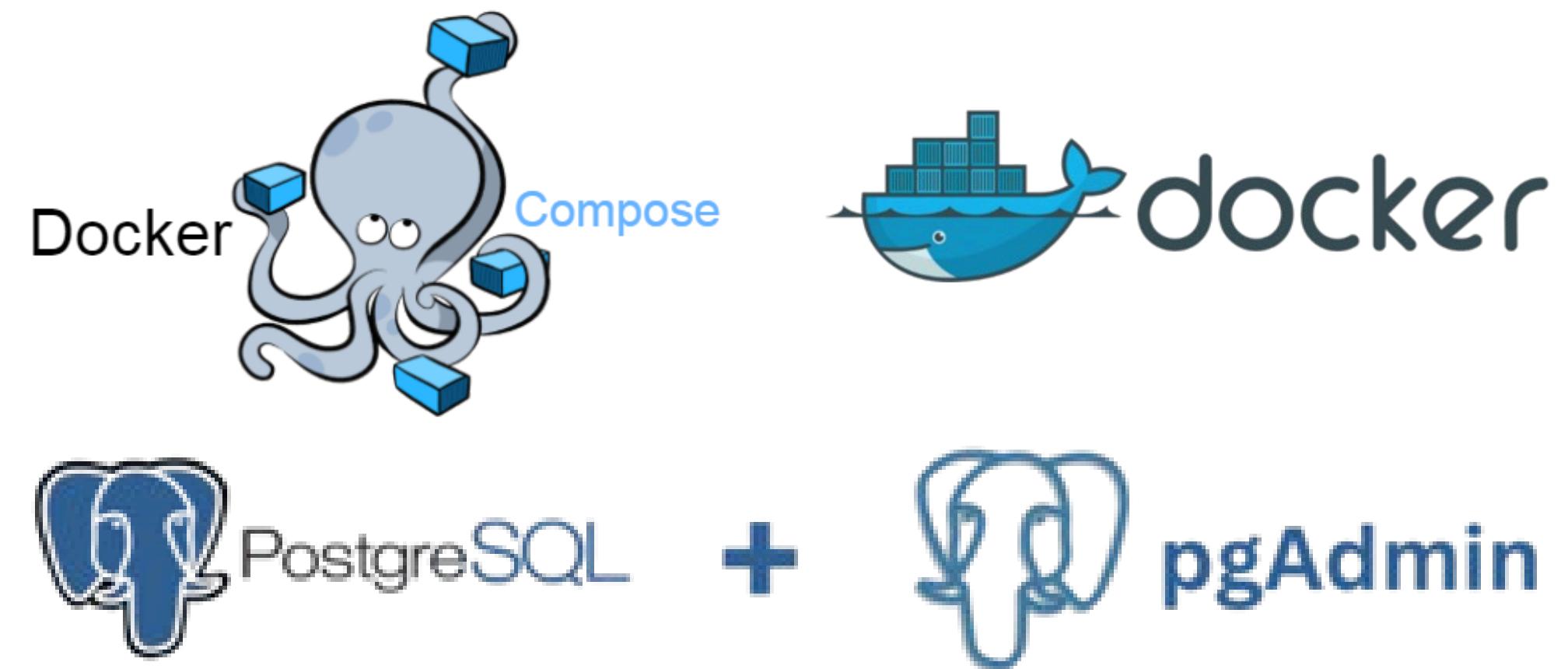


DESARROLLO PROFESIONAL EN odoo

Restaurante



EQUIPO



GABRIEL
MENDEZ

PLATOS



SOLEDAD
MORAL
CATEGORÍAS



RUBEN
LLANES

RESERVAS



PAULA
CARDENAS

EQUIPO

OBJETIVO

El objetivo de nuestro proyecto es desarrollar una aplicación completa de gestión empresarial utilizando Odoo, que simule una solución real usada por una empresa en nuestro caso un restaurante.

TECNOLOGIAS USADAS

ODOO
PYTHON
XML
CSV
DOCKER
POSTGRESQL

MÓDULO 1

PLATOS

En este módulo el cliente puede ver todos los platos en la carta, y cada plato viene con sus detalles como:

- Precio
- Una descripción breve
- Imagen del plato (opcional)
- Disponibilidad
- Mas información que quisiéramos añadir en un futuro

HTTP://LOCALHOST:8069/CARTA/CHULETA-POLLO-21

Volver

Chuleta pollo

Precio: 20.0
Disponible: True
Descripción: chuleta de pollo



LOCALHOST:8069/CARTA

Your Logo Inicio Contáctenos

Carta

Chuleta pollo chuleta de pollo Precio: 20.0€ Ver detalles	Banan babana Precio: 1.5€ Ver detalles	Causa de pollo causita de pollo ñam ñam Precio: 11.99€ Ver detalles
Lentejas un plato lentejas de to la vida Precio: 15.0€ Ver detalles	Pipas pipas Precio: 1.2€ Ver detalles	

MÓDULO 1

PLATOS

En el backend de Odoo, podemos añadir la información de los platos para, posteriormente, mostrársela a los clientes en la web

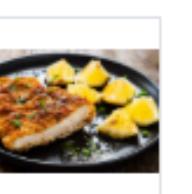
VISTA FORM

Gestion de platos Productos Internos

Nuevo Platitos Chuleta pollo

Nombre	Chuleta pollo
Precio	20,00
Disponible	<input checked="" type="checkbox"/>
Descripción	chuleta de pollo

Imagen



VISTA TREE

Gestion de platos Productos Internos

Nuevo Platitos

1-5 / 5 < >

Nombre	Precio	Disponible	Imagen
Chuleta pollo	20,00	<input checked="" type="checkbox"/>	
Banan	1,50	<input checked="" type="checkbox"/>	
Causa de pollo	11,99	<input checked="" type="checkbox"/>	
Lentejas	15,00	<input checked="" type="checkbox"/>	
Pipas	1,20	<input checked="" type="checkbox"/>	

MÓDULO 2

CATEGORÍAS - BACKEND

Este módulo permite crear y gestionar las distintas categorías del restaurante desde el ERP de Odoo.

Cada categoría contiene los siguientes campos:

- Nombre (campo obligatorio)
- Descripción
- Activo (booleano para activar o desactivar categoría)

El módulo incluye:

- Vista lista, donde se muestran todas las categorías registradas.
- Vista formulario, que permite crear o editar una categoría.
- Acción asociada para acceder al modelo.

Desde el backend es posible crear nuevas categorías, modificar las existentes y desactivarlas si es necesario.

Vista lista:

The screenshot shows the Odoo Backend interface for managing restaurant categories. The top navigation bar has tabs for 'Categorías Restaurante' and 'Categorías del Restaurante'. Below the tabs, there's a 'New' button and a search bar labeled 'Search...'. A table lists five categories: 'Nombre' (Entrantes), 'Entrantes' (checked), 'Platos Principales' (checked), 'Postres' (checked), and 'Bebidas' (checked). To the right of each category name is a 'Activo' column with a checked checkbox. The entire interface is styled with a purple header and light gray background.

Vista formulario:

The screenshot shows the Odoo Backend interface for creating or editing a category. The top navigation bar has tabs for 'Categorías Restaurante' and 'Categorías del Restaurante'. Below the tabs, there's a 'New' button and a 'Categorías del Restaurante' tab. The main form contains fields for 'Nombre' (Entrantes), 'Entrantes' (checked), 'Activo' (checked), 'Descripción' (Platos para comenzar la comida), and a 'Platos para comenzar la comida' note. The interface is styled with a purple header and light gray background.

MÓDULO 2

CATEGORÍAS - WEBSITE

El módulo dispone de una parte web pública desarrollada mediante:

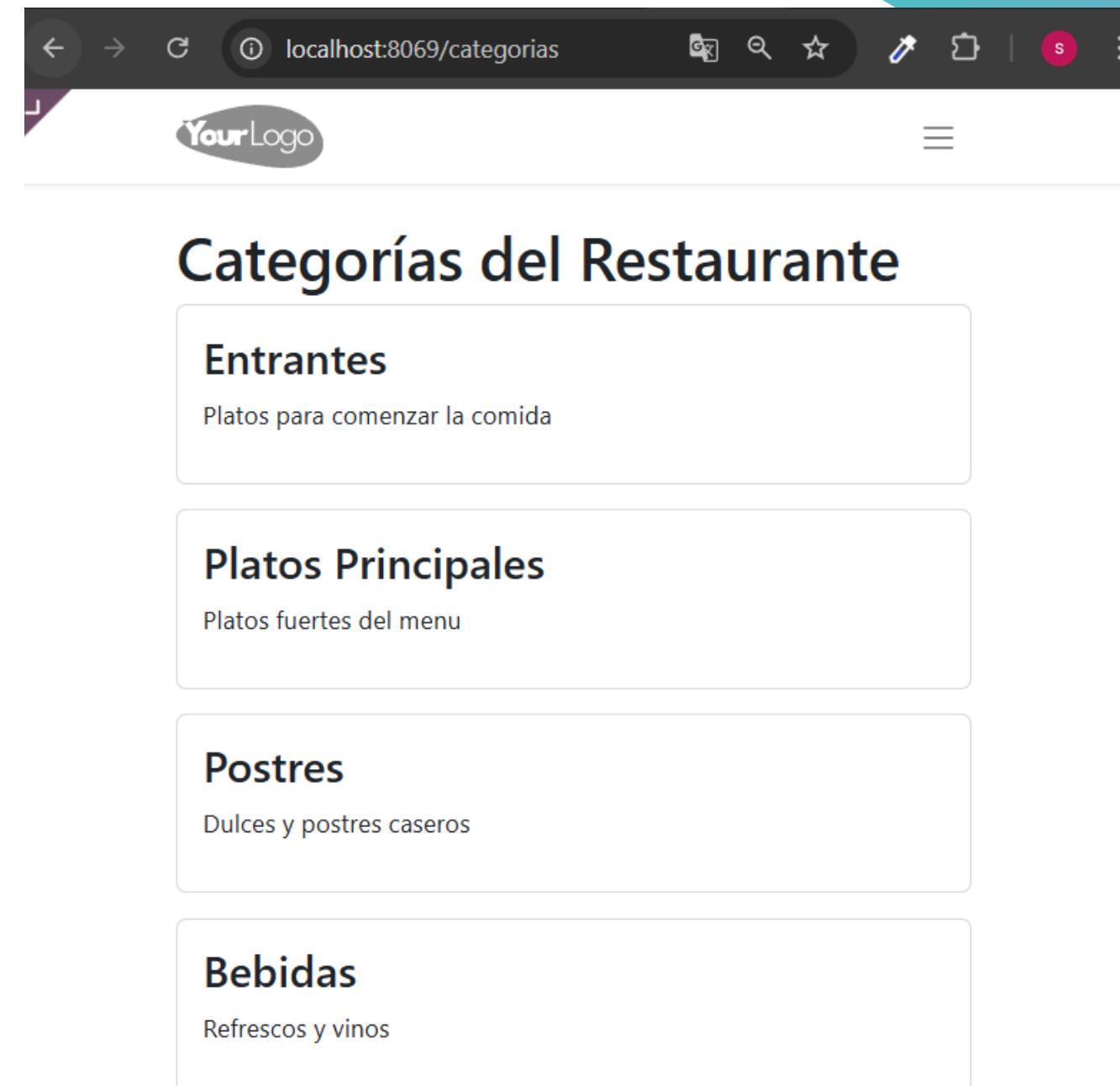
- Controller con @http.route
- Plantilla QWeb
- Ruta pública (auth='public', website=True)

En esta página se muestran únicamente las categorías activas del restaurante.

La información se presenta en formato tarjeta, mostrando:

- Nombre de la categoría
- Descripción

Los dato que aparecen en la web son los mismos creados previamente desde el backend del ERP.



MÓDULO 3

RESERVAS

Modulo el cual se encarga de crear y mostrar las reservas del restaurante, tiene estos valores:

- Nombre de la persona que reserva la mesa
- Fecha y hora de la reserva
- Numero de comensales
- Descripción o añadido voluntario sobre algun aspecto de necesidad extra

Reservas

Nombre: Juan
Fecha y hora: 2026-02-20 21:15:00
Comensales: 2
Descripción: Cerca de la ventana

Nombre: Andrea
Fecha y hora: 2026-02-20 20:30:00
Comensales: 4
Descripción: Cerca del baño

Nombre: Daniel
Fecha y hora: 2026-02-21 13:45:00
Comensales: 5
Descripción: A poder ser con una trona para bebes

Nombre: Daniela
Fecha y hora: 2026-02-20 14:50:00
Comensales: 3
Descripción: Nada que añadir

Nombre: Alberto
Fecha y hora: 2026-02-21 20:45:00
Comensales: 2
Descripción: Buen espacio para salir



The screenshot shows a web-based reservation management system. At the top, there's a purple header bar with a grid icon, a message center icon with '3' notifications, a clock icon, and a square icon. Below the header, the title 'Reservas' is displayed with a 'New' button and a gear icon. On the right, there are navigation buttons for page 1-5 / 5, a search icon, and a refresh icon.

<input type="checkbox"/>	Reserva a nombre de	Fecha y hora de la reserva	Numero de comensales
<input type="checkbox"/>	Juan	02/20/2026 21:15:00	2
<input type="checkbox"/>	Andrea	02/20/2026 20:30:00	4
<input type="checkbox"/>	Daniel	02/21/2026 13:45:00	5
<input type="checkbox"/>	Daniela	02/20/2026 14:50:00	3
<input type="checkbox"/>	Alberto	02/21/2026 20:45:00	2

MÓDULO 3

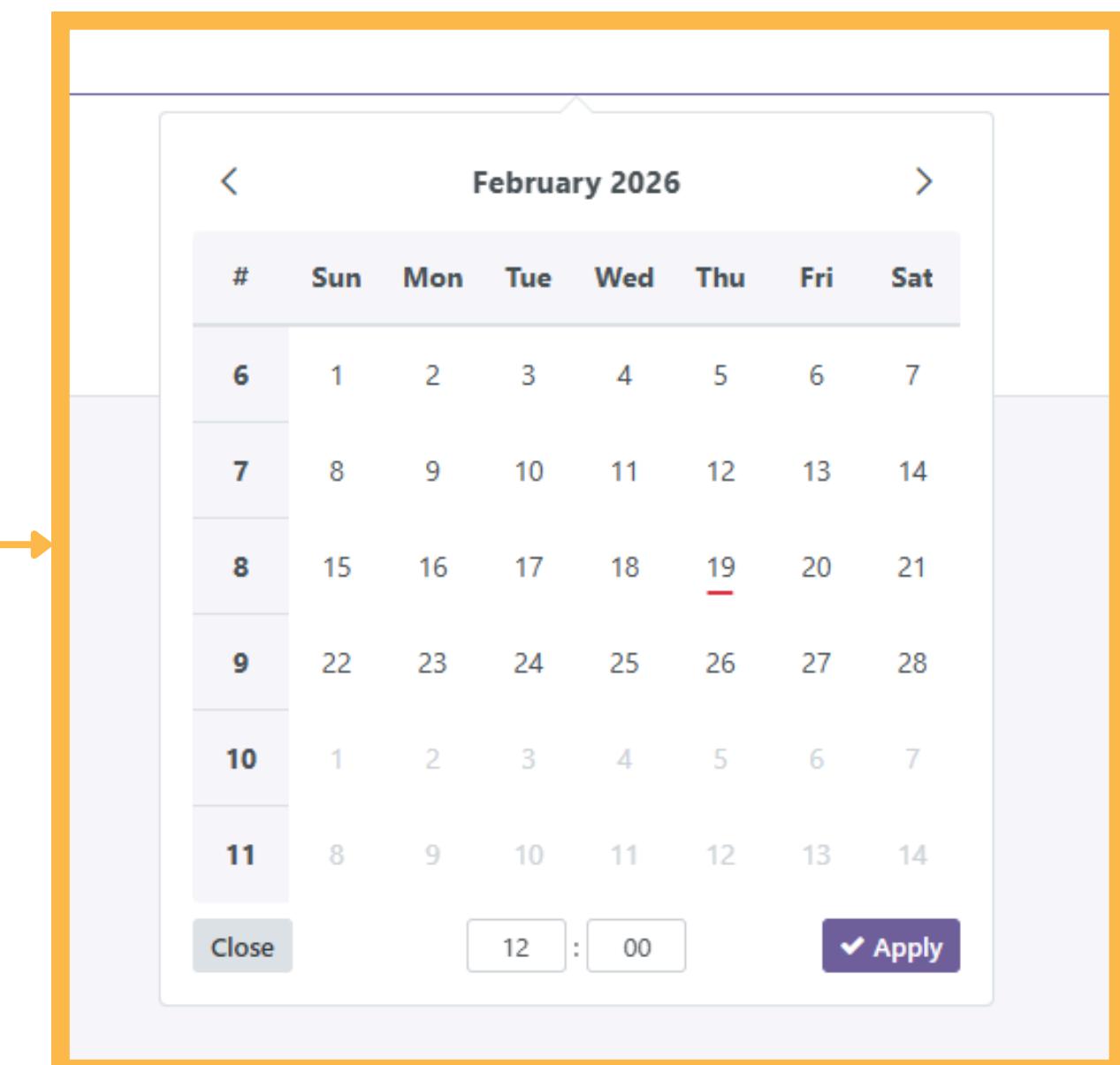
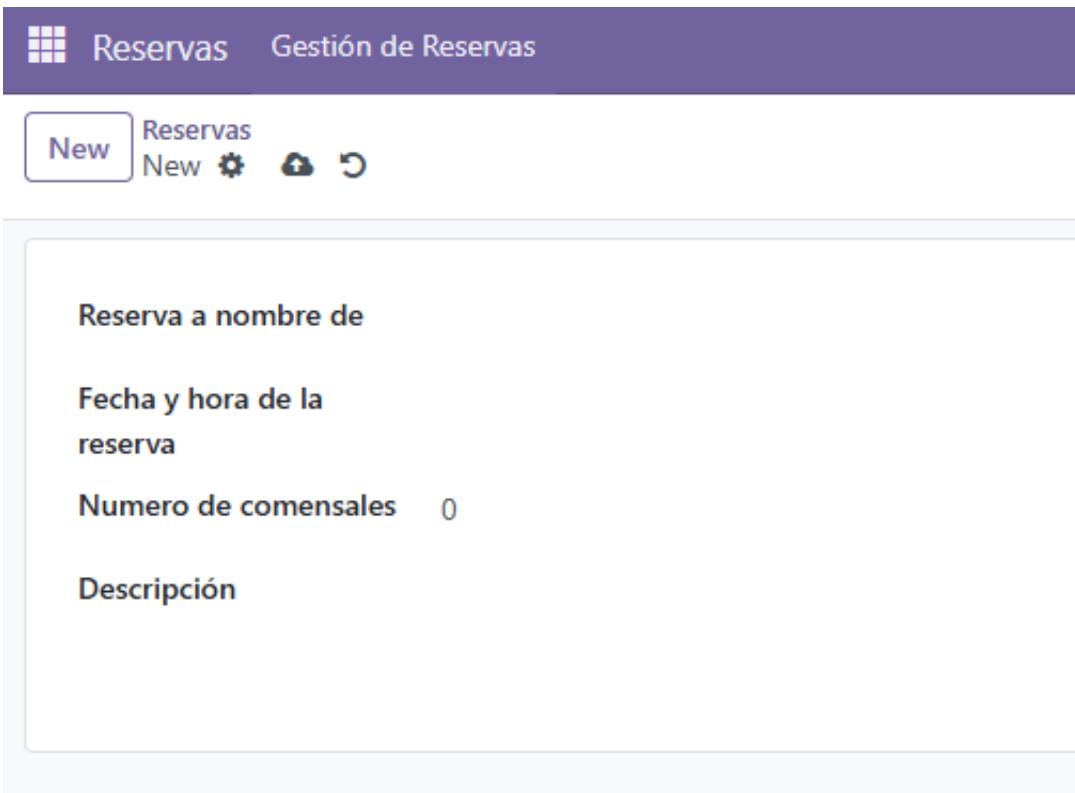
RESERVAS

A la hora de crear una nueva reserva se piden 4 datos de los cuales tres de ellos son obligatorios: Nombre, fecha y hora y numero de comensales.

Otra función a destacar es el “*Datetime*” el cual te permite mediante un calendario escoger fecha, permitiendo elegir hasta mes o año, y una hora en formato de 24h.

```
class modulo_reservas(models.Model):
    _name = 'modulo_reservas.modulo_reservas'
    _description = 'Reservas'

    name = fields.Char(string="Reserva a nombre de", required=True)
    fecha_reserva = fields.Datetime(string="Fecha y hora de la reserva", required=True)
```



MODULO EQUIPO

Su función principal es mostrar la información de cada miembro tanto en el backend del ERP como en la web pública, simulando una sección real de presentación de equipo en una empresa.

The screenshot shows a web browser window with the URL `localhost:8069/equipo` in the address bar. The page title is "Equipo". It lists four team members in a grid:

- Soledad Moral** (RolModulo Categorias)
[ver detalle](#)
- Ruben Llanes** (RolModulo Reservas)
[ver detalle](#)
- Gabriel Mendez** (RolModulo Platos)
[ver detalle](#)
- Paula Cardenas** (RolModulo Equipo)
[ver detalle](#)

The browser interface includes standard navigation buttons (back, forward, search, etc.) and a toolbar with links to "IP Campus", "Google", "Office 365", and "Traductor de Google".

MODULO EQUIPO

```
miembro.py X  
addons > modulo_equipos > models > miembro.py  
1  from odoo import models, fields  
2  
3  class EquipoMiembro(models.Model):  
4      _name = "equipo.miembro"  
5      _description = "Miembro del equipo"  
6  
7      name = fields.Char(string="Nombre", required=True)  
8      rol = fields.Char(string="Rol (modulo)", required=True)  
9      description = fields.Text(string = "Descripcion")  
10     foto = fields.Image(string="Fotografia")
```

Cada campo se convierte automáticamente en una columna en la base de datos.

Paula Cardenas

- Acción que abre las vistas del modelo.
- Indica qué modelo se va a mostrar.
- Crea los menús visibles en el sistema.

Paula Cardenas

```
<record id="action_equipo_miembro" model="ir.actions.act_window">  
    <field name="name">Miembros</field>  
    <field name="res_model">equipo.miembro</field>  
    <field name="view_mode">tree,form</field>  
</record>  
  
<!-- Menu principal -->  
<menuitem id="menu_equipo_root"  
        name="Equipo"  
        action="action_equipo_miembro"  
        sequence="10" />
```

Esto conecta el modelo con la interfaz gráfica.

Paula Cardenas

MODULO EQUIPO

- Esta vista muestra los registros en formato tabla.
- El widget image permite mostrar imágenes en miniatura dentro de la tabla.

Paula Cardenas

```
<record id="view_equipo_miembro_tree" model="ir.ui.view">
    <field name="name">equipo.miembro.tree</field>
    <field name="model">equipo.miembro</field>
    <field name="arch" type="xml">
        <tree>
            <field name="name"/>
            <field name="rol"/>
            <field name="foto" widget="image" options="{'size':[50,50]}"/>
        </tree>
    </field>
</record>
```

- Esta vista permite crear y editar registros.
- Aplica estilo visual para que la imagen se muestre como avatar.

Paula Cardenas

```
<form>
    <sheets>
        <sheet>
            <group>
                <field name="name"/>
                <field name="rol"/>
            </group>
            <group>
                <field name="description"/>
                <field name="foto" widget="image" class="oe_avatar"/>
            </group>
        </sheet>
    </sheets>
</form>
```



AHORA ARRANCAR
DOCKER