Rapport de Projet :

Module de Gestion de Réservation de Salles sous Odoo 17

Réalisé par : Yassine BEKKOUCH

Encadré par : M. Mohammed AITDAOUD

5éme Année en Ingénierie Informatique et Réseaux spécialite MIAGE

Année universitaire : 2023/2024

Table des matières

1. Introduction

Ce projet vise à développer un module personnalisé pour l'ERP Odoo (version 17) permettant la gestion complète de salles de réunion et de leurs réservations. Le module a été développé dans un environnement conteneurisé Docker, respectant l'architecture MVC (Modèle-Vue-Contrôleur) propre au framework Odoo.

Objectifs du module :

• Gérer un parc de salles (capacité, équipements).

• Planifier des réservations via un calendrier.

• Suivre le statut des demandes (Brouillon, Confirmé, Annulé).

2. Environnement Technique et Initialisation

Le projet repose sur l'utilisation de Docker pour l'orchestration des conteneurs (PostgreSQL et Odoo) et de Visual Studio Code pour le développement.

2.1 Structure du Module

La première étape a consisté à créer l'arborescence standard d'un module Odoo. La logique métier (models), les interfaces (views) et la sécurité (security) ont été séparées.

Illustration : Création de la structure via le terminal

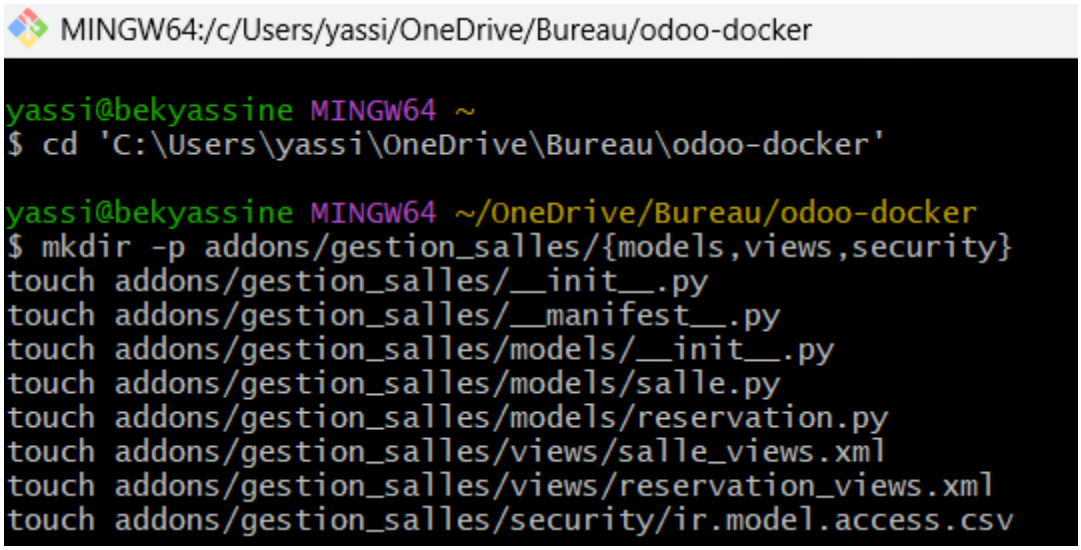
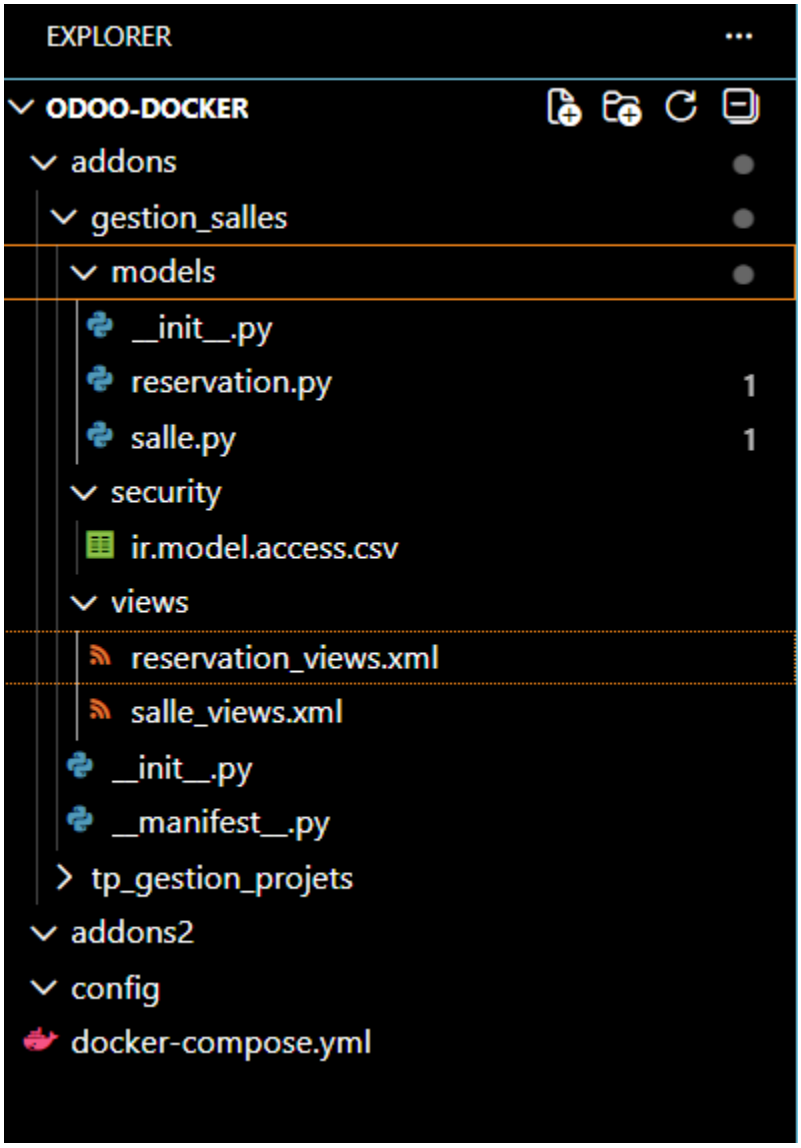


Illustration : Vue de l'explorateur de fichiers



3. Développement Backend (Modèles de Données)

Le cœur du module repose sur deux classes Python principales définies dans le dossier models :

• gestion.salle : Définit les caractéristiques d'une salle (Nom, Capacité, Projecteur, Climatisation).

• gestion.reservation : Gère les événements (Date de début, Date de fin, Salle liée, Responsable).

Les fichiers __manifest__.py et __init__.py ont été configurés pour déclarer le module et charger les fichiers Python.

3.1 Code models

salle.py 1

addons > gestion_salles > models > salle.py > Salle

```python
from odoo import models, fields

class Salle(models.Model):
    _name = "gestion.salle"
    _description = "Salle de réunion"

    name = fields.Char(string="Nom de la salle", required=True)
    capacite = fields.Integer(string="Capacité (personnes)")
    a_projecteur = fields.Boolean(string="Projecteur vidéo disponible ?")
    a_climatisation = fields.Boolean(string="Climatisation ?")
    description = fields.Text(string="Description / Notes")
```

__init__.py

addons > gestion_salles > models > __init__.py

```python
from . import salle
from . import reservation
```

reservation.py 1

addons > gestion_salles > models > reservation.py > Reservation

```python
from odoo import models, fields

class Reservation(models.Model):
    _name = "gestion.reservation"
    _description = "Réservation de salle"

    name = fields.Char(string="Objet de la réunion", required=True)
    salle_id = fields.Many2one("gestion.salle", string="Salle", required=True)
    date_debut = fields.Datetime(string="Début", required=True)
    date_fin = fields.Datetime(string="Fin", required=True)
    responsable_id = fields.Many2one("res.users", string="Organisateur", default=lambda self: self.env.user)
    statut = fields.Selection([
        ('brouillon', 'Brouillon'),
        ('confirme', 'Confirmé'),
        ('annule', 'Annulé'),
    ], string="Statut", default='brouillon')
```

3.2 Code security

ir.model.access.csv

addons > gestion_salles > security > ir.model.access.csv

```
id,name,model_id:id,group_id:id,perm_read,perm_write,perm_create,perm_unlink
access_gestion_salle,gestion.salle,model_gestion_salle,,1,1,1,1
access_gestion_reservation,gestion.reservation,model_gestion_reservation,,1,1,1,1
```

3.3 Code views

```xml
<?xml version="1.0" encoding="UTF-8"?>
<odoo>
    <record id="view_reservation_calendar" model="ir.ui.view">
        <field name="name">gestion.reservation.calendar</field>
        <field name="model">gestion.reservation</field>
        <field name="arch" type="xml">
            <calendar string="Réservations" date_start="date_debut" date_stop="date_fin" color="salle_id">
                <field name="name"/>
                <field name="salle_id"/>
            </calendar>
        </field>
    </record>

    <record id="view_reservation_tree" model="ir.ui.view">
        <field name="name">gestion.reservation.tree</field>
        <field name="model">gestion.reservation</field>
        <field name="arch" type="xml">
            <tree decoration-info="statut == 'brouillon'" decoration-success="statut == 'confirmé'">
                <field name="name"/>
                <field name="salle_id"/>
                <field name="date_debut"/>
                <field name="date_fin"/>
                <field name="statut" widget="badge"/>
            </tree>
        </field>
    </record>

    <record id="view_reservation_form" model="ir.ui.view">
        <field name="name">gestion.reservation.form</field>
        <field name="model">gestion.reservation</field>
        <field name="arch" type="xml">
            <form>
                <header>
                    <field name="statut" widget="statusbar" options="{'clickable': '1'}"/>
                </header>
                <sheet>
                    <group>
                        <field name="name"/>
                        <field name="salle_id"/>
                    </group>
                    <group>
                        <field name="date_debut"/>
                        <field name="date_fin"/>
                        <field name="responsable_id"/>
                    </group>
                </sheet>
            </form>
        </field>
    </record>

    <record id="action_gestion_reservation" model="ir.actions.act_window">
        <field name="name">Réservations</field>
        <field name="res_model">gestion.reservation</field>
        <field name="view_mode">calendar,tree,form</field>
    </record>

    <menuitem id="menu_reservations" parent="menu_gestion_root" action="action_gestion_reservation" sequence="20"/>
</odoo>
```

```xml
<?xml version="1.0" encoding="UTF-8"?>
<odoo>
    <record id="view_salle_form" model="ir.ui.view">
        <field name="name">gestion.salle.form</field>
        <field name="model">gestion.salle</field>
        <field name="arch" type="xml">
            <form>
                <sheet>
                    <div class="oe_title">
                        <h1><field name="name" placeholder="Ex: Salle de Conférence A"/></h1>
                    </div>
                    <group>
                        <group>
                            <field name="capacite"/>
                        </group>
                        <group>
                            <field name="a_projecteur"/>
                            <field name="a_climatisation"/>
                        </group>
                    </group>
                    <field name="description" placeholder="Notes sur la salle..."/>
                </sheet>
            </form>
        </field>
    </record>

    <record id="action_gestion_salle" model="ir.actions.act_window">
        <field name="name">Salles</field>
        <field name="res_model">gestion.salle</field>
        <field name="view_mode">tree,form</field>
    </record>

    <menuitem id="menu_gestion_root" name="Réservation Salles" sequence="10"/>
    <menuitem id="menu_salles" parent="menu_gestion_root" action="action_gestion_salle" sequence="10"/>
</odoo>
```

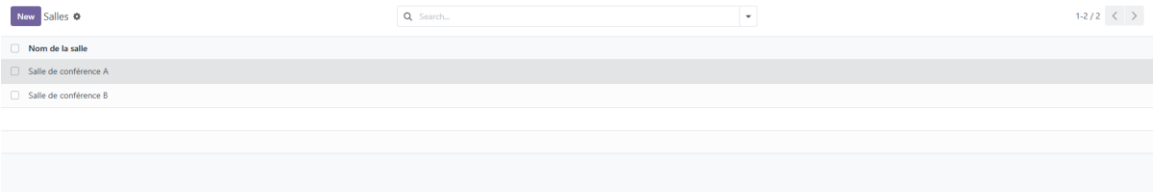
4. Développement Frontend (Vues XML et Menus)

L'interface utilisateur a été conçue en XML pour offrir plusieurs modes de visualisation : Liste (Tree), Formulaire (Form) et Calendrier.

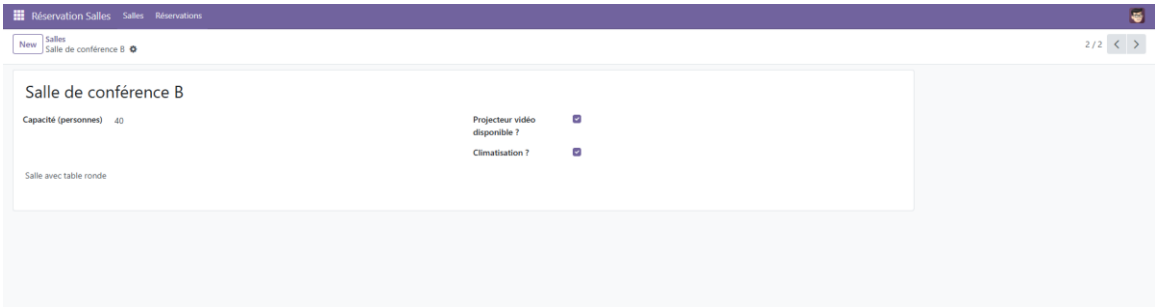
4.1 Gestion des Salles

Une interface permet de lister les salles et de consulter leurs détails techniques (équipements disponibles).

Illustrations prévues :
- Vue liste des salles.



- Vue formulaire détaillée d'une salle.

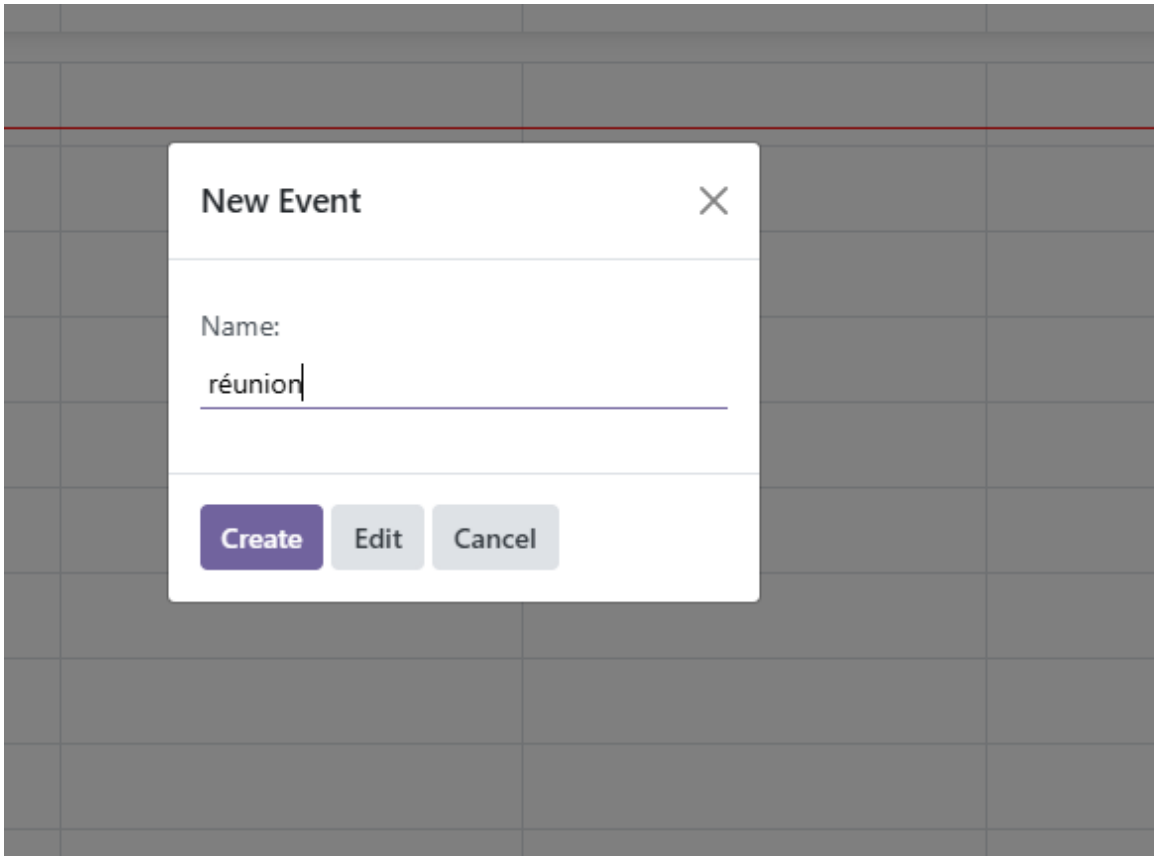


4.2 Gestion des Réservations

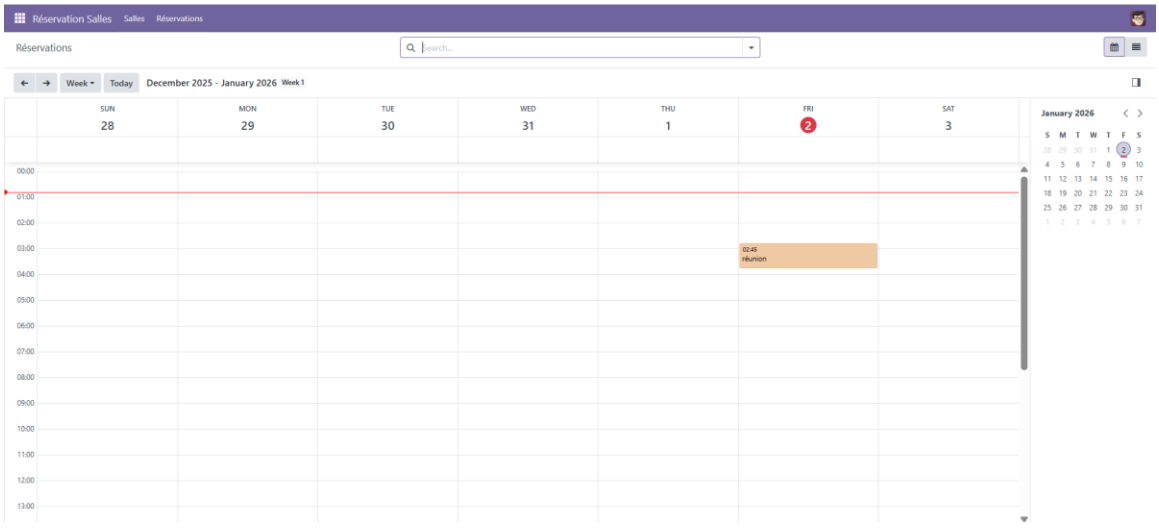
Une vue Calendrier a été intégrée afin de faciliter la visualisation et la planification des réservations.

Illustrations prévues :

- Création rapide d'une réservation depuis le calendrier.



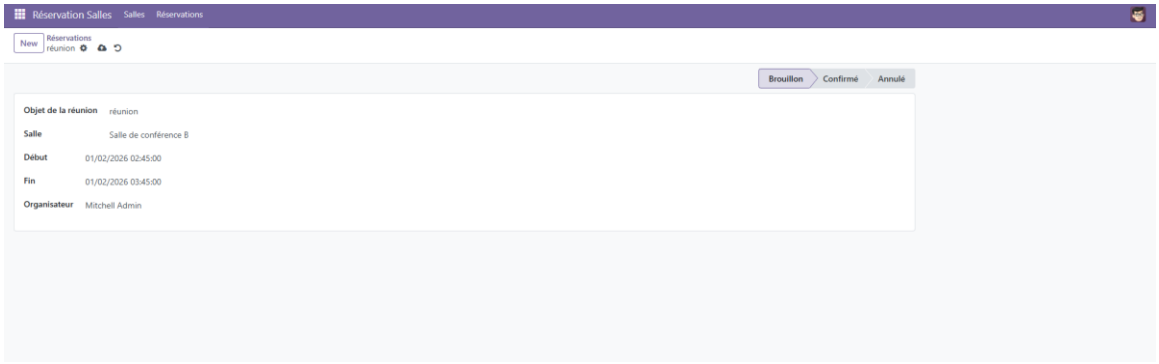
- Vue calendrier hebdomadaire globale.



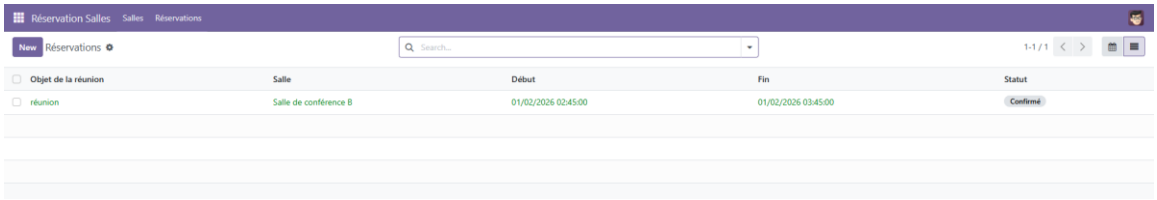
4.3 Suivi des Statuts (Workflow)

Un workflow simple a été mis en place avec une barre d'état permettant de suivre l'évolution des réservations (Brouillon, Confirmé, Annulé).

Illustrations prévues :
- Workflow de validation.



- Liste synthétique des réservations.



5. Sécurité et Déploiement

Les droits d'accès ont été configurés dans le fichier ir.model.access.csv afin de définir les permissions CRUD (Create, Read, Update, Delete) aux utilisateurs.

Le déploiement a nécessité la configuration du docker-compose.yml pour monter correctement le volume des extra-addons afin qu'Odoo détecte le nouveau module personnalisé.

6. Conclusion

Ce projet a permis de mettre en pratique l'architecture modulaire d'Odoo. Le résultat est un module fonctionnel de gestion de réservation de salles, intégré nativement à l'ERP et offrant une interface fluide et adaptée aux besoins de l'organisation.