

# *Manuel Utilisateur et Description Fonctionnelle*

**Master 1 SRT – 2024-2025**

**Projet : Plateforme de gestion d'exercices avec correction automatique par IA**

## **Description fonctionnelle du projet**

### **Objectif du projet**

Ce projet a pour objectif de proposer une plateforme de gestion des exercices entre professeurs et étudiants, intégrant l'intelligence artificielle pour la correction automatique des devoirs soumis au format PDF.

### **Interface Étudiant**

#### ✓ **Accès sécurisé :**

- Authentification via un système de jetons (token JWT) pour accéder au tableau de bord.

#### ✓ **Consultation des exercices :**

- L'étudiant peut visualiser la liste des exercices publiés par les professeurs.
- Chaque exercice comporte un titre, une description, une date de publication et éventuellement un fichier téléchargeable.

#### ✓ **Soumission d'une réponse :**

- L'étudiant peut téléverser un fichier PDF contenant sa réponse.
- Dès l'envoi, une requête est envoyée à l'IA (DeepSeek via Ollama) pour correction automatique.

#### ✓ **Correction automatique :**

- L'IA génère un feedback détaillé, identifie les erreurs et attribue une note sur 20.

#### ✓ **Affichage de la note et feedback :**

- L'étudiant peut consulter la note obtenue et le commentaire de correction généré automatiquement.

### **Interface Professeur**

- **Accès sécurisé :**

- Authentification via token pour accéder à son tableau de bord.

- **Publication d'un exercice :**

- Le professeur peut créer un exercice en renseignant un titre, une description, et/ou un contenu textuel (optionnellement un fichier).
- Lors de la publication, tous les étudiants enregistrés dans la base reçoivent automatiquement l'exercice (pas besoin de sélectionner les destinataires).

- **Suivi des soumissions :**

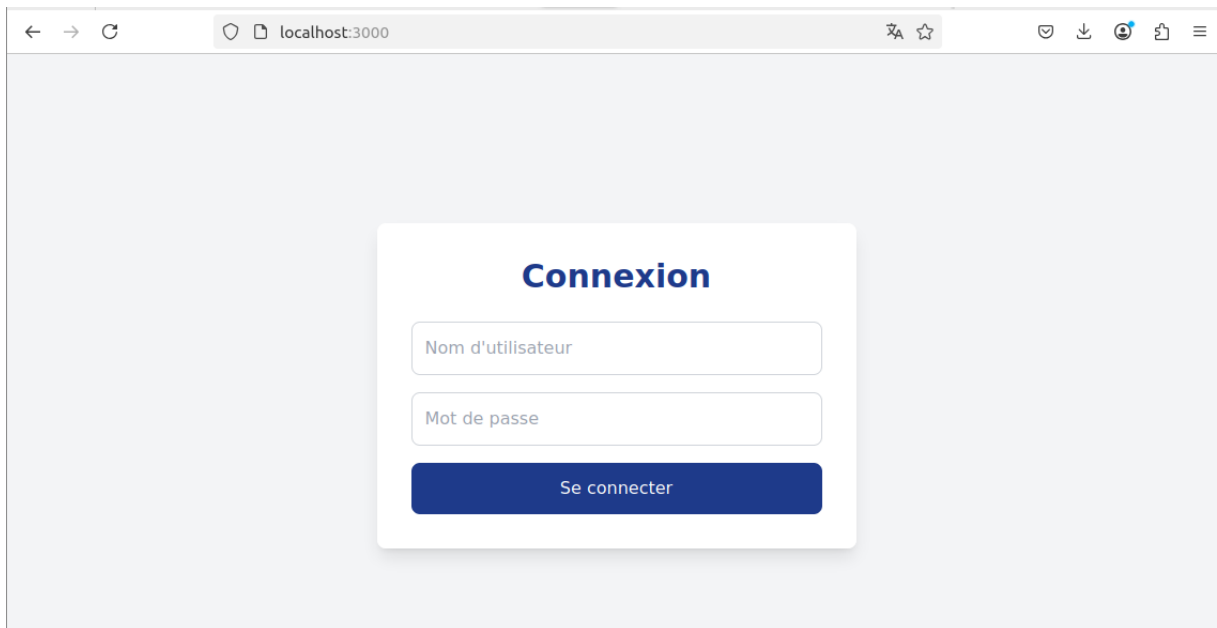
- Visualisation de la liste des soumissions reçues (étudiant, titre de l'exercice).
- Chaque soumission affiche automatiquement la note attribuée par l'IA et le retour fourni à l'étudiant.

## **Architecture technique**

- **Backend** : Django + Django REST Framework
- **Frontend** : React.js + TailwindCSS
- **IA / NPL** : Ollama (modèle DeepSeek)
- **Stockage fichiers** : MinIO (non encore intégré)
- **Sécurité** :
  - Authentification JWT
  - Fichiers PDF uniquement
  - Objectif futur : ajout de la détection de plagiat
- **Déploiement** :
  - Prévu via AWS EC2 (avec NGINX + Gunicorn)
  - GitHub pour le suivi des versions

## Manuel utilisateur

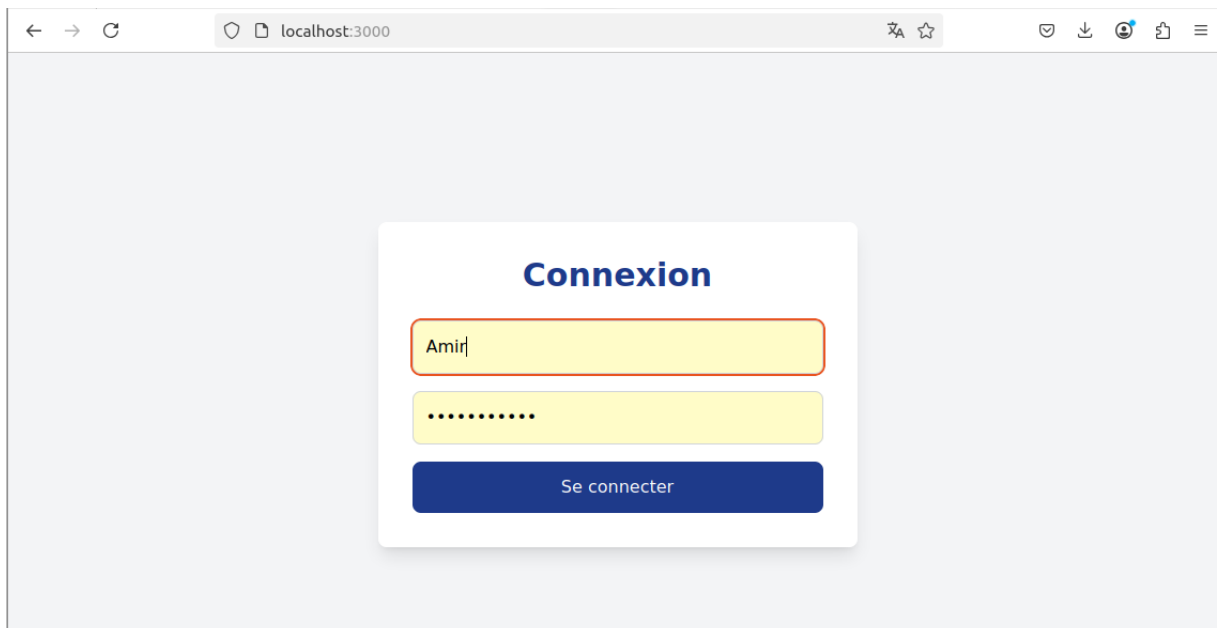
Allons sur le navigateur, tapons <http://localhost:3000> et nous avons ceci :



### Côté Étudiant

#### ○ Connexion


- Accès par identifiant « **Amir** » qui un étudiant puis tapons le mode passe+ mot de passe



- Redirection vers le tableau de bord étudiant


← → ↻ localhost:3000/etudiant

## Tableau de Bord Étudiant

 **Soumettre une réponse**

-- Choisissez un exercice --

No file selected.

 **Mes Corrections**

Aucune soumission pour l'instant.

### ○ Consulter les exercices

- Liste des exercices reçus avec titre, contenu et date

### ○ Soumettre une réponse

- Sélection d'un fichier PDF
- Bouton « Soumettre » pour envoyer le fichier

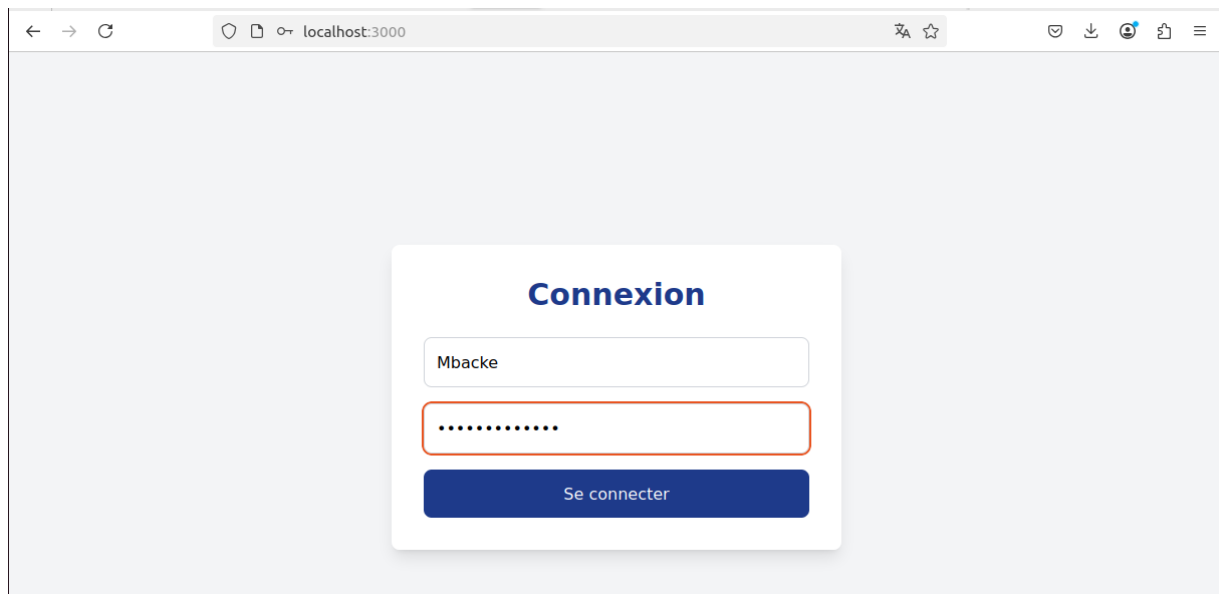
### ○ Voir sa note et le feedback IA

- Une fois l'exercice corrigé automatiquement, la note et le commentaire sont affichés dans le tableau de bord

## Côté Professeur

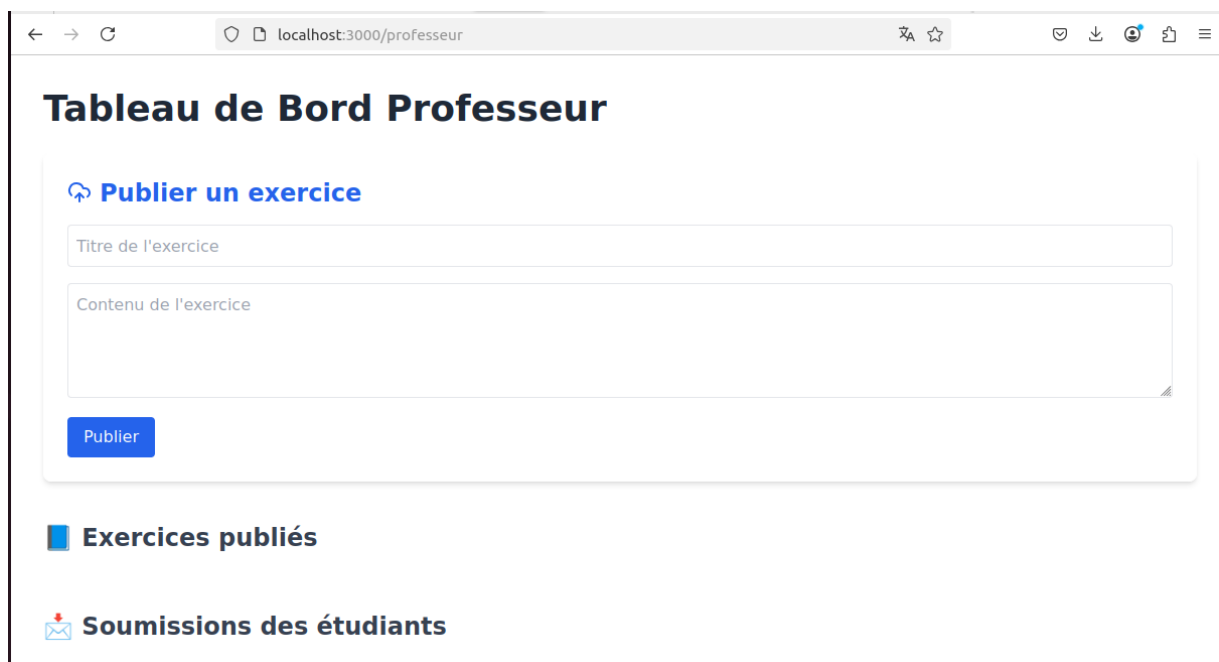
### ✓ Rapport

- Accès par identifiant + mot de passe



- Redirection vers le tableau de bord professeur

Nous avons été rediriger vers la page du professeur



### ✓ Publier un exercice

- Saisie du titre
- Description de l'exercice (champ texte)
- Bouton « Publier »

localhost:3000/professeur

# Tableau de Bord Professeur

Publier un exercice

TD

Écrivez une requête SQL permettant d'afficher tous les enregistrements de la table 'clients' ayant effectué une commande supérieure à 100 euros.

Publier

Exercices publiés

Soumissions des étudiants