

Dossier de gestion de projet SAE S3 (Mussto)



Cahier des charges	3
Présentation du projet	3
Enjeux	3
Contraintes	3
A. Temporelles	3
B. Qualitatives	3
C. Juridiques	3
Fonctionnalités	4
A. Fonctionnalités Étudiant	4
B. Fonctionnalités Professeur	4
C. Fonctionnalités Administrateur	4
Identification EPIC	5
Rédaction EPIC	5
1. Information	5
2. Authentification	5
3. Navigation	5
4. Administration	5
5. Communication	6
6. Gestion des accès	6
Risques	6
Définition des User Stories	6
a. User Stories	6
b. Tâches	10
Planification du projet	12
A. Gantt	12
Spécifications Formelles	12
Plan du site	12
Maquettage	13
Étudiant	13
Enseignant	14
Administrateur	16
Technologies utilisées	16
Diagramme de Package	16
Diagrammes de Cas d'utilisation	17
Package scolarité	17
Package gestion	17
Package contact	18
Diagrammes de Classe	18
Package scolarité	18
Package gestion	19
Package contact	19

Cahier des charges

Présentation du projet

Suite à une obsolescence de leur ancien site web de gestion de note nommé Tomuss, l'Université Claude Bernard Lyon 1 a décidé de faire un appel d'offre afin d'améliorer et de remettre à neuf leur site web. Notre entreprise ayant été choisie suite à l'appel d'offre, nous sommes donc chargés du projet.

Celui-ci consiste à la création d'une application web permettant la gestion des informations de scolarité (note, sondage, date d'un devoir surveillé, module et groupe) des étudiants par les professeurs et la diffusion de celles-ci aux étudiants de l'Université Claude Bernard Lyon 1.

Notre application se limitera à l'enregistrement par les professeurs des notes de chaque étudiant, des devoirs à venir et à l'affichage pour chaque étudiant des informations de sa scolarité.

Enjeux

L'aboutissement du projet est important afin de garantir aux étudiants la capacité de suivre leur scolarité dans de meilleures conditions que celles de l'ancien site web ainsi de permettre une manipulation simple et rapide pour faire gagner du temps aux professeurs .

Contraintes

A. Temporelles

Le projet doit être terminé le 19 janvier 2023.

B. Qualitatives

Le projet doit être hébergé sur le serveur web de l'IUT Informatique.

C. Juridiques

La protection des données des étudiants doit être garantie.

Fonctionnalités

A. Fonctionnalités Étudiant

- **Consultation des modules** : L'étudiant devra avoir la possibilité d'avoir accès à tous ces modules
- **Consultation des notes** : L'étudiant devra avoir la possibilité d'avoir accès à toutes ces notes qui seront classées par modules
- **Sondage** : L'étudiant devra avoir la possibilité de répondre aux sondages qu'il a reçu
- **Date des devoirs surveillés** : L'étudiant devra avoir possibilité de connaître les dates auxquelles il a des devoirs surveillés dans un module

B. Fonctionnalités Professeur

- **Visualiser les modules** : Le professeur doit avoir la possibilité de visualiser les modules auxquels il est affecté
- **Ajouter des notes** : Le professeur doit avoir la possible d'ajouter des notes aux étudiants sur ses modules
- **Créer des sondages** : Le professeur doit avoir la possibilité de créer des sondages à faire remplir par les élèves
- **Programmer un DS** : Le professeur doit pouvoir programmer un devoir surveillé sur un module pour que les étudiants puissent savoir quand ils auront ce devoir
- **Contacter les administrateurs** : Le professeur doit pouvoir contacter un administrateur en cas de besoin

C. Fonctionnalités Administrateur

- **Création et gestion d'utilisateurs** : L'administrateur doit avoir la possibilité de créer des comptes utilisateurs ainsi que de modifier les informations de ces derniers
- **Création et gestion des modules** : L'administrateur doit avoir la possibilité de créer de nouveaux modules et d'affecter les enseignants ainsi que les groupes à ces derniers
- **Création et gestion des groupes classe** : L'administrateur doit pouvoir créer les différents groupes classes et affecter étudiants à ces derniers
- **Contacter les professeurs** : L'administrateur doit pouvoir contacter un professeur si celui-ci a un problème

Identification EPIC

1. **Information** : stockage des modules / notes / groupes / devoir surveillé dans la base de données lors de leur création/modification d'un compte
2. **Authentification** : gestion de la connexion d'un utilisateur à un compte existant, affichage de messages d'erreurs en cas de mot de passe incorrect ou d'identifiant introuvable.
3. **Navigation** : gestion de la navigation d'un utilisateur entre les pages auxquelles il a accès en fonction de son statut (professeur / étudiant / administrateur) et affichage des informations le concernant en fonction de la page auquel il se trouve
4. **Administration** : page de gestion (création de compte / assignation module / affectation à un groupe)
5. **Communication** : sondages entre les professeurs et les élèves, et messages entre professeurs et administrateurs
6. **Gestion des accès** : gestion des accès en fonction du type de compte

Rédaction EPIC

1. Information

Les informations du site seront stockées dans une base de données, chaque type d'utilisateur (administrateur, professeur, étudiant) fera l'objet d'une table SQL pour les différencier. De même pour les groupes, modules, notes et devoirs surveillés qui auront tous une table SQL dédiée. Toutes les tables seront dans une base de données relationnelle.

2. Authentification

L'utilisateur aura la possibilité de se connecter sur notre plateforme via une page de connexion, le compte devra se verrouiller au-delà du troisième essai de connexion non concluant. Si la connexion s'est bien effectuée, le site devra rediriger l'utilisateur sur la bonne interface en fonction du type de compte. Enfin, si le compte en lui-même n'est pas reconnu il y aura un message d'erreur

3. Navigation

Un utilisateur pourra naviguer entre les pages auxquelles il a accès en fonction de son statut via un menu. Chacune des pages affichera le contenu correspondant à celle-ci. Un étudiant aura accès à 4 types de pages : une page d'accueil, une page de notes et une page de modules pour voir ceux auxquels il est inscrit. Un professeur aura accès à une page module permettant l'ajout de note ainsi que la création d'un sondage en fonction du module sélectionné et une page discussion. L'administrateur aura accès à une page de gestion permettant la création d'utilisateur, de groupe et de module, une page de gestion d'utilisateur, de module et de groupe et une page de discussion.

4. Administration

Le site disposera de plusieurs pages de gestion auxquelles seuls les administrateurs auront accès. Celles-ci permettent de créer des comptes (étudiants / professeurs / administrateurs (un compte administrateurs sera créé avant la livraison du produit afin de permettre le bon fonctionnement du site)), des modules et des groupes.

De plus, l'administrateur aura accès à la liste des utilisateurs du site et peut modifier les informations les concernant (groupe, module, etc...).

5. Communication

Un professeur peut créer des sondages pour un module qui lui est affecté et peut les voir. Un étudiant peut voir le sondage s'il y est affecté ou s'il appartient à un groupe qui y est affecté, il peut aussi y répondre et le professeur pourra voir sa réponse. Un professeur peut envoyer un message à un administrateur via une page de contact.

6. Gestion des accès

Un utilisateur n'aura pas les mêmes droits en fonction de son statut dans la base de données (étudiant / professeur / administrateur). Un étudiant ne peut que visualiser ses notes alors que le professeur peut en rentrer.

Risques

- Git : Sauvegarde corrompue.
- Serveur : Serveur web de l'IUT inaccessible.
- Humain : Problème de santé d'un développeur retardant les tâches sur lesquelles il travaillait.
- Scolaire : Impossibilité pour un développeur de travailler à cause d'un surplus de travail scolaire.
- Matériel : Problème de connexion ou de matériel d'un développeur l'empêchant de communiquer avec les autres développeurs ou de travailler sur le projet.
- Relationnel : Incapacité du groupe à se mettre d'accord sur l'une des manières de réaliser le projet pouvant mener à un retard dans le développement.

Définition des User Stories

a. User Stories

Référence de l'US : 1.1			
Titre de l'US : Base de données	Priorité : Très haute	Valeur : 4/10	Effort : 5j/h
En tant que : système Je souhaite que : stocker les informations de chaque utilisateur dans une base de données Afin que : je puisse les obtenir facilement à la demande des utilisateurs			
Critère d'acceptation : Etant donné que : un utilisateur utilise le site Lorsque : j'accède à ma session Alors : j'ai accès à toute les informations me concernant			

Référence de l'US : 2.1			
Titre de l'US : Connexion	Priorité : Très Haute	Valeur : 4/10	Effort : 2j/h
En tant que : administrateur du système Je souhaite que : l'accès aux compte de l'étudiant et du professeur soit sécurisé Afin que : les informations les concernant restent privés			
Critère d'acceptation : Etant donné que : je suis sur la page de connexion Lorsque : je rentre et que je valide mon identifiant Alors : une vérification de mon identifiant m'informe si oui ou non il existe Etant donné que : je suis sur le système Lorsque : je vérifie la connexion de l'utilisateur Alors : si le nombre d'essais de connexion est supérieur à 3 alors le compte est verrouillé			

Référence de l'US : 3.1			
Titre de l'US : Utilisation du menu	Priorité : Haute	Valeur : 4/10	Effort : 10j/h
<p>En tant que : utilisateur</p> <p>Je souhaite que : naviguer entre les pages facilement</p> <p>Afin que : je puisse voir les informations proposés me concernant</p>			
<p>Critère d'acceptation :</p> <p>Etant donné que : je suis étudiant</p> <p>Lorsque : je navigue dans le site via le menu</p> <p>Alors : je peux accéder à la page listant les notes que j'ai eu et consulter la liste de mes camarades de groupe.</p> <p>Etant donné que : je suis professeur</p> <p>Lorsque : je navigue dans le site via le menu</p> <p>Alors : je peux accéder à la page listant les modules auxquels je suis affecté et contacter les administrateurs.</p> <p>Etant donné que : je suis administrateur</p> <p>Lorsque : je navigue dans le site via le menu</p> <p>Alors : je peux accéder à la page me permettant de créer un nouvel utilisateur, une page de gestion d'utilisateur et de contacter les professeurs qui ont des problèmes sur le site</p>			

Référence de l'US : 4.1			
Titre de l'US : Création groupe, utilisateur et module	Priorité : moyenne	Valeur : 6/10	Effort : 4j/h
<p>En tant que : administrateur du système</p> <p>Je souhaite que : ajouter un nouvel utilisateur/ nouveau groupe/nouveau module</p> <p>Afin que : qu'il puisse utiliser cet "outil" proposé par l'IUT/ l'on puisse ajouter des étudiants à un groupe et un module (idem pour le professeur)</p>			
<p>Critère d'acceptation :</p> <p>Etant donné que : je suis sur la gestion des utilisateurs</p> <p>Lorsque : je rentre l'identifiant et le mot de passe du nouvel utilisateur ainsi que son status (étudiant, professeur, administration)</p> <p>Alors : le système inscrit le nouvel utilisateur dans la base de donnée</p> <p>Etant donné que : je suis sur la gestion des utilisateurs</p> <p>Lorsque : je rentre le nom du nouveau groupe classe</p> <p>Alors : le système inscrit le nouveau groupe dans la base de donnée</p> <p>Etant donné que : je suis sur la gestion des utilisateurs</p> <p>Lorsque : je rentre le nom du nouveau module</p> <p>Alors : le système inscrit le nouveau module dans la base de donnée</p>			

Référence de l'US : 4.2			
Titre de l'US : Gestion groupe, utilisateur et module	Priorité : moyenne	Valeur : 6/10	Effort : 8j/h
<p>En tant que : administrateur du système</p> <p>Je souhaite que : consulter la liste des utilisateurs du site</p> <p>Afin que : je puisse modifier le groupe d'étudiant, le module affecté à un professeur, etc</p>			
<p>Critère d'acceptation :</p> <p>Etant donné que : je suis sur la gestion des utilisateurs</p> <p>Lorsque : je modifie des informations d'un utilisateurs (comme le groupe d'un étudiant ou les modules qu'un professeur enseigne ou les modules suivie par les étudiants)</p> <p>Alors : le système met à jour la base de donnée</p>			

Référence de l'US : 5.1			
Titre de l'US : Contact	Priorité : Basse	Valeur : 2/10	Effort : 1j/h
<p>En tant que : professeur/administrateur système/étudiant</p> <p>Je souhaite que : pouvoir communiquer avec un administrateur/professeur/étudiant</p> <p>Afin que : je puisse signaler d' éventuelles difficultés d'utilisation du site/répondre au problème des professeurs/ envoyer ou répondre à des sondages</p>			
<p>Critère d'acceptation :</p> <p>Etant donné que : je suis professeur</p> <p>Lorsque : j'envoie un message à un administrateur</p> <p>Alors : le message est inséré dans la base de donnée et est visible par l'administrateur</p> <p>Etant donné que : je suis administrateur</p> <p>Lorsque : je consulte la page de contact</p> <p>Alors : je peux voir et répondre aux différents messages</p> <p>Etant donné que : je suis professeur</p> <p>Lorsque : je publie un sondage destiné à tous les groupes d'un module</p> <p>Alors : les étudiants concernés peuvent y répondre</p> <p>Etant donné que : je suis étudiant</p> <p>Lorsque : je répond à un sondage</p> <p>Alors : le professeur qui l'a créé reçoit ma réponse</p>			

Référence de l'US : 6.1			
Titre de l'US : Gestion d'accès/droit	Priorité : Très haute	Valeur : 7/10	Effort : 4j/h
<p>En tant que : système</p> <p>Je souhaite que : chaque types d'utilisateur aient des droits différents</p> <p>Afin que : chaque types d'utilisateur aient une apparence du site différente et accès à des fonctionnalités uniques pour chaque types</p>			
<p>Critère d'acceptation :</p> <p>Etant donné que : je suis le système</p> <p>Lorsque : un utilisateur se connecte</p> <p>Alors : je vérifie son type par rapport à sa présence dans les tables "étudiant", "professeur" et "administrateur", valide son identification et le redirige vers les pages réservés à son type</p> <p>Etant donné que : je suis étudiant</p> <p>Lorsque : je me connecte</p> <p>Alors : je suis redirigé vers la page d'accueil étudiant et accès aux fonctionnalités réservés aux étudiants</p> <p>Etant donné que : je suis professeur</p> <p>Lorsque : je me connecte</p> <p>Alors : je suis redirigé vers la page d'accueil professeur et accès aux fonctionnalités réservés aux professeurs</p> <p>Etant donné que : je suis administrateur</p> <p>Lorsque : je me connecte</p> <p>Alors : je suis redirigé vers la page d'accueil administrateur et accès aux fonctionnalités réservés aux administrateurs</p>			

b. Tâches

US 1.1

Tâches :

- réalisation du Modèle Conceptuelle de Données
- rédaction du script SQL
- mise en place de la base de donnée
- préparation des requêtes SQL qui seront réalisé par le système

US 2.1

Tâches :

- organisation du répertoire du projet sous le paterne MVC
- réalisation d'un fichier de vue v_connexion.php avec un fichier CSS associé
- réalisation d'un contrôleur c_connexion.php
- réalisation d'un modèle m_connexion.php

US 3.1

Tâches :

- réalisation d'un fichier de vue v_menu.php avec un fichier CSS associé (on vérifiera aussi le type d'utilisateur pour afficher les éléments du menu par rapport à cela)
- réalisation des pages d'accueil, de module, de note, etc adapté selon le type d'utilisateur (utilisation des variables de session) avec un fichier CSS associé à chaque page ainsi que les modèles et contrôleurs associés
- réalisation de la page index.php

US 4.1

Tâches :

- formulaire pour ajout d'utilisateur dans v_gestion_utilisateur.php et ajout des éléments de création d'utilisateurs dans m_gestion_utilisateur
- formulaire pour ajout de module dans v_gestion_module.php et ajout des éléments de création de module dans m_gestion_module
- formulaire pour ajout de groupe dans v_gestion_groupe.php et ajout des éléments de création de groupe dans m_gestion_groupe

US 4.2

Tâches :

- réalisation d'un fichier de vue v_gestion_utilisateur.php avec un fichier CSS associé
- réalisation d'un contrôleur c_gestion_utilisateur.php
- réalisation d'un modèle m_gestion_utilisateur.php
- réalisation d'un fichier de vue v_gestion_groupe.php avec un fichier CSS associé
- réalisation d'un contrôleur c_gestion_groupe.php

- réalisation d'un modèle m_gestion_groupe.php
- réalisation d'un fichier de vue v_gestion_module.php avec un fichier CSS associé
- réalisation d'un contrôleur c_gestion_module.php
- réalisation d'un modèle m_gestion_module.php

US 5.1

Tâches :

- réalisation d'un fichier de vue v_contact.php avec un fichier CSS associé
- réalisation d'un contrôleur c_contact.php
- réalisation d'un modèle m_contact.php

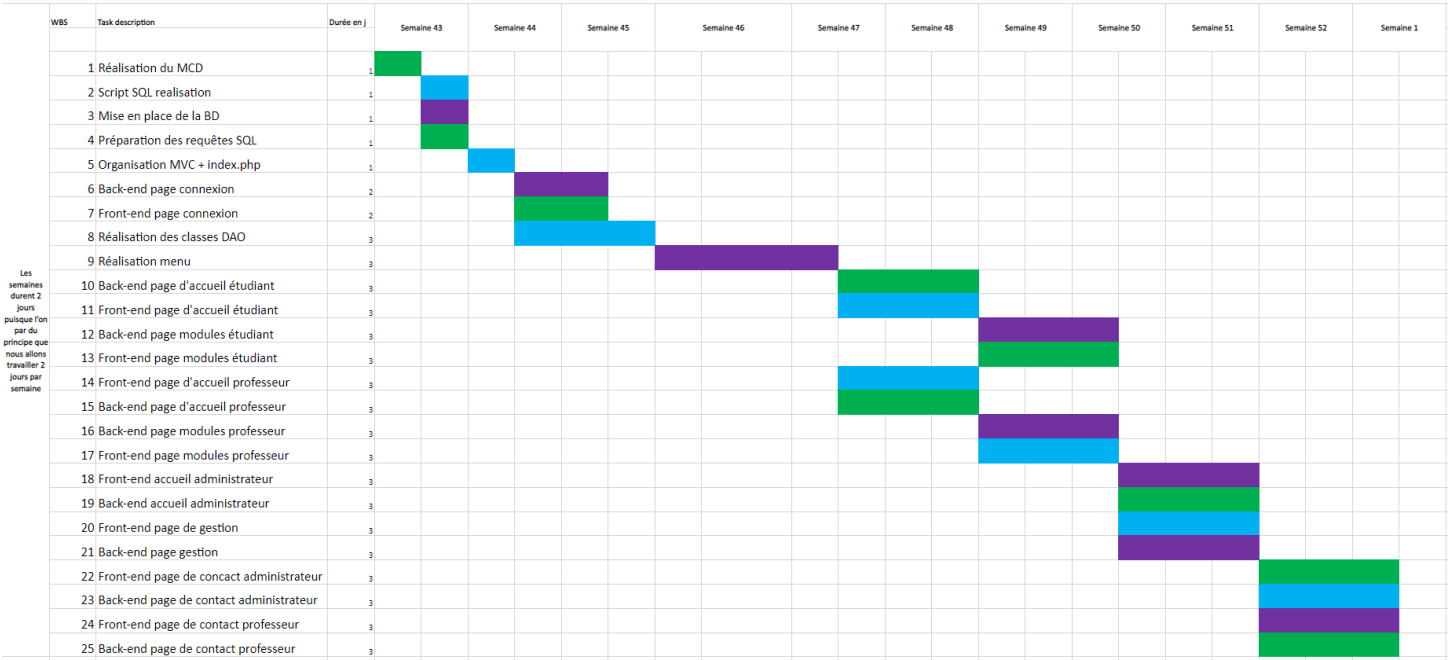
US 6.1

Tâches :

- réalisation des fichiers dans le modèle contenant les classes DAO (Data Access Object) permettant la connexion à la base de donnée et de manipuler celle-ci

Planification du projet

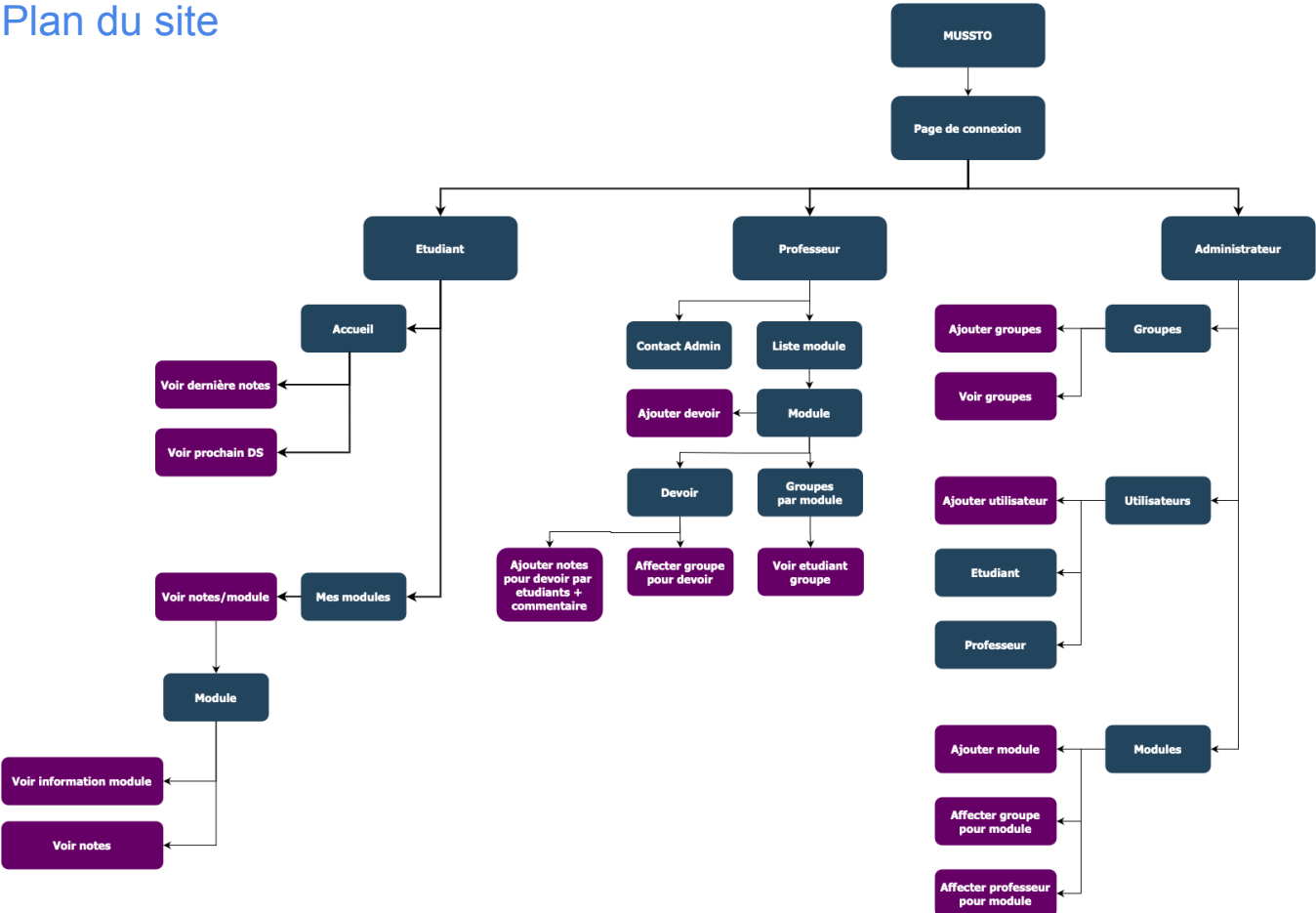
A. Gantt



Pour l'affectation des développeurs aux tâches, nous travaillerons en binôme quand le Gantt nous permet de faire des tâches en parallèle. Sinon nous travaillerons seul ou en groupe en fonction de la situation.

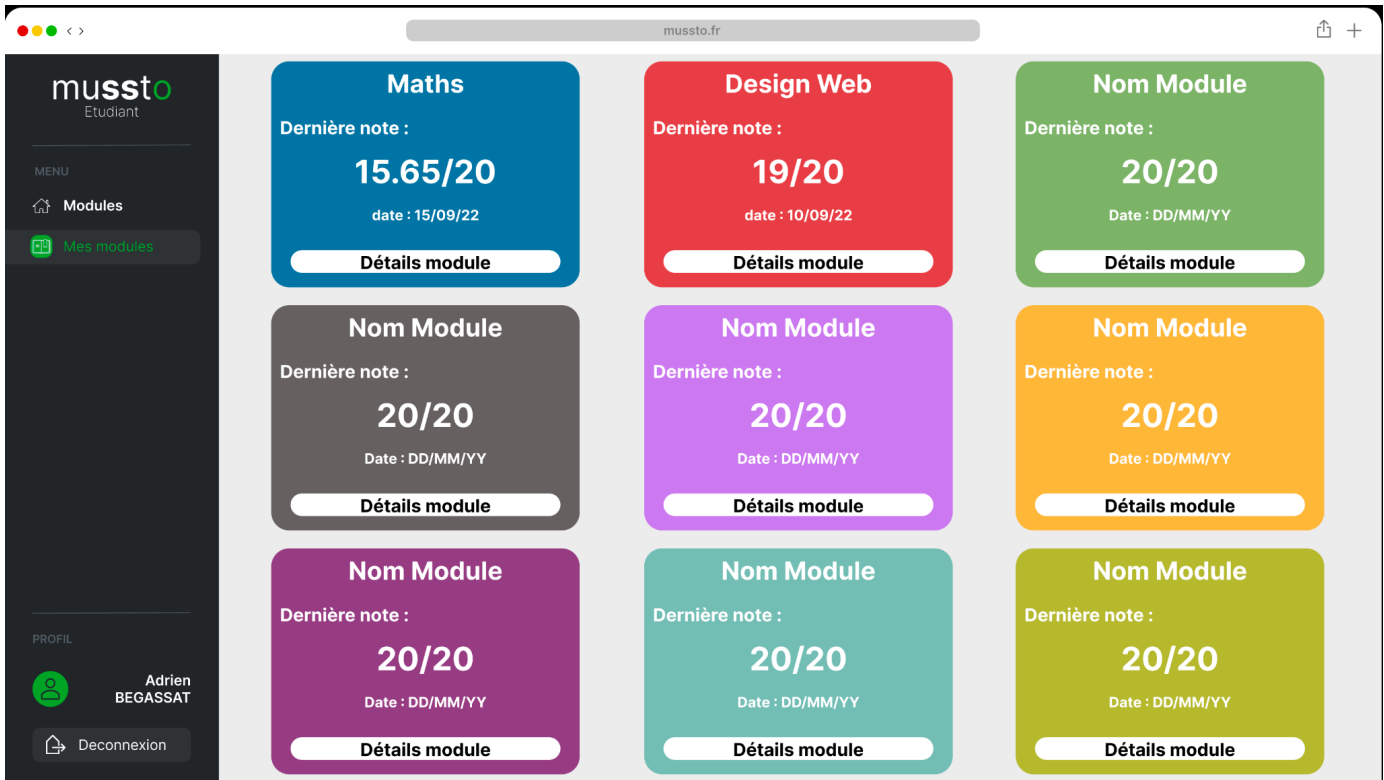
Spécifications Formelles

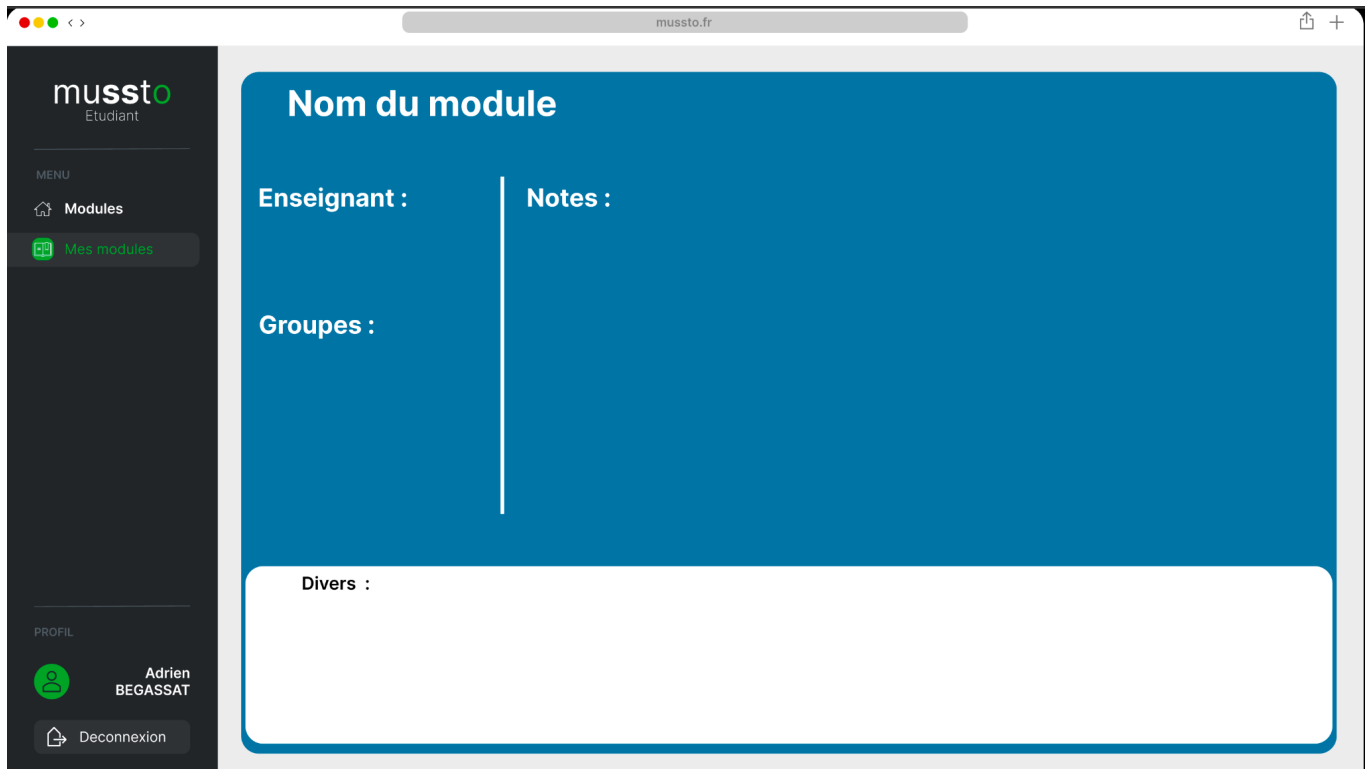
Plan du site



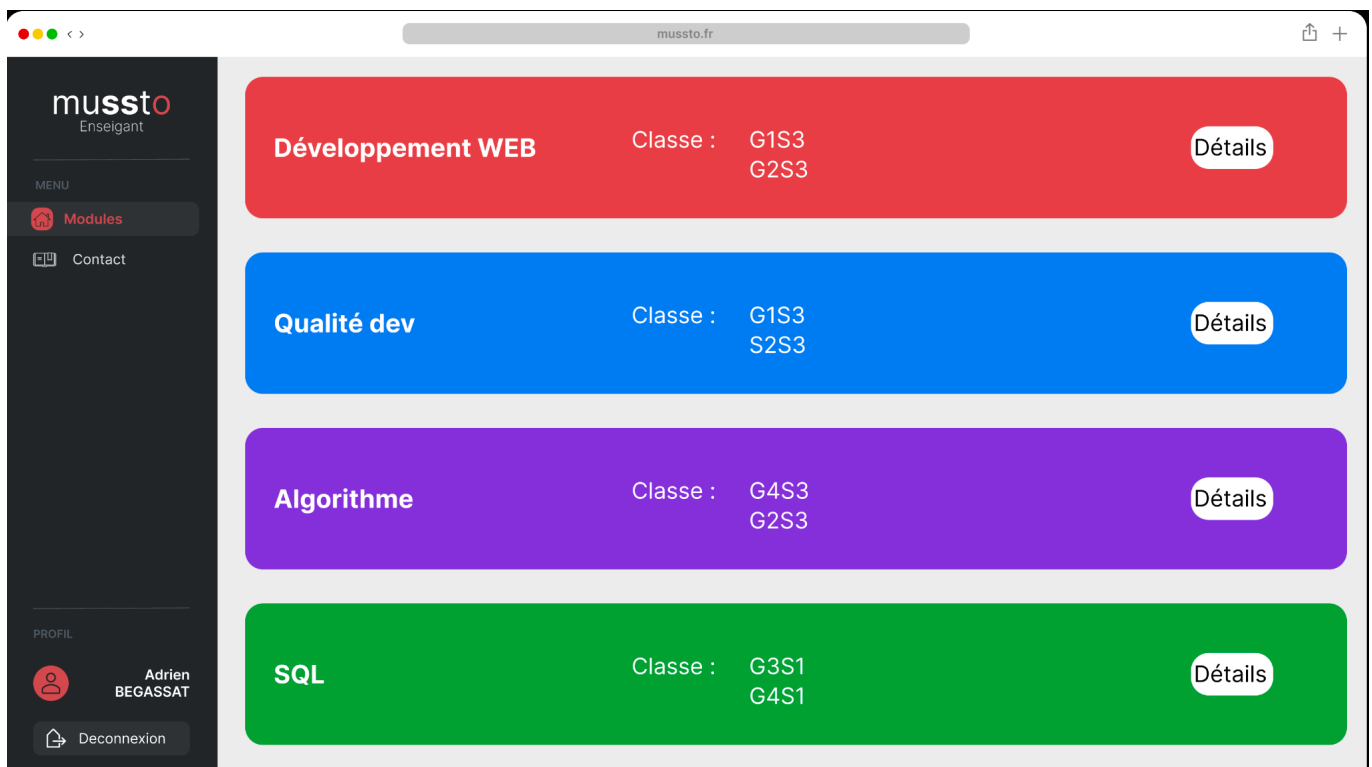
Maquettage

Étudiant





Professeur



mussto

Enseignant

MENU

Modules

Contact

PROFIL

Adrien
BEGASSAT

Deconnexion

mussto.fr

+

Créer un DS

Nom du DS

Saisissez la date du DS

G1S3 Création d'application

G2S3 Création d'application

G4S3 Gestion base de données

Valider

Annuler

mussto

Enseignant

MENU

Modules

Contact

PROFIL

Adrien
BEGASSAT

Deconnexion

mussto.fr

+

Ajouter une note

Nom du DS

Numéro de l'étudiant

Note

Valider

Administrateur

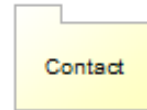


Technologies utilisées

- HTML/CSS et Javascript : front-end
- Mysql : base de donnée
- PHP: back-end
- SQL : manipulation des donnée
- Figma : maquette

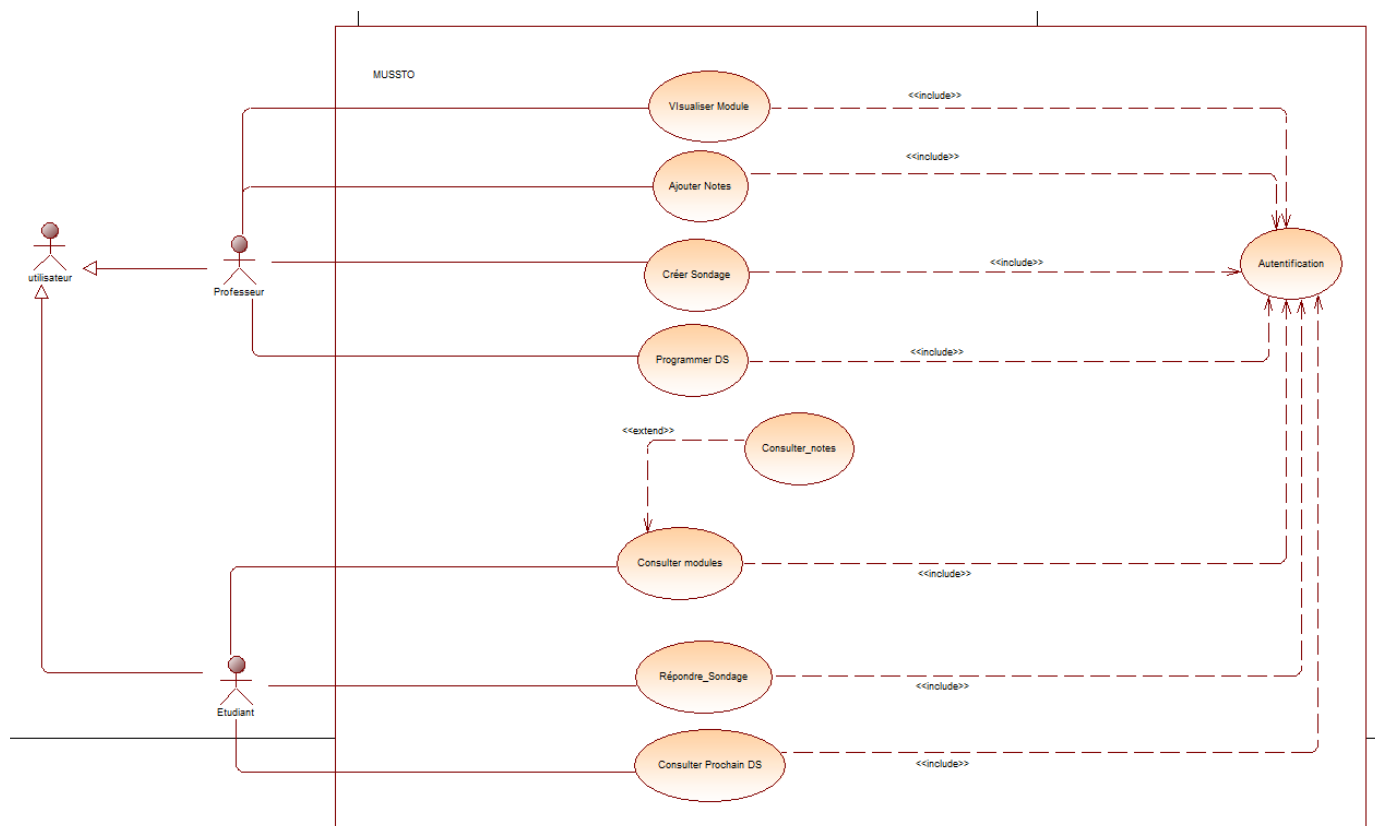
Annexe : diagramme et script SQL

Diagramme de Package

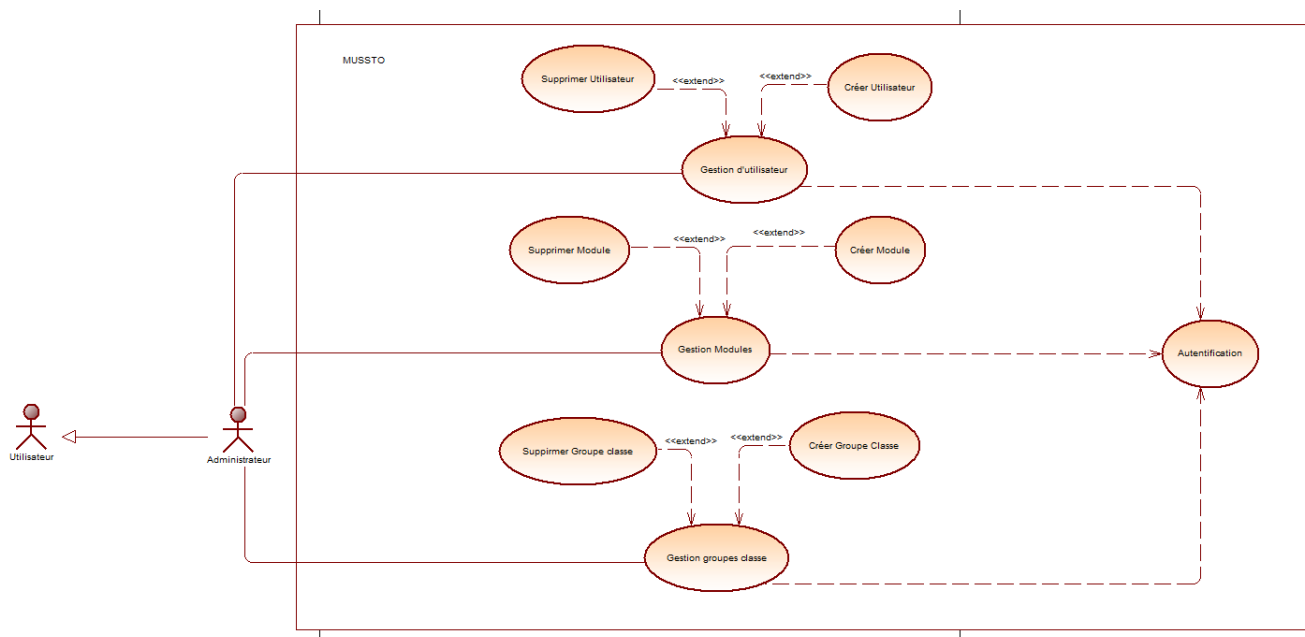


Diagrammes de Cas d'utilisation

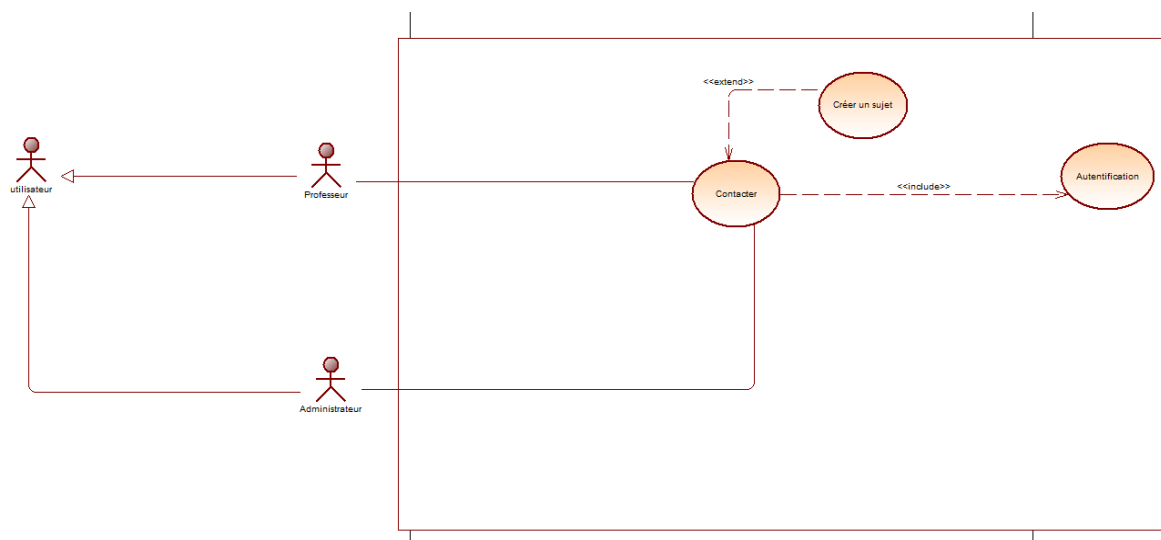
Package scolarité



Package gestion



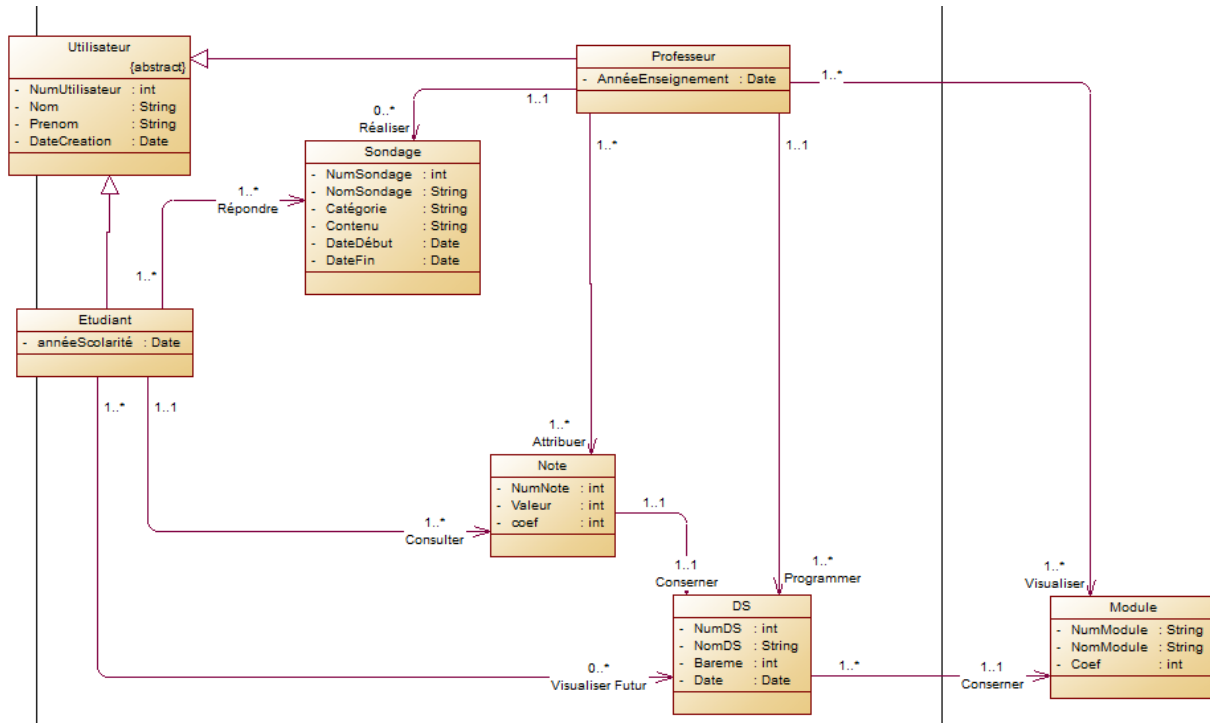
Package contact



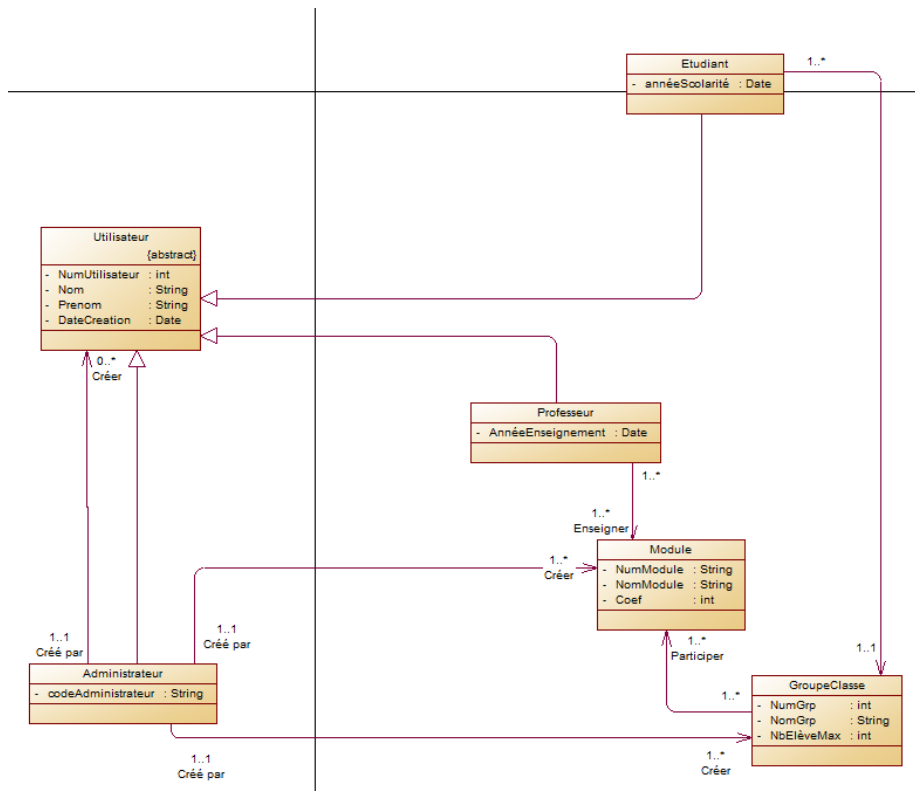
On remarque que le contenu des DCU correspond aux différentes fonctionnalités décrites dans le cahier des charges.

Diagrammes de Classe

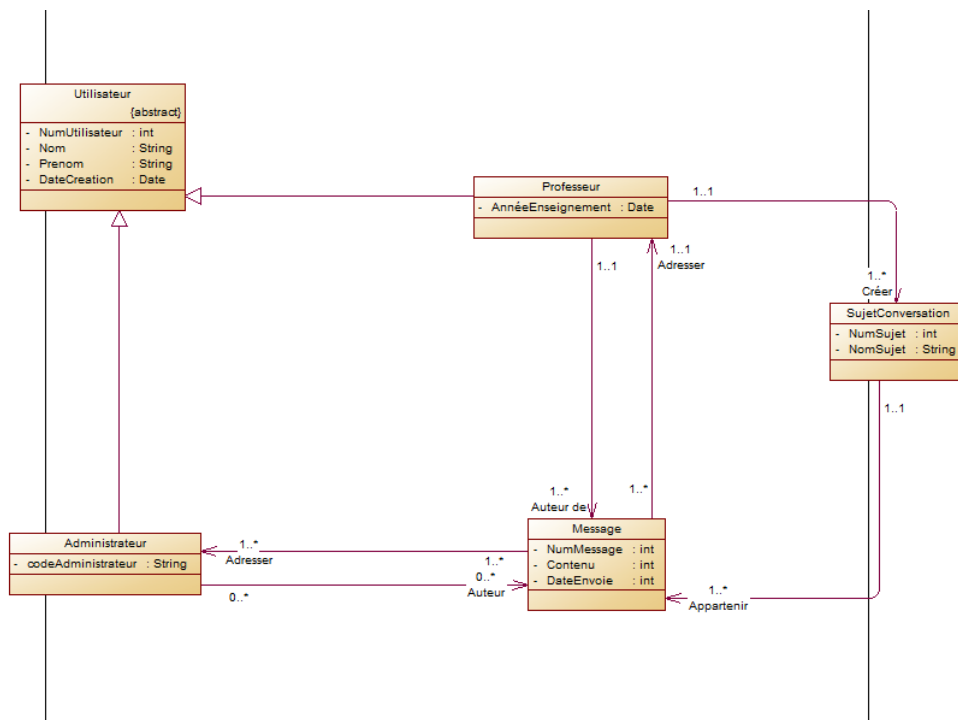
Package scolarité



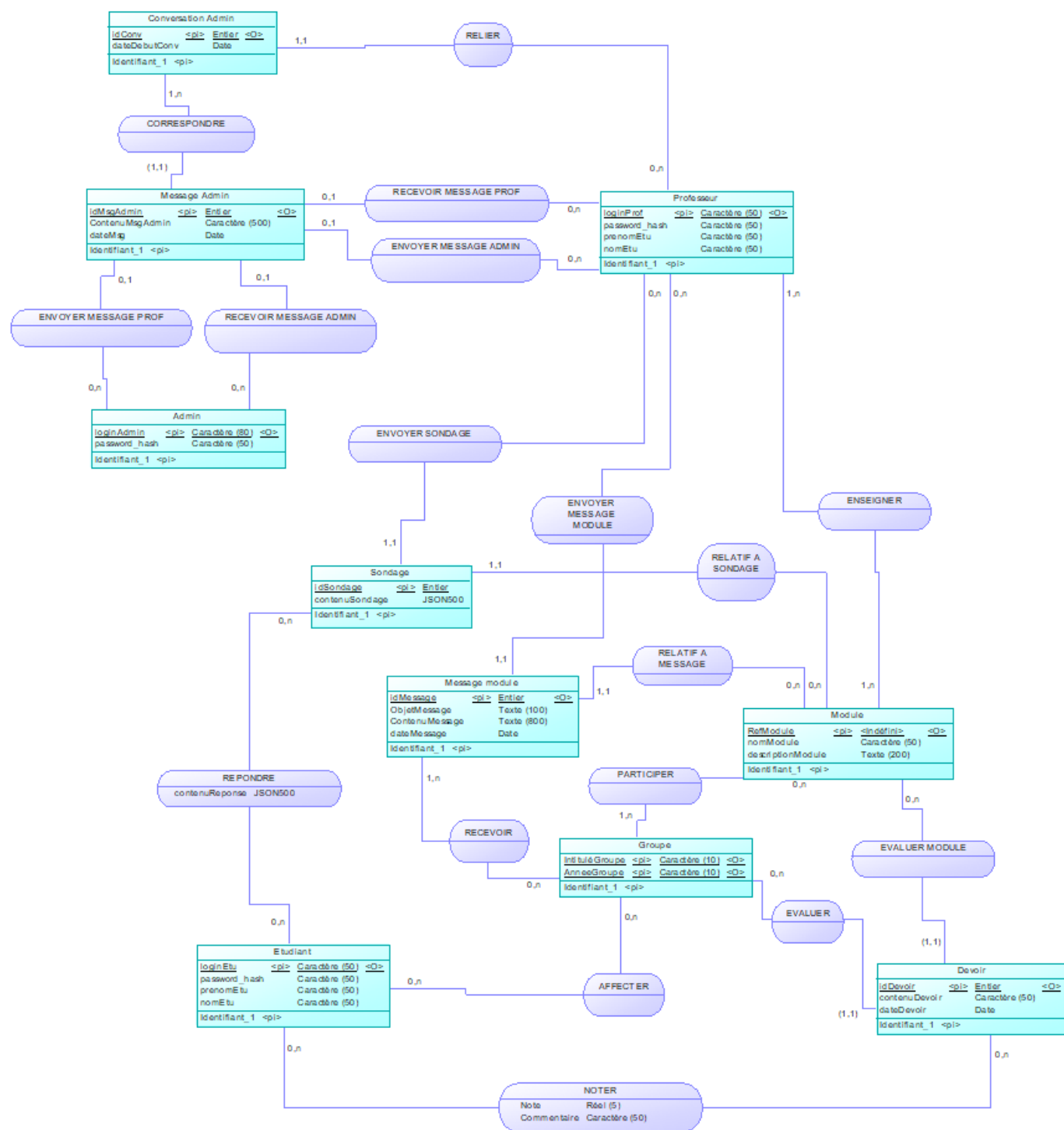
Package gestion



Package contact



MCD



Script SQL

```
--
-- Base de données : `mussto`
--

-- -----

--
-- Structure de la table `ADMIN`
--

CREATE TABLE `ADMIN` (
  `LOGINADMIN` varchar(80) NOT NULL,
  `PASSWORD_HASH` varchar(100) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

-- -----

--
-- Structure de la table `AFFECTER`
--

CREATE TABLE `AFFECTER` (
  `LOGINETU` varchar(50) NOT NULL,
  `INTITULEGROUPE` varchar(10) NOT NULL,
  `ANNEEGROUPE` varchar(10) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

-- -----

--
-- Structure de la table `CONVERSATION_ADMIN`
--

CREATE TABLE `CONVERSATION_ADMIN` (
  `IDCONV` int(11) NOT NULL,
  `LOGINPROF` varchar(50) NOT NULL,
  `DATEDEBUTCONV` date DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

-- -----

--
-- Structure de la table `DEVOIR`
--

CREATE TABLE `DEVOIR` (
  `INTITULEGROUPE` varchar(10) NOT NULL,
  `ANNEEGROUPE` varchar(10) NOT NULL,
  `REFMODULE` varchar(10) NOT NULL,
  `IDDEVOIR` int(11) NOT NULL,
  `CONTENUDEVOIR` varchar(50) DEFAULT NULL,
  `DATEDEVOIR` date DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

-- -----

--
-- Structure de la table `ENSEIGNER`
--

CREATE TABLE `ENSEIGNER` (
  `LOGINPROF` varchar(50) NOT NULL,
  `REFMODULE` varchar(10) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

-- -----

--
-- Structure de la table `ETUDIANT`
--
```



```
CREATE TABLE `ETUDIANT` (  
  `LOGINETU` varchar(50) NOT NULL,  
  `PASSWORD_HASH` varchar(100) DEFAULT NULL,  
  `PRENOMETU` varchar(50) DEFAULT NULL,  
  `NOMETU` varchar(50) DEFAULT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

```
--  
-- Structure de la table `GROUPE`  
--
```

```
CREATE TABLE `GROUPE` (  
  `INTITULEGROUPE` varchar(10) NOT NULL,  
  `ANNEEGROUPE` varchar(10) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

```
--  
-- Structure de la table `MESSAGE_ADMIN`  
--
```

```
CREATE TABLE `MESSAGE_ADMIN` (  
  `IDCONV` int(11) NOT NULL,  
  `IDMSGADMIN` int(11) NOT NULL,  
  `LOGINADMIN` varchar(80) DEFAULT NULL,  
  `LOGINPROF` varchar(50) DEFAULT NULL,  
  `ADM_LOGINADMIN` varchar(80) DEFAULT NULL,  
  `PRO_LOGINPROF` varchar(50) DEFAULT NULL,  
  `CONTENUMSGADMIN` varchar(500) DEFAULT NULL,  
  `DATEMSG` date DEFAULT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

```
--  
-- Structure de la table `MESSAGE_MODULE`  
--
```

```
CREATE TABLE `MESSAGE_MODULE` (  
  `IDMESSAGE` int(11) NOT NULL,  
  `LOGINPROF` varchar(50) NOT NULL,  
  `REFMODULE` varchar(10) NOT NULL,  
  `OBJETMESSAGE` text DEFAULT NULL,  
  `CONTENUMESSAGE` text DEFAULT NULL,  
  `DATEMESSAGE` date DEFAULT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

```
--  
-- Structure de la table `MODULE`  
--
```

```
CREATE TABLE `MODULE` (  
  `REFMODULE` varchar(10) NOT NULL,  
  `NOMMODULE` varchar(50) DEFAULT NULL,  
  `DESCRIPTIONMODULE` text DEFAULT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

```
--  
-- Structure de la table `NOTER`  
--
```

```
CREATE TABLE `NOTER` (  
  `LOGINETU` varchar(50) NOT NULL,  
  `INTITULEGROUPE` varchar(10) NOT NULL,  
  `ANNEEGROUPE` varchar(10) NOT NULL,  
  `REFMODULE` varchar(10) NOT NULL,  
  `IDDEVOIR` int(11) NOT NULL,
```

```
`NOTE` float DEFAULT NULL,  
`COMMENTAIRE` varchar(50) DEFAULT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

```
-- -----
```

```
--  
-- Structure de la table `PARTICIPER`  
--
```

```
CREATE TABLE `PARTICIPER` (  
  `INTITULEGROUPE` varchar(10) NOT NULL,  
  `ANNEEGROUPE` varchar(10) NOT NULL,  
  `REFMODULE` varchar(10) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

```
-- -----
```

```
--  
-- Structure de la table `PROFESSEUR`  
--
```

```
CREATE TABLE `PROFESSEUR` (  
  `LOGINPROF` varchar(50) NOT NULL,  
  `PASSWORD_HASH` varchar(80) DEFAULT NULL,  
  `PRENOMPROF` varchar(50) DEFAULT NULL,  
  `NOMEPROF` varchar(50) DEFAULT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

```
-- -----
```

```
--  
-- Structure de la table `RECEVOIR`  
--
```

```
CREATE TABLE `RECEVOIR` (  
  `IDMESSAGE` int(11) NOT NULL,  
  `INTITULEGROUPE` varchar(10) NOT NULL,  
  `ANNEEGROUPE` varchar(10) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

```
-- -----
```

```
--  
-- Structure de la table `REPONDRE`  
--
```

```
CREATE TABLE `REPONDRE` (  
  `LOGINETU` varchar(50) NOT NULL,  
  `IDSONDAGE` int(11) NOT NULL,  
  `CONTENUREPONSE` varchar(500) DEFAULT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

```
-- -----
```

```
--  
-- Structure de la table `SONDAGE`  
--
```

```
CREATE TABLE `SONDAGE` (  
  `IDSONDAGE` int(11) NOT NULL,  
  `REFMODULE` varchar(10) NOT NULL,  
  `LOGINPROF` varchar(50) NOT NULL,  
  `CONTENUSONDAGE` varchar(500) DEFAULT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

```
--  
-- Index pour les tables déchargées  
--
```

```
--  
-- Index pour la table `ADMIN`  
--
```

```
ALTER TABLE `ADMIN`
```

```

ADD PRIMARY KEY ('LOGINADMIN');

--
-- Index pour la table `AFFECTER`
--
ALTER TABLE `AFFECTER`
  ADD PRIMARY KEY ('LOGINETU','INTITULEGROUPE','ANNEEGROUPE'),
  ADD KEY `AFFECTER_FK` ('LOGINETU'),
  ADD KEY `AFFECTER2_FK` ('INTITULEGROUPE','ANNEEGROUPE');

--
-- Index pour la table `CONVERSATION_ADMIN`
--
ALTER TABLE `CONVERSATION_ADMIN`
  ADD PRIMARY KEY ('IDCONV'),
  ADD KEY `RELIER_FK` ('LOGINPROF');

--
-- Index pour la table `DEVOIR`
--
ALTER TABLE `DEVOIR`
  ADD PRIMARY KEY ('INTITULEGROUPE','ANNEEGROUPE','REFMODULE','IDDEVOIR'),
  ADD KEY `EVALUER_FK` ('INTITULEGROUPE','ANNEEGROUPE'),
  ADD KEY `EVALUER_MODULE_FK` ('REFMODULE');

--
-- Index pour la table `ENSEIGNER`
--
ALTER TABLE `ENSEIGNER`
  ADD PRIMARY KEY ('LOGINPROF','REFMODULE'),
  ADD KEY `ENSEIGNER_FK` ('LOGINPROF'),
  ADD KEY `ENSEIGNER2_FK` ('REFMODULE');

--
-- Index pour la table `ETUDIANT`
--
ALTER TABLE `ETUDIANT`
  ADD PRIMARY KEY ('LOGINETU');

--
-- Index pour la table `GROUPE`
--
ALTER TABLE `GROUPE`
  ADD PRIMARY KEY ('INTITULEGROUPE','ANNEEGROUPE');

--
-- Index pour la table `MESSAGE_ADMIN`
--
ALTER TABLE `MESSAGE_ADMIN`
  ADD PRIMARY KEY ('IDCONV','IDMSGADMIN'),
  ADD KEY `CORRESPONDRE_FK` ('IDCONV'),
  ADD KEY `ENVOYER_MESSAGE_PROF_FK` ('ADM_LOGINADMIN'),
  ADD KEY `RECEVOIR_MESSAGE_ADMIN_FK` ('LOGINADMIN'),
  ADD KEY `RECEVOIR_MESSAGE_PROF_FK` ('LOGINPROF'),
  ADD KEY `ENVOYER_MESSAGE_ADMIN_FK` ('PRO_LOGINPROF');

--
-- Index pour la table `MESSAGE_MODULE`
--
ALTER TABLE `MESSAGE_MODULE`
  ADD PRIMARY KEY ('IDMESSAGE'),
  ADD KEY `ENVOYER_MESSAGE_MODULE_FK` ('LOGINPROF'),
  ADD KEY `RELATIF_A_MESSAGE_FK` ('REFMODULE');

--
-- Index pour la table `MODULE`
--
ALTER TABLE `MODULE`
  ADD PRIMARY KEY ('REFMODULE');

--
-- Index pour la table `NOTER`
--
ALTER TABLE `NOTER`
  ADD PRIMARY KEY ('INTITULEGROUPE','ANNEEGROUPE','REFMODULE','LOGINETU','IDDEVOIR'),

```

```

ADD KEY `NOTER_FK` ('LOGINETU'),
ADD KEY `NOTER2_FK` ('INTITULEGROUPE', 'ANNEEGROUPE', 'REFMODULE', 'IDDEVOIR');

--
-- Index pour la table `PARTICIPER`
--
ALTER TABLE `PARTICIPER`
  ADD PRIMARY KEY ('INTITULEGROUPE', 'ANNEEGROUPE', 'REFMODULE'),
  ADD KEY `PARTICIPER_FK` ('INTITULEGROUPE', 'ANNEEGROUPE'),
  ADD KEY `PARTICIPER2_FK` ('REFMODULE');

--
-- Index pour la table `PROFESSEUR`
--
ALTER TABLE `PROFESSEUR`
  ADD PRIMARY KEY ('LOGINPROF');

--
-- Index pour la table `RECEVOIR`
--
ALTER TABLE `RECEVOIR`
  ADD PRIMARY KEY ('IDMESSAGE', 'INTITULEGROUPE', 'ANNEEGROUPE'),
  ADD KEY `RECEVOIR_FK` ('IDMESSAGE'),
  ADD KEY `RECEVOIR2_FK` ('INTITULEGROUPE', 'ANNEEGROUPE');

--
-- Index pour la table `REPONDRE`
--
ALTER TABLE `REPONDRE`
  ADD PRIMARY KEY ('LOGINETU', 'IDSONDAGE'),
  ADD KEY `REPONDRE_FK` ('LOGINETU'),
  ADD KEY `REPONDRE2_FK` ('IDSONDAGE');

--
-- Index pour la table `SONDAGE`
--
ALTER TABLE `SONDAGE`
  ADD PRIMARY KEY ('IDSONDAGE'),
  ADD KEY `ENVOYER_SONDAGE_FK` ('LOGINPROF'),
  ADD KEY `RELATIF_A_SONDAGE_FK` ('REFMODULE');

--
-- Contraintes pour les tables déchargées
--

--
-- Contraintes pour la table `AFFECTER`
--
ALTER TABLE `AFFECTER`
  ADD CONSTRAINT `FK_AFFECTER` FOREIGN KEY ('LOGINETU') REFERENCES `ETUDIANT` ('LOGINETU'),
  ADD CONSTRAINT `FK_AFFECTER2` FOREIGN KEY ('INTITULEGROUPE', 'ANNEEGROUPE') REFERENCES `GROUPE` ('INTITULEGROUPE', 'ANNEEGROUPE');

--
-- Contraintes pour la table `CONVERSATION_ADMIN`
--
ALTER TABLE `CONVERSATION_ADMIN`
  ADD CONSTRAINT `FK_RELIER` FOREIGN KEY ('LOGINPROF') REFERENCES `PROFESSEUR` ('LOGINPROF');

--
-- Contraintes pour la table `DEVOIR`
--
ALTER TABLE `DEVOIR`
  ADD CONSTRAINT `FK_EVALUER` FOREIGN KEY ('INTITULEGROUPE', 'ANNEEGROUPE') REFERENCES `GROUPE` ('INTITULEGROUPE', 'ANNEEGROUPE'),
  ADD CONSTRAINT `FK_EVALUER_MODULE` FOREIGN KEY ('REFMODULE') REFERENCES `MODULE` ('REFMODULE');

--
-- Contraintes pour la table `ENSEIGNER`
--
ALTER TABLE `ENSEIGNER`
  ADD CONSTRAINT `FK_ENSEIGNER` FOREIGN KEY ('LOGINPROF') REFERENCES `PROFESSEUR` ('LOGINPROF'),
  ADD CONSTRAINT `FK_ENSEIGNER2` FOREIGN KEY ('REFMODULE') REFERENCES `MODULE` ('REFMODULE');

--

```

```

-- Contraintes pour la table 'MESSAGE_ADMIN'
--
ALTER TABLE 'MESSAGE_ADMIN'
ADD CONSTRAINT 'FK_CORRESPONDRE' FOREIGN KEY ('IDCONV') REFERENCES 'CONVERSATION_ADMIN' ('IDCONV'),
ADD CONSTRAINT 'FK_ENVOYER_MESSAGE_ADMIN' FOREIGN KEY ('PRO_LOGINPROF') REFERENCES 'PROFESSEUR' ('LOGINPROF'),
ADD CONSTRAINT 'FK_ENVOYER_MESSAGE_PROF' FOREIGN KEY ('ADM_LOGINADMIN') REFERENCES 'ADMIN' ('LOGINADMIN'),
ADD CONSTRAINT 'FK_RECEVOIR_MESSAGE_ADMIN' FOREIGN KEY ('LOGINADMIN') REFERENCES 'ADMIN' ('LOGINADMIN'),
ADD CONSTRAINT 'FK_RECEVOIR_MESSAGE_PROF' FOREIGN KEY ('LOGINPROF') REFERENCES 'PROFESSEUR' ('LOGINPROF');

--
-- Contraintes pour la table 'MESSAGE_MODULE'
--
ALTER TABLE 'MESSAGE_MODULE'
ADD CONSTRAINT 'FK_ENVOYER_MESSAGE_MODULE' FOREIGN KEY ('LOGINPROF') REFERENCES 'PROFESSEUR' ('LOGINPROF'),
ADD CONSTRAINT 'FK_RELATIF_A_MESSAGE' FOREIGN KEY ('REFMODULE') REFERENCES 'MODULE' ('REFMODULE');

--
-- Contraintes pour la table 'NOTER'
--
ALTER TABLE 'NOTER'
ADD CONSTRAINT 'FK_NOTER' FOREIGN KEY ('LOGINETU') REFERENCES 'ETUDIANT' ('LOGINETU'),
ADD CONSTRAINT 'FK_NOTER2' FOREIGN KEY ('INTITULEGROUPE', 'ANNEEGROUPE', 'REFMODULE', 'IDDEVOIR') REFERENCES 'DEVOIR'
('INTITULEGROUPE', 'ANNEEGROUPE', 'REFMODULE', 'IDDEVOIR');

--
-- Contraintes pour la table 'PARTICIPER'
--
ALTER TABLE 'PARTICIPER'
ADD CONSTRAINT 'FK_PARTICIPER' FOREIGN KEY ('INTITULEGROUPE', 'ANNEEGROUPE') REFERENCES 'GROUPE' ('INTITULEGROUPE',
'ANNEEGROUPE'),
ADD CONSTRAINT 'FK_PARTICIPER2' FOREIGN KEY ('REFMODULE') REFERENCES 'MODULE' ('REFMODULE');

--
-- Contraintes pour la table 'RECEVOIR'
--
ALTER TABLE 'RECEVOIR'
ADD CONSTRAINT 'FK_RECEVOIR' FOREIGN KEY ('IDMESSAGE') REFERENCES 'MESSAGE_MODULE' ('IDMESSAGE'),
ADD CONSTRAINT 'FK_RECEVOIR2' FOREIGN KEY ('INTITULEGROUPE', 'ANNEEGROUPE') REFERENCES 'GROUPE' ('INTITULEGROUPE',
'ANNEEGROUPE');

--
-- Contraintes pour la table 'REPONDRE'
--
ALTER TABLE 'REPONDRE'
ADD CONSTRAINT 'FK_REPONDRE' FOREIGN KEY ('LOGINETU') REFERENCES 'ETUDIANT' ('LOGINETU'),
ADD CONSTRAINT 'FK_REPONDRE2' FOREIGN KEY ('IDSONDAGE') REFERENCES 'SONDAGE' ('IDSONDAGE');

--
-- Contraintes pour la table 'SONDAGE'
--
ALTER TABLE 'SONDAGE'
ADD CONSTRAINT 'FK_ENVOYER_SONDAGE' FOREIGN KEY ('LOGINPROF') REFERENCES 'PROFESSEUR' ('LOGINPROF'),
ADD CONSTRAINT 'FK_RELATIF_A_SONDAGE' FOREIGN KEY ('REFMODULE') REFERENCES 'MODULE' ('REFMODULE');
COMMIT;

```