

CRÉATION D'UN MODULE ODOO 2025/2026



A photograph of the EMSI building, a modern white structure with large glass windows. A sign on top reads "Ecole Marocaine des Sciences de l'Ingénieur". Below it, a vertical sign indicates "Pôle Finance" and "Pôle Ingénierie". The building is set against a clear blue sky with palm trees in the foreground.

GESTION DES COURRIERS ENTRANTS ET SORTANTS

PREPARED BY:
KABIL YOUSSEF

PRESENTED TO:
**MOHAMMED
AITDAOUD**

Table of Content

INTRODUCTION GÉNÉRALE	03
CHAPITRE 1 : PRÉSENTATION DU PROJET	04
CHAPITRE 2 : ENVIRONNEMENT DE TRAVAIL	05
CHAPITRE 3 : CRÉATION DU MODULE ODOO	06
CHAPITRE 4 : CRÉATION DES VUES	07
STRATEGIC REPORT	08
FUTURE OUTLOOK AND GOALS	09
CONTACT US	10

INTRODUCTION GÉNÉRALE



Dans le cadre du LAB 2 : Création d'un module Odoo, ce projet consiste à concevoir et développer un module personnalisé dédié à la gestion des **courriers entrants et sortants**. La gestion des courriers représente une fonction essentielle dans toute organisation administrative, permettant d'assurer le suivi, la traçabilité et l'archivage des échanges officiels.

Face aux limites des méthodes de gestion manuelle, l'informatisation du processus devient une nécessité afin d'améliorer l'efficacité et la fiabilité du traitement des courriers. Dans ce contexte, la plateforme Odoo, en tant qu'ERP open source modulaire, offre un environnement adapté pour le développement de solutions personnalisées répondant à des besoins réels.

L'objectif de ce projet est de mettre en place un module Odoo permettant l'enregistrement, la consultation et le suivi des courriers, tout en distinguant les courriers entrants et sortants ainsi que leurs différents états. Ce rapport présente les différentes étapes de réalisation du projet, depuis la présentation générale jusqu'aux perspectives d'amélioration.

CHAPITRE 1 : PRÉSENTATION DU PROJET

1.1 Contexte du projet

Dans le cadre du module de travaux pratiques LAB 2 : Création d'un module Odoo, il a été demandé de concevoir un module personnalisé répondant à un besoin réel de gestion. Le thème attribué pour ce projet est la gestion des courriers entrants et sortants, un processus essentiel dans toute organisation administrative.

La gestion des courriers permet de centraliser les échanges officiels, d'assurer leur suivi et de garantir une meilleure organisation des documents. Ce projet s'inscrit dans une démarche pédagogique visant à appliquer les concepts théoriques liés à Odoo à travers un cas pratique concret.

1.2 Objectifs du projet

L'objectif principal de ce projet est de développer un module Odoo permettant la gestion des courriers entrants et sortants. Les objectifs spécifiques sont les suivants :

- Créer un module Odoo personnalisé
- Gérer les courriers entrants et sortants
- Assurer le suivi des états des courriers
- Faciliter la consultation et l'organisation des courriers
- Mettre en pratique les notions étudiées dans le cadre du module Odoo



CHAPITRE 2 : ENVIRONNEMENT DE TRAVAIL

2.1 Environnement matériel et logiciel

Le développement de ce projet a été réalisé sur un ordinateur personnel fonctionnant sous le système d'exploitation Windows 11. Ce choix permet de bénéficier d'un environnement stable et compatible avec les outils modernes de développement. Afin d'assurer la portabilité et la facilité de déploiement de l'application, la technologie Docker a été utilisée. Docker permet de créer des conteneurs isolés contenant l'ensemble des dépendances nécessaires au bon fonctionnement d'Odoo, évitant ainsi les problèmes liés à la configuration manuelle de l'environnement.

2.2 Outils utilisés

Plusieurs outils ont été utilisés pour la réalisation du projet, notamment :

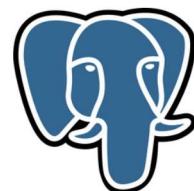
- Docker Desktop : pour la gestion des conteneurs
- Docker Compose : pour l'orchestration des services Odoo et PostgreSQL
- Odoo : plateforme ERP open source utilisée pour le développement du module
- PostgreSQL : système de gestion de base de données utilisé par Odoo
- Visual Studio Code : éditeur de code pour le développement du module
- Navigateur Web : pour l'accès à l'interface Odoo

2.3 Architecture technique

L'architecture technique du projet repose sur une approche basée sur des conteneurs Docker. Deux services principaux ont été mis en place :

- Un conteneur Odoo, responsable de l'interface utilisateur et de la logique applicative
- Un conteneur PostgreSQL, chargé du stockage des données

Ces services communiquent entre eux via un réseau interne Docker, garantissant une séparation claire entre l'application et la base de données.



CHAPITRE 2 : ENVIRONNEMENT DE TRAVAIL

The screenshot shows the Docker Desktop interface. At the top, there's a list of running Docker containers:

Container	Image	Command	Created	Status	Ports
odoo-docker	-	-	-	-	-
db-1	38596cd842af	postgres:11	-	-	-
odoo-1	52ac15d56245	odoo:17	8069:8069	Up 35 minutes	0.0.0.0:80 69->8069/tcp, [::]:8069->8069/tcp

Below the container list is a "Showing 7 items" message. Underneath, there's a "Terminal" section with the following command output:

```
PS C:\Users\TUFF> docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS
52ac15d56245 odoo:17 "/entrypoint.sh odoo" 59 minutes ago Up 35 minutes 0.0.0.0:80
69->8069/tcp, [::]:8069->8069/tcp odoo-docker-odoo-1
38596cd842af postgres:15 "docker-entrypoint.s..." 59 minutes ago Up 35 minutes 5432/tcp
odoo-docker-db-1
PS C:\Users\TUFF>
```

Figure 2.1 : Conteneurs Odoo et PostgreSQL en cours d'exécution dans Docker Desktop

The screenshot shows two side-by-side Odoo web pages.

Left Page (Database Creation):

Odoo logo

Warning, your Odoo database manager is not protected. To secure it, we have generated the following master password for it:
66nm-c7gx-xx5t

You can change it below but be sure to remember it, it will be asked for future operations on databases.

Form fields:

- Master Password: (with eye icon)
- Database Name: gestion_courriers
- Email: kabilyoussef86@gmail.com
- Password: (with eye icon)
- Phone Number: 0638461783
- Language: English (US)
- Country: Morocco
- Demo Data:

Buttons: Create database or restore a database

Right Page (Login):

Your logo

Email: kabilyoussef86@gmail.com

Password: (with eye icon)

Log in

Manage Databases | Powered by Odoo

Figure 2.2 : Lancement de la plateforme Odoo

CHAPITRE 3 : CRÉATION DU MODULE ODOO

3.1 Présentation du module

Ce chapitre est consacré à la création du module Odoo dédié à la gestion des courriers entrants et sortants. Le module développé permet de centraliser les informations relatives aux courriers, de distinguer leur type et de suivre leur état tout au long de leur cycle de vie.

La création du module repose sur l'architecture standard d'Odoo, qui s'appuie sur le modèle MVC (Model–View–Controller). Cette architecture facilite la séparation entre la logique métier, l'interface utilisateur et la gestion des données.

3.2 Structure du module

Le module a été créé sous forme d'un dossier personnalisé placé dans le répertoire des modules additionnels d'Odoo. Il contient plusieurs sous-dossiers permettant d'organiser le code de manière claire et structurée. La structure générale du module comprend :

- Un fichier manifeste décrivant le module
- Un dossier pour les modèles Python
- Un dossier pour les vues XML
- Un dossier dédié à la sécurité et aux droits d'accès

Cette organisation permet une maintenance facilitée et respecte les bonnes pratiques recommandées par Odoo.

📁 .idea	1/2/2026 9:24 PM	File folder	
📁 __pycache__	1/2/2026 8:56 PM	File folder	
📁 models	1/2/2026 8:56 PM	File folder	
📁 security	1/2/2026 8:46 PM	File folder	
📁 views	1/2/2026 8:44 PM	File folder	
📄 __init__.py	1/2/2026 8:55 PM	Python File	1 KB
📄 __manifest__.py	1/2/2026 8:44 PM	Python File	1 KB
CSV data_demo	1/2/2026 8:58 PM	Microsoft Excel Comma...	1 KB

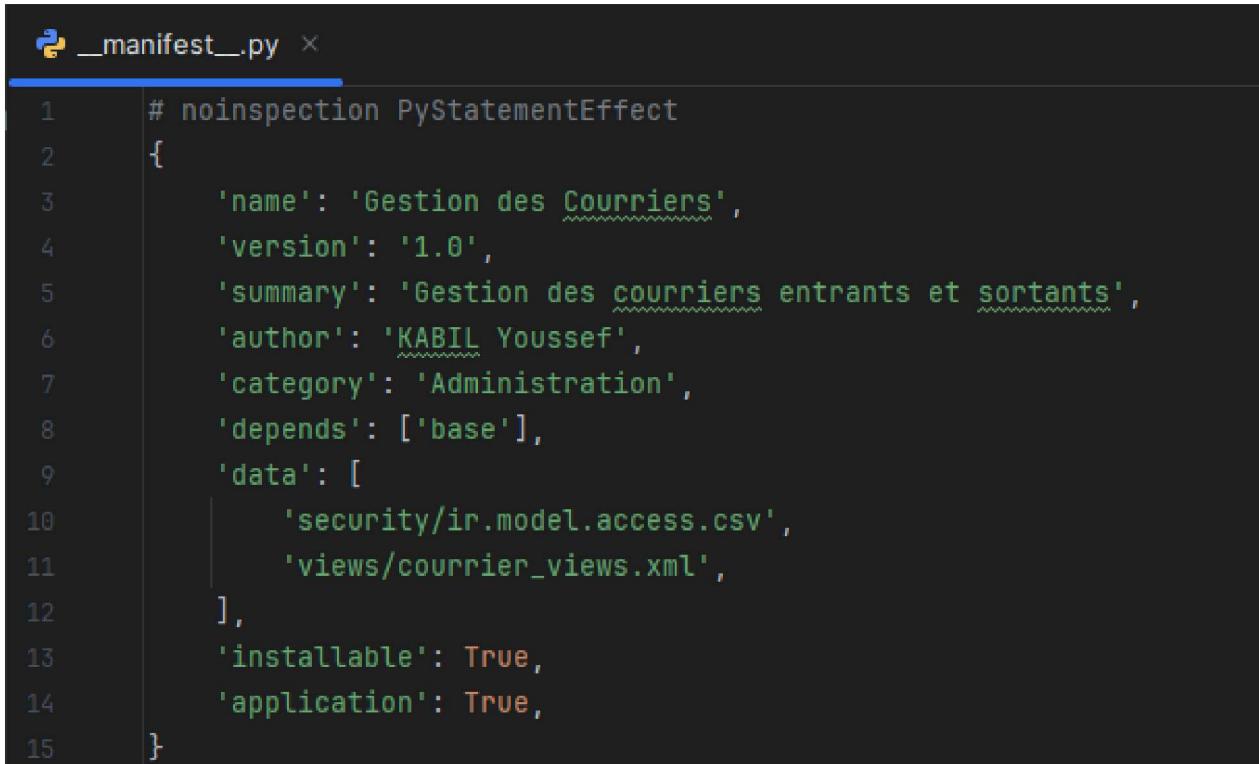
Figure 3.1 : Arborescence du module courrier_management

3.3 Fichier manifeste du module

Le fichier manifeste (`__manifest__.py`) joue un rôle essentiel dans la déclaration du module. Il contient les informations générales telles que le nom du module, l'auteur, la version, les dépendances ainsi que les fichiers à charger lors de l'installation.

Ce fichier permet à Odoo d'identifier le module et de gérer correctement son installation et son intégration dans le système.

CHAPITRE 3 : CRÉATION DU MODULE ODOO



```
# noinspection PyStatementEffect
{
    'name': 'Gestion des Courriers',
    'version': '1.0',
    'summary': 'Gestion des courriers entrants et sortants',
    'author': 'KABIL Youssef',
    'category': 'Administration',
    'depends': ['base'],
    'data': [
        'security/ir.model.access.csv',
        'views/courrier_views.xml',
    ],
    'installable': True,
    'application': True,
}
```

Figure 3.2 :Contenu du fichier __manifest__.py

3.4 Modélisation des données

La modélisation des données constitue une étape clé dans le développement du module. Un modèle principal a été créé afin de représenter un courrier. Ce modèle permet de stocker les informations essentielles telles que la référence, le type de courrier, l'expéditeur, le destinataire, la date, l'objet et l'état du courrier.

La définition du modèle a été réalisée en langage Python en utilisant l'ORM d'Odoo, ce qui facilite la manipulation des données et leur persistance dans la base de données PostgreSQL.

```
class Courrier(models.Model):
    _name = 'courrier.management'
    _description = 'Gestion des Courriers'
    reference = fields.Char(string="Référence", required=True)
    type = fields.Selection([('entrant', 'Courrier Entrant'), ('sortant', 'Courrier Sortant')], string="Type")
    sender = fields.Char(string="Expéditeur")
    receiver = fields.Char(string="Destinataire")
    subject = fields.Char(string="Objet")
    date = fields.Date(string="Date", default=fields.Date.today)
    status = fields.Selection([('draft', 'Brouillon'), ('sent', 'Envoyé'), ('received', 'Reçu'),
                               ('archived', 'Archivé')], default='draft')
    description = fields.Text(string="Description")
```

Figure 3.3 :Contenu du fichier models/courrier.py

CHAPITRE 3 : CRÉATION DU MODULE ODOO

3.5 Crédation des vues

Afin de permettre l'interaction avec les données, plusieurs vues ont été créées en utilisant le langage XML. Deux vues principales ont été mises en place :

- Une vue liste, permettant d'afficher l'ensemble des courriers sous forme de tableau
- Une vue formulaire, permettant la consultation et la modification des informations d'un courrier

Ces vues offrent une interface simple et intuitive pour la gestion des courriers par les utilisateurs.

```
<odoo>
    <!-- List View -->
    <record id="view_courrier_tree" model="ir.ui.view">
        <field name="name">courrier.tree</field>
        <field name="model">courrier.management</field>
        <field name="arch" type="xml">
            <tree>
                <field name="reference"/>
                <field name="type"/>
                <field name="sender"/>
                <field name="receiver"/>
                <field name="date"/>
                <field name="status"/>
            </tree>
        </field>
    </record>
```

Figure 3.4 : Vue liste des courriers dans Odoo

```
<!-- Form View -->
<record id="view_courrier_form" model="ir.ui.view">
    <field name="name">courrier.form</field>
    <field name="model">courrier.management</field>
    <field name="arch" type="xml">
        <form><sheet><group><field name="reference"/>
            <field name="type"/>
            <field name="status"/></group>
            <group><field name="sender"/>
                <field name="receiver"/>
                <field name="date"/></group>
            <group><field name="subject"/>
                <field name="description"/></group></sheet>
        </form>
    </field>
</record>
```

Figure 3.5 : Vue formulaire d'un courrier

CHAPITRE 3 : CRÉATION DU MODULE ODOO

3.6 Menus et actions

Pour rendre le module accessible depuis l'interface Odoo, des menus et des actions ont été définis. Un menu principal « Gestion Courriers » a été créé, contenant un sous-menu permettant d'accéder à la liste des courriers.

Cette configuration permet une navigation fluide et une intégration naturelle du module au sein de l'interface Odoo.

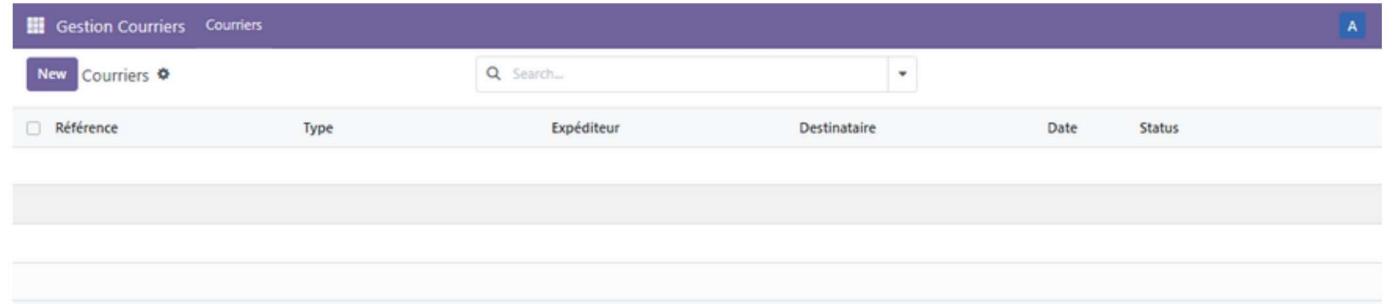


Figure 3.6 : Menu « Gestion Courriers » dans l'interface Odoo

3.7 Sécurité et droits d'accès

La sécurité du module a été assurée par la définition des droits d'accès à l'aide d'un fichier de configuration spécifique. Ce fichier permet d'autoriser les utilisateurs à lire, créer, modifier et supprimer des courriers, selon les règles définies.

La mise en place de ces droits garantit un accès contrôlé aux données et évite toute manipulation non autorisée.

CHAPITRE 4 : TESTS ET RÉSULTATS

4.1 TESTS

4.1.1. Création Manuelle

Dans cette première phase, nous avons vérifié l'ergonomie des vues et le bon fonctionnement des contraintes définies dans le modèle (champs obligatoires, valeurs par défaut).

Le test consiste à créer un courrier unitaire en utilisant le formulaire développé. Nous avons validé les points suivants :

- Champs Requis : Le système empêche la sauvegarde si le champ « Objet du Courrier » ou « Type » est vide.
- Widgets de Sélection : Le choix entre « Entrant » et « Sortant » s'effectue correctement via le widget de sélection.
- Valeurs par Défaut : La date se remplit automatiquement avec la date du jour et le statut est initialisé à « Brouillon ».

Référence	
Type	
Status	Courrier Entrant Courrier Sortant
Expéditeur	
Destinataire	
Date	01/02/2026
Objet	
Description	

Figure 4.1 : Formulaire de création d'un nouveau courrier entrant.

4.1.2. Test par Importation de Données (Fichier CSV)

Afin de tester la robustesse de l'application avec un plus grand volume d'enregistrements et d'éviter une saisie répétitive, nous avons élaboré un scénario d'importation de masse.

Nous avons créé un fichier `test_courriers.csv` contenant un jeu de données varié (courriers entrants, sortants, différents statuts). La structure du fichier respecte strictement les identifiants techniques du modèle Odoo (`name`, `expediteur`, `type_courrier`, `statut`).

L'importation a été réalisée via la fonctionnalité native d'Odoo (« Favoris » > « Importer des enregistrements »), permettant de valider le mapping automatique des colonnes et l'intégrité des données importées.

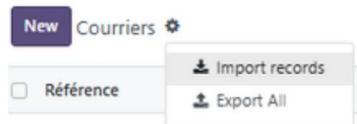


Figure 4.2 : Interface d'importation des données CSV.

CHAPITRE 4 : TESTS ET RÉSULTATS

reference	type	sender	receiver	subject	date	status	description
CE-2025-001	entrant	Ministère de l'Éducation	Service Administratif	Demande de rapport annuel	1/5/2025	received	Demande officielle du rapport annuel.
CE-2025-002	entrant	Société Maroc Telecom	Direction Informatique	Facture Internet Décembre	1/3/2025	received	Facture mensuelle Internet.
CE-2025-003	entrant	Ahmed El Mansouri	Administration Scolaire	Demande de relevé de notes	1/2/2025	draft	Demande d'accès à un relevé de notes.
CS-2025-001	sortant	Administration Centrale	Ministère de l'Éducation	Transmission du rapport annuel	1/8/2025	sent	Envoi du rapport annuel.
CS-2025-002	sortant	Direction Générale	Personnel Administratif	Convocation à l'union interne	1/10/2025	sent	Convocation à une réunion interne.
CS-2025-003	sortant	Service Scolaire	Ahmed El Mansouri	Relevé de notes disponible	1/12/2025	sent	Notification de disponibilité du relevé.

Figure 4.3 : contenu du fichier csv

4.2. RÉSULTATS OBTENUS

Suite à l'exécution des scénarios de test définis précédemment, nous avons pu valider le bon fonctionnement du module. Cette section présente les résultats de l'intégration des données de test et l'affichage final dans l'interface Odoo.

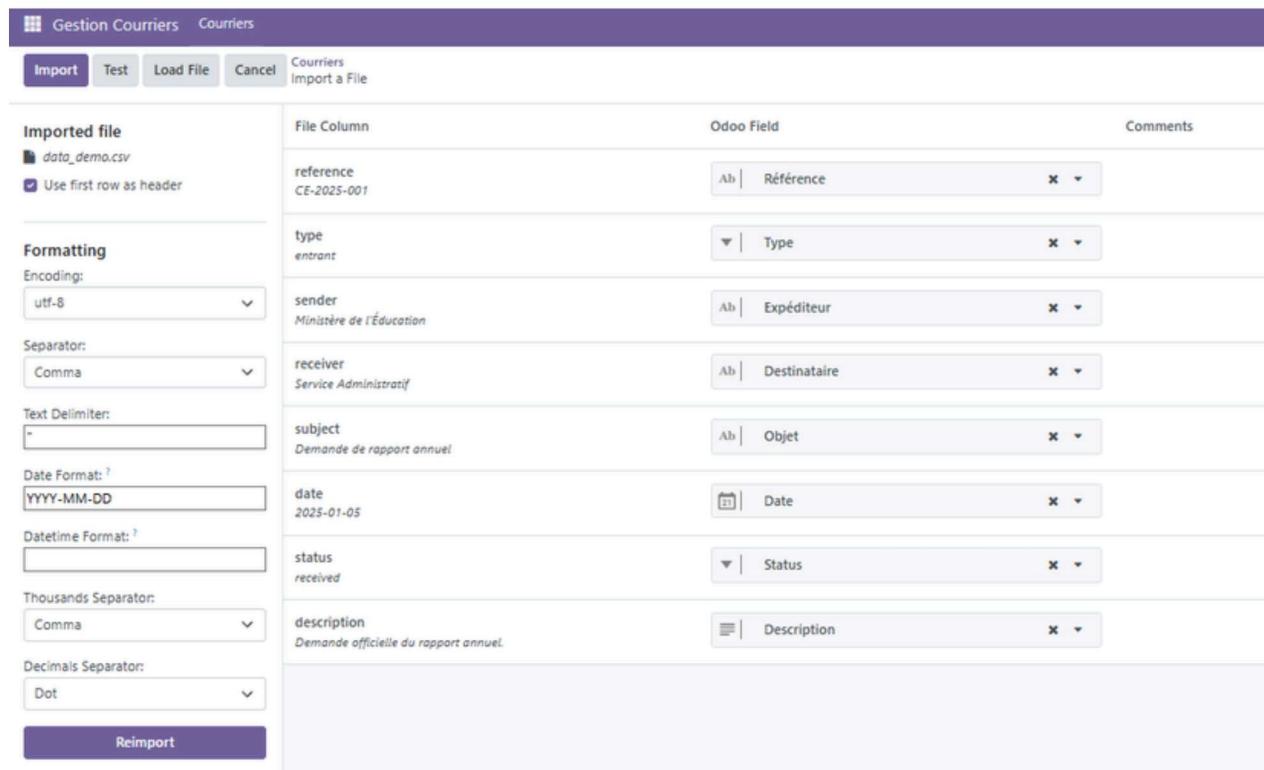


Figure 4.4 : Interface de mappage des champs lors de l'importation CSV.

Nous observons que :

- Les champs de sélection (Type et Status) sont reconnus sans erreur.
- Les dates sont correctement interprétées au format YYYY-MM-DD.
- L'encodage UTF-8 est bien pris en compte, préservant les accents (ex: "Ministère de l'Éducation").

Une fois l'importation validée, les données sont immédiatement accessibles via le menu « Gestion Courriers ». La vue liste (tree view) que nous avons développée permet une visualisation synthétique de l'ensemble des enregistrements.

CHAPITRE 4 : TESTS ET RÉSULTATS

Gestion Courriers					
Courriers		Search...		1-6 / 6	
New	Courriers	Expéditeur	Destinataire	Date	Status
<input type="checkbox"/>	Référence	Type	Expéditeur	Destinataire	Date
<input type="checkbox"/>	CE-2025-001	Courrier Entrant	Ministère de l'Éducation	Service Administratif	01/05/2025
<input type="checkbox"/>	CE-2025-002	Courrier Entrant	Société Maroc Telecom	Direction Informatique	01/03/2025
<input type="checkbox"/>	CE-2025-003	Courrier Entrant	Ahmed El Mansouri	Administration Scolarité	01/02/2025
<input type="checkbox"/>	CS-2025-001	Courrier Sortant	Administration Centrale	Ministère de l'Éducation	01/08/2025
<input type="checkbox"/>	CS-2025-002	Courrier Sortant	Direction Générale	Personnel Administratif	01/10/2025
<input type="checkbox"/>	CS-2025-003	Courrier Sortant	Service Scolarité	Ahmed El Mansouri	01/12/2025

Figure 4.4 : Vue liste finale affichant les courriers entrants et sortants importés.