# Analyse et conception

1. Introduction :

Dans cette partie, on va analyser et modéliser les besoins du client avec le langage UML.

L’activité d’analyse et de conception permet de traduire les besoins fonctionnels et les contraintes issues du cahier des charges et de la spécification des exigences dans un langage plus professionnel et compréhensible par tous les individus intervenants dans la réalisation et l’utilisation de l’application.

2.UML :

1. صورة تحتوي على نص, قصاصة فنية

   تم إنشاء الوصف تلقائياًChoix d’UML

UML (*Unified Modeling Language*, que l'on peut traduire par "*langage de modélisation unifié*) est une notation permettant de modéliser un problème de façon standard. Ce langage est né de la fusion de plusieurs méthodes existantes auparavant, et est devenu désormais la référence en termes de modélisation objet, à un tel point que sa connaissance est souvent nécessaire pour obtenir un poste de développeur objet.

1. صورة تحتوي على نص, قصاصة فنية

   تم إنشاء الوصف تلقائياًPourquoi modéliser ?

De la même façon qu'il vaut mieux dessiner une maison avant de la construire, il vaut mieux modéliser   
un système avant de le réaliser.

Modéliser, c’est décrire de manière visuelle et graphique les besoins, les solutions fonctionnelles et techniques du projet.

Modéliser pour :   
• Obtenir une modélisation de très haut niveau indépendante des langages et des   
environnements.   
• Faire collaborer des participants de tous horizons autour d'un même document de   
synthèse.   
• Faire des simulations avant de construire un système.

3.Présentation des outils :

**EDraw Max** est une solution professionnelle qui permet de concevoir de nombreux modèles de diagrammes et graphiques utiles pour de nombreux besoins : gestion de projet, architecture, développement ou encore dans l'éducation. Mention spéciale aux nombreux modèles disponibles qui couvrent de nombreux besoins.

4.Identification des acteurs :

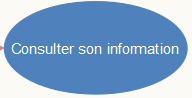
1. Acteurs

Un acteur est l'idéalisation d'un rôle joué par une personne externe, un processus ou une chose qui interagit avec un système.   
Il se représente par un petit bonhomme (figure 1) avec son nom (son rôle) inscrit dessous.

**Figure 1 : Exemple de représentation d’un acteur**

1. Cas d’utilisation

Un cas d’utilisation est un service rendu à un acteur : c’est une fonctionnalité de son point de vue.   
Un cas d'utilisation se représente par une ellipse (figure 2) contenant le nom du cas (un verbe à l'infinitif), et optionnellement, au-dessus du nom, un stéréotype.



**Figure 2 : Exemple de représentation d’un cas d’utilisation**

c. Acteur direct

Les acteurs directs, ce sont les utilisateurs de l’application, qui touchent directement l’application.

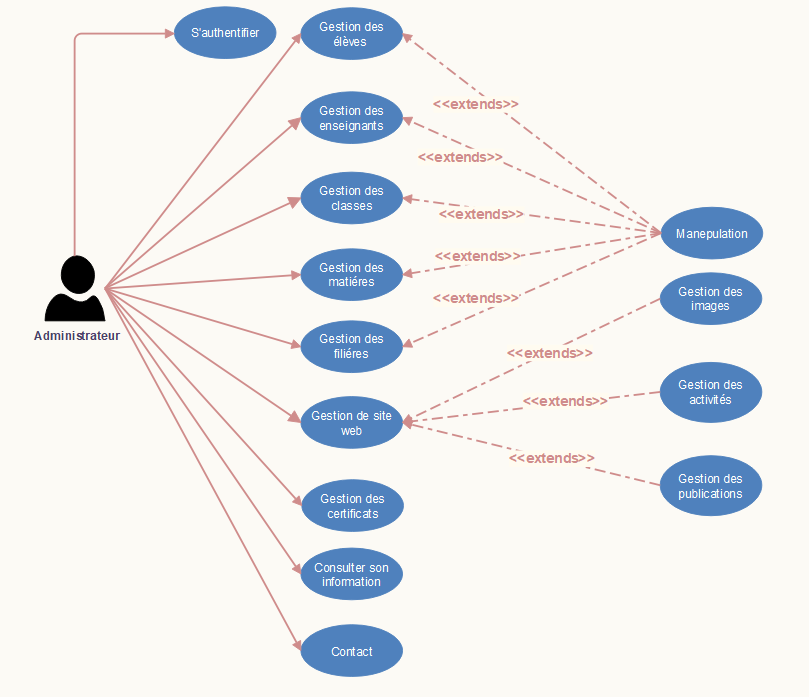
|  |  |
| --- | --- |
| Type d’acteur | Description fonctionnelle |
|  | L’acteur le plus important, qui aura les fonctionnalités suivantes :   * Consulter ses leçons. * Demander certificat scolaire. * Etc… |
|  | A pour fonctionnalités :   * Manipuler des leçons. * Consulter ses informations. * Etc… |
|  | Ce type d’acteur aura les fonctionnalités suivantes :   * Gestion des élèves. * Gestion des certificats. * Gestion de site web. * Etc.… |

**Tableau 1 : Acteurs primaires**

5.Diagramme de cas d’utilisation :

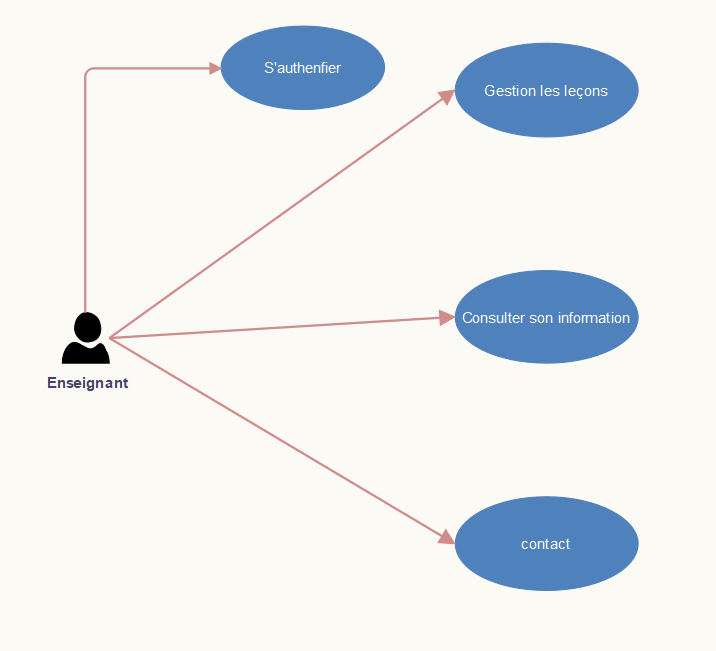
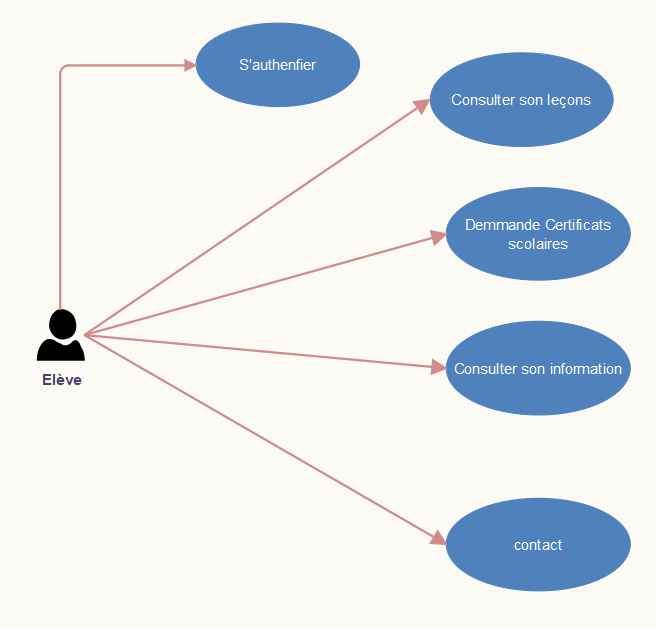
Le diagramme Use-Case, appelé diagramme de cas d’utilisation en français, fait partie des diagrammes de comportement du langage [Unified Modelling Language](https://www.ionos.fr/digitalguide/sites-internet/developpement-web/uml-un-langage-de-modelisation-pour-la-programmation-orientee-objet/), UML en abrégé, avec les systèmes et processus de programmation objet ou encore les processus métier. UML n’est donc pas un langage de programmation, mais un langage de modélisation. C’est une méthode standardisée qui décrit un système en cours de conception, ou déjà existant. Cela se fait à l’aide de diagrammes, dans lesquels tous les objets impliqués sont structurés et liés les uns aux autres.

1. Diagramme de cas d’utilisation d’acteur <<Administrateur>>



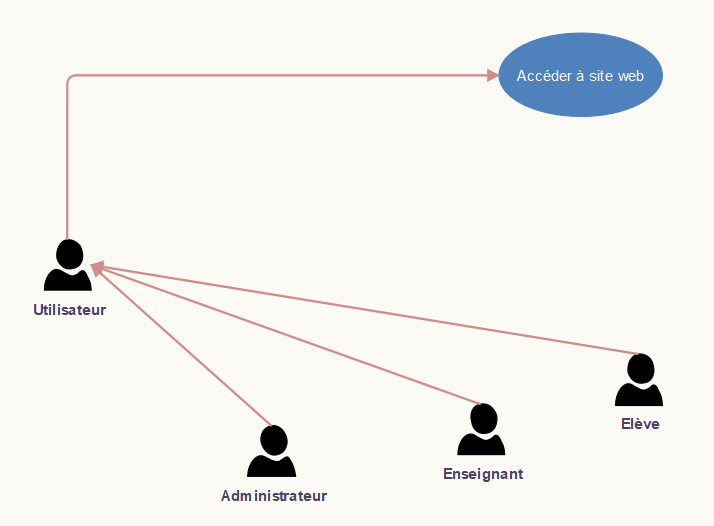
**Figure 3 : Diagramme de cas d’utilisation d’acteur <<Administrateur>>**

1. Diagramme de cas d’utilisation d’acteur <<Elève>>

**  
 Figure 4 : Diagramme de cas d’utilisation d’acteur <<Elève>>**c. Diagramme de cas d’utilisation d’acteur <<Enseignant>>

**Figure 5 : Diagramme de cas d’utilisation d’acteur <<Enseignant>>**

d. Diagramme de cas d’utilisation <<Héritage>>



**Figure 6 : Diagramme de cas d’utilisation <<Héritage>>**

6.Diagramme de séquence :

Les diagrammes de séquence sont une solution populaire de modélisation dynamique en langage UML, car ils se concentrent plus précisément sur les lignes de vie, les processus et les objets qui vivent simultanément, et les messages qu'ils échangent entre eux pour exercer une fonction avant la fin de la ligne de vie.

* Fiche de description << Authentification >> :

**Sommaire d’identification :**

• **Titre :** Authentification.

• **Résume** : entrer un mail et mot de passe pour accéder à l’application.

• **Acteurs** : Administrateur – système.

**Description des enchainements :**

• **Enchainements :**

* **Scénario :**

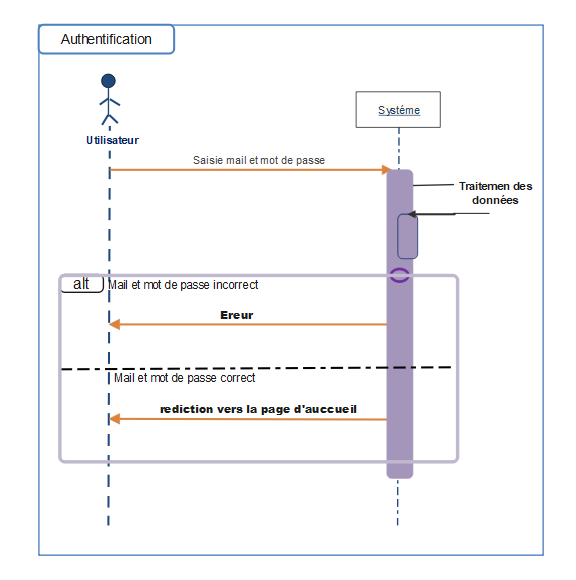
1. Saisie le login et mot de passe.

2. Envoyer login et mot de passe.

3. Traitement des informations envoyées.

4. En cas d’erreur, l’authentification est rejetée.

5. Si non, redirection d’utilisateur vers ses rôles

 **Tableau 2: Description textuelle d’authentification**

**Figure 7 : Diagramme de séquence d’authentification**

* Fiche de description << Gestion des données >> :

**Sommaire d’identification :**

• **Titre** : Gestion des données.

• **Résume** : Permettre de manipuler toutes les informations concernant un élève

• **Acteurs** : administrateur – système.

**Description des enchainements :**

• **Pré conditions :**

* Authentification.

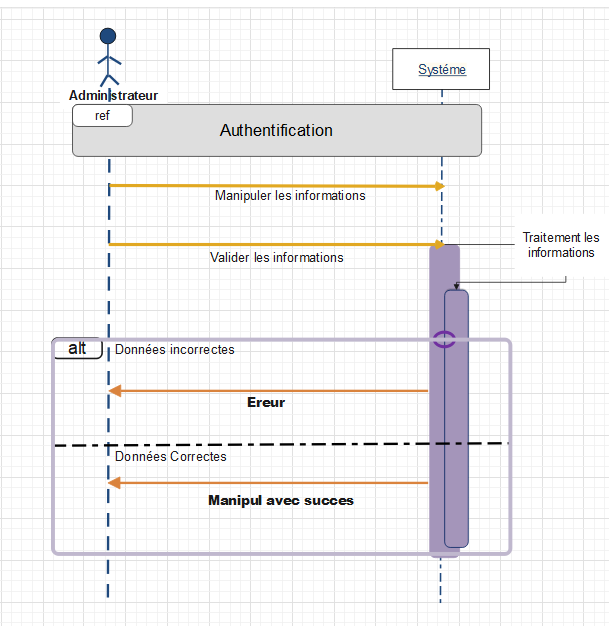
• **Enchainements :**

* **Scénario :**

1. manipuler les informations.

2. validation les informations

3. Traitement les informations

 **Tableau 3 : Description textuelle de gestion des données**

**Figure 8 : Diagramme de séquence de Gestion des données**

• Fiche de description << Gestion des images de site web >> :

**Sommaire d’identification :**

• **Titre** : Gestion des images de site web.

• **Résume** : Permettre d’ajouter des images

• **Acteurs** : administrateur – système.

**Description des enchainements :**

**• Pré conditions :**

**.** Authentification.

• **Enchainements :**

* **Scénario :**

1. manipuler les images et les informations.

2. Validation de manipulation.

3. Traitement des informations envoyé.

صورة تحتوي على نص, داخلي, لقطة شاشة

تم إنشاء الوصف تلقائياً **Tableau 4 : Description textuelle de gestion des images de site web**

**Figure 9 : Diagramme de séquence de Gestion des images de site web**

• Fiche de description << Demande des certificats >> :

**Sommaire d’identification :**

• **Titre :** Demander le certificat scolaire

• **Résume** : ce cas d’utilisation permet de demander le certificat scolaire pour un élève.

• **Acteurs** : élève – système.

**Description des enchainements** :

• **Pré conditions :**

1. Authentification.

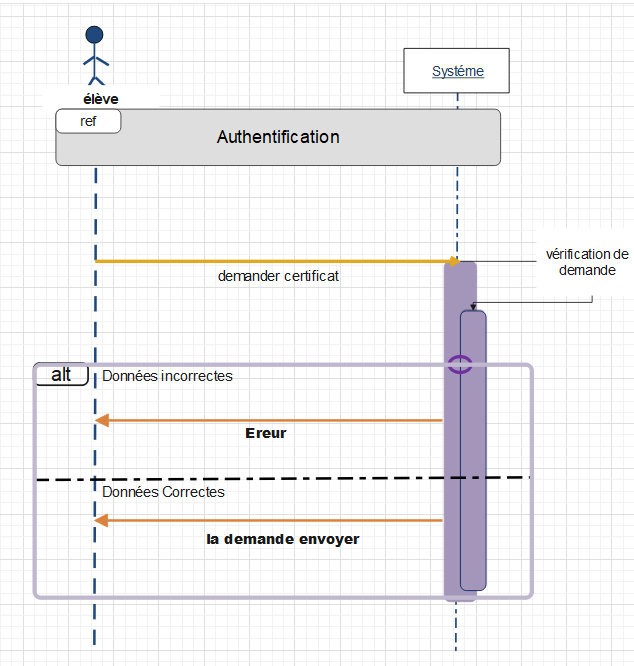
• **Enchainements :**

* **Scénario :**

1. l’élève demander un certificat scolaire

2. vérification de demande

3. envoyer la demande

 **Tableau 5 : Description textuelle de demande des certificats**

**Figure 10 : Diagramme de séquence de demande de certificat**

• Fiche de description << Accéder aux leçons >> :

**Sommaire d’identification :**

• **Titre :** Accéder aux leçons

• **Résume** : ce cas d’utilisation permet d’accéder aux leçons.

• **Acteurs** : élève – système.

**Description des enchainements** :

• **Pré conditions :**

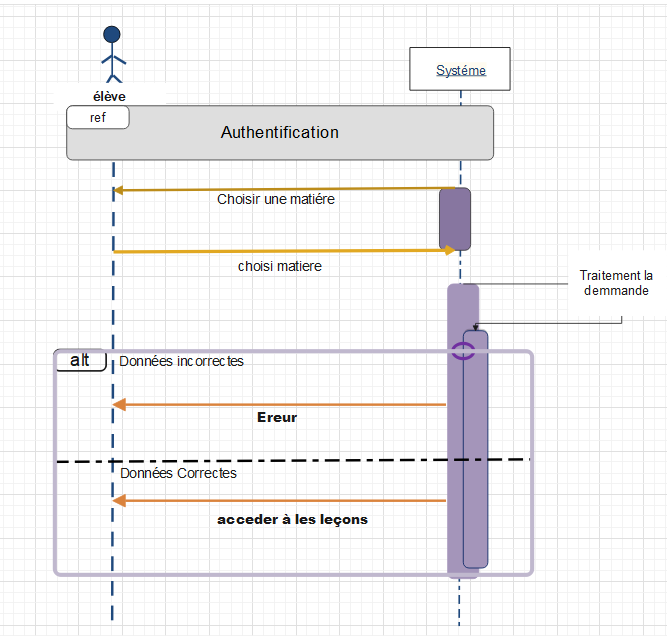
1.Authentification.

• **Enchainements :**

* **Scénario :**

1. choisi la matière

2. accéder pour regarder les fichiers des leçons.

 **Tableau 6 : Description textuelle d’Accéder à leçons**

**Figure 11 : Diagramme de séquence d’accéder aux leçons**

• Fiche de description << gestion des leçons >> :

**Sommaire d’identification :**

• **Titre :** gestion des leçons

• **Résume** : ce cas d’utilisation permet de gérer les leçons.

• **Acteurs** : enseignant – système.

**Description des enchainements** :

• **Pré conditions :**

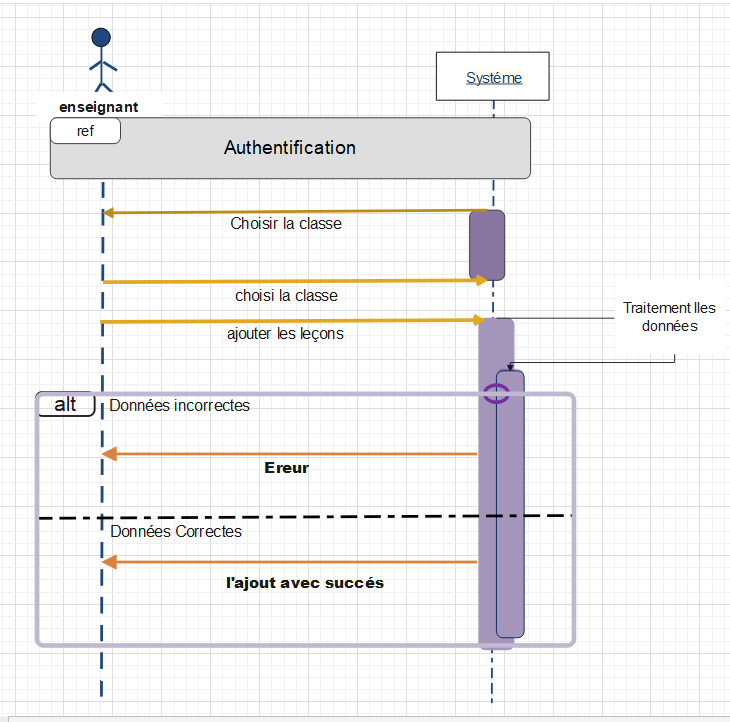
1.Authentification.

• **Enchainements :**

* **Scénario :**

1. choisi la classe

2. ajouter les fichiers des leçons.

 **Tableau 7 : Description textuelle de gestion les leçons**

**Figure 12 : Diagramme de séquence de gestion les leçons**

• Fiche de description << contact >> :

**Sommaire d’identification :**

• **Titre :** contacts.

• **Résume :** Ce cas d’utilisation permet d’envoyer et recevoir les messages

• **Acteurs :** utilisateur, destinataire.

**Description des enchainements :**

• **Pré conditions :**

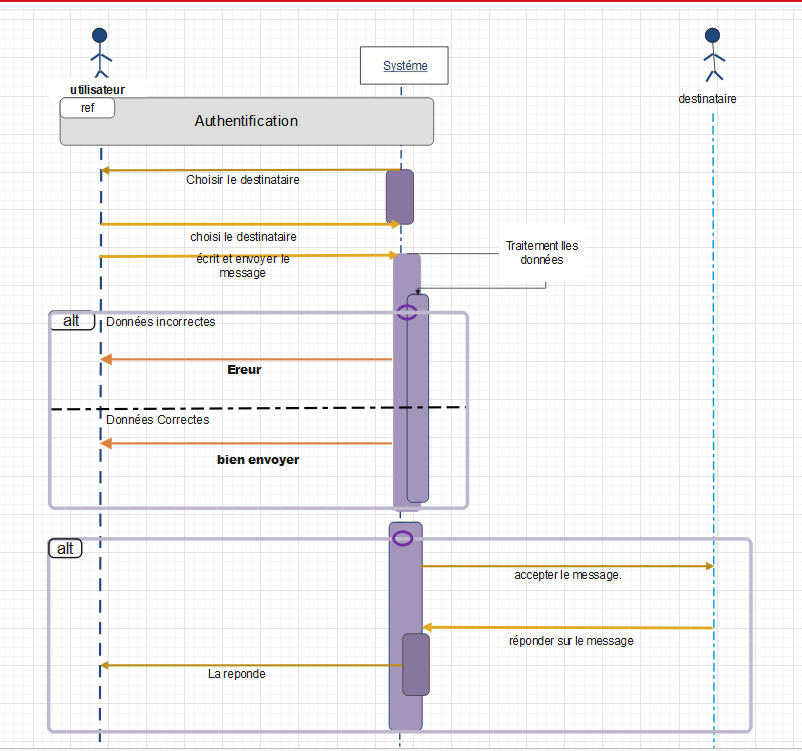
1. Authentification.

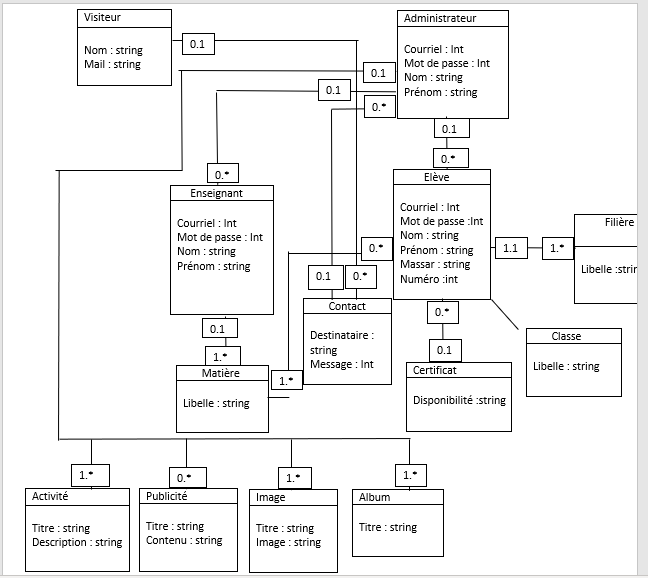
• **Enchainements :**

* **Scénario :**

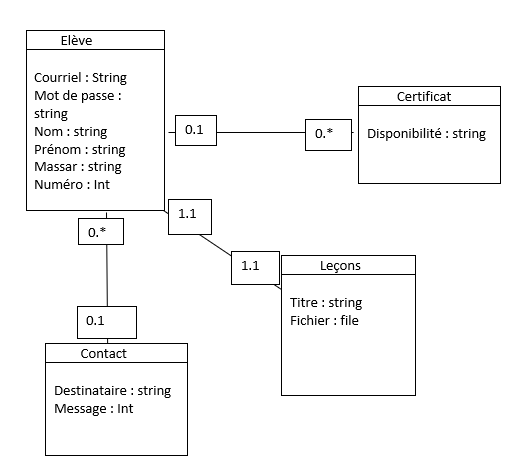
1. L’utilisateur choisi le destinataire.
2. L’utilisateur écrit le message.
3. L’utilisateur envoyer le message.
4. Le destinataire accepter le message.
5. Le destinataire répond

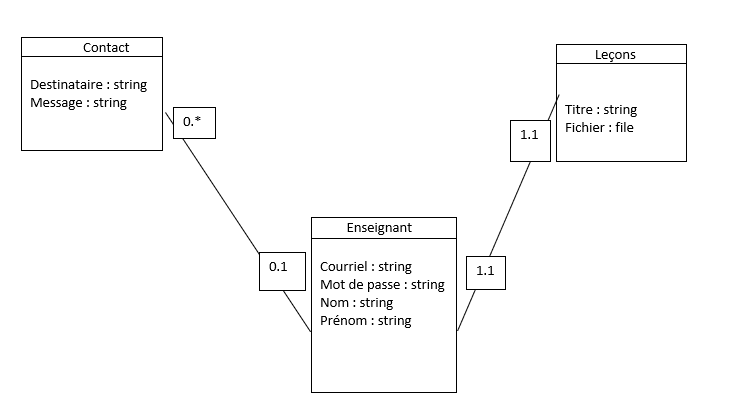
**Tableau 8 : Description textuelle de contact**

**Figure 13 : Diagramme de séquence de contact** 

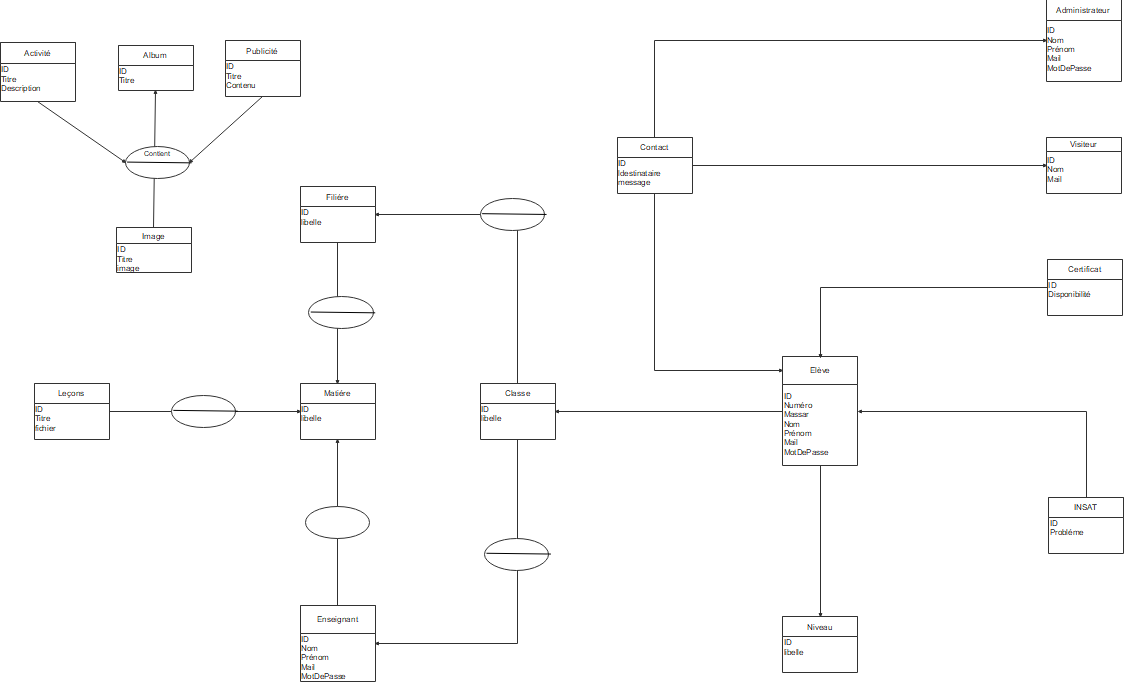
7.Diagramme de classe :

**Figure 14 : Diagramme de classe de gestion les données**

**Figure 15 : Diagramme de classe d’élève** 



**Figure 16 : Diagramme de classe d’enseignant**

8.Merise : a. MCD

0.\*

a

0.1

0.\*

1.\*

1.\*

0.1

0.\*

a

0.1

0.1

1.\*

a

0.1

1.\*

a

a

**Figure 17 : Modèle conceptuel de données**

0.\*

0.\*

1.1

1.1

1.1

1.1

1.1

1.1

1.\*

0.1

1.\*

0.1

0.\*

1.\*

1.\*

1.\*

a

a

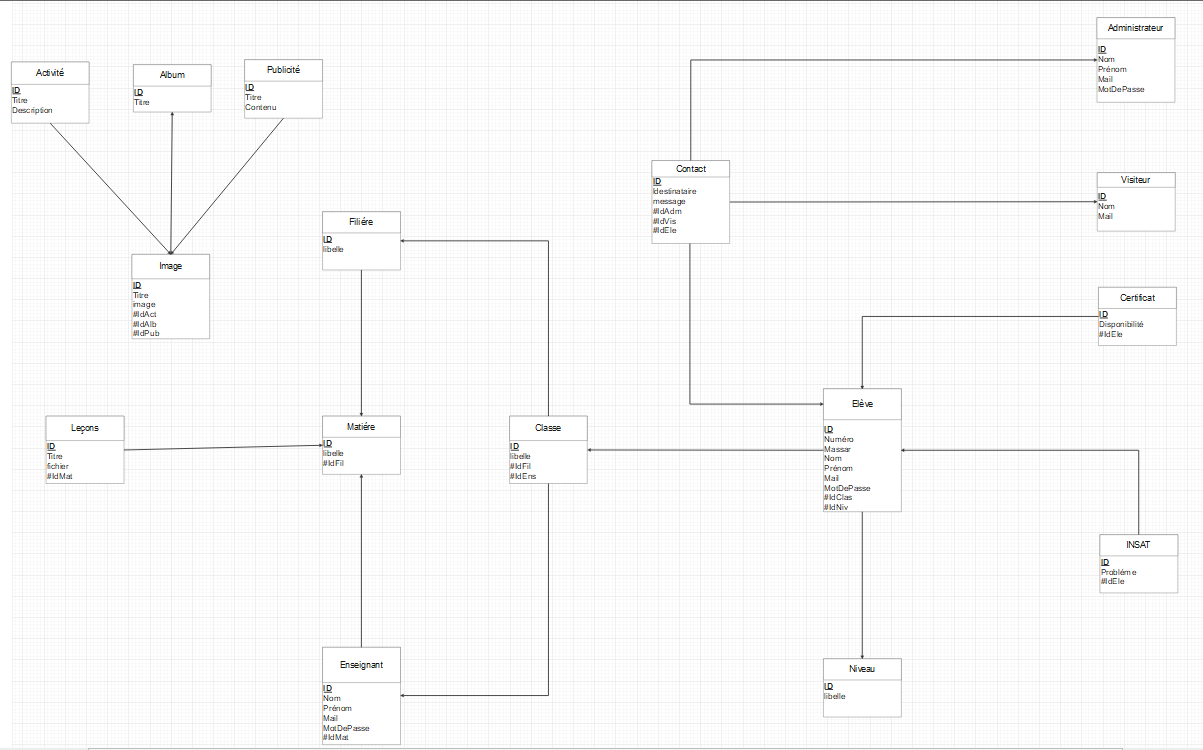
a

a

a

a

a

**b.MLD :**

**Figure 17 : Modèle Logique de données**