

Gestion des inscriptions :

Inscriptions simples



malak ifiss

Résumé :

Ce projet consiste en la conception et le développement d'un module personnalisé sous **Odoo 17** dédié à la **gestion des inscriptions simples**. L'objectif principal est de proposer une solution numérique permettant de centraliser les données des inscrits, de faciliter leur création, leur consultation et leur suivi à travers un système de statuts (Nouveau, Validé, Annulé).

Le module a été développé en respectant l'architecture standard d'Odoo et en utilisant des technologies telles que Python pour la logique métier, XML pour l'interface utilisateur, PostgreSQL pour la gestion des données et Docker pour l'environnement d'exécution. Ce projet, réalisé dans un cadre académique, a permis de renforcer la compréhension pratique du développement de modules Odoo ainsi que des principes fondamentaux des systèmes ERP.

Mots-clés :

Odoo 17, ERP, gestion des inscriptions, module personnalisé, Python, XML, Docker, PostgreSQL

Abstract:

This project focuses on the design and development of a custom module using Odoo 17 for simple registration management. The main objective is to provide a digital solution that centralizes registration data and facilitates their creation, consultation, and monitoring through a status-based workflow (New, Validated, Cancelled). The module was developed following Odoo's standard architecture and using technologies such as Python for business logic, XML for the user interface, PostgreSQL for data management, and Docker for the execution environment. Conducted in an academic context, this project helped strengthen practical skills in Odoo module development and enhanced the understanding of core ERP concepts.

Keywords:

Odoo 17, ERP, registration management, custom module, Python, XML, Docker, PostgreSQL

Table des matières

Résumé	1
Mots-clés :	1
Abstract:	2
Keywords:.....	2
Table des figures :	4
INTRODUCTION GÉNÉRALE.....	5
CHAPITRE 1 : PRÉSENTATION DU PROJET.....	6
1.1 Contexte du projet.....	6
1.2 Problématique.....	6
1.3 Objectifs du projet	6
Conclusion	7
CHAPITRE 2 : ARCHITECTURE ET TECHNOLOGIES UTILISÉES.....	8
2.1 Architecture générale de l'application	8
2.2 Structure du module.....	8
2.3 Technologies utilisées	9
2.3.1 Odoo 17 – Framework ERP Open Source.....	9
2.3.2 Python – Logique métier et modèles	9
2.3.3 XML – Interface utilisateur	9
2.3.4 PostgreSQL – Base de données	9
2.3.5 Docker – Environnement d'exécution.....	10
2.3.6.Git (Git Bash)	10
2.4 Réalisation du projet.....	10
2.4.1. Interface de connexion à Odoo	10
2.4.2. Redémarrage de l'environnement Odoo via Docker	11
2.4.3. Docker Desktop – Images, Volumes et Conteneurs	12
2.4.4.Home page	13
2.4.5.Vue liste des inscriptions.....	14
2.4.6. Formulaire de création d'une nouvelle inscription.....	14
2.4.7. Formulaire d'une inscription validée	15
2.4.8. Conclusion	15
CONCLUSION GÉNÉRALE	16

Table des figures :

Figure 1:Architetcture du projet	8
Figure 2:logo de odoo.....	9
Figure 3:logo de python	9
Figure 4:logo de xml	9
Figure 5:logo de PostgreSQL.....	9
Figure 6:logo de docker	10
Figure 7:logo de git.....	10
Figure 8: Interface de connexion à Odoo.....	11
Figure 9 :Redémarrage de l'environnement Odoo via Docker.....	11
Figure 10:docker container	12
Figure 11:docker images	12
Figure 12:docker volumes	12
Figure 13:Home page	13
Figure 14:Vue liste des inscriptions	14
Figure 15:Formulaire de création d'une nouvelle inscriptionj.....	14
Figure 16:Formulaire d'une inscription validée	15

INTRODUCTION GÉNÉRALE

La transformation numérique des organisations a profondément modifié la manière dont les processus administratifs sont conçus et gérés. Les systèmes de gestion intégrés (ERP) jouent aujourd'hui un rôle central dans l'automatisation des tâches, la centralisation des données et l'amélioration de la productivité des utilisateurs.

Dans ce contexte, Odoo s'impose comme une solution ERP open source largement adoptée, offrant une architecture modulaire et flexible permettant le développement d'applications métiers personnalisées.

Ce projet académique s'inscrit dans le cadre de l'apprentissage du développement de modules personnalisés sous Odoo 17. Il consiste à concevoir et implémenter un module simple mais fonctionnel dédié à la gestion des inscriptions. Le module permet de créer, consulter, modifier et suivre des inscriptions via une interface intuitive intégrée à Odoo.

Ce projet met en pratique les concepts fondamentaux d'Odoo tels que la définition des modèles, la création des vues, la gestion des menus, les actions métier et la gestion des états (workflow).

CHAPITRE 1 : PRÉSENTATION DU PROJET

1.1 Contexte du projet

Dans de nombreux établissements (écoles, centres de formation, organisations ou événements), la gestion des inscriptions est souvent réalisée de manière manuelle ou à l'aide de fichiers dispersés (Excel, documents papier). Cette approche entraîne des problèmes de suivi, des erreurs de saisie et un manque de centralisation des informations.

L'objectif de ce projet est de proposer une solution numérique simple intégrée à un ERP permettant de centraliser la gestion des inscriptions et de faciliter leur suivi à travers différents états (nouvelle, validée, annulée).

Ce projet a été réalisé dans un cadre académique afin de renforcer la maîtrise pratique du framework Odoo et de ses mécanismes internes

1.2 Problématique

La gestion manuelle ou semi-automatisée des inscriptions présente plusieurs limites :

- Absence de centralisation des données des inscrits
- Difficulté de suivi de l'état des inscriptions
- Risque d'erreurs lors de la saisie des informations
- Perte de temps dans le traitement administratif

La problématique principale est donc la suivante :

Comment concevoir un module Odoo simple, fiable et intuitif permettant de gérer efficacement les inscriptions et leur cycle de vie au sein d'un système ERP ?

1.3 Objectifs du projet

Les objectifs principaux de ce projet sont :

- Concevoir un module Odoo personnalisé pour la gestion des inscriptions
- Permettre la création, la modification et la consultation des inscriptions
- Mettre en place un système de statuts (Nouveau / Validé / Annulé)
- Améliorer l'interface utilisateur pour une meilleure expérience
- Implémenter une logique métier simple à travers des actions (valider, annuler)
- Appliquer les bonnes pratiques de développement sous Odoo 17

Conclusion

À travers ce premier chapitre, le contexte général du projet a été présenté ainsi que la problématique liée à la gestion manuelle ou semi-automatisée des inscriptions. Cette analyse a permis de mettre en évidence les limites des méthodes traditionnelles, notamment le manque de centralisation des données, les difficultés de suivi et les risques d'erreurs lors du traitement administratif.

Face à ces constats, les objectifs du projet ont été clairement définis afin de concevoir un module Odoo simple, fiable et intuitif, capable de répondre aux besoins de gestion des inscriptions. Ce chapitre a ainsi permis de poser les bases conceptuelles du projet et de justifier le choix d'une solution ERP comme Odoo pour digitaliser et optimiser ce processus.

Les éléments présentés dans ce chapitre constituent le socle de réflexion ayant guidé la conception et le développement du module, qui seront détaillés dans les chapitres suivants à travers l'architecture, les technologies utilisées et la réalisation concrète du projet.

CHAPITRE 2 : ARCHITECTURE ET TECHNOLOGIES UTILISÉES

2.1 Architecture générale de l'application

Le module développé repose sur l'architecture standard d'Odoo, inspirée du modèle MVC (Model – View – Controller) :

- Model : définition des données et de la logique métier des inscriptions
- View : interface utilisateur (vues liste et formulaire)
- Actions : interaction entre l'utilisateur et le système (boutons, workflows)

Le module est intégré sous forme d'un add-on indépendant, facilitant sa maintenance et son évolution.

2.2 Structure du module

Le module est organisé selon la structure standard recommandée par Odoo :

- models/ : définition du modèle tp.inscription
- views/ : vues XML (liste et formulaire des inscriptions)
- security/ : gestion des droits d'accès
- __manifest__.py : description et configuration du module

Cette structure assure une séparation claire des responsabilités et améliore la lisibilité du code.

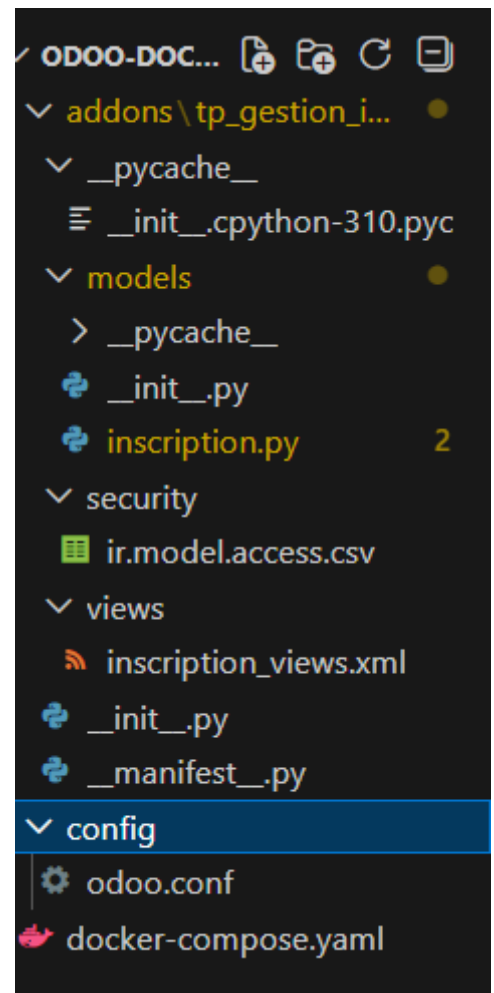


Figure 1: Architecture du projet

2.3 Technologies utilisées

2.3.1 Odoo 17 – Framework ERP Open Source

Odoo 17 est un framework ERP open source puissant et modulaire permettant de développer des applications métiers personnalisées. Il fournit des fonctionnalités avancées telles que la gestion des utilisateurs, des droits d'accès, des vues dynamiques et des workflows métiers.



Figure 2: logo de odoo

Grâce à son architecture modulaire, Odoo permet d'ajouter facilement de nouvelles fonctionnalités sous forme de modules, comme le module de gestion des inscriptions développé dans ce projet.

2.3.2 Python – Logique métier et modèles

Python est le langage principal utilisé pour définir les modèles et la logique métier sous Odoo.

Dans ce projet, Python est utilisé pour :

- Définir le modèle des inscriptions
- Gérer les états (nouveau, validé, annulé)
- Implémenter les actions métier (valider, annuler)
- Assurer la cohérence et la validation des données



Figure 3: logo de python

2.3.3 XML – Interface utilisateur

Le langage XML est utilisé pour définir l'interface graphique du module, notamment :

- Les vues formulaire et liste
- Les menus de navigation
- Les actions associées au modèle



XML permet de structurer l'interface utilisateur de manière claire et cohérente, sans modifier la logique métier.

Figure 4: logo de xml

2.3.4 PostgreSQL – Base de données

PostgreSQL est le système de gestion de base de données relationnelle utilisé par Odoo.

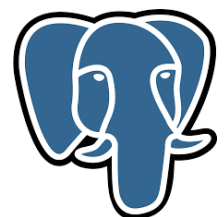


Figure 5: logo de PostgreSQL

Il assure le stockage fiable et performant des données relatives aux inscriptions.

2.3.5 Docker – Environnement d'exécution

Docker est utilisé pour déployer et exécuter l'environnement Odoo de manière isolée. Il permet :

- Une installation simplifiée
- Une meilleure portabilité du projet
- Un environnement stable et reproductible



Figure 6: logo de docker

2.3.6. Git (Git Bash)

Dans le cadre de ce projet, Git a été utilisé principalement à travers Git Bash comme interface en ligne de commande. Git Bash a servi d'environnement pour exécuter les commandes nécessaires à la gestion et au déploiement de l'application Odoo via Docker. Grâce à cet outil, il a été possible de lancer des commandes telles que le démarrage des conteneurs, l'arrêt des services, le redémarrage de l'environnement ou encore la mise à jour des modules Odoo. L'utilisation de Git Bash a permis de travailler dans un terminal compatible avec les commandes Linux, même sous Windows, facilitant ainsi l'exécution des scripts Docker et la gestion du projet de manière efficace et structurée.



Figure 7: logo de git

2.4 Réalisation du projet

2.4.1. Interface de connexion à Odoo

Cette figure illustre l'interface de connexion à la plateforme Odoo. L'utilisateur est invité à saisir son identifiant (email ou nom d'utilisateur) ainsi que son mot de passe afin d'accéder au système. Cette étape constitue le point d'entrée sécurisé vers l'ERP et permet de garantir que seules les personnes autorisées peuvent accéder aux fonctionnalités du module de gestion des inscriptions.

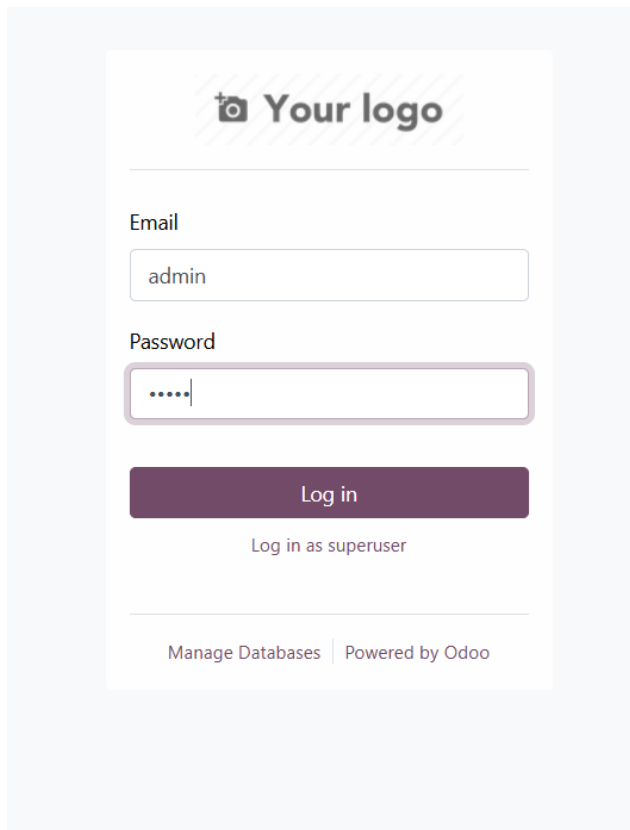


Figure 8: Interface de connexion à Odoo

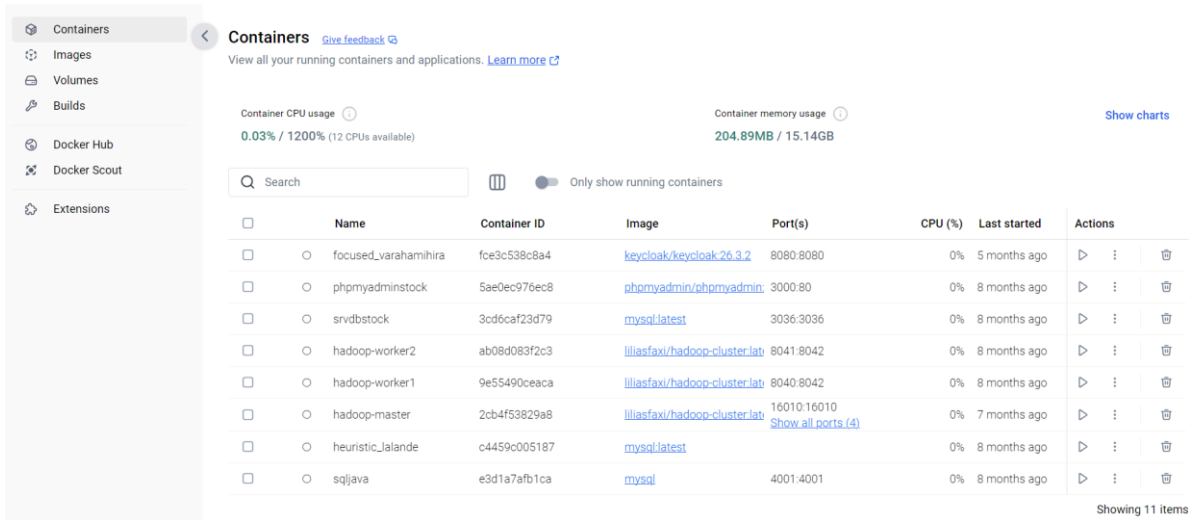
2.4.2. Redémarrage de l'environnement Odoo via Docker

Cette capture montre l'exécution de la commande docker compose restart dans le terminal. Elle confirme le redémarrage réussi des conteneurs Odoo et PostgreSQL. Cette étape est essentielle après toute modification du code du module afin de recharger les changements et d'assurer leur prise en compte par le système.

```
malak@thinkpad MINGW64 ~/OneDrive/Bureau/odoo-docker (main)
$ docker compose restart
time="2026-01-01T16:53:00+01:00" level=warning msg="C:\\Users\\malak\\OneDrive\\Bureau\\odoo-docker\\docker-compose.yaml: the attribute 'version' is obsolete and it will be ignored, please remove it to avoid potential confusion"
[+] Restarting 2/2
  ✓ Container odoo_app Started
  ✓ Container odoo_db Started
```

Figure 9 :Redémarrage de l'environnement Odoo via Docker

2.4.3. Docker Desktop – Images, Volumes et Conteneurs

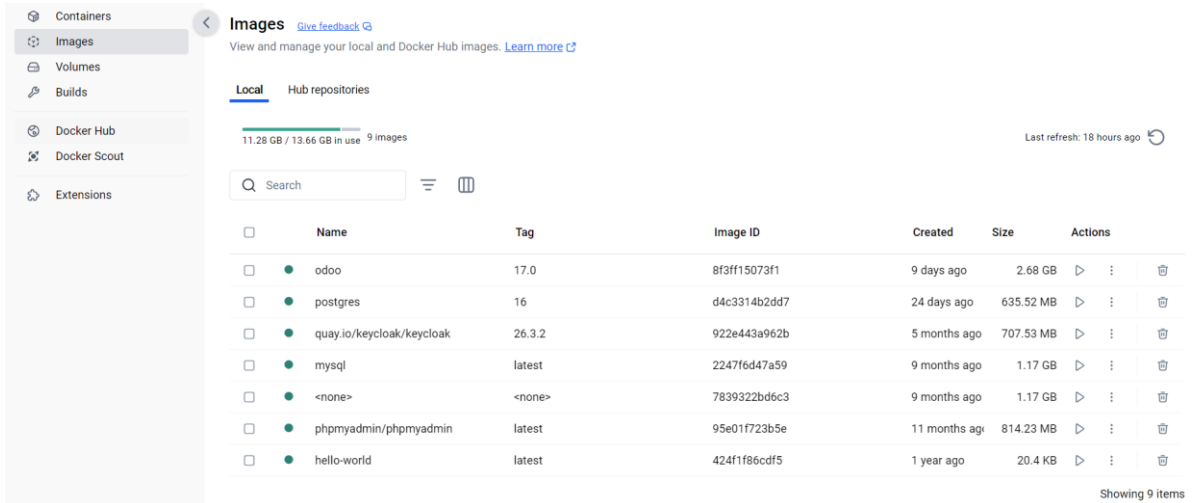


The screenshot shows the Docker Desktop interface with the 'Containers' tab selected. The left sidebar contains navigation options: Containers, Images, Volumes, Builds, Docker Hub, Docker Scout, and Extensions. The main area displays 'Containers' with a subtitle 'View all your running containers and applications. [Learn more](#)'. It shows 'Container CPU usage' at 0.03% / 1200% (12 CPUs available) and 'Container memory usage' at 204.89MB / 15.14GB. A search bar and a toggle for 'Only show running containers' are present. Below is a table of 11 containers.

<input type="checkbox"/>	Name	Container ID	Image	Port(s)	CPU (%)	Last started	Actions
<input type="checkbox"/>	focused_varahamihira	fce3c538c8a4	keycloak/keycloak/26.3.2	8080:8080	0%	5 months ago	▶ ⋮ 🗑
<input type="checkbox"/>	phpmyadminstock	5ae0ec976ec8	phpmyadmin/phpmyadmin	3000:80	0%	8 months ago	▶ ⋮ 🗑
<input type="checkbox"/>	snvdbstock	3cd6caf23d79	mysql/latest	3036:3036	0%	8 months ago	▶ ⋮ 🗑
<input type="checkbox"/>	hadoop-worker2	ab08d083f2c3	lillasfai/hadoop-cluster-lat	8041:8042	0%	8 months ago	▶ ⋮ 🗑
<input type="checkbox"/>	hadoop-worker1	9e55490ceaca	lillasfai/hadoop-cluster-lat	8040:8042	0%	8 months ago	▶ ⋮ 🗑
<input type="checkbox"/>	hadoop-master	2cb4f53829a8	lillasfai/hadoop-cluster-lat	16010:16010 Show all ports (4)	0%	7 months ago	▶ ⋮ 🗑
<input type="checkbox"/>	heuristic_lalande	c4459c005187	mysql/latest		0%	8 months ago	▶ ⋮ 🗑
<input type="checkbox"/>	sqljava	e3d1a7afb1ca	mysql	4001:4001	0%	8 months ago	▶ ⋮ 🗑

Showing 11 items

Figure 10:docker container

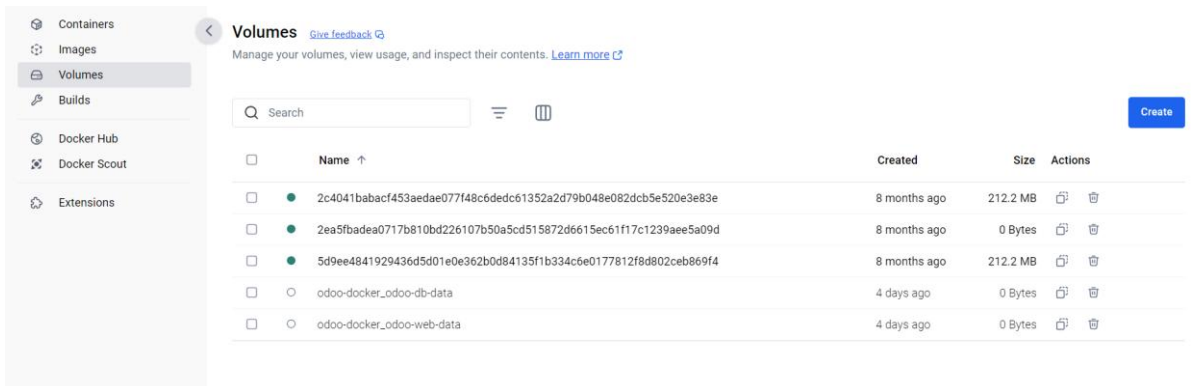


The screenshot shows the Docker Desktop interface with the 'Images' tab selected. The left sidebar is the same as in Figure 10. The main area displays 'Images' with a subtitle 'View and manage your local and Docker Hub images. [Learn more](#)'. It shows 'Local' and 'Hub repositories' tabs. A progress bar indicates '11.28 GB / 13.66 GB in use' for 9 images. A search bar and icons for list and grid views are present. Below is a table of 9 images.

<input type="checkbox"/>	Name	Tag	Image ID	Created	Size	Actions
<input type="checkbox"/>	odoo	17.0	8f3ff15073f1	9 days ago	2.68 GB	▶ ⋮ 🗑
<input type="checkbox"/>	postgres	16	d4c3314b2dd7	24 days ago	635.52 MB	▶ ⋮ 🗑
<input type="checkbox"/>	quay.io/keycloak/keycloak	26.3.2	922e443a962b	5 months ago	707.53 MB	▶ ⋮ 🗑
<input type="checkbox"/>	mysql	latest	2247f6d47a59	9 months ago	1.17 GB	▶ ⋮ 🗑
<input type="checkbox"/>	<none>	<none>	7839322bd6c3	9 months ago	1.17 GB	▶ ⋮ 🗑
<input type="checkbox"/>	phpmyadmin/phpmyadmin	latest	95e01f723b5e	11 months ago	814.23 MB	▶ ⋮ 🗑
<input type="checkbox"/>	hello-world	latest	424f1f86cdf5	1 year ago	20.4 KB	▶ ⋮ 🗑

Showing 9 items

Figure 11:docker images



The screenshot shows the Docker Desktop interface with the 'Volumes' tab selected. The left sidebar is the same as in Figure 10. The main area displays 'Volumes' with a subtitle 'Manage your volumes, view usage, and inspect their contents. [Learn more](#)'. It shows a search bar, list and grid view icons, and a 'Create' button. Below is a table of 6 volumes.

<input type="checkbox"/>	Name ↑	Created	Size	Actions
<input type="checkbox"/>	2c4041babacf453aedae077f48c6dedc61352a2d79b048e082dc5e520e3e83e	8 months ago	212.2 MB	📁 🗑
<input type="checkbox"/>	2ea5fbadea0717b810bd226107b50a5cd515872d6615ec61f17c1239aee5a09d	8 months ago	0 Bytes	📁 🗑
<input type="checkbox"/>	5d9ee4841929436d5d01e0e362b0d84135f1b334c6e0177812f8d802ceb869f4	8 months ago	212.2 MB	📁 🗑
<input type="checkbox"/>	odoo-docker_odoo-db-data	4 days ago	0 Bytes	📁 🗑
<input type="checkbox"/>	odoo-docker_odoo-web-data	4 days ago	0 Bytes	📁 🗑

Figure 12:docker volumes

Cette figure illustre l'interface de gestion de **Docker Desktop**, montrant les **images** et **volumes** utilisés dans l'environnement du projet Odoo. Elle met en évidence les images nécessaires au fonctionnement de l'application, notamment **Odoo 17** et **PostgreSQL**, ainsi que les volumes Docker dédiés à la persistance des données (base de données et fichiers applicatifs). L'utilisation conjointe des images et des volumes garantit un environnement d'exécution isolé, stable et reproductible, tout en assurant la conservation des données même après le redémarrage des conteneurs.

2.4.4.Home page

Cette figure montre l'interface **Apps** d'Odoo 17, qui permet de consulter et gérer les modules disponibles. Les applications sont organisées par catégories et peuvent être activées ou mises à jour selon les besoins. Cette interface a été utilisée pour installer et gérer le module personnalisé **Gestion des inscriptions**, illustrant la modularité et la flexibilité de la plateforme Odoo.

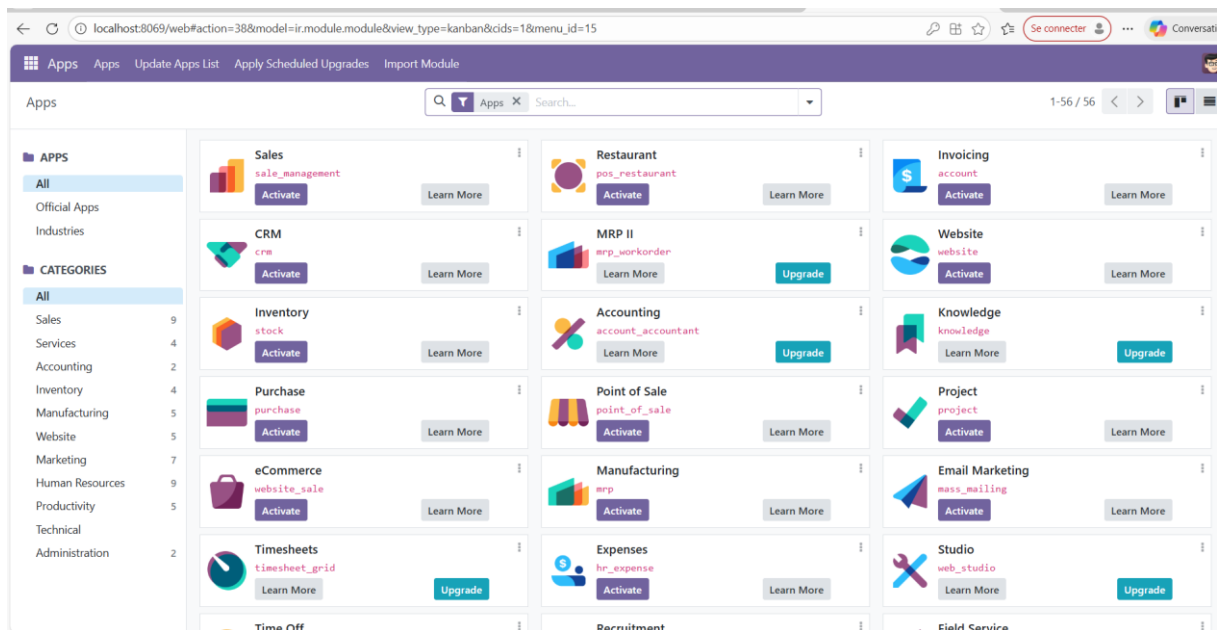


Figure 13:Home page

2.4.5. Vue liste des inscriptions

Cette figure présente la vue liste du module **Gestion des inscriptions – Inscriptions simples**. Elle affiche l'ensemble des inscriptions enregistrées dans le système sous forme de tableau, avec les principales informations : nom complet, email, téléphone, date d'inscription et statut. Cette vue permet à l'utilisateur d'avoir une vision globale et synthétique des inscriptions ainsi que de suivre leur état (nouveau, validé ou annulé).



The screenshot shows a web application interface for 'Gestion des inscriptions'. At the top, there's a header with the title 'Gestion des inscriptions' and a sub-header 'Inscriptions simples'. Below the header, there's a navigation bar with a 'New' button and a search bar. The main content area displays a table with the following columns: 'Nom complet', 'Email', 'Téléphone', 'Date d'inscription', and 'Statut'. The table contains two rows of data.

Nom complet	Email	Téléphone	Date d'inscription	Statut
ifiss malak	ifissmalak@gmail.com		01/01/2026	Annulé
imane ifiss	imaneifiss@gmail.com		01/01/2026	Validé

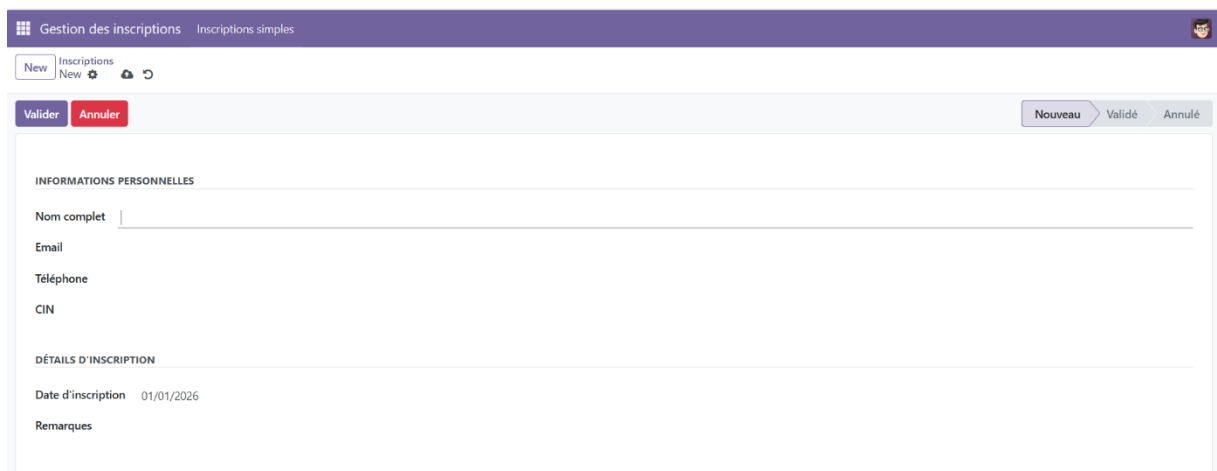
Figure 14: Vue liste des inscriptions

2.4.6. Formulaire de création d'une nouvelle inscription

Cette capture illustre le formulaire de création d'une nouvelle inscription. L'interface est structurée en deux parties principales :

- **Informations personnelles** (nom complet, email, téléphone, CIN)
- **Détails d'inscription** (date d'inscription et remarques)

Les boutons d'action *Valider* et *Annuler*, ainsi que la barre de statut, permettent de gérer le cycle de vie de l'inscription de manière intuitive.



The screenshot shows the 'Gestion des inscriptions' application with the 'Inscriptions simples' sub-header. The main content area is a form for creating a new registration. At the top, there's a navigation bar with 'New' and 'Inscriptions' buttons. Below the navigation bar, there's a status bar with 'Valider' and 'Annuler' buttons. The form is divided into two sections: 'INFORMATIONS PERSONNELLES' and 'DÉTAILS D'INSCRIPTION'. The 'INFORMATIONS PERSONNELLES' section contains fields for 'Nom complet', 'Email', 'Téléphone', and 'CIN'. The 'DÉTAILS D'INSCRIPTION' section contains fields for 'Date d'inscription' (set to 01/01/2026) and 'Remarques'. At the bottom right, there's a status bar with 'Nouveau', 'Validé', and 'Annulé' buttons.

Figure 15: Formulaire de création d'une nouvelle inscription

2.4.7. Formulaire d'une inscription validée

Le statut est clairement visible grâce à la barre de statut, ce qui facilite le suivi de l'état de chaque inscription. Cette vue confirme le bon fonctionnement du workflow implémenté dans le module et démontre la capacité du système à gérer les transitions entre les différents états, on peut changer l'état de l'inscription vers annulé en cliquant sur le bouton annulé en haut à droite

The screenshot displays the 'Gestion des inscriptions' (Registration Management) form in Odoo. The form is titled 'Inscriptions simples' and shows a registration for 'amine ifiss'. The status bar at the top right indicates the registration is 'Validé' (Validated), with buttons for 'Nouveau' (New), 'Validé', and 'Annulé' (Cancelled). The form is divided into two sections: 'INFORMATIONS PERSONNELLES' (Personal Information) and 'DÉTAILS D'INSCRIPTION' (Registration Details). The personal information section includes fields for 'Nom complet' (Full Name), 'Email', 'Téléphone' (Phone), and 'CIN'. The registration details section includes fields for 'Date d'inscription' (Registration Date) and 'Remarques' (Remarks).

INFORMATIONS PERSONNELLES	
Nom complet	amine ifiss
Email	amineifiss@gmail.com
Téléphone	
CIN	

DÉTAILS D'INSCRIPTION	
Date d'inscription	01/01/2026
Remarques	

Figure 16:Formulaire d'une inscription validée

2.4.8. Conclusion

Ce deuxième chapitre a permis de détailler l'architecture du module de gestion des inscriptions ainsi que les technologies mises en œuvre pour sa réalisation. L'architecture standard d'Odoo, inspirée du modèle MVC, a offert un cadre structuré et cohérent pour la séparation entre la logique métier, l'interface utilisateur et les actions interactives.

La présentation de la structure du module a mis en évidence l'organisation claire des composants, facilitant la lisibilité, la maintenance et l'évolution du projet. Le choix des technologies, notamment Odoo 17, Python, XML, PostgreSQL, Docker et Git Bash, a permis de mettre en place un environnement de développement stable, performant et conforme aux bonnes pratiques professionnelles.

Enfin, la description des différentes interfaces réalisées a illustré concrètement le fonctionnement du module et confirmé l'atteinte des objectifs fixés. Ce chapitre démontre ainsi que les choix architecturaux et technologiques effectués ont contribué à la mise en place d'une solution fonctionnelle, évolutive et adaptée à la gestion des inscriptions au sein d'un système ERP.

CONCLUSION GÉNÉRALE

Ce projet avait pour objectif principal la conception et le développement d'un module Odoo personnalisé dédié à la gestion des inscriptions simples. Réalisé dans un cadre académique, il a permis de mettre en pratique les notions fondamentales liées au développement sous Odoo 17, tout en répondant à une problématique réelle de gestion administrative.

À travers ce travail, un module fonctionnel et structuré a été implémenté, permettant la création, la consultation, la modification et le suivi des inscriptions via une interface claire et intuitive. L'intégration d'un système de statuts (Nouveau, Validé, Annulé) ainsi que des actions métier associées a permis de modéliser un cycle de vie simple mais efficace pour chaque inscription.

Ce projet a également permis de renforcer la compréhension de l'architecture d'Odoo, notamment la séparation entre les modèles Python, les vues XML et la logique métier. L'utilisation de technologies telles que Python, XML, PostgreSQL et Docker a contribué à la mise en place d'un environnement de développement stable, moderne et conforme aux bonnes pratiques professionnelles.

Au-delà de l'aspect technique, ce projet a constitué une expérience enrichissante en termes d'analyse, de résolution de problèmes et d'adaptation aux évolutions du framework, notamment les changements introduits par Odoo 17. Il a ainsi permis de développer des compétences solides en développement ERP et de mieux appréhender les enjeux liés à la digitalisation des processus administratifs.

Enfin, ce module peut servir de base à des évolutions futures, telles que la gestion avancée des utilisateurs, l'ajout de statistiques, la génération de rapports ou l'intégration avec d'autres modules Odoo. Ce projet représente donc une étape importante dans l'apprentissage du développement de solutions ERP personnalisées et constitue une réalisation concrète et valorisante sur le plan académique et technique.