|  |
| --- |
| [Nom de la société] |
| Spécification technique et de conception de la base de données bd\_gest\_project |
| [Sous-titre du document] |

|  |
| --- |
| Christian OUEDRAOGO  28/06/2025 |

1. **Objectif du système**

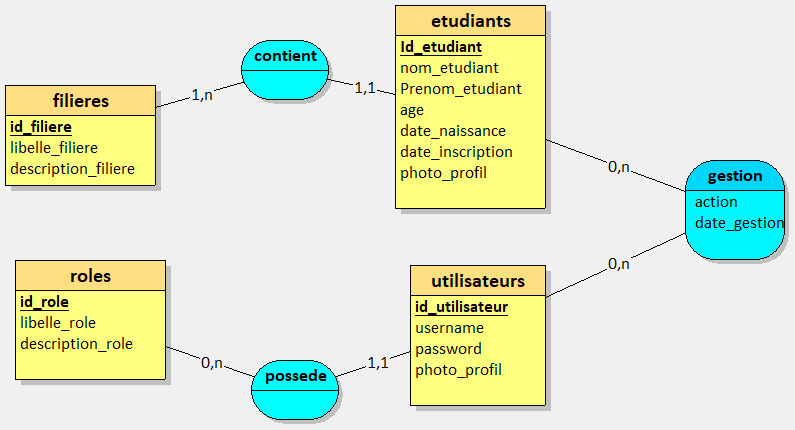
Le système a pour but de gérer l’ensemble des informations liées aux étudiants, aux filières d’enseignement, aux utilisateurs du système, à leurs rôles respectifs, ainsi qu’à toutes les opérations de gestion effectuées dans une école ou un établissement d’enseignement.

1. **Modèle conceptuel (MCD)**

**Le modele conceptuel de donné se présente comme suit :**

* **Entités principales :**

|  |  |
| --- | --- |
| Entités principales | Relations |
| Filière | * Une filière contient plusieurs étudiants. * Un utilisateur a un rôle. * Un utilisateur peut effectuer plusieurs actions (création, modification, suppression) sur les étudiants. * Une action de gestion est liée à un utilisateur et un étudiant. |
| Étudiant |
| Utilisateur |
| Rôle |
| Gestion (historique des actions) |



1. **Modèle logique (MLD)**

* **Table filieres**

|  |  |  |  |
| --- | --- | --- | --- |
| Champ | Type | Clé | Contraintes |
| id\_filiere | INT | PK | AUTO\_INCREMENT |
| libelle\_filiere | VARCHAR(50) |  | NOT NULL |
| description\_filiere | VARCHAR(255) |  | NULL |

* **Table etudiants**

|  |  |  |  |
| --- | --- | --- | --- |
| Champ | Type | Clé | Contraintes |
| Id\_etudiant | INT | PK | AUTO\_INCREMENT |
| nom\_etudiant | VARCHAR(50) |  | NOT NULL |
| Prenom\_etudiant | VARCHAR(50) |  | NOT NULL |
| age | INT |  | NOT NULL |
| date\_naissance | DATE |  | NOT NULL |
| filiere | INT | FK | REFERENCES filieres(id\_filiere) |
| date\_inscription | DATE |  | NULL |
| photo\_profil | BLOB |  | NULL |

* **Table roles**

|  |  |  |  |
| --- | --- | --- | --- |
| Champ | Type | Clé | Contraintes |
| id\_role | INT | PK | AUTO\_INCREMENT |
| libelle\_role | VARCHAR(50) |  | UNIQUE, NOT NULL |
| description\_role | VARCHAR(255) |  | NULL |

* **Table utilisateurs**

|  |  |  |  |
| --- | --- | --- | --- |
| Champ | Type | Clé | Contraintes |
| id\_utilisateur | INT | PK | AUTO\_INCREMENT |
| username | VARCHAR(45) |  | NOT NULL |
| password | VARCHAR(60) |  | NOT NULL (haché avec bcrypt côté backend) |
| Id\_role | INT | FK | REFERENCES roles(id\_role) |
| photo\_profil | BLOB |  | NULL |

* **Table gestion**

|  |  |  |  |
| --- | --- | --- | --- |
| Champ | Type | Clé | Contraintes |
| id\_gestion | INT | PK | AUTO\_INCREMENT |
| id\_utilisateur | INT | FK | REFERENCES utilisateurs(id\_utilisateur) |
| id\_etudiant | INT | FK | REFERENCES etudiants(Id\_etudiant) |
| action | ENUM |  | ENUM('Creation', 'Modification', 'Suppression') NOT NULL |
| date\_gestion | DATE |  | NOT NULL |

Une image contenant texte, capture d’écran, diagramme, ligne

Le contenu généré par l’IA peut être incorrect.

1. **Contraintes d’intégrité**

* **Clés primaires** sur toutes les entités pour garantir l’unicité.
* **Clés étrangères** pour assurer la cohérence des relations entre les tables.
* **Énumération des actions** pour limiter les types d’opérations dans la table gestion.
* **Indexation** automatique sur les champs de jointure (filière, rôle, utilisateur, étudiant).

**5. Sécurité**

* Les mots de passe sont **hachés avec bcrypt** avant insertion.
* Les rôles d’utilisateur permettent une **gestion des autorisations** :
  + **Administrateur** : accès total
  + **Secrétaire** : gestion des inscriptions
  + **Professeur** : consultation uniquement

**6. Exploitation et évolutivité**

* La structure permet de générer des statistiques :
  + Nombre d’étudiants par filière
  + Historique des modifications par utilisateur
* La base peut être étendue avec :
  + Des modules de notes
  + Des emplois du temps
  + Une messagerie interne