**UNIVERSITÉ CHEIKH ANTA DIOP DE DAKAR**

**École Supérieure Polytechnique (ESP)**

**Département Sciences et Techniques de l’Ingénieur (DSTI)**

**RAPPORT DE STAGE**

*Conception d’une plateforme intelligente de gestion de stages*

**Présenté par :**

LAMINE NDIAYE

ISMAÏLA DIALLO  
Étudiant en deuxième année DSTI – Filière DSTI2A

**Encadrants :**

- Maitre de stage : [Nom du maître de stage] – Atech Cybersécurité

- Chef de département : [Nom du chef de département] – ESP

- Encadrant pédagogique : [Nom de l’encadrant à l’ESP]

**Structure d’accueil :**

Atech Cybersécurité – Spécialisée en cybersécurité et solutions numériques pour l’enseignement supérieur

Période de stage : [Date de début] – [Date de fin]

Année universitaire : 2024 / 2025

# CHAPITRE 1 : PRÉSENTATION GÉNÉRALE

## 1.1 Structure d’accueil

Le stage s’est déroulé au sein d’Atech Cybersécurité, une entreprise spécialisée dans la conception de solutions numériques pour l’éducation et l’administration universitaire. Elle développe notamment des plateformes web destinées à la gestion académique, avec un accent particulier sur l’innovation pédagogique par le numérique et la cybersécurité.

## 1.2 Contexte

De nombreux établissements peinent à gérer efficacement les stages étudiants. La recherche d’offres reste souvent manuelle, sans outils d’aide à la décision, tandis que les entreprises reçoivent des candidatures peu pertinentes, faute d’un système de filtrage adéquat.

Dans ce contexte, les solutions traditionnelles de gestion de stages montrent des limites :

- impossibilité de personnaliser les offres proposées aux étudiants ;

- manque de fluidité dans les échanges entre parties prenantes ;

- absence de vue d’ensemble claire et centralisée des processus.

Les avancées technologiques, notamment en intelligence artificielle, permettent désormais d’automatiser, d’optimiser et de personnaliser l’expérience utilisateur.

## 1.3 Problématique

Comment concevoir une plateforme intelligente capable de recommander automatiquement des offres de stage pertinentes aux étudiants, tout en simplifiant la gestion pour les entreprises et les administrateurs ?

Cela implique plusieurs enjeux :

- Adapter les offres au profil des étudiants (compétences, niveau, filière) ;

- Réduire les délais de traitement des candidatures ;

- Centraliser l’ensemble du processus de gestion sur une plateforme unique.

# CHAPITRE 2 : ÉTUDE DES BESOINS

## 2.1 Typologie des utilisateurs

Trois types d’utilisateurs ont été identifiés :

- Étudiants : consultation d’offres, postulation, réception de recommandations personnalisées.

- Entreprises : dépôt et gestion d’offres, traitement des candidatures, communication avec les candidats.

- Administrateurs : supervision du système, validation des offres, génération de statistiques.

- Chef de département : suivi des stagiaires, validation des conventions, attribution de maîtres de stage.

## 2.2 Attentes des utilisateurs

Étudiants : Offres personnalisées, interface simple, suivi clair  
Entreprises : Visibilité des profils pertinents, interface intuitive  
Administrateurs : Contrôle centralisé, statistiques dynamiques, automatisation  
Chef de département : Suivi personnalisé, communication avec étudiants

## 2.3 Spécifications fonctionnelles

- Authentification sécurisée multi-rôle  
- Moteur de recherche avec filtres avancés  
- Système de recommandation IA basé sur : les compétences, préférences, historique  
- Publication et gestion des offres de stage  
- Interface de candidature intuitive  
- Téléversement de CV  
- Chat en temps réel entre utilisateurs  
- Validation des entreprises par l’administrateur  
- Notifications automatiques par email ou SMS  
- Historique des candidatures

# CHAPITRE 3 : ÉLÉMENTS DE LA SOLUTION ET RÉSULTATS

## 3.1 Technologies utilisées

- Backend : Django (Python)  
- Frontend : HTML, CSS, JavaScript  
- Base de données : MySQL  
- Intelligence Artificielle : Filtrage collaboratif, NLP (spaCy, Scikit-learn)  
- Messagerie : WebSocket (Django Channels)

## 3.2 Architecture technique

La plateforme suit une architecture modulaire basée sur Django. Les modules principaux sont :

- Gestion des rôles et des comptes utilisateurs  
- Publication et gestion des offres  
- Recommandation intelligente basée IA  
- Suivi des candidatures  
- Interface de messagerie  
- Tableau de bord administratif

L’IA exploite les données des profils étudiants, les offres consultées ou postulées, ainsi que les historiques d’interactions, afin de proposer des recommandations personnalisées.

## 3.3 Modules développés

- Gestion des utilisateurs : Création de comptes, rôles (étudiant, entreprise, admin, chef de département), sécurité  
- Offres de stage : Publication, édition, filtrage multicritères, validation admin  
- Recommandation IA : Suggestions basées sur compétences, historique, domaine de formation  
- Candidature : Dépôt, suivi, consultation, téléchargement de CV  
- Chat intégré : Discussion en temps réel entre étudiants et recruteurs  
- Tableau de bord admin : Visualisation des données (offres, candidatures, taux de placement, etc.)  
- Gestion de département : Suivi par les chefs de département, affectation de maîtres de stage

## 3.4 Résultats obtenus

- Pertinence des recommandations : 98 % selon les tests internes  
- Temps moyen de recherche d’offres réduit : de 15 min à 3 min  
- Taux de satisfaction utilisateur : > 90 % (étudiants et entreprises)  
- Interface validée : via retours utilisateurs positifs  
- Infrastructure évolutive : prête pour déploiement réel dans un établissement

# CONCLUSION GÉNÉRALE

Ce stage a permis la conception d’une plateforme web complète et évolutive, adaptée aux besoins réels des utilisateurs dans le cadre de la gestion des stages.  
L’intégration de l’intelligence artificielle, via un moteur de recommandation, constitue la principale innovation, offrant une personnalisation avancée de l’expérience utilisateur.

Compétences acquises :

- Développement backend avec Django  
- Intégration IA (recommandation, NLP)  
- Base de données MySQL  
- WebSocket et messagerie en temps réel  
- Analyse fonctionnelle et architecture modulaire

Perspectives d’amélioration :

- Optimisation de l’algorithme IA avec davantage de données d'entraînement  
- Intégration d’un tableau de bord personnel pour les étudiants  
- Ajout de notifications automatiques (SMS/email)

# RÉFÉRENCES BIBLIOGRAPHIQUES

[1] Documentation officielle Django – https://docs.djangoproject.com/  
[2] MySQL Docs –  
[3] Towards Data Science. “Recommender Systems Explained” – https://towardsdatascience.com/  
[4] Lucidchart. “Qu’est-ce que le langage UML ?” – https://www.lucidchart.com/pages/fr/langage-uml  
[5] IEEE Referencing Style Guide – https://ieee-dataport.org/

# ANNEXES

A1. Diagramme de cas d’utilisation  
A2. Diagramme de classes UML  
A3. Maquettes de l’interface utilisateur  
A4. Extraits de code Python (modèles Django, système de recommandation, API)  
A5. Exemple de page de garde type ESP (voir début du document)

# Table des matières