

MÓDULO

ODOO



Autores: Aritz Ayensa, Anthuan Lugris, Sergio Ajona

Asignatura: Sistemas de gestión empresarial

Fecha: 02/11/2024

Grupo: DAM 2A

Índice

Introducción	2
Modelos Implementados	4
1. Modelo res.coche	4
2. Modelo res.booking	4
3. Modelo res.partner (Heredado como Cliente)	5
Automatizaciones y Cron Jobs	7
Vistas en XML	8
1. Vista de Clientes (res.partner)	8
2. Vista de Coches (res.coche)	8
3. Vista de Reservas (res.booking)	8
Errores	9
Error 1: Campo numbercall no válido en ir.cron	9
Error 2: Incompatibilidad de la etiqueta <tree> en Odoo 18	9
Conclusión	10

Introducción

Este documento describe el desarrollo de un módulo de Odoo para la gestión de reservas de coches. El módulo permite registrar coches disponibles, gestionar reservas, y generar facturas automáticamente. Además, proporciona una verificación de solapamiento de reservas y un sistema para cancelar reservas pendientes.

Gestion Reservas

Descripción breve del módulo

Por Bugisoft S.A

Actualizar

Desinstalar

Información

Datos técnicos

Características instaladas

MENÚS CREADOS

- Cientes
- Coches
- Coches/Coches
- Reservas
- Reservas/Calendario de Reservas
- Reservas/Lista de Reservas
- Reservas/Reservas Kanban

VISTAS CREADAS

* INHERIT cliente.form.inherit (form)

cliente.list (list)

coche.form (form)

coche.list (list)

res.booking.calendar (calendar)

res.booking.kanban (kanban)

view.reserva.form (form)

view.reserva.list (list)

Modelos Implementados

1. Modelo res.coche

Este modelo gestiona la información de los coches disponibles para la reserva.

Campos:

- **name** (Char): Modelo del coche.
- **marca** (Char): Marca del coche.
- **precio_por_dia** (Float): Precio por día de alquiler.
- **matricula** (Char): Matrícula única.
- **tipo_combustible** (Selection): Tipo de combustible (Gasolina, Diésel, Híbrido, Eléctrico).
- **num_asientos** (Integer): Número de asientos (por defecto 5).
- **transmision** (Selection): Tipo de transmisión (Manual, Automática).
- **kilometraje** (Float): Kilometraje actual.
- **color** (Char): Color del coche.
- **reservation_ids** (One2many): Relación con el modelo res.booking.

Reglas de Negocio:

- Se verifica que la matrícula sea única.
- No se permite eliminar coches con reservas activas.

2. Modelo res.booking

Este modelo gestiona las reservas de los coches.

Campos:

- **name** (Char): Nombre de la reserva.
- **description** (Text): Descripción de la reserva.
- **duracion** (Integer): Duración en días.
- **fecha** (Datetime): Fecha y hora de inicio.
- **estado** (Selection): Estado de la reserva (Pendiente, Confirmada, Cancelada).
- **precio** (Float): Precio calculado automáticamente.
- **coche_id** (Many2one): Relación con el coche reservado.
- **cliente_id** (Many2one): Relación con el cliente que reserva.
- **factura_id** (Many2one): Relación con la factura generada.

Reglas de Negocio:

- Se calcula el precio automáticamente, aplicando un descuento del 10% a clientes VIP.
- Se verifica que no haya solapamientos de reservas para el mismo coche.
- La fecha de inicio no puede ser anterior a la actual.

Acciones y Métodos:

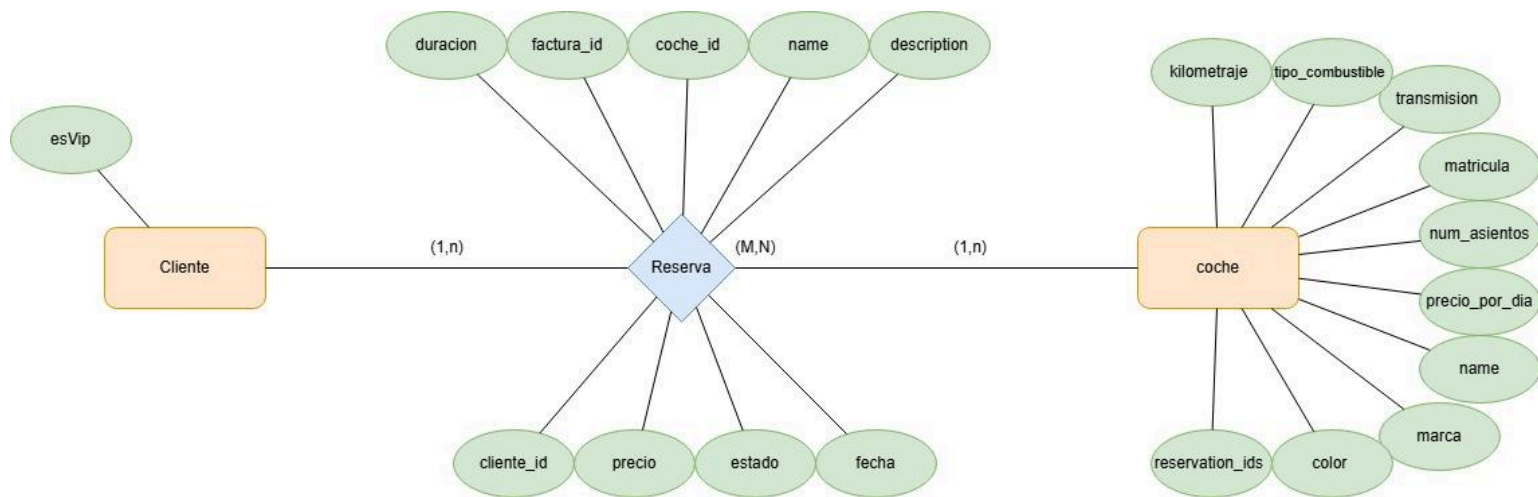
- **action_confirmar_reserva()**: Genera una factura y cambia el estado a "Confirmada".
- **action_cancelar_reserva()**: Cambia el estado a "Cancelada".
- **cancelar_reservas_pendientes()**: Cancela reservas en estado "Pendiente" por más de 24 horas.
- **crear_cron_job()**: Programa un cron job para cancelar reservas pendientes.

3. Modelo res.partner (Heredado como Cliente)

Este modelo extiende **res.partner** para incluir información específica de clientes.

Campos:

- **is_vip** (Boolean): Indica si el cliente es VIP.
- **reserva_ids** (One2many): Relación con reservas realizadas por el cliente.



Automatizaciones y Cron Jobs

Se implementa un cron job para cancelar reservas en estado "Pendiente" con más de 24 horas de antigüedad.

```
<?xml version="1.0" encoding="utf-8"?>
<odoo>
  <data noupdate="1">
    <record id="ir_cron_cancelar_reservas_pendientes" model="ir.cron">
      <field name="name">Cancelar reservas pendientes</field>
      <field name="model_id" ref="model_res_booking"/>
      <field name="state">code</field>
      <field name="code">model.cancelar_reservas_pendientes()</field>
      <field name="interval_number">1</field>
      <field name="interval_type">hours</field>
      <field name="numbercall">-1</field> <!-- -1 significa que se
ejecutará indefinidamente -->
      <field name="active" eval="True"/>
    </record>
  </data>
</odoo>
```

Vistas en XML

El módulo incluye varias vistas en XML para gestionar clientes, coches y reservas de manera eficiente.

1. Vista de Clientes (**res.partner**)

- **Lista de clientes:** Muestra los clientes con su nombre, correo, teléfono y si son VIP.
- **Formulario de clientes:** Modifica la vista original para incluir un campo de VIP.
- **Menú de clientes:** Agrega un menú de acceso a la lista de clientes.

2. Vista de Coches (**res.coche**)

- **Lista de coches:** Muestra todos los coches disponibles con detalles como modelo, marca, precio, matrícula, etc.
- **Formulario de coches:** Permite agregar nuevos coches con sus características.
- **Menú de coches:** Agrega un menú de acceso a la lista de coches.

3. Vista de Reservas (**res.booking**)

- **Lista de reservas:** Muestra las reservas con información clave como nombre, fecha, estado y cliente.
- **Formulario de reservas:** Permite gestionar las reservas con botones para confirmar o cancelar.
- **Calendario de reservas:** Muestra las reservas en una vista de calendario según la fecha de inicio.
- **Vista Kanban:** Permite visualizar las reservas agrupadas por estado.
- **Menú de reservas:** Agrega accesos directos a las diferentes vistas de reservas (lista, kanban, calendario)

Errores

Error 1: Campo numbercall no válido en ir.cron

Descripción:

Durante la carga de datos en Odoo, se generó el siguiente error:

ValueError: Invalid field 'numbercall' on model 'ir.cron'

El error indica que el modelo ir.cron no reconoce el campo numbercall. En versiones anteriores de Odoo, este campo se usaba para definir la cantidad de veces que se ejecutaría un cron job, pero ha sido eliminado en versiones recientes.

Solución:

Modificar el archivo cron_job.xml eliminando el campo numbercall.

Error 2: Incompatibilidad de la etiqueta <tree> en Odoo 18

Descripción del error: En Odoo 18, se utilizaba la etiqueta <tree> para crear vistas de tipo lista. Sin embargo, esta etiqueta fue eliminada y reemplazada por <list>. Al intentar cargar una vista con la etiqueta <tree>, el sistema generó un error indicando que la etiqueta no es compatible.

Causa del error:

- En versiones anteriores de Odoo, <tree> era válida, pero en Odoo 18 fue reemplazada por <list>.
- Al usar <tree>, el sistema no reconoció la etiqueta y lanzó un error.

Solución:

- Se sustituyó la etiqueta <tree> por <list> en el archivo XML de la vista

Conclusión

Este módulo optimiza la gestión de reservas de coches, asegurando disponibilidad, evitando solapamientos y automatizando la facturación. Además, mejora la experiencia del cliente al ofrecer descuentos a usuarios VIP y mantener un control eficiente de las reservas en curso.