



**ECOLE MAROCAINE DES
SCIENCES DE L'INGENIEUR**
Membre de **HONORIS UNITED UNIVERSITIES**

Rapport de Mini-projet Création d'un nouveau module (Gestion de projets étudiants)

Élaboré par :
KHAIRY Sami (G3)

Professeur :
Pr AIT DAOUD Mohammed

Lien Github :
<https://github.com/Sami-khairy/Student-Project-Management---Odoo-module>

Année universitaire : 2025/2026

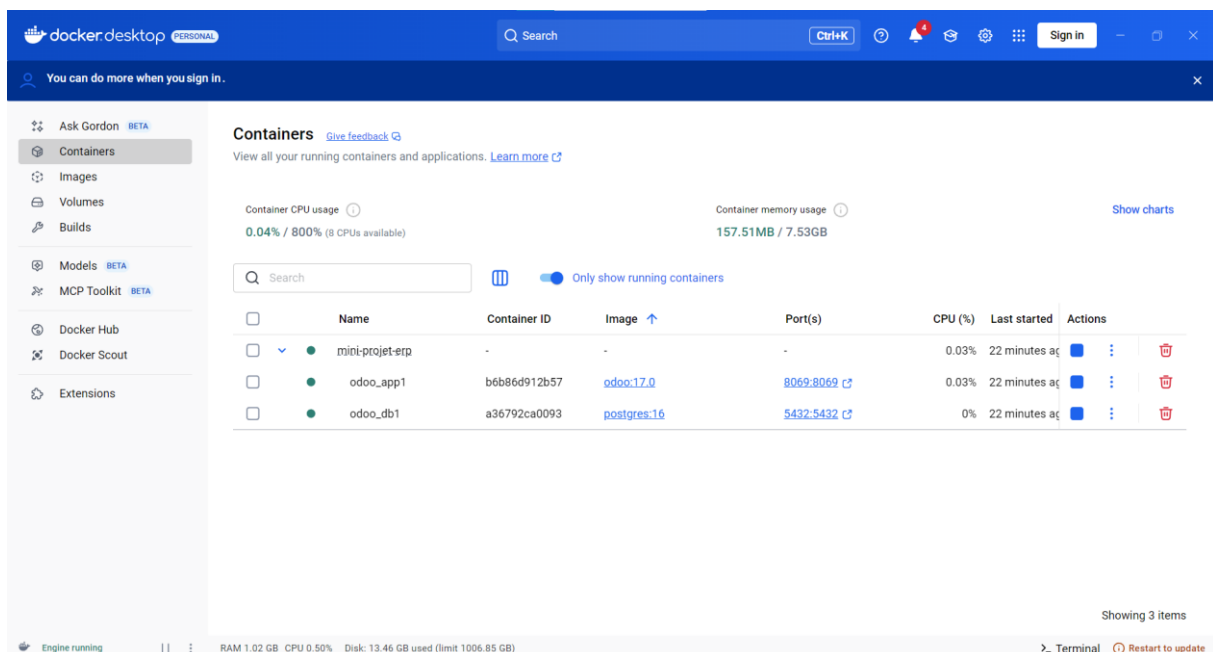
Objectif du Module

Créer un module Odoo technique nommé `gestion_projets_etudiants` visant à digitaliser l'intégralité du processus de suivi académique. Ce module permet de :

- **Gérer le cycle de vie complet** des projets (PFE, PFA, Stages) à travers un workflow strict et validé : *Brouillon* -> *Validation* -> *En cours* -> *Rendu* -> *Noté*.
- **Suivre l'avancement technique** via un système de "**Livrables & Jalons**", intégrant une barre de progression calculée automatiquement et le dépôt des fichiers (rapports, code source).
- **Organiser les soutenances** en gérant la planification (Date, Lieu), la composition des membres du Jury, et l'envoi automatisé des **convocations par email**.
- **Appliquer une notation multicritères** précise (Rapport/10, Présentation/5, Technique/5) avec un calcul automatique et sécurisé de la note finale (/20).
- **Sécuriser l'accès aux données** grâce à trois niveaux de permissions : **Étudiant** (vue limitée aux siens), **Tuteur** (supervision et notation), et **Responsable Pédagogique** (administration).
- **Piloter l'activité** grâce à des **Tableaux de bord et Statistiques** (Vues Graphiques et Pivot) permettant d'analyser la répartition des notes et la charge des tuteurs, tout en conservant les vues Kanban et Liste pour la gestion quotidienne.

Étape 1 : Préparation de l'environnement Docker

Nous allons utiliser un fichier `docker-compose.yml` configuré pour Odoo 17 avec une base PostgreSQL 16.



The screenshot shows the Docker Desktop interface. The left sidebar contains navigation options: Containers (selected), Images, Volumes, Builds, Models (BETA), MCP Toolkit (BETA), Docker Hub, Docker Scout, and Extensions. The main area displays the 'Containers' section with a search bar and a toggle for 'Only show running containers'. Below this is a table of running containers:

	Name	Container ID	Image	Port(s)	CPU (%)	Last started	Actions
<input type="checkbox"/>	mini-projet-erp	-	-	-	0.03%	22 minutes ago	
<input type="checkbox"/>	odoo_app1	b6b86d912b57	odoo:17.0	8069:8069	0.03%	22 minutes ago	
<input type="checkbox"/>	odoo_db1	a36792ca0093	postgres:16	5432:5432	0%	22 minutes ago	

At the bottom, the status bar shows 'Engine running', system resources (RAM 1.02 GB, CPU 0.50%, Disk: 13.46 GB used / 1006.85 GB limit), and a 'Terminal' button with a 'Restart to update' link.

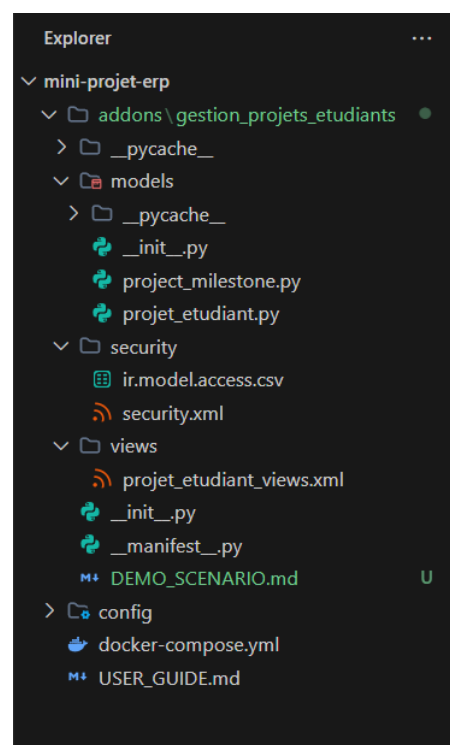
```

docker-compose.yml X
docker-compose.yml
1  version: "3.8"
2
3  services:
4    db:
5      image: postgres:16
6      container_name: odoo_db1
7      ports:
8        - "5432:5432"
9      environment:
10       POSTGRES_USER: odoo
11       POSTGRES_PASSWORD: odoo
12       PGDATA: /var/lib/postgresql/data/pgdata
13      volumes:
14        - odoo-db-data1:/var/lib/postgresql/data
15
16    odoo17:
17      image: odoo:17.0
18      container_name: odoo_app1
19      depends_on:
20        - db
21      ports:
22        - "8069:8069"
23      volumes:
24        - odoo-web-data1:/var/lib/odoo
25        - ./config:/etc/odoo
26        - ./addons:/mnt/extra-addons
27      command: >
28        odoo -d odoo_db1
29        -u gestion_projets_etudiants
30        --dev=xml
31        --db_user=odoo
32        --db_password=odoo
33        --db_host=db
34
35  volumes:
36    odoo-web-data1:
37    odoo-db-data1:

```

Etape 2 : Création de la structure du module

Dans le dossier addons créez l'arborescence suivante:



Étape 3 : Définition du module (Manifeste)

Le manifeste décrit le module et ses dépendances (mail, contacts).

Fichier : addons/gestion_projets_etudiants/__manifest__.py

```
__manifest__.py X
addons > gestion_projets_etudiants > __manifest__.py
1  {
2      'name': 'Student Project Management',
3      'version': '1.0',
4      'summary': 'Academic Project Management System (PFE, PFA, etc.)',
5      'category': 'Education',
6      'author': 'KHAIRY Sami',
7      'depends': ['base', 'mail', 'contacts'],
8      'data': [
9          'security/security.xml',
10         'security/ir.model.access.csv',
11         'views/projet_etudiant_views.xml',
12     ],
13     'installable': True,
14     'application': True,
15 }
```

Fichier : addons/gestion_projets_etudiants/__init__.py

```
__init__.py X
addons > gestion_projets_etudiants > __init__.py
1  from . import models
2  |
```

Etape 4 : Le Modèle Python (Back-end)

Nous allons créer la logique métier avec les contraintes et l'héritage du système de messagerie (mail.thread).

1. Init du dossier models :

Fichier : addons/gestion_projets_etudiants/models/__init__.py

```
_init_.py X
addons > gestion_projets_etudiants > models > _init_.py
1 from . import projet_etudiant
2 from . import project_milestone
```

Fichier : addons/gestion_projets_etudiants/models/projet_etudiant.py

Ce fichier définit le modèle principal **StudentProject**, qui constitue le cœur de la base de données du module. Il structure l'enregistrement des projets en définissant les champs essentiels (titre, dates, liens Git) et gère les relations entre les différents acteurs (étudiants, tuteurs et membres du jury), tout en pilotant le cycle de vie du projet à travers ses différents états (brouillon, en cours, rendu, noté).

Au-delà du stockage, ce fichier contient toute la **logique métier et l'automatisation**. Il intègre des fonctions Python pour calculer dynamiquement la note finale et la progression en temps réel, vérifier la cohérence des saisies (contraintes de notes) et automatiser l'envoi des emails de convocation pour les soutenances.

```
projet_etudiant.py X
addons > gestion_projets_etudiants > models > projet_etudiant.py > StudentProject > student_ids
1 from odoo import models, fields, api
2
3 class StudentProject(models.Model):
4     _name = "student.project"
5     _description = "Student Project Management"
6     _inherit = ['mail.thread', 'mail.activity.mixin']
7
8     name = fields.Char(string="Titre du projet", required=True, tracking=True)
9     description = fields.Html(string="Description du sujet")
10
11     project_type = fields.Selection([
12         ('pfe', 'PFE'),
13         ('pfa', 'PFA'),
14         ('mini_projet', 'Mini-projet'),
15         ('stage', 'Stage')
16     ], string="Type de Projet", default='mini_projet', tracking=True)
17
18     # Relation avec les étudiants (res.partner)
19     student_ids = fields.Many2many(
20         'res.partner',
21         string="Étudiant(s)",
22         help="Sélectionnez les étudiants assignés à ce projet."
23     )
24
25     # Relation avec le tuteur (Supervisor)
26     supervisor_id = fields.Many2one(
27         'res.partner',
28         string="Tuteur (Professeur)",
29         tracking=True
30     )
31
32     date_deadline = fields.Date(string="Date limite", tracking=True)
33     repository_url = fields.Char(string="Lien Repository (Git)")
34
```

```
projet_etudiant.py X
addons > gestion_projets_etudiants > models > projet_etudiant.py > StudentProject > student_ids
3 class StudentProject(models.Model):
34
35     # Champs pour la Soutenance
36     defense_date = fields.DateTimeField(string="Date de Soutenance", tracking=True)
37     defense_location = fields.CharField(string="Salle / Lieu")
38     jury_ids = fields.Many2many(
39         'res.partner',
40         'student_project_jury_rel',
41         'project_id',
42         'partner_id',
43         string="Membres du Jury",
44         help="Sélectionnez les membres du jury (en plus du tuteur)."
45     )
46
47     # Jalons & Livrables
48     milestone_ids = fields.One2many(
49         'student.project.milestone',
50         'project_id',
51         string="Livrables & Jalons"
52     )
53     progress = fields.Float(string="Progression", compute="_compute_progress", store=True)
54
55     # Évaluation Détaillée
56     grade_report = fields.Float(string="Note Rapport (/10)", default=0.0, tracking=True)
57     grade_oral = fields.Float(string="Note Présentation (/5)", default=0.0, tracking=True)
58     grade_technical = fields.Float(string="Qualité Technique (/5)", default=0.0, tracking=True)
59
60     grade = fields.Float(string="Note Finale (/20)", compute="_compute_total_grade", store=True, tracking=True)
61
62     state = fields.Selection([
63         ('draft', 'Brouillon'),
64         ('validation', 'Validation'),
65         ('in_progress', 'En cours'),
66         ('submitted', 'Rendu'),
67         ('done', 'Noté')
68     ], string="État", default='draft', tracking=True, group_expand='_expand_states')
69
```

```
projet_etudiant.py X
addons > gestion_projets_etudiants > models > projet_etudiant.py > StudentProject > student_ids
3 class StudentProject(models.Model):
70
71     def _expand_states(self, states, domain, order):
72         return [key for key, val in type(self).state.selection]
73
74     # Workflow Actions
75     def action_draft(self):
76         self.state = 'draft'
77
78     def action_validate(self):
79         self.state = 'validation'
80
81     def action_in_progress(self):
82         self.state = 'in_progress'
83
84     def action_submit(self):
85         self.state = 'submitted'
86
87     def action_done(self):
88         self.state = 'done'
89
90     def action_send_defense_invitation(self):
91         """ Envoie une convocation via le Chatter à tous les participants """
92         for record in self:
93             if not record.defense_date:
94                 raise models.ValidationError("Veuillez définir une date de soutenance avant d'envoyer les convocations.")
95
96             partners = record.student_ids | record.jury_ids | record.supervisor_id
97             if not partners:
98                 raise models.ValidationError("Aucun participant (étudiant, tuteur ou jury) n'est assigné.")
99
100             subject = f"Convocation Soutenance : {record.name}"
101             body = f"Vous êtes invité à la soutenance du projet du {record.defense_date}. Lieu : {record.defense_location or 'Non spécifié'}."
102             record.message_post(body=body, subject=subject, partner_ids=partners.ids)
```

```
projet_etudiant.py X
addons > gestion_projets_etudiants > models > projet_etudiant.py > StudentProject > _check_sub_grades
3 class StudentProject(models.Model):
104     @api.depends('milestone_ids', 'milestone_ids.is_done')
105     def _compute_progress(self):
106         for record in self:
107             if not record.milestone_ids:
108                 record.progress = 0.0
109             else:
110                 total = len(record.milestone_ids)
111                 done = len(record.milestone_ids.filtered(lambda m: m.is_done))
112                 record.progress = (done / total) * 100
113
114     @api.depends('grade_report', 'grade_oral', 'grade_technical')
115     def _compute_total_grade(self):
116         for record in self:
117             record.grade = record.grade_report + record.grade_oral + record.grade_technical
118
119     @api.constrains('grade_report', 'grade_oral', 'grade_technical')
120     def _check_sub_grades(self):
121         for record in self:
122             if not (0 <= record.grade_report <= 10):
123                 raise models.ValidationError("La note du rapport doit être entre 0 et 10.")
124             if not (0 <= record.grade_oral <= 5):
125                 raise models.ValidationError("La note orale doit être entre 0 et 5.")
126             if not (0 <= record.grade_technical <= 5):
127                 raise models.ValidationError("La note technique doit être entre 0 et 5.")
128
```

Fichier : addons/gestion_projets_etudiants/models/project_milestone.py

Ce fichier contient la définition du modèle **ProjectMilestone**, qui représente les sous-tâches ou les livrables obligatoires liés à un projet principal. Il agit comme une table "enfant" dans la base de données, directement reliée au modèle principal `student.project` via une relation Many2one.

Ce fichier gère la granularité du suivi de projet en permettant de définir des étapes clés (jalons) avec leurs propres échéances et descriptions. Il implémente une sécurité de données via l'instruction `ondelete='cascade'`, assurant que si un projet est supprimé, tous ses livrables associés le sont aussi.

De plus, ce modèle intègre les fonctionnalités de soumission de travail. Grâce aux champs `is_done` (booléen) et `file_attachment` (binaire), il permet aux étudiants de marquer une étape comme terminée et de téléverser directement leurs rapports ou codes sources, ce qui alimente ensuite le calcul automatique de la progression globale du projet.

```

project_milestone.py X
addons > gestion_projets_etudiants > models > project_milestone.py > ...
1 from odoo import models, fields, api
2
3 class ProjectMilestone(models.Model):
4     _name = "student.project.milestone"
5     _description = "Jalon / Livrable du Projet"
6     _order = "deadline, id"
7
8     project_id = fields.Many2one('student.project', string="Projet", required=True, ondelete='cascade')
9     name = fields.Char(string="Titre du Livrable", required=True)
10    description = fields.Text(string="Description")
11    deadline = fields.Date(string="Date Limite")
12    is_done = fields.Boolean(string="Terminé")
13    file_attachment = fields.Binary(string="Fichier Joint")
14    file_name = fields.Char(string="Nom du Fichier")
15

```

Etape 5 : Sécurité et Droits d'accès

Cette étape est cruciale pour définir **qui** peut voir **quoi**.

1. Droits d'accès CRUD (CSV) :

Fichier : addons/gestion_projets_etudiants/security/ir.model.access.csv

Ce fichier **ir.model.access.csv** constitue la couche de sécurité essentielle du module (ACL - Access Control Lists). Il fait l'interface entre les modèles de données définis en Python et les groupes d'utilisateurs, dictant précisément les opérations CRUD (Create, Read, Update, Delete) autorisées pour chaque rôle au sein de l'application.

Le fichier configure une hiérarchie de permissions stricte pour protéger l'intégrité des données. Il accorde aux **Étudiants** des droits de travail (lecture, écriture, création) tout en leur interdisant formellement la suppression de projets ou de jalons (le 0 dans la colonne perm_unlink), évitant ainsi les pertes accidentelles. À l'inverse, les **Tuteurs et Managers** se voient attribuer des droits d'administration complets (1,1,1,1), leur permettant de gérer le cycle de vie des données sans restriction.

```

ir.model.access.csv X
addons > gestion_projets_etudiants > security > ir.model.access.csv
1 id,name,model_id:id,group_id:id,perm_read,perm_write,perm_create,perm_unlink
2 access_student_project_student,student.project.student,model_student_project,group_student_project_student,1,1,1,0
3 access_student_project_tutor,student.project.tutor,model_student_project,group_student_project_tutor,1,1,1,1
4 access_student_project_manager,student.project.manager,model_student_project,group_student_project_manager,1,1,1,1
5 access_student_project_milestone_student,student.project.milestone.student,model_student_project_milestone,group_student_project_milestone,1,1,1,0
6 access_student_project_milestone_tutor,student.project.milestone.tutor,model_student_project_milestone,group_student_project_milestone,1,1,1,1
7 access_student_project_milestone_manager,student.project.milestone.manager,model_student_project_milestone,group_student_project_milestone,1,1,1,1

```

2. Définition des groupes et règles :

Fichier : addons/gestion_projets_etudiants/security/security.xml

Ce fichier XML configure l'architecture des **Rôles et Groupes** utilisateurs au sein de l'interface Odoo. Il définit formellement les trois profils d'acteurs du module (Étudiant, Tuteur, Responsable Pédagogique) et établit une hiérarchie de droits par héritage (implied_ids), garantissant que chaque niveau supérieur possède automatiquement toutes les prérogatives du niveau inférieur.

Au-delà de la simple définition des groupes, ce script implémente les **Règles d'Enregistrement (Record Rules)**, qui assurent la confidentialité des données ligne par ligne. Il applique un filtre dynamique (domain_force) qui restreint la vue des étudiants uniquement aux projets où ils sont assignés, tout en accordant aux tuteurs une visibilité globale ([1, '=', 1]) sur l'ensemble des projets pour permettre leur supervision.

```
security.xml x
addons > gestion_projets_etudiants > security > security.xml
1 <?xml version="1.0" encoding="utf-8"?>
2 <odoo>
3   <data>
4     <!-- CATEGORY (Module Category in Users form) -->
5     <record model="ir.module.category" id="module_category_student_project">
6       <field name="name">Gestion des Projets</field>
7       <field name="description">Niveaux d'accès pour les projets étudiants.</field>
8       <field name="sequence">10</field>
9     </record>
10
11     <!-- GROUP: STUDENT -->
12     <record id="group_student_project_student" model="res.groups">
13       <field name="name">Étudiant</field>
14       <field name="category_id" ref="module_category_student_project"/>
15       <field name="implied_ids" eval="[(4, ref('base.group_user'))]"/>
16       <field name="comment">Peut voir uniquement ses propres projets.</field>
17     </record>
18
19     <!-- GROUP: TUTOR (TEACHER) -->
20     <record id="group_student_project_tutor" model="res.groups">
21       <field name="name">Tuteur / Professeur</field>
22       <field name="category_id" ref="module_category_student_project"/>
23       <field name="implied_ids" eval="[(4, ref('group_student_project_student'))]"/>
24       <field name="comment">Peut voir tous les projets et les valider.</field>
25     </record>
26
27     <!-- GROUP: MANAGER (ADMIN) -->
28     <record id="group_student_project_manager" model="res.groups">
29       <field name="name">Responsable Pédagogique</field>
30       <field name="category_id" ref="module_category_student_project"/>
31       <field name="implied_ids" eval="[(4, ref('group_student_project_tutor'))]"/>
32       <field name="users" eval="[(4, ref('base.user_root')), (4, ref('base.user_admin'))]"/>
33     </record>
34
```

```
security.xml X
addons > gestion_projets_etudiants > security > security.xml
2  <odoo>
3  <data>
34
35      <!-- RECORD RULES (Row Level Security) -->
36
37      <!-- Student Rule: Can only see projects where they are assigned -->
38      <record id="rule_student_project_student_own" model="ir.rule">
39          <field name="name">Projets Étudiant: Seulement les siens</field>
40          <field name="model_id" ref="model_student_project"/>
41          <field name="groups" eval="[(4, ref('group_student_project_student'))]" />
42          <field name="domain_force">[('student_ids', 'in', user.partner_id.id)]</field>
43          <field name="perm_read" eval="True"/>
44          <field name="perm_write" eval="True"/>
45          <field name="perm_create" eval="True"/>
46          <field name="perm_unlink" eval="False"/> <!-- Students usually can't delete -->
47      </record>
48
49      <!-- Tutor Rule: See ALL projects (or refine to supervised only if needed, currently ALL for visibility) -->
50      <record id="rule_student_project_tutor_all" model="ir.rule">
51          <field name="name">Projets Tuteur: Tous</field>
52          <field name="model_id" ref="model_student_project"/>
53          <field name="groups" eval="[(4, ref('group_student_project_tutor'))]" />
54          <field name="domain_force">[(1, '=', 1)]</field>
55          <field name="perm_read" eval="True"/>
56          <field name="perm_write" eval="True"/>
57          <field name="perm_create" eval="True"/>
58          <field name="perm_unlink" eval="True"/>
59      </record>
60
61  </data>
62 </odoo>
```

Etape 6 : Interface Utilisateur (Vues)

Nous définissons ici les vues Kanban, Liste, Formulaire et Recherche etc.

Fichier : addons/gestion_projets_etudiants/views/projet_etudiant_views.xml

Ce fichier XML est l'architecte de l'**interface utilisateur (UI)** du module. Il définit l'ensemble des vues interactives (Kanban, Liste, Formulaire, Calendrier) qui permettent aux utilisateurs de gérer les projets, ainsi que la structure de navigation (Menus et Actions) nécessaire pour accéder à ces écrans.

Le code configure une expérience riche en intégrant des **composants visuels dynamiques** tels que la barre d'état du workflow, des indicateurs de couleur pour les retards et une jauge pour la note finale. De plus, il inclut des vues décisionnelles (Graphique et Pivot) qui offrent aux responsables un tableau de bord statistique instantané sur la répartition et la notation des projets.

```

proj etudiant_views.xml X
addons > gestion_projets_etudiants > views > projet_etudiant_views.xml
1  <?xml version="1.0" encoding="UTF-8"?>
2  <odoo>
3
4      <!-- SEARCH VIEW -->
5      <record id="view_student_project_search" model="ir.ui.view">
6          <field name="name">student.project.search</field>
7          <field name="model">student.project</field>
8          <field name="arch" type="xml">
9              <search>
10                 <field name="name"/>
11                 <field name="student_ids"/>
12                 <field name="supervisor_id"/>
13                 <field name="project_type"/>
14                 <filter string="Mes Projets" name="my_projects" domain="[('student_ids', 'in', uid)]"/>
15                 <filter string="PFE Uniquement" name="pfe_only" domain="[('project_type', '=', 'pfe')]/>
16                 <filter string="Projets non notés" name="not_graded" domain="[('grade', '=', 0), ('state', '=', 'done')]/>
17                 <group expand="1" string="Regrouper par">
18                     <filter string="Tuteur" name="groupby_supervisor" context="{ 'group_by': 'supervisor_id' }"/>
19                     <filter string="État" name="groupby_state" context="{ 'group_by': 'state' }"/>
20                     <filter string="Type" name="groupby_type" context="{ 'group_by': 'project_type' }"/>
21                 </group>
22             </search>
23         </field>
24     </record>

```

```

proj etudiant_views.xml X
addons > gestion_projets_etudiants > views > projet_etudiant_views.xml
2  <odoo>
26
27      <!-- KANBAN VIEW -->
28      <record id="view_student_project_kanban" model="ir.ui.view">
29          <field name="name">student.project.kanban</field>
30          <field name="model">student.project</field>
31          <field name="arch" type="xml">
32              <kanban default_group_by="state" class="o_kanban_small_column" on_create="quick_create">
33                  <field name="name"/>
34                  <field name="student_ids"/>
35                  <field name="grade"/>
36                  <field name="project_type"/>
37                  <field name="state"/>
38                  <field name="id"/>
39                  <field name="activity_ids"/>
40                  <field name="activity_state"/>
41                  <templates>
42                      <t t-name="kanban-box">
43                          <div t-attf-class="oe_kanban_global_click oe_kanban_card">
44                              <div class="oe_kanban_details">
45                                  <strong class="o_kanban_record_title">
46                                      <field name="name"/>
47                                  </strong>
48                                  <div>
49                                      <field name="student_ids" widget="many2many_tags" options="{ 'color_field': 'color' }"/>
50                                  </div>
51                                  <div class="o_kanban_record_bottom mt-2">
52                                      <div class="oe_kanban_bottom_left">
53                                          <field name="activity_ids" widget="kanban_activity"/>
54                                          <field name="project_type" widget="label_selection"/>
55                                      </div>
56                                      <div class="oe_kanban_bottom_right">
57                                          <field name="progress" widget="progressbar"/>
58                                          <span t-if="record.grade.raw_value > 0" class="badge rounded-pill text-bg-success">
59                                              <field name="grade"/>/20
60                                          </span>
61                                      </div>
62                                  </div>
63                              </div>
64                          </t>
65                      </templates>
66                  </kanban>
67              </field>
68          </record>

```

```
proj_etudiant_views.xml X
addons > gestion_projets_etudiants > views > projet_etudiant_views.xml
2 <odoo>
69
70 <!-- TREE VIEW -->
71 <record id="view_student_project_tree" model="ir.ui.view">
72   <field name="name">student.project.tree</field>
73   <field name="model">student.project</field>
74   <field name="arch" type="xml">
75     <tree decoration-danger="date_deadline &lt; current_date and state != 'done'"
76           decoration-success="state == 'done'"
77           decoration-info="state == 'in_progress'">
78       <field name="name"/>
79       <field name="student_ids" widget="many2many_tags"/>
80       <field name="supervisor_id"/>
81       <field name="project_type"/>
82       <field name="date_deadline"/>
83       <field name="progress" widget="progressbar" optional="show"/>
84       <field name="defense_date" optional="hide"/>
85       <field name="grade" optional="show"/>
86       <field name="state" widget="badge"
87             decoration-success="state == 'done'"
88             decoration-info="state == 'in_progress'"
89             decoration-warning="state == 'submitted'"/>
90     </tree>
91   </field>
92 </record>
93
```

```
proj_etudiant_views.xml X
addons > gestion_projets_etudiants > views > projet_etudiant_views.xml
2 <odoo>
94 <!-- FORM VIEW -->
95 <record id="view_student_project_form" model="ir.ui.view">
96   <field name="name">student.project.form</field>
97   <field name="model">student.project</field>
98   <field name="arch" type="xml">
99     <form>
100       <header>
101         <button name="action_validate" string="Valider" type="object" class="oe_highlight" invisible="state != 'draft'"/>
102         <button name="action_in_progress" string="Commencer" type="object" class="oe_highlight" invisible="state != 'validation'"/>
103         <button name="action_submit" string="Rendre le Projet" type="object" class="oe_highlight" invisible="state != 'in_progress'"/>
104         <button name="action_done" string="Noter & Clôturer" type="object" class="oe_highlight" invisible="state != 'submitted'"/>
105         <button name="action_send_defense_invitation" string="Envoyer Convocation" type="object" invisible="not defense_date"/>
106         <button name="action_draft" string="Remettre en Brouillon" type="object" invisible="state == 'draft'"/>
107         <field name="state" widget="statusbar" statusbar_visible="draft,validation,in_progress,submitted,done"/>
108       </header>
109       <sheet>
110         <div class="oe_title">
111           <label for="name" string="Titre du Projet"/>
112           <h1>
113             <field name="name" placeholder="Ex: Mise en place d'un Pipeline DevOps..."/>
114           </h1>
115         </div>
116         <group>
117           <group>
118             <field name="student_ids" widget="many2many_tags"/>
119             <field name="supervisor_id"/>
120             <field name="project_type"/>
121           </group>
122           <group>
123             <field name="date_deadline"/>
124             <field name="repository_url" widget="url"/>
125             <field name="grade" widget="progressbar" options="{ 'max_value': 20 }" readonly="1"/>
126           </group>
127         </group>
128
```

```
proj etudiant_views.xml X
addons > gestion_projets_etudiants > views > projet_etudiant_views.xml
2 <odoo>
95 <record id="view_student_project_form" model="ir.ui.view">
98 <field name="arch" type="xml">
99 <form>
100 <sheet>
129 <notebook>
130 <page string="Description & Détails">
131 <field name="description" placeholder="Détails du sujet, contraintes techniques..."/>
132 </page>
133 <page string="Soutenance & Jury" invisible="state == 'draft'">
134 <group>
135 <group>
136 <field name="defense_date"/>
137 <field name="defense_location"/>
138 </group>
139 <group>
140 <field name="jury_ids" widget="many2many_tags"/>
141 </group>
142 </group>
143 <group string="Evaluation" invisible="state not in ['submitted', 'done']">
144 <group>
145 <field name="grade_report"/>
146 <field name="grade_oral"/>
147 </group>
148 <group>
149 <field name="grade_technical"/>
150 <field name="grade" readonly="1" decoration-bf="1" string="Total / 20" widget="gauge" style="width:120px; height:90px; cursor:pointer;" />
151 </group>
152 </group>
153 </page>
154 <page string="Livrables & Jalons">
155 <group>
156 <field name="progress" widget="progressbar"/>
157 </group>
158 </page>
159 </notebook>
160 </sheet>
161 </form>
162 </record>
163 </odoo>
```

```
proj etudiant_views.xml X
add: D:\SIR\erp\mini-projet\mini-projet-erp\addons\gestion_projets_etudiants\views\projet_etudiant_views.xml (preview @)
2 <record id="view_student_project_form" model="ir.ui.view">
98 <field name="arch" type="xml">
99 <form>
100 <sheet>
158 <field name="milestone_ids">
159 <tree editable="bottom" decoration-success="is_done">
160 <field name="is_done"/>
161 <field name="name"/>
162 <field name="deadline"/>
163 <field name="file_attachment" filename="file_name" widget="binary"/>
164 <field name="file_name" column_invisible="True"/>
165 </tree>
166 </field>
167 </page>
168 </notebook>
169 </sheet>
170 <!-- CHATTER -->
171 <div class="oe_chatter">
172 <field name="message_follower_ids"/>
173 <field name="activity_ids"/>
174 <field name="message_ids"/>
175 </div>
176 </form>
177 </field>
178 </record>
179 </odoo>
180 <!-- CALENDAR VIEW -->
181 <record id="view_student_project_calendar" model="ir.ui.view">
182 <field name="name">student.project.calendar</field>
183 <field name="model">student.project</field>
184 <field name="arch" type="xml">
185 <calendar string="Calendrier des Soutenances" date_start="defense_date" color="project_type" mode="month">
186 <field name="name"/>
187 <field name="student_ids"/>
188 <field name="defense_location"/>
189 </calendar>
```

proj et_etudiant_views.xml X

addons > gestion_projets_etudiants > views > projet_etudiant_views.xml

```
2 <odoo>
191 </record>
192
193 <!-- GRAPH VIEW -->
194 <record id="view_student_project_graph" model="ir.ui.view">
195   <field name="name">student.project.graph</field>
196   <field name="model">student.project</field>
197   <field name="arch" type="xml">
198     <graph string="Statistiques Projets" type="bar">
199       <field name="project_type"/>
200       <field name="grade" type="measure"/>
201     </graph>
202   </field>
203 </record>
204
205 <!-- PIVOT VIEW -->
206 <record id="view_student_project_pivot" model="ir.ui.view">
207   <field name="name">student.project.pivot</field>
208   <field name="model">student.project</field>
209   <field name="arch" type="xml">
210     <pivot string="Analyse des Projets">
211       <field name="project_type" type="row"/>
212       <field name="supervisor_id" type="row"/>
213       <field name="state" type="col"/>
214       <field name="grade" type="measure"/>
215     </pivot>
216   </field>
217 </record>
218
```

2 <odoo>

217 </record>

218

219 <!-- ACTIONS -->

220 <record id="action_student_project" model="ir.actions.act_window">

221 <field name="name">Projets Étudiants</field>

222 <field name="res_model">student.project</field>

223 <field name="view_mode">kanban,tree,form,calendar</field>

224 <field name="help" type="html">

225 <p class="o_view_nocontent_smiling_face">

226 Créez le premier projet étudiant !

227 </p>

228 </field>

229 </record>

230

231 <!-- MENUS -->

232 <menuitem id="menu_school_root" name="Student Project" sequence="10"/>

233

234 <menuitem id="menu_student_projects"

235 name="Mes Projets"

236 parent="menu_school_root"

237 action="action_student_project"

238 sequence="10"/>

239

240 <!-- REPORTING MENU -->

241 <record id="action_student_project_reporting" model="ir.actions.act_window">

242 <field name="name">Tableau de Bord</field>

243 <field name="res_model">student.project</field>

244 <field name="view_mode">graph,pivot</field>

245 <field name="help" type="html">

246 <p class="o_view_nocontent_smiling_face">

247 Aucune donnée pour les statistiques actuellement.

248 </p>

249 </field>

250 </record>

251

252 <menuitem id="menu_student_project_reporting"

253 name="Statistiques"

254 parent="menu_school_root"

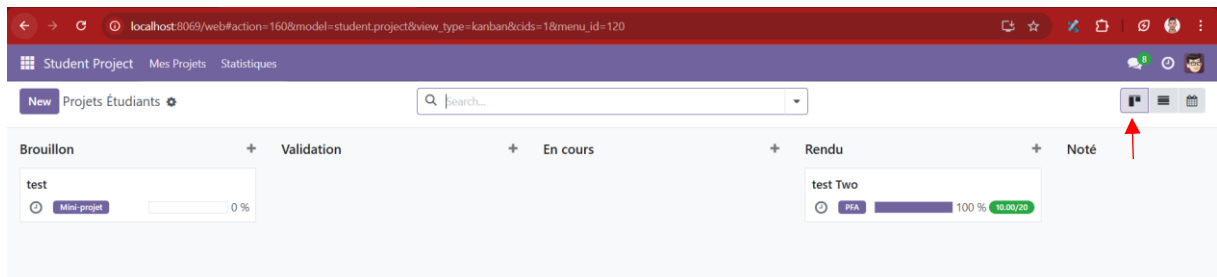
255 action="action_student_project_reporting"

256 sequence="20"/>

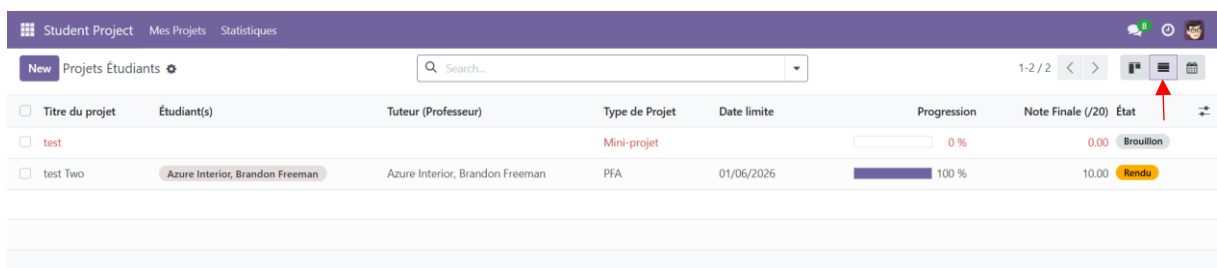
257

258 </odoo>

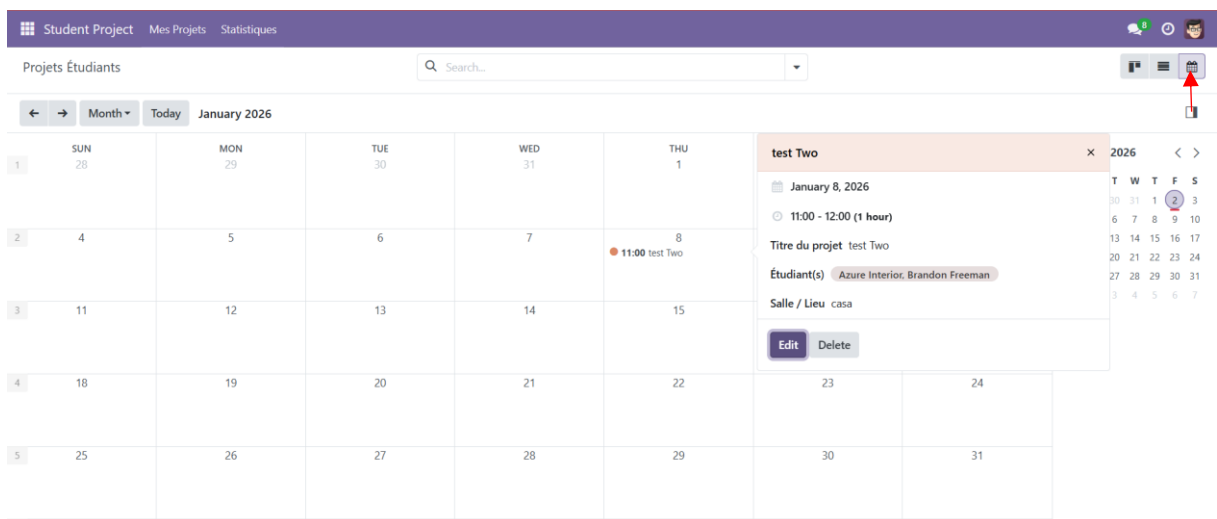
Étape 7 : Tester le module



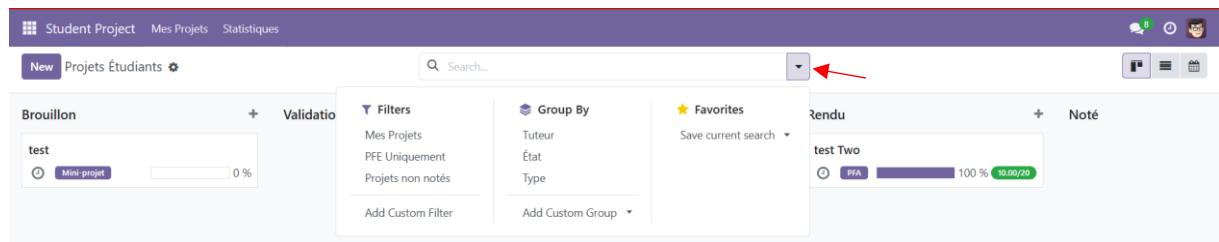
Vue Kanban (Pipeline) : Tableau de bord visuel organisant les projets par colonnes selon leur étape dans le workflow (Brouillon, En cours, Rendu...), offrant une vue d'ensemble immédiate de l'avancement global.



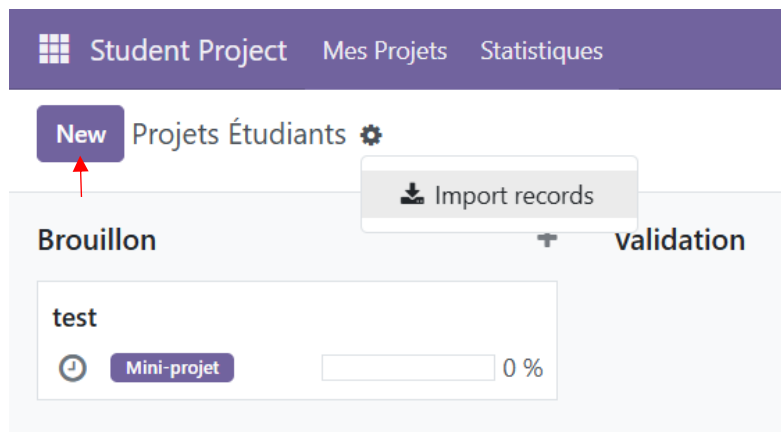
Vue Liste Détaillée : Affichage tabulaire complet des projets permettant de comparer rapidement les informations clés (Type, Tuteur, Date limite) et de visualiser la progression technique et la note finale via des indicateurs graphiques.



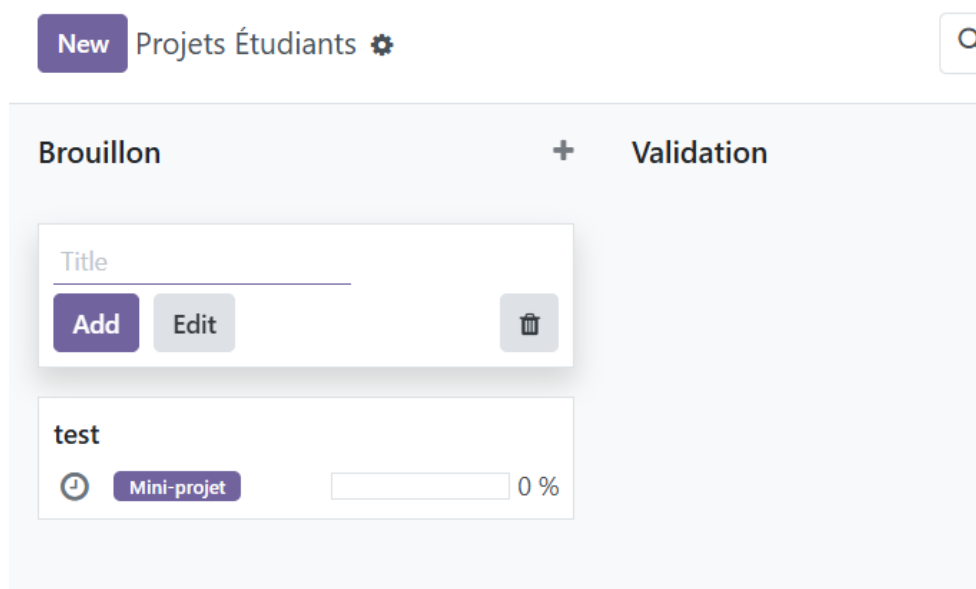
Vue Calendrier des Soutenances : Interface de planification visuelle permettant de consulter les dates de soutenance. Le pop-up affiche les détails essentiels (horaire, étudiants concernés, lieu) sans avoir à ouvrir la fiche projet.



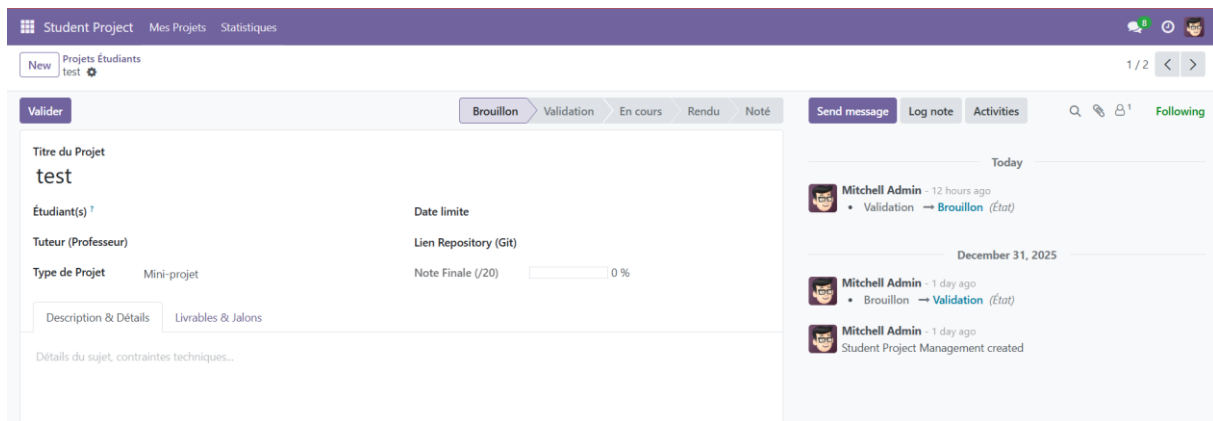
Outils de Recherche Avancée : Démonstration des capacités de tri avec la barre de recherche déployée, montrant les filtres prédéfinis (ex: "PFE Uniquement") et les options de regroupement pour organiser les données efficacement.



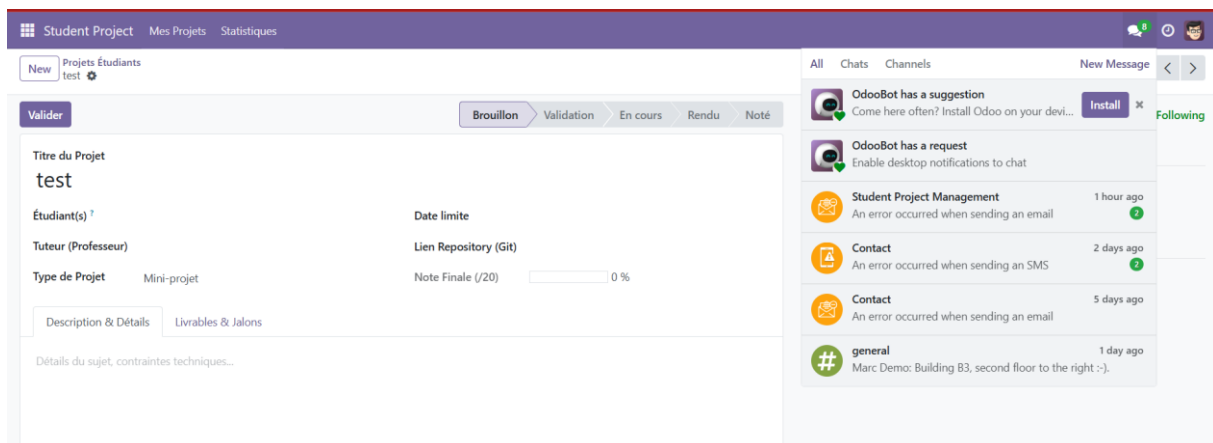
Fonctionnalité d'importation : Aperçu du menu permettant l'importation de masse (via fichiers CSV ou Excel), facilitant la création rapide de multiples fiches projets sans saisie manuelle.



Quick Create (Kanban) : Fonctionnalité permettant d'ajouter un nouveau projet à la volée directement depuis la colonne "Brouillon" du Kanban, en saisissant simplement le titre, sans ouvrir le formulaire complet.



Interface de Notation et Chatter : Vue complète d'un projet "Rendu". À gauche, la grille de notation (Rapport/Oral/Technique) calculant la note finale. À droite, le "Chatter" qui historique toutes les actions et communications.



Affiche le menu des messages système, alertant l'utilisateur des erreurs (ex: envoi d'email échoué) ou des actions requises.

Student Project

Mes Projets

Statistiques

New

Projets Étudiants

test

Valider

Brouillon

Val

Titre du Projet

test

Étudiant(s) ?

Tuteur (Professeur)

Type de Projet

Description & Détails

Détails du sujet, contenu

Azure Interior

Azure Interior, Brandon Freeman

Azure Interior, Colleen Diaz

Azure Interior, Nicole Ford

Deco Addict

Deco Addict, Addison Olson

Deco Addict, Douglas Fletcher

Deco Addict, Floyd Steward

Search More...

Start typing...

Date limite

Lien Repository (Git)

Note Finale (/20)

Démonstration du champ de sélection multiple (Many2many) permettant d'assigner facilement un groupe d'étudiants au projet.

Student Project

Mes Projets

Statistiques

New

Projets Étudiants

test

Valider

Brouillon

Validation

En cours

Rendu

Noté

Titre du Projet

test

Étudiant(s) ?

Date limite

Tuteur (Professeur)

Lien Repository (Git)

Type de Projet

Mini-projet

Note Finale (/20)

0 %

Description & Détails

Livrables & Jalons

Progression

0 %

Ter...

Titre du Livrable

Date Lim...

Fichier Joint

☐

Upload your file

Add a line

Onglet Livrables & Jalons : Interface permettant de définir les étapes techniques du projet. C'est ici que les étudiants peuvent téléverser leurs fichiers, ce qui met à jour automatiquement la barre de progression globale.

Student Project Mes Projets Statistiques

New Projets Étudiants test Two 2 / 2

Noter & Clôturer Envoyer Convocation Remettre en Brouillon Brouillon Validation En cours Rendu Noté

Titre du Projet
test Two

Étudiant(s) ? Azure Interior, Brandon Freeman X

Tuteur (Professeur) Azure Interior, Brandon Freeman

Type de Projet PFA

Date limite 01/06/2026

Lien Repository (Git)

Note Finale (/20) 10 %

Description & Détails Soutenance & Jury Livrables & Jalons

Date de Soutenance 01/08/2026 11:00:00

Membres du Jury ? Azure Interior, Colleen Diaz X

Salle / Lieu casa

ÉVALUATION

Note Rapport (/10)	10.00	Qualité Technique (/5)	0.00
Note Présentation (/5)	0.00	Total / 20	Note Finale (/20)

10

Send message Log note Activities

Today

Mitchell Admin - 52 minutes ago

- 4.00 → 10.00 (Note Finale (/20))
- 0.00 → 10.00 (Note Rapport (/10))

Mitchell Admin - 1 hour ago

Vous êtes invité à la soutenance du projet du 2026-01-08 10:00:00. Lieu : casa.

Mitchell Admin - 1 hour ago

- None → 01/08/2026 11:00:00 (Date de Soutenance)

Mitchell Admin - 12 hours ago

- Validation → Rendu (État)

Mitchell Admin - 12 hours ago

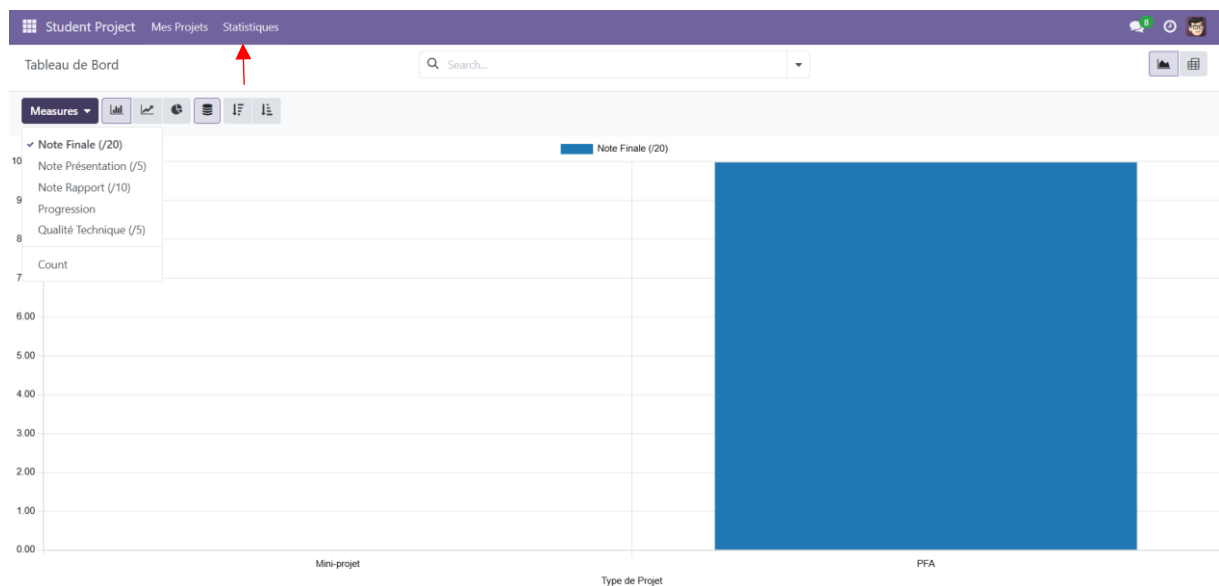
- Brouillon → Validation (État)

Mitchell Admin - 12 hours ago

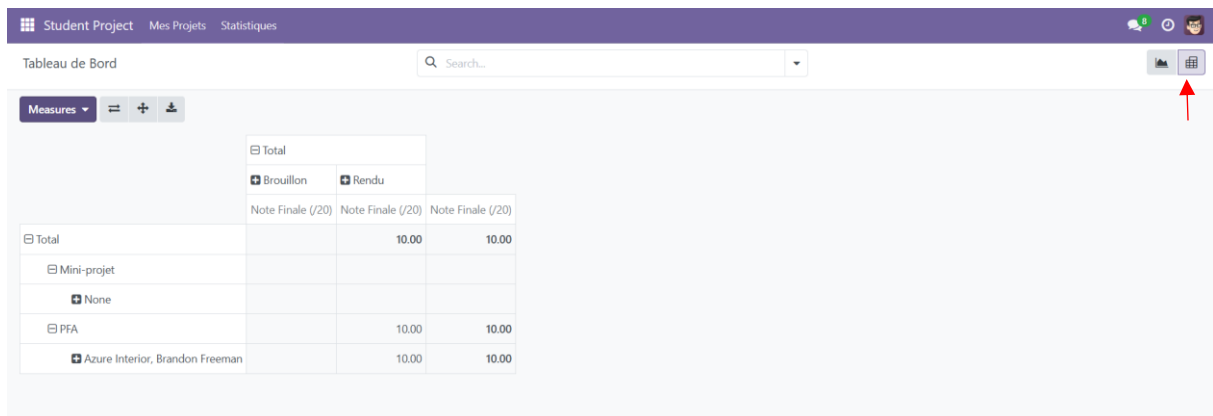
- None → 01/06/2026 (Date limite)
- 0.00 → 4.00 (Note Finale (/20))
- Mini-projet → PFA (Type de Projet)
- None → Azure Interior, Brandon Freeman (Tuteur (Professeur))

Mitchell Admin - 12 hours ago

L'écran permet de gérer toute la logistique de la soutenance (date, lieu, membres du jury) et d'accéder à la **grille d'évaluation**. On y voit le calcul détaillé de la note (Rapport/Oral/Technique) synthétisé par une jauge visuelle dynamique. Sur la droite, le **Chatter** (historique) joue un rôle crucial de traçabilité, enregistrant chaque modification de note, changement d'état et l'envoi automatique des emails de convocation aux participants.



Présente l'outil de reporting visuel, ici un diagramme en barres comparant la "Note Finale" par "Type de Projet", avec un menu permettant de changer les mesures analysées.



Montre la vue tableau croisé dynamique, idéale pour les responsables pédagogiques afin d'analyser les performances (notes moyennes) regroupées par tuteur, type de projet et état d'avancement.

Name	Login	Language	Latest authentication	Status
Marc Demo	demo	English (US)		Never Connected
Mitchell Admin	admin	English (US)	01/01/2026 23:38:09	Confirmed
Professeur1	professeur1@gmail.com	English (US)		Never Connected
student1	student1@gmail.com	English (US)	01/01/2026 23:37:10	Confirmed

Affiche la liste des utilisateurs du système, permettant à l'administrateur de gérer les comptes (étudiants, professeurs) et de vérifier leur statut de connexion.

Send an Invitation Email

Name: Professeur1

Email Address: professeur1@gmail.com

Access Rights: Preferences: Account Security

ADMINISTRATION: Administration

OTHER: Gestion des Projets: Tuteur / Professeur

Étudiant

Tuteur / Professeur

Responsable Pédagogique

Illustre la fiche de configuration d'un utilisateur où l'on attribue son niveau de droits spécifique pour le module (Étudiant, Tuteur ou Responsable Pédagogique).