

|  |
| --- |
| Projet base de données |
| Hugues Bégeot  Célia Bunouf  L3 Int – Groupe 4 |

Une image contenant objet

Description générée automatiquement

Table des matières

[Introduction 3](#_Toc8160287)

[Dictionnaire de données 4](#_Toc8160288)

[MCD [Modèle conceptuel de données] : 6](#_Toc8160289)

[MLD [Modèle Logique de Données] : 7](#_Toc8160290)

[Scénario d’exécution : 7](#_Toc8160291)

# 

# Introduction

Dans le cours de Base de données, nous avions pour projet de produire un programme capable d’effectuer la gestion administrative d’une école avec ses élèves, professeurs, cours et groupes. Ainsi, que s’occuper de la gestion des notes des étudiants.

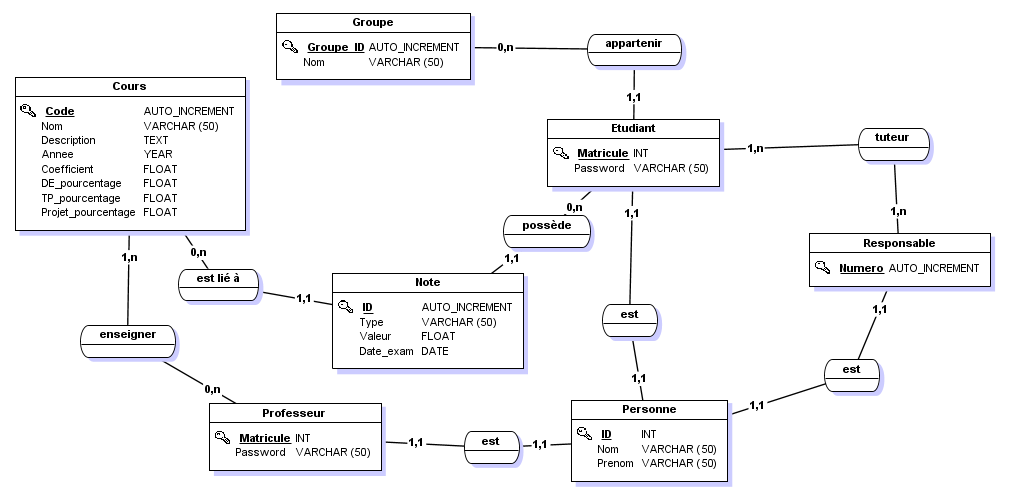
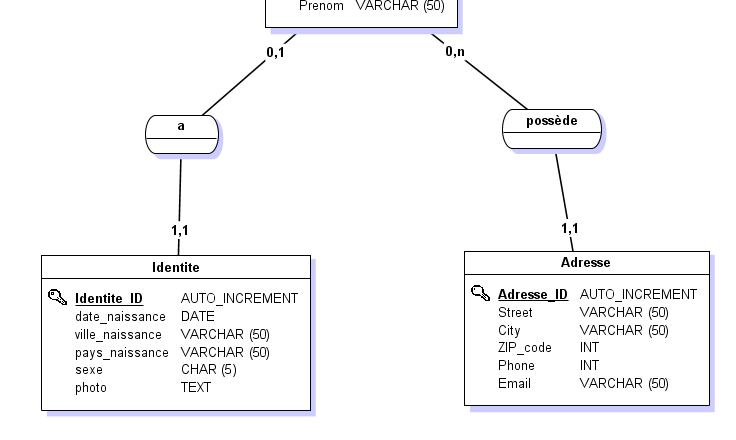
L’objectif final du projet est de créer une interface graphique permettant toutes les fonctionnalités décrites précédemment.

Le projet a été réalisé en JAVA, avec une base de données MySQL. L’interface a été réalisée à l’aide de JavaSwing.

# Dictionnaire de données

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Table** | **Nom** | **Description** | **Domaine ou type** | **Taille** | **Commentaire** | **Contraintes, règles de calcul** |
| Cours | Code | Code du cours | Auto\_increment |  |  | Automatique |
| Nom | Nom du cours | Varchar | 50 |  | Obligatoire |
| Description | Description du cours | Text |  |  |  |
| Annee | Année d'enseignement du cours | Year |  |  |  |
| Coefficient | coefficient du cours | Float |  |  |  |
| DE\_pourcentage | Pourcentage du DE associé au cours | Float |  |  |  |
| TP\_pourcentage | Pourcentage du TP associé au cours | Float |  |  |  |
| Projet\_pourcentage | Pourcentage du Projet associé au cours | Float |  |  |  |
| Personne | ID | Index de la personne | Auto\_increment |  |  | Automatique |
| Nom | Nom du cours | Varchar | 50 |  | Obligatoire |
| Prenom | Prénom de la personne | Varchar | 50 |  |  |
| Adresse | Adresse\_ID | Identifiant de l'adresse | Auto\_increment |  |  |  |
| Street | rue de l'adresse | Varchar | 50 |  |  |
| City | ville de l'adresse | Varchar | 50 |  |  |
| ZIP\_code | code postal de la ville | Int | 50 |  |  |
| Email | email de la personne | Varchar | 50 |  |  |
| Phone | numero de téléphone de la personne | Int |  |  |  |
| Identité | identite\_ID | Identifiant de l'identité | Auto\_increment |  |  | Automatique |
| date\_naissance | date de naissance | Date |  |  |  |
| ville\_naissance | ville de naissance | Varchar | 50 |  |  |
| pays\_naissance | pays de naissance | Varchar | 50 |  |  |
| sexe | sexe de la personne | Char | 5 |  |  |
| photo | photo de la personne | Text | 50 |  |  |
| Professeur/ Etudiant | Matricule | Matricule du professeur/eleve | Int |  |  |  |
| Password | mot de passe du professeur/eleve | Varchar | 50 |  |  |
| Responsable | Numero | numero du responsible | Int |  |  | Automatique |
| Password | mot de passe de la personne | Varchar | 50 |  |  |
| Note | ID | Index de la personne | Auto\_increment |  |  | Automatique |
| Valeur | valeur de la note obtenue | Float |  |  |  |
| Type | type de note | Varchar | 50 |  |  |
| Date\_exam | date de l'examen | Date |  |  |  |

# MCD [Modèle conceptuel de données] :

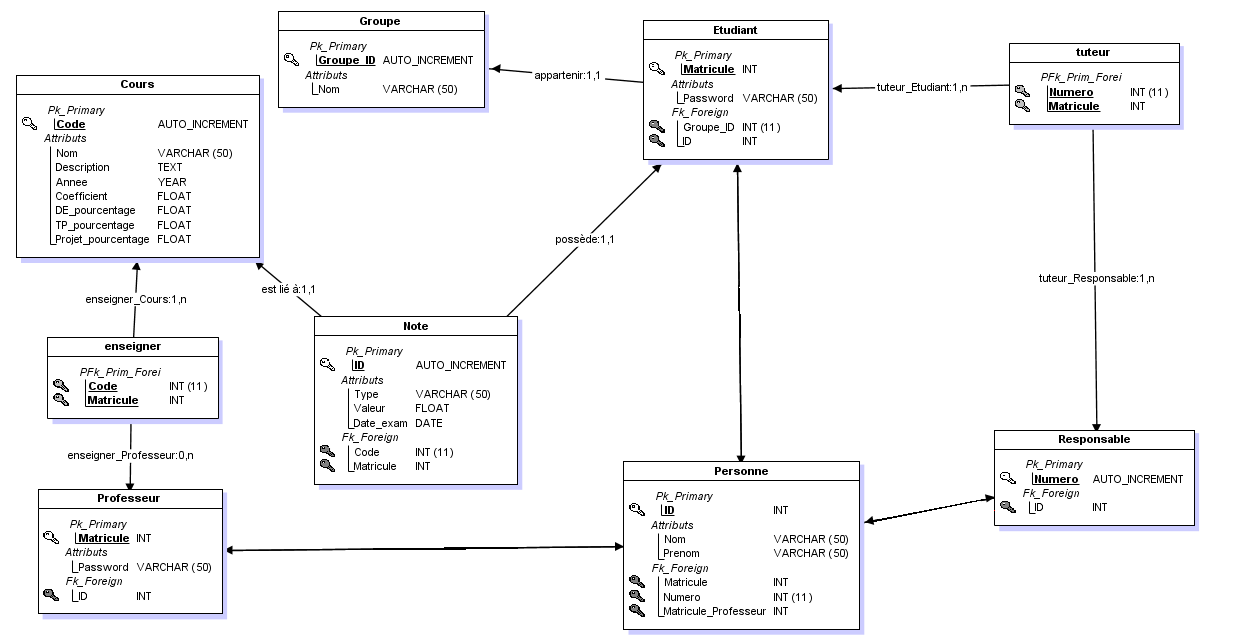
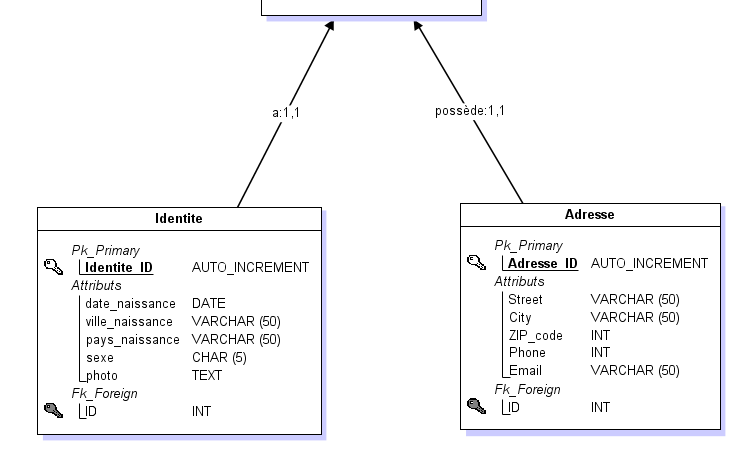
Lié aux limites de JMerise version étudiante toutes les classes n’ont pu être représentées sur le même MCD. De nombreuses classes et relations ont ainsi due être codé manuellement.

Nous avons ainsi regroupé plusieurs MCD pour tenter de représenter la base de données :

Nous avons donc décidé de partir sur un ensemble de 9 classes principales pour représenter l’ensemble d’une gestion d’école. Par la suite une 10ième classe à été rajouté, la classe administration qui possède les mêmes caractéristiques qu’un professeur (matricule et password). Nous pensons que c’est une représentation plutôt correcte de la réalité.

# MLD [Modèle Logique de Données] :

Tout comme le MCD, le MLD n’a pu être réaliser entièrement automatiquement et à donc été implémenté manuellement.

 Voici une représentation MLD obtenu en combinant plusieurs MLD crées automatiquement :

Des tables ont donc été modifiées manuellement.

# Scénario d’exécution :

Tout d’abord, la personne utilisant l’application doit se connecter à l’aide de son matricule et son mot de passe. Ainsi, chaque matricule est unique et un mot de passe lui est associé.

[Schéma LOGIN]

Ensuite en fonction du statut de la personne dans l’établissement différente interface s’offre à l’utilisateur. Ainsi 3 cas différents :

* L’utilisateur fait partie de l’administration
* L’utilisateur est un étudiant
* L’utilisateur est un professeur

# Difficultés rencontrées :

La difficulté rencontrée a été la connexion à la base de données au début car nous avions connecté notre IDE à la base de données et non le programme avant de comprendre qu’il suffisait d’inclure un fichier de connexion java.

# Points forts :

Pour ce projet nous avons tenter de procurer une facilité d’utilisation et une plaisance visuelle de l’application. Nous pensons alors que l’un des points forts de notre projet est sa qualité d’interface, et sa praticité.

# Points faibles :

Après implémentation de la base de données avec Java, nous avons remarqué que certains choix fait au départ n’étaient peut-être pas judicieux, notamment le fait d’avoir un ID pour le groupe car il n’est pas possible d’avoir deux groupes possédant le même nom, de ce fait, le nom doit être unique et non null et aurait pu être la seule clé primaire.