

## **Membres du Groupe**

**Aminata Mbaye**

**Ndeye Soukeye Lo**

**Woury Ba**

**Fatou Faye**

**Mamadou Seydou Kande**

## **Projet : Gestion des Inscriptions des Nouveaux Bacheliers à l'ISEP de Thiès**

---

### **Introduction**

L'Institut Supérieur d'Enseignement Professionnel (ISEP) de Thiès joue un rôle crucial dans la formation des jeunes bacheliers. Face à l'augmentation annuelle du nombre d'étudiants, il est primordial de mettre en place un système performant pour optimiser le processus d'inscription.

Ce projet a pour objectif de développer une application intuitive et fiable qui facilite la gestion des inscriptions des nouveaux bacheliers. En garantissant une gestion fluide et efficace, cette application contribuera à l'amélioration des opérations administratives de l'ISEP et à une meilleure organisation des étudiants.

---

# **Gestion des Enregistrements des Nouveaux Bacheliers Orientés à l'ISEP de Thiès**

## **1. Recueil des Besoins**

Pour garantir que l'application répond aux attentes des utilisateurs finaux, il est essentiel de recueillir leurs besoins à travers une approche collaborative et structurée. Voici les principales étapes du recueil des besoins :

### **1.1 Responsable pédagogique / Direction de l'ISEP**

- **Besoins :**

- Suivi des inscriptions : Accéder à un tableau de bord pour suivre les inscriptions des nouveaux bacheliers dans les différentes filières.

- Orientation : Gérer l'orientation des étudiants dans les filières appropriées en fonction de leurs résultats et des critères académiques.

- Rapports statistiques : Générer des rapports sur les étudiants inscrits, leur répartition dans les différentes filières, etc.

- Suivi des capacités des filières : Suivre les capacités des filières pour éviter la surcharge.

### **1.2 Entretien avec les responsables administratifs de l'ISEP**

- **Objectif :** Comprendre en détail le processus d'inscription actuel et identifier les informations nécessaires pour chaque bachelier.
- **Aspects couverts :**
  - Les documents requis pour l'inscription (copies des diplômes, pièce d'identité, photos, etc.).
  - Les critères de validation des inscriptions (par exemple, paiement des frais, dossier complet).
  - Les défis rencontrés dans le processus manuel actuel, tels que les retards, les doublons ou les pertes d'informations.

### **1.3 Consultation avec les professeurs**

- **Objectif :** Identifier les besoins pédagogiques liés à l'inscription et à la gestion des filières.
- **Aspects couverts :**
  - Organisation des filières en fonction des spécialités des bacheliers.
  - Gestion des effectifs par filière pour équilibrer la répartition des étudiants.
  - Suivi des performances académiques des étudiants en fonction de leur inscription initiale.

### **1.4 Consultation avec le service informatique**

- **Objectif :** Comprendre les exigences techniques nécessaires à la mise en œuvre de l'application.
- **Aspects couverts :**
  - Les contraintes techniques, comme la compatibilité avec les systèmes existants et la sécurité des données.
  - La base de données à utiliser, avec une structure permettant une gestion efficace des informations.
  - Les fonctionnalités attendues de l'application, telles que l'import/export de données, les sauvegardes, et l'accessibilité via une interface web ou mobile.

### **1.5 Consultation avec le service médical**

- **Objectif :** Intégrer les besoins liés à la santé des nouveaux bacheliers dans le processus d'inscription.
- **Aspects couverts :**
  - Organisation d'une visite médicale obligatoire pour chaque bachelier lors de son inscription.
  - Saisie des données médicales essentielles dans le système (antécédents médicaux, vaccinations, allergies, etc.), tout en respectant la confidentialité des données personnelles.
  - Coordination entre le service médical et le service administratif pour confirmer que les bacheliers ont passé leur consultation.

## **1.6 Consultation avec les nouveaux bacheliers**

- **Objectif :** Comprendre les besoins et attentes des étudiants pour une application intuitive et facile à utiliser.
- **Aspects couverts :**
  - Les défis rencontrés lors de leur inscription actuelle (complexité, manque d'information, files d'attente).
  - Les fonctionnalités qu'ils souhaiteraient voir dans une application (notifications, suivi du statut de leur inscription, interface mobile).
  - Leur capacité à fournir les informations nécessaires à l'inscription en ligne.

## **1.7 Consultation avec le personnel de la scolarité**

- **Objectif :** Faciliter les tâches administratives pour les gestionnaires et réduire leur charge de travail.
- **Aspects couverts :**
  - Les fonctionnalités attendues, comme la validation rapide des inscriptions et l'édition des certificats de scolarité.
  - Les outils nécessaires pour gérer les cas particuliers (inscriptions tardives, réclamations).

### **1.7 Technicien informatique**

- **Besoins :**

Maintenance du système : Assurer la maintenance et la gestion des bases de données.

Support technique : Résoudre les problèmes techniques (erreurs de l'application, pannes, etc.).

Mise à jour de l'application : Mettre à jour le système pour ajouter de nouvelles fonctionnalités ou corriger des bogues.

## **1.8 Enseignants**

- **Besoins :**

Accéder à la liste des étudiants : Les enseignants doivent pouvoir consulter les étudiants affectés à leurs matières ou départements.

Suivi des étudiants : Suivre les progrès des étudiants inscrits dans leur département

---

Ces éléments permettent d'identifier et de formaliser les besoins des différents acteurs impliqués dans le processus d'inscription, garantissant que l'application réponde à toutes les attentes. Cette démarche exhaustive assure que l'application sera non seulement utile mais aussi adaptée à l'écosystème de l'ISEP de Thiès.

## **. Analyse des Entités, Attributs et Relations**

Après avoir recueilli les besoins, une analyse approfondie des entités du système, de leurs attributs et des relations entre elles est nécessaire. Voici une proposition :

### **1. Bachelier**

**Représente les étudiants candidats à l'inscription.**

#### **. Attributs :**

- **ID\_bachelier (int) : Identifiant unique du bachelier.**
- **Nom (String) : Nom de famille du bachelier.**
- **Prénom (String) : Prénom du bachelier.**
- **Date\_de\_naissance (Date) : Date de naissance du bachelier.**
- **Adresse (String) : Adresse complète de résidence.**
- **Filière (String) : Filière pour laquelle il postule.**
- **Téléphone (String) : Numéro de téléphone.**
- **Email (String) : Adresse e-mail.**
- **Statut\_medical (String) : État médical après visite ("apte", "inapte").**
- **Type\_bachelier (String) : Catégorie du bachelier (nouveau, réorienté, autre).**

### **2. Inscription**

**Gère le processus d'enregistrement des bacheliers.**

- **Attributs :**

- **ID\_inscription (int) :** Identifiant unique de l'inscription.
- **Date\_inscription (Date) :** Date de soumission de l'inscription.
- **Statut (String) :** Statut actuel de l'inscription ("validé", "en attente", "refusé").
- **Bachelier\_ID (int) :** Identifiant du bachelier associé.
- **Documents\_complets (bool) :** Indique si les documents requis sont fournis.
- **Statut\_confirmation (bool) :** Statut de confirmation après communication.
- **Observations (String) :** Notes sur le dossier.

### **3. Filière**

**Représente les filières de l'ISEP.**

- **Attributs :**

- **ID\_filière (int) :** Identifiant unique.
- **Nom\_filière (String) :** Nom de la filière (ex. "Informatique").
- **Description (String) :** Description générale de la filière.
- **Capacité (int) :** Nombre maximum d'étudiants.



- **Département\_ID (int) : Référence au département correspondant.**

#### **4. Département**

**Structure qui regroupe plusieurs filières sous une entité commune.**

- **Attributs :**
  - **ID\_département (int) : Identifiant unique.**
  - **Nom\_département (String) : Nom du département (ex. "Sciences et Technologies").**
  - **Responsable (String) : Nom du chef de département.**
  - **Description (String) : Brève description des activités.**

#### **5. Utilisateur (Administrateur, Professeur, etc.)**

**Représente les personnes intervenant dans la gestion des inscriptions.**

- **Attributs :**
  - **ID\_utilisateur (int) : Identifiant unique.**
  - **Nom (String) : Nom complet de l'utilisateur.**
  - **Rôle (String) : Type d'utilisateur ("administrateur", "enseignant", etc.).**
  - **Email (String) : Adresse e-mail professionnelle.**
  - **Téléphone (String) : Numéro de contact.**

- **Département\_ID (int) : Département associé (si applicable).**

## **6. Communication**

**Gère les interactions avec les étudiants.**

- **Attributs :**

- **ID\_communication (int) : Identifiant unique.**
- **ID\_bachelier (int) : Référence au bachelier concerné.**
- **Date\_communication (Date) : Date de l'envoi de l'e-mail ou de l'appel.**
- **Type\_communication (String) : Nature de la communication ("e-mail", "appel", etc.).**
- **Statut (String) : Résultat ("effectué", "non joignable", etc.).**

## **7. Visite Médicale**

**Représente les consultations médicales obligatoires.**

- **Attributs :**

- **ID\_visite (int) : Identifiant unique.**
- **ID\_bachelier (int) : Référence au bachelier.**
- **Date\_visite (Date) : Date de la visite médicale.**
- **Résultat (String) : Résultat de la visite ("apte", "inapte").**
- **Observations (String) : Notes du médecin.**

---

## Relations : Version Enrichie

1. Un Bachelier peut avoir une ou plusieurs Inscriptions.
  - Relation : *Bachelier(1) → Inscription(n)*
2. Une Inscription est liée à une Filière.
  - Relation : *Inscription(1) → Filière(1)*
3. Une Filière appartient à un Département.
  - Relation : *Filière(n) → Département(1)*
4. Un Utilisateur peut intervenir dans la gestion d'une Inscription.
  - Relation : *Utilisateur(1) → Inscription(n)*
5. Une Communication est liée à un Bachelier.
  - Relation : *Communication(n) → Bachelier(1)*
6. Une Visite Médicale est associée à un Bachelier.
  - Relation : *Visite Médicale(1) → Bachelier(1)*
7. Un Département peut être géré par un Utilisateur (chef de département).
  - Relation : *Utilisateur(1) → Département(n)*

---

## Ajouts Proposés

- **Documents requis pour inscription : Ajout d'un statut "Documents complets".**
  - **Statut médical obligatoire : Intégration du statut médical pour le suivi.**
  - **Système de communication : Suivi des interactions avec les bacheliers.**
  - **Liens renforcés avec les départements et utilisateurs : Gestion hiérarchique mieux définie.**
- 

### **3. Diagramme des Cas d'Utilisation**

Le diagramme des cas d'utilisation identifie les fonctionnalités principales de l'application et les acteurs associés.

Bien sûr ! Voici comment les cas d'utilisation sont appliqués par les différents acteurs dans le diagramme :

#### **Acteurs et leurs Cas d'Utilisation**

##### **1. Administrateur :**

- Gestion des Étudiants: Ajouter, modifier, supprimer et afficher les informations des étudiants.
- Gestion des Filières : Ajouter, supprimer et afficher les filières.

- Gestion des Départements : Ajouter, supprimer et afficher les départements.
- Gestion des Techniciens Informatiques : Gérer les techniciens en charge du système.

## **2. Étudiant :**

- Enrôlement des Étudiants : S'inscrire dans une filière.
- Consulter les Informations : Afficher les informations sur sa filière et son orientation.

## **3. Enseignant :**

- Affectation des Enseignants : Accepter une affectation à un département ou une filière.
- Consulter les Informations : Afficher des informations sur son département et sa filière.

## **4. Technicien Informatique :**

- Gestion des Techniciens Informatique : Gérer les aspects techniques du système et assurer son bon fonctionnement.

## **4. Diagramme de Classe**

Le diagramme de classe précise les entités et leurs relations en détail. Voici une proposition :

## 1. Table Étudiant

Représente les étudiants qui s'inscrivent.

- **Attributs :**

- id\_etudiant (int) : Identifiant unique.
  - nom (String) : Nom de l'étudiant.
  - prenom (String) : Prénom de l'étudiant.
  - date\_de\_naissance (Date) : Date de naissance.
  - email (String) : Adresse email.
  - telephone (String) : Numéro de téléphone.
  - adresse (String) : Adresse personnelle.
  - filiere\_id (int) : Référence à la filière.
  - status\_medical (String) : Statut médical ("apte", "en attente").
- 

## 2. Table Filière

Représente les filières proposées par l'ISEP.

- **Attributs :**

- id\_filiere (int) : Identifiant unique.
- nom\_filiere (String) : Nom de la filière.

- description (String) : Brève description.
  - departement\_id (int) : Référence au département associé.
  - **Méthodes :**
    - ajouterEtudiant() : Ajouter un étudiant à la filière.
    - retirerEtudiant() : Retirer un étudiant.
- 

### 3. Table Département

Représente les départements de l'ISEP (regroupant plusieurs filières).

- **Attributs :**
    - id\_departement (int) : Identifiant unique.
    - nom\_departement (String) : Nom du département.
    - description (String) : Description générale.
  - **Méthodes :**
    - ajouterFiliere() : Ajouter une filière au département.
    - supprimerFiliere() : Supprimer une filière.
- 

### 4. Table GestionEnrollement

Gère les inscriptions des bacheliers.

- **Attributs :**

- id\_enrollement (int) : Identifiant unique.
  - date\_enrollement (Date) : Date de l'inscription.
  - id\_etudiant (int) : Référence à l'étudiant inscrit.
  - id\_filiere (int) : Référence à la filière choisie.
  - statut\_inscription (String) : Statut de l'inscription ("validé", "en attente").
  - statut\_communication (String) : Indique si l'étudiant a été contacté par email ou téléphone.
  - statut\_visite\_medicale (String) : Indique si la visite médicale a été complétée.
- 

## 5. Table Enseignant

Représente les enseignants associés aux départements.

- **Attributs :**

- id\_enseignant (int) : Identifiant unique.
- nom (String) : Nom de l'enseignant.
- prenom (String) : Prénom de l'enseignant.
- email (String) : Adresse email.
- telephone (String) : Contact téléphonique.
- departement\_id (int) : Référence au département associé.



- **Méthodes :**

- ajouterCours() : Associer un cours à l'enseignant.
- consulterListeEtudiants() : Consulter les étudiants dans sa filière.

---

## 6. Table Technicien Informatique

Représente les techniciens informatiques de l'ISEP.

- **Attributs :**

- id\_technicien (int) : Identifiant unique.
- nom (String) : Nom du technicien.
- prenom (String) : Prénom du technicien.
- domaine\_expertise (String) : Spécialité (réseaux, maintenance, etc.).
- telephone (String) : Numéro de téléphone.

- **Méthodes :**

- resoudreProbleme() : Résoudre un problème technique.

---

## 7. Table Communication

Gère les communications avec les étudiants (emails ou appels).

- **Attributs :**

- id\_communication (int) : Identifiant unique.
  - id\_etudiant (int) : Référence à l'étudiant.
  - type\_communication (String) : Type de communication ("email", "appel").
  - statut (String) : Indique si la communication a été effectuée.
  - date\_communication (Date) : Date de l'interaction.
- 

## 8. Table Visite Médicale

Gère les consultations médicales des étudiants.

- **Attributs :**

- id\_visite (int) : Identifiant unique.
- id\_etudiant (int) : Référence à l'étudiant.
- date\_visite (Date) : Date de la visite médicale.
- resultat (String) : Résultat médical ("apte", "inapte").

- **Méthodes :**

- effectuerVisite() : Effectuer une visite médicale pour un étudiant.
- 

## Relations entre les Tables

1. **Étudiant** est relié à **Filière** (relation 1-n).

2. **Filière** appartient à un **Département** (relation 1-n).
3. **Enseignant** est associé à un **Département** (relation 1-n).
4. **Technicien Informatique** est une entité indépendante mais intervient dans la gestion des **problèmes techniques**.
5. **GestionEnrollement** relie **Étudiant** et **Filière** pour gérer les inscriptions.
6. **Visite Médicale** est reliée à **Étudiant** pour gérer leur état de santé.
7. **Communication** est reliée à **Étudiant** pour assurer le suivi des notifications.

## **Conclusion**

Ce projet offre une solution complète pour la gestion des inscriptions des nouveaux bacheliers à l'ISEP de Thiès. Il s'articule autour de la collecte des besoins, de la conception logique et de l'implémentation technique.

La prochaine étape consistera à développer le code correspondant aux entités et aux fonctionnalités identifiées, tout en effectuant des tests rigoureux pour garantir la fiabilité de l'application. En outre, l'expérience utilisateur sera optimisée pour garantir une adoption facile par les différents acteurs.



