

S2.04 Exploitation d'une Base de Données - Partie 1

1) Cahier des charges

- Environnement:

L'environnement du projet est l'IUT de Villetaneuse.

- Interlocuteur(s):

L'interlocuteur principal du projet est le corps enseignant de l'IUT, car la gestion des notes sera principalement effectuée par les enseignants.

- Contexte:

L'objectif de ce projet est de faciliter la gestion des notes en mettant en place une base de données qui permettra de gérer facilement les notes des étudiants. Cette base de données devra permettre d'ajouter, de modifier et de supprimer les notes.

- Exigences fonctionnelles:

Création d'un modèle de données adapté à la gestion des notes des étudiants en BUT.

Mise en place d'une interface utilisateur conviviale pour faciliter l'ajout, la modification et la suppression des notes.

Gestion des relevés de notes et des bilans.

Mise en œuvre des restrictions d'accès aux données en fonction des rôles des utilisateurs (étudiant, enseignant, responsable de matière, etc.).

- Exigences non fonctionnelles:

- Technique:

Utilisation d'un système de gestion de base de données relationnelle pour stocker les données.

Le langage de programmation SQL sera utilisé à travers le SGBD PostgresQL.

- Performance:

Temps de réponse rapide lors de l'accès et de la manipulation des données.

Gestion efficace de la volumétrie des données pour assurer des performances optimales.

Il faudra respecter les normes de gestion et de création de base de données et des différents niveaux d'accès de la BD avec des rôles pour chaque utilisateurs.

- Ergonomie:

Interface utilisateur intuitive et conviviale pour faciliter l'utilisation de l'application par les enseignants.

Prise en compte des bonnes pratiques en matière de design d'interface utilisateur.

- Volumétrie:

La base de données devra pouvoir gérer un volume important de données, en prenant en compte le nombre d'étudiants et de matières.

- Sécurité:

Mise en place de mesures de sécurité pour protéger les données des étudiants, en assurant l'accès restreint aux personnes autorisées.

Utilisation de techniques de chiffrement si nécessaire pour sécuriser les données sensibles.

- Risque(s) à ne pas faire le projet:

Difficulté accrue dans la gestion manuelle des notes des étudiants, pouvant entraîner des erreurs et une perte de temps pour les enseignants.

Manque de centralisation et de traçabilité des données relatives aux notes des étudiants.

- Principaux jalons:

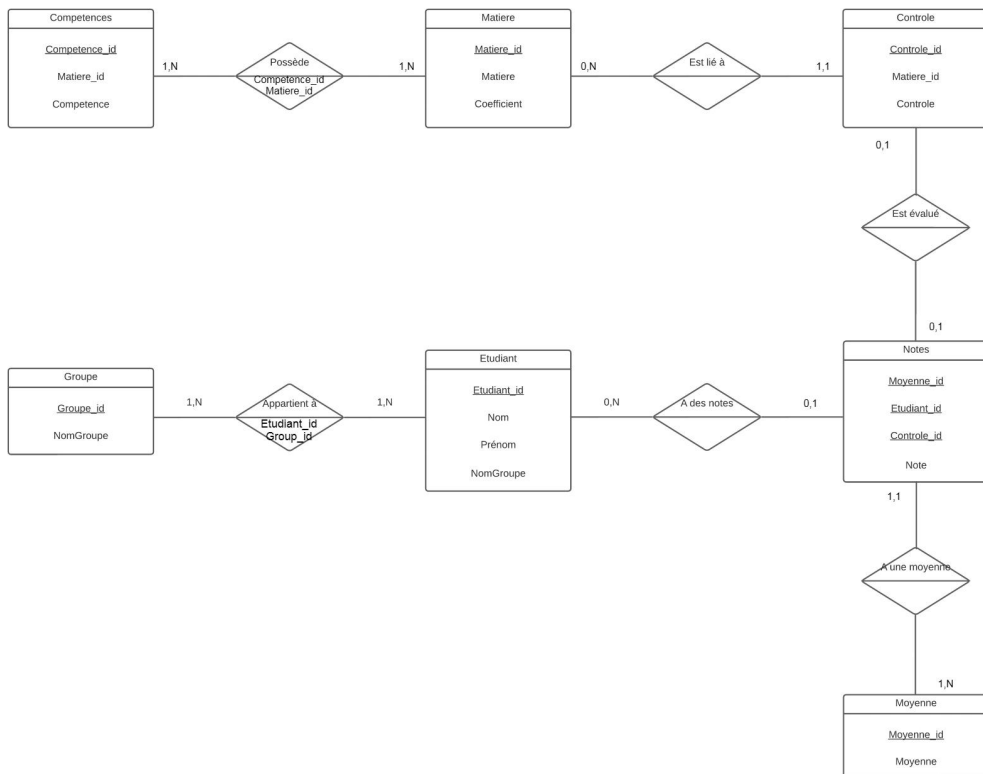
Lundi 22 mai 2023 → Modélisation des données et Création de la base de données à partir de cette modélisation.

- Limites:

Ce projet ne couvre que la gestion des notes des étudiants en BUT à l'IUT de Villetaneuse et ne prend pas en compte d'autres aspects de la gestion académique.

La base de données ne stockera que les informations relatives aux notes des étudiants et ne gèrera pas d'autres types de données administratives.

2) Modèle de données



Le modèle entité-association a été utilisé pour représenter toutes les données dont nous aurons besoin. Nous avons pris la décision de créer plusieurs relations entre les tables qui sont pour nous nécessaires.

La table Etudiant représente les informations sur les étudiants, chaque étudiant est identifié de manière unique par une clé primaire (Etudiant_id).

La table Moyenne stocke les moyennes des étudiants. Chaque moyenne est associée à une clé primaire (Moyenne_id) et est représentée par un INTEGER.

La table Matiere stocke les informations sur les matières enseignées. Chaque matière possède une clé primaire (Matiere_id), un nom et un coefficient.

La table Controle représente les différents controles effectués pour chaque matière. Chaque controle est identifié avec une clé primaire (Controle_id) et est associé à une matière spécifique avec une clé étrangère (Matiere_id).

La table Notes stocke les notes des étudiants pour chaque controle. Les notes sont liées à un étudiant spécifique (Etudiant_id), un controle spécifique (Controle_id) et

une moyenne spécifique (Moyenne_id). La table contient aussi un attribut Note qui représente la valeur de la note obtenue par l'étudiant.

La table Groupe représente les différents groupes. Chaque groupe est identifié avec une clé primaire (Groupe_id) et est associé à un nom de groupe spécifique (NomGroupe).

La table Competences stocke les compétences associé à chaque matière. Chaque compétence est identifié avec une clé primaire (Compétence_id) et est liée à une matière spécifique (Matiere_id).

Ces différentes tables sont liées entre elles par des clés primaires et étrangères ce qui permet de gérer de manière structurée les informations concernant les étudiants, les groupes, les matieres, les controles, les notes, les compétences.

3) Règle de gestion de ces données et leurs mises en œuvre par des procédures stockée

Les règles de gestion des données pour la gestion des notes des étudiants en BUT peuvent être implémentées à l'aide de procédures stockées dans la base de données. Ces procédures stockées permettent de mettre en œuvre différentes règles, telles que la vérification des données pour s'assurer que les notes sont valides et numériques, le calcul automatique des moyennes en tenant compte des coefficients des matières, la restriction d'accès pour permettre aux enseignants d'accéder uniquement aux notes des étudiants inscrits dans les matières qu'ils enseignent, le contrôle des autorisations pour limiter les droits de modification et de suppression aux enseignants, responsables de matière et administrateurs, et l'enregistrement de l'historique des modifications pour garder une trace des modifications apportées aux notes. En utilisant des procédures stockées, on centralise la logique métier et on assure la cohérence et l'intégrité des données. De plus, cela peut contribuer à améliorer les performances en réduisant les échanges de données entre le serveur de base de données et l'application. La création et l'implémentation de ces procédures stockées dépendent du système de gestion de base de données utilisé et nécessitent de se référer à la documentation spécifique du système pour obtenir des détails sur leur création et leur exécution

4) Script de création de base de données

```
CREATE TABLE Etudiant (  
  
    Etudiant_id serial primary key,  
  
    Nom varchar(50) ,  
  
    Prenom varchar(50) ,
```

```

        NomGroupe varchar(50)

    );

CREATE TABLE Moyenne (

    Moyenne_id int primary key,

    Moyenne float

);

CREATE TABLE Matiere(

    Matiere_id varchar(10) primary key,

    Matiere varchar,

    Coefficient int

);

CREATE TABLE Controle (

    Controle_id serial primary key,

    Matiere_id varchar(10) references Matiere(Matiere_id),

    Controle varchar

);

CREATE TABLE Notes (

    Etudiant_id int references Etudiant(Etudiant_id),

    Controle_id int references Controle(Controle_id),

    Moyenne_id int references Moyenne(Moyenne_id),

    Note decimal(4,2),

```

```
primary key(Etudiant_id,Controle_id,Moyenne_id)
);

CREATE TABLE Groupe (

    Groupe_id int primary key,

    NomGroupe varchar(50)

);

CREATE TABLE Competences(

    Competence_id int primary key,

    Matiere_id varchar(10) references Matiere(Matiere_id),

    Competence varchar(50)

);
```