***C:\Users\HP\Desktop\conception\logo.jpg***

**Mini Projet :**

Application de gestion d’un établissement d’enseignant supérieur

***Réalisé par :***

**Lajili Hela**

**Chaieb Sirine**

**Soussi Oussama**

**Moussa Sahar**

**Lajjem Siwar**

***2 LAIG***

Table des matières

[1 Introduction sur le projet : 3](#_Toc7645694)

[2 Sprint 1 : 4](#_Toc7645695)

[2.1 Spécification des besoins : 4](#_Toc7645696)

[**2.1.1** **Diagramme de cas d’utilisation :** 4](#_Toc7645697)

[**2.1.2** **Diagramme de séquence système :** 9](#_Toc7645698)

[**2.1.3** **Marquette d’interfaces :** 13](#_Toc7645699)

[2.2 Conception : 18](#_Toc7645700)

[**2.2.1** **Diagramme de classe :** 18](#_Toc7645701)

[2.3 Scrum : 19](#_Toc7645702)

[**2.3.1** **Sprint backlog :** 19](#_Toc7645703)

[**2.3.2** **Scrum board :** 19](#_Toc7645704)

[**2.3.3** **Burndown chart :** 21](#_Toc7645705)

[3 Sprint 2 : 22](#_Toc7645706)

[3.1 Spécification des besoins : 22](#_Toc7645707)

[**3.1.1** **Diagramme de cas d’utilisation :** 22](#_Toc7645708)

[**3.1.2** **Diagramme de séquence système :** 23](#_Toc7645709)

[**3.1.3** **Marquette d’interfaces :** 23](#_Toc7645710)

[3.2 Conception : 23](#_Toc7645711)

[**3.2.1** **Diagramme de classe :** 23](#_Toc7645712)

[3.3 Scrum : 23](#_Toc7645713)

[**3.3.1** **Sprint backlog :** 23](#_Toc7645714)

[**3.3.2** **Scrum board :** 24](#_Toc7645715)

[**3.3.3** **Burndown chart :** 26](#_Toc7645716)

[4 Réalisation : 26](#_Toc7645717)

[4.1 Langage de programmation : 26](#_Toc7645718)

[4.2 Méthodes et Logiciels : 27](#_Toc7645719)

[5 Conclusion : 29](#_Toc7645720)

# ***Introduction sur le projet :***

Devant l'établissement qui dépasse les capacités du point de vue effectif des étudiants et enseignants. Dans ce cas, le travail manuel de préparation de l’emploi du temps devient quasi-impossible et les responsables se trouvent obligés de faire fonctionner leurs établissements jusqu’aux heures tardives et aussi, à répartir les emplois du temps des enseignants sur plusieurs journées, ce qui les empêche de se concentrer sur leurs activités de recherche ou bien même de ramener des étudiants pour ne suivre qu’une seule Séance pendant toute une journée.

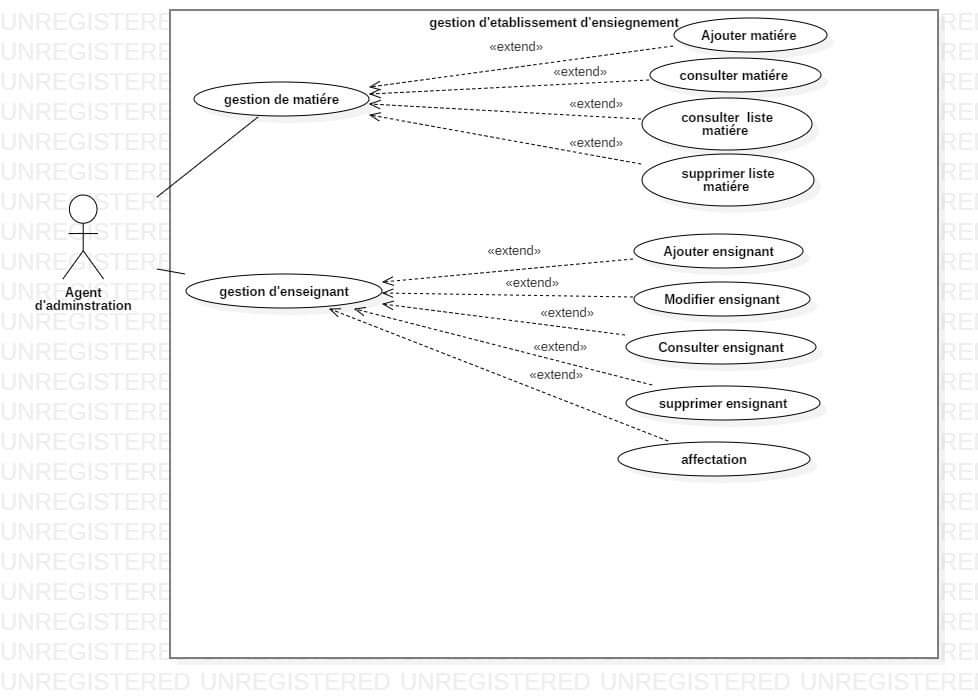
Dans ce cadre notre système permet de préparer et de gérer les emplois aussi bien des enseignants, que des étudiants et aussi avoir tous les états possibles concernant l’occupation des salles en prendront les bonnes décisions.

# ***Sprint 1 :***

## ***Spécification des besoins :***

Pour atteindre nos objectifs dans des meilleures conditions, nous devons spécifier les besoins de notre application.

### **Diagramme de cas d’utilisation :**



#### ** Description textuelle du cas d’utilisation « ajouter Enseignant »**

|  |  |
| --- | --- |
| **Cas d’utilisation** | **Ajouter Enseignant** |
| **Acteurs** | Agent administrative |
| **Précondition** | Authentification préalable |
| **Scénario nominal** | 1-l’agent demande la formulaire d’ajout  2-le système affiche le formulaire  3-l’agent saisit les informations correspondantes au nouvel enseignant.  4-le système vérifie les données saisies  5-le système enregistre le nouveau parcours et affiche un message de succès |
| **Scénario alternatif** | 4-a-l’agent saisie des données manquantes.  1-le système affiche un message d’erreur  2-retour à l’étape 3.  4-b-l’agent ajoute un enseignant existe déjà.  1-le système affiche un message d’erreur  2-retour à l’étape 3. |
| **Post condition** | L’ajout d’un nouvel enseignant est réussi avec sucées. |

#### ** Description textuelle du cas d’utilisation « Modifier Enseignant »**

|  |  |
| --- | --- |
| **Cas d’utilisation** | **Modifier Enseignant** |
| **Acteurs** | Agent administrative |
| **Précondition** | Authentification préalable |
| **Scénario nominal** | 1-l’agent demande le formulaire de modification  2-le système affiche le formulaire  3-l’agent modifie les informations et valide.  4-le système vérifie les données saisies.  5-le système enregistre les données et affiche un message de succès. |
| **Scénario alternatif** | 4-a-l’agent saisie des données manquantes  1-le système affiche un message d’erreur  2-reprise de l’étape 3 du scénario nominal  4-b- L’agent réécrit les mêmes coordonnées d’un enseignant déjà existant.  1-le système demande à l’administrateur de modifier les informations.  2-reprise de l’étape 3 du scénario nominal |
| **Post condition** | La modification de l’enseignant déjà inscrit est réussie avec succès. |

#### ** Description textuelle du cas d’utilisation « Supprimer Enseignant »**

|  |  |
| --- | --- |
| **Cas d’utilisation** | **Supprimer Enseignant** |
| **Acteurs** | Agent administrative |
| **Précondition** | Authentification préalable |
| **Scénario nominal** | 1-l’agent demande le formulaire de suppression.  2-le système affiche le formulaire.  3-l’agent choisi l’enseignant à supprimer et valide son choix.  4-le système affiche un message de confirmation.  5-le système supprime l’enseignant et affiche un message de succès. |
| **Scénario alternatif** | 3-a-l’agent annule son choix  1-le système annule la suppression |
| **Post condition** | L’enseignant a bien été supprimé. |

#### ** Description textuelle du cas d’utilisation « Consulter Enseignant »**

|  |  |
| --- | --- |
| **Cas d’utilisation** | **Consulter Enseignant** |
| **Acteurs** | Agent administrative |
| **Précondition** | Authentification préalable |
| **Scénario nominal** | 1-l’agent demande l’affichage de la liste des enseignant.  2-le système affiche la liste des enseignants. |
| **Scénario alternatif** | 3-a-l’agent annule son choix  1-le système annule la suppression |
| **Post condition** | 2-a-aucun résultat.  1-le système affiche un message de type « aucun résultat trouvé ». |

#### ** Description textuelle du cas d’utilisation « Ajouter Matières »**

|  |  |
| --- | --- |
| **Cas d’utilisation** | **Ajouter Matières** |
| **Acteurs** | Agent administrative |
| **Précondition** | Authentification préalable |
| **Scénario nominal** | 1. L’agent saisit les informations correspondant à la nouvelle matière et confirme sa demande d’ajout. 2. Le système ajoute la nouvelle matière et envoie un message indiquant le succès de l’opération. |
| **Scénario alternatif** | 4-a-l’agent saisie des données manquantes.  1-le système affiche un message d’erreur  2-retour à l’étape 3.  4-b-l’agent ajoute une matière existe déjà.  1-le système affiche un message d’erreur  2-retour à l’étape 3. |
| **Post condition** | L’ajout d’une nouvelle matière est réussi. |

#### ** Description textuelle du cas d’utilisation « Modifier Matières »**

|  |  |
| --- | --- |
| **Cas d’utilisation** | **Modifier Matières** |
| **Acteurs** | Agent administrative |
| **Précondition** | Authentification préalable |
| **Scénario nominal** | 1-l’agent demande le formulaire de modification  2-le système affiche le formulaire  3-l’agent saisit les informations correspondant à la matière choisie et confirme sa demande de modification.  4-le système vérifie les données saisies.  5-le système enregistre les données et affiche un message de succès. |
| **Scénario alternatif** | 4-a-l’agent saisie des données manquantes  1-le système affiche un message d’erreur  2-reprise de l’étape 3 du scénario nominal  4-b- L’agent réécrit les mêmes coordonnées d’une matière déjà existant.  1-le système demande à l’agent de modifier les informations.  2-reprise de l’étape 3 du scénario nominal |
| **Post condition** | La modification des matières est réussie. |

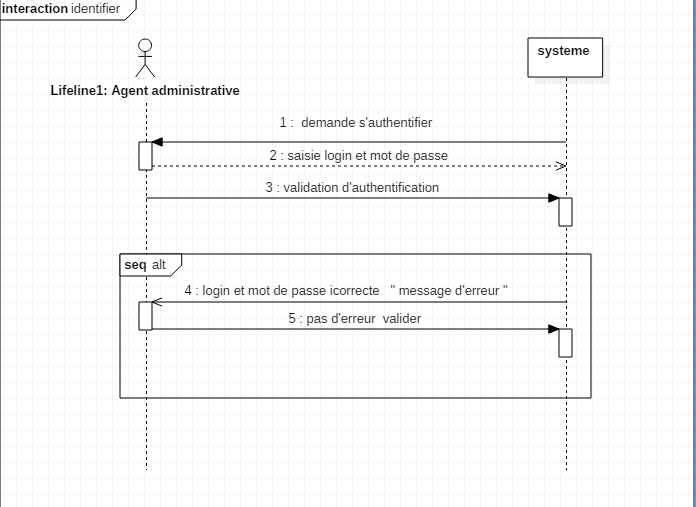
#### ** Description textuelle du cas d’utilisation « Supprimer Matières »**

|  |  |
| --- | --- |
| **Cas d’utilisation** | **Supprimer Matières** |
| **Acteurs** | Agent administrative |
| **Précondition** | Authentification préalable |
| **Scénario nominal** | 1-l’agent demande le formulaire de suppression.  2-le système affiche le formulaire.  3-l’agent choisi la matière à supprimer et valide son choix.  4-le système affiche un message de confirmation.  5-le système supprime la matière et affiche un message de succès. |
| **Scénario alternatif** | 3-a-l’agent annule son choix  1-le système annule la suppression |
| **Post condition** | La matière a bien été supprimée. |

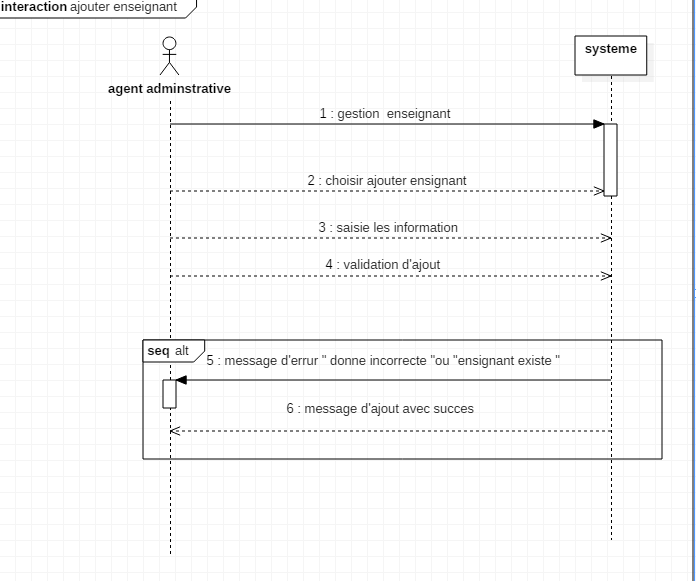
#### ** Description textuelle du cas d’utilisation « Consulter Matières »**

|  |  |
| --- | --- |
| **Cas d’utilisation** | **Consulter Matières** |
| **Acteurs** | Agent administrative |
| **Précondition** | Authentification préalable |
| **Scénario nominal** | 1-L’agent clique sur gestion matière puis consulter matières.  2-Le système affiche toute la liste des matières.  3-L’agent clique sur la matière choisie.  4-Le système affiche les informations de la matière. |
| **Scénario alternatif** | 4-a-aucun résultat.  1-le système affiche un message de type « aucun résultat trouvé ». |
| **Post condition** | Consultation de la liste des matières réussie. |

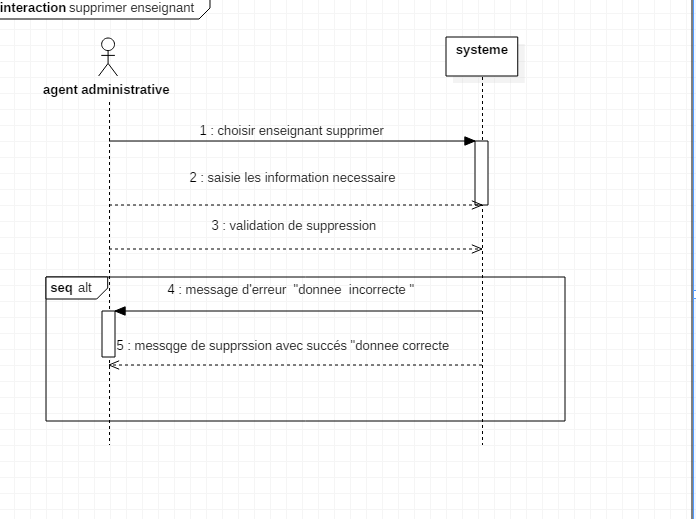
### **Diagramme de séquence système :**



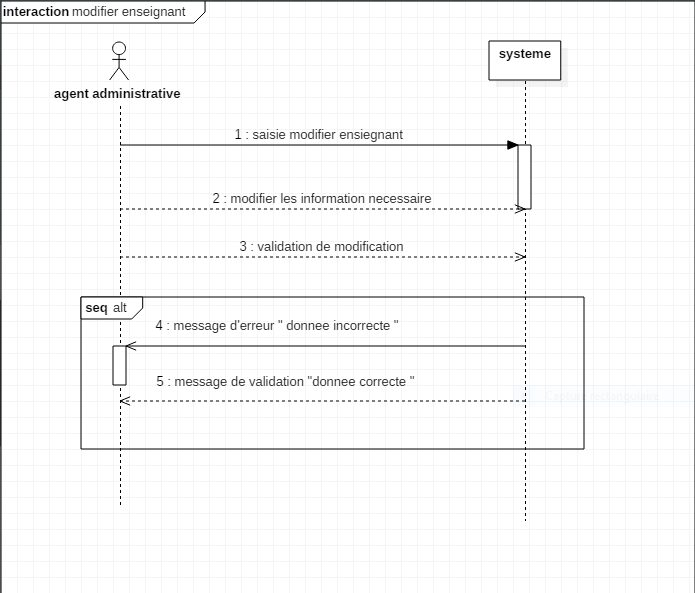
**Figure1 :** Diagramme de séquence du cas « authentifier »



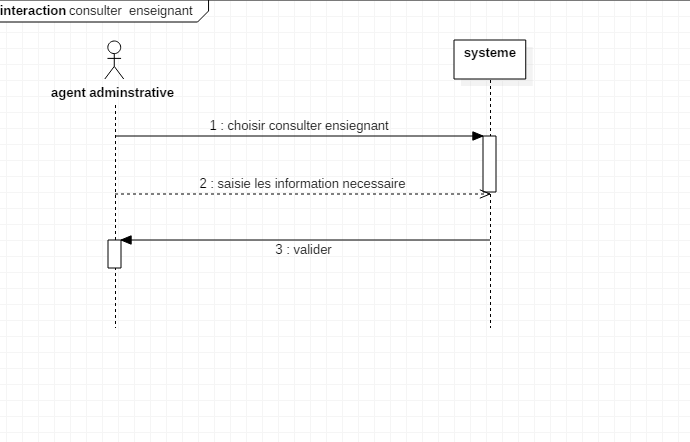
**Figure2 :** Diagramme de séquence du cas « ajouter enseignant »



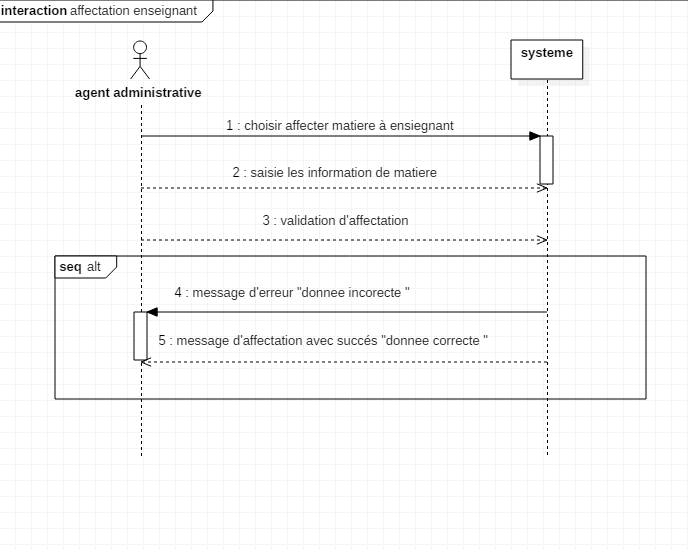
**Figure3 :** Diagramme de séquence du cas « supprimer enseignant »



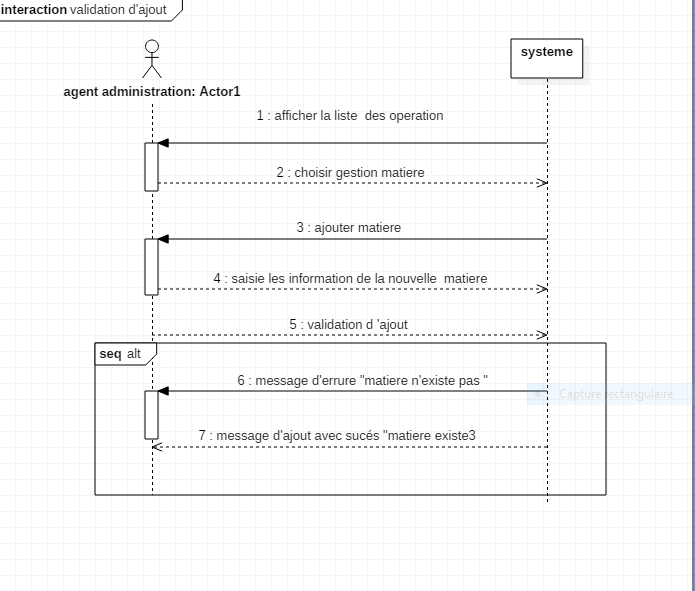
**Figure4 :** Diagramme de séquence du cas « modifier enseignant »



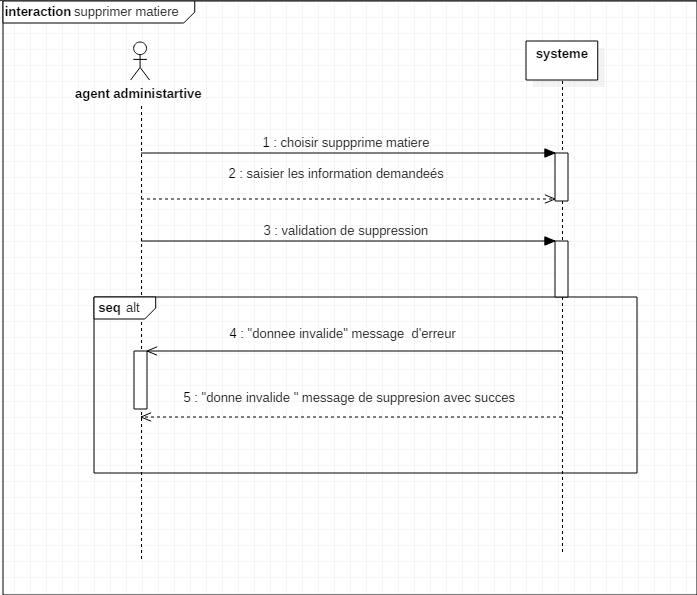
**Figure5 :** Diagramme de séquence du cas « consulter enseignant »



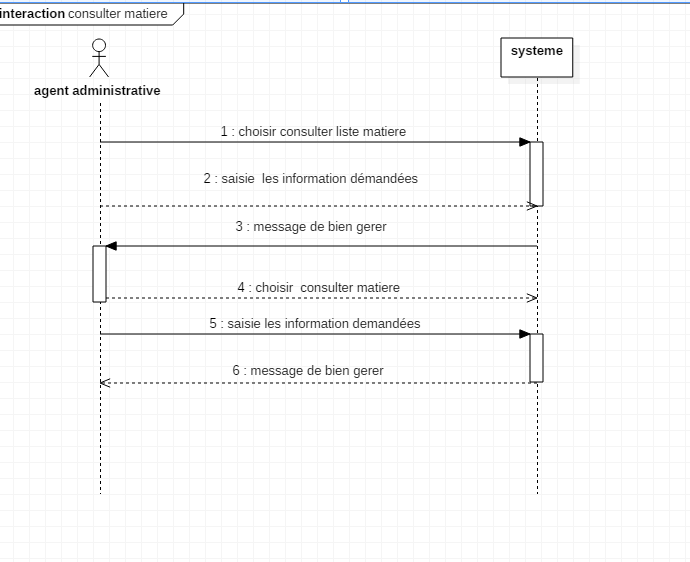
**Figure6 :** Diagramme de séquence du cas « ajouter matière »



**Figure7 :** Diagramme de séquence du cas « supprimer matière »

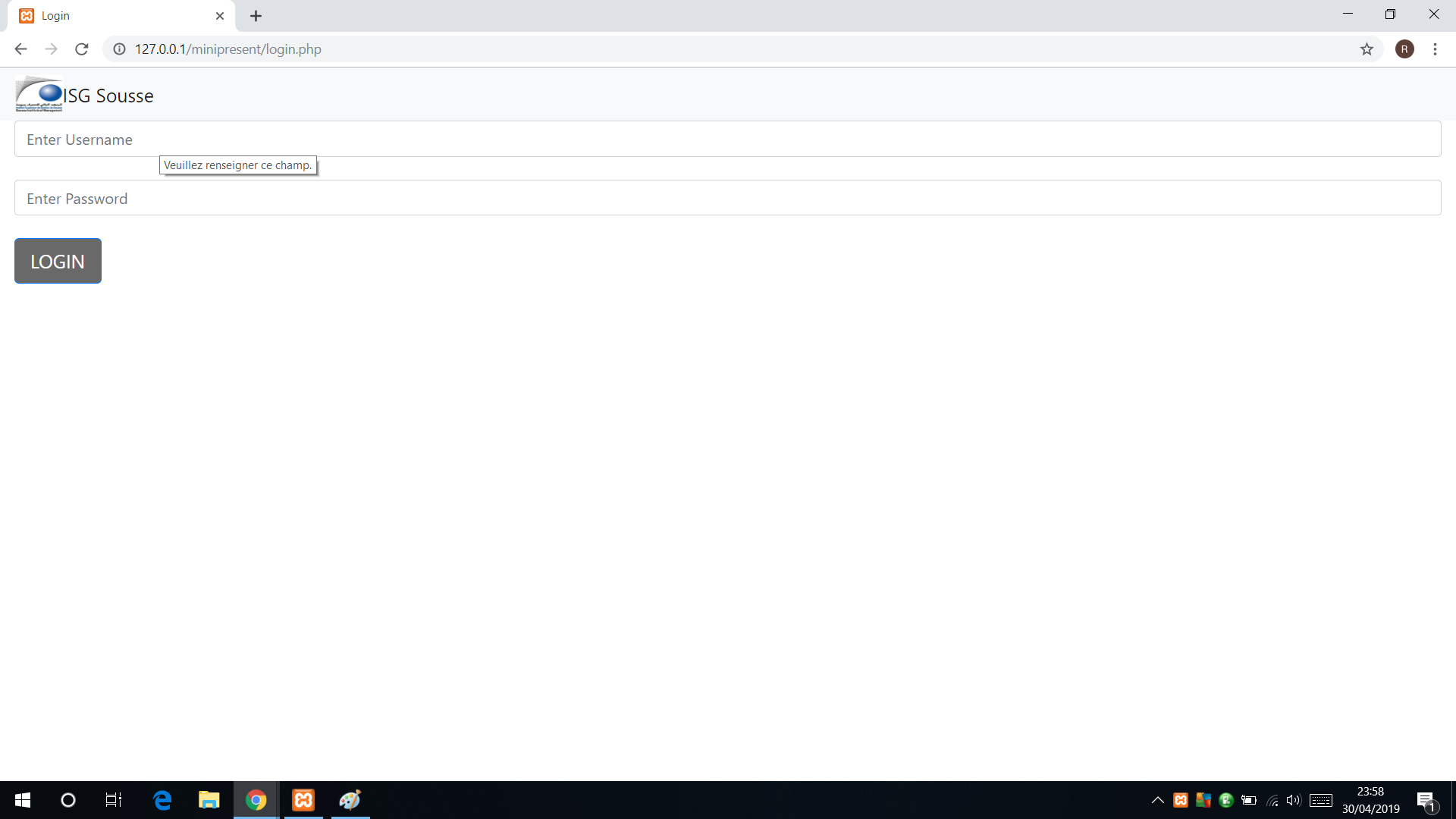


**Figure8 :** Diagramme de séquence du cas « consulter matière »



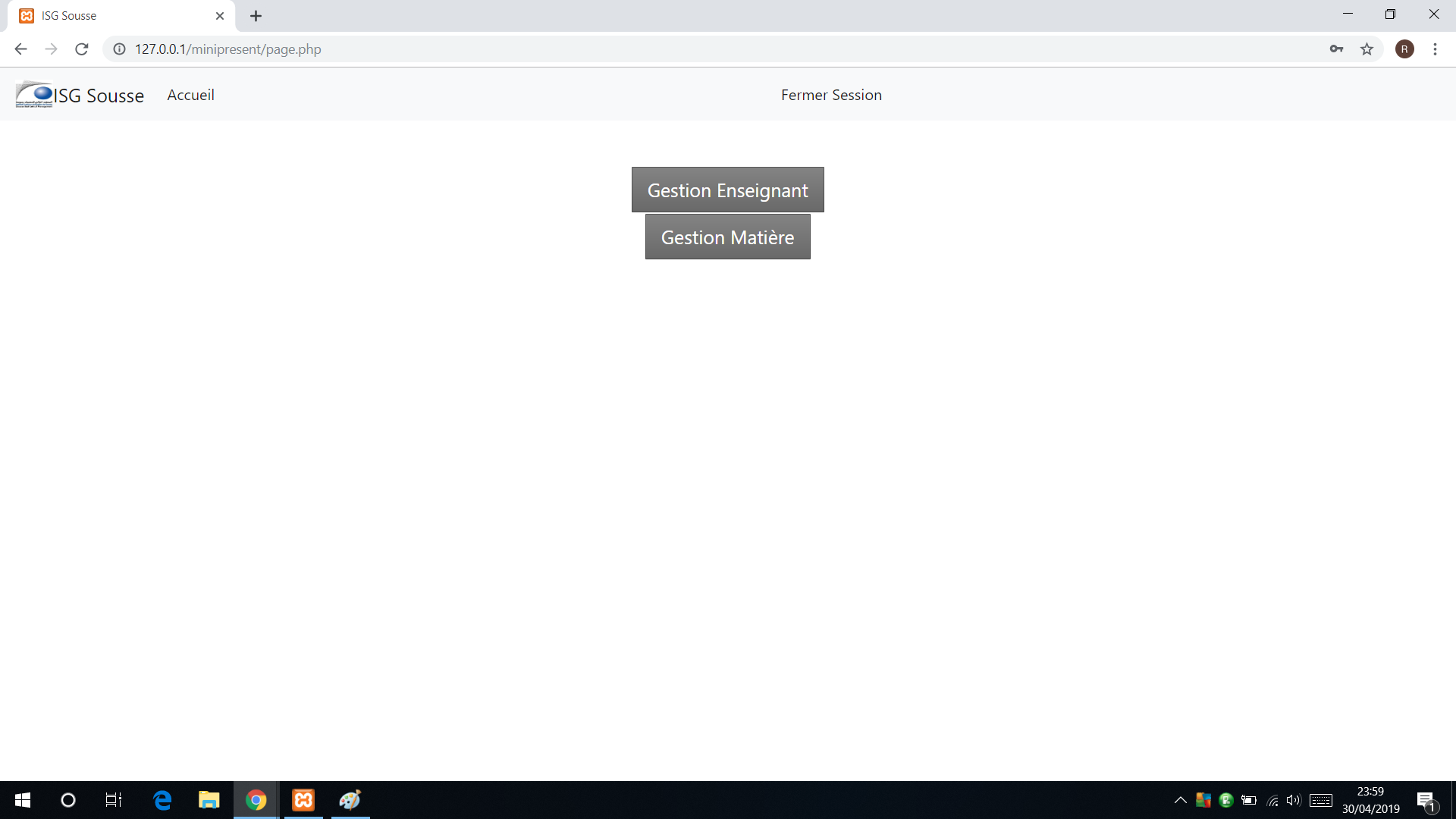
### **Marquette d’interfaces :**

***Figure 1 :*** présente l’interface Authentification de l’application. Cette interface est décomposée deux champs pour la saisie d’utilisateur et du mot de passe pour pouvoir accéder aux autres interfaces.



**Figure 1 : Login**

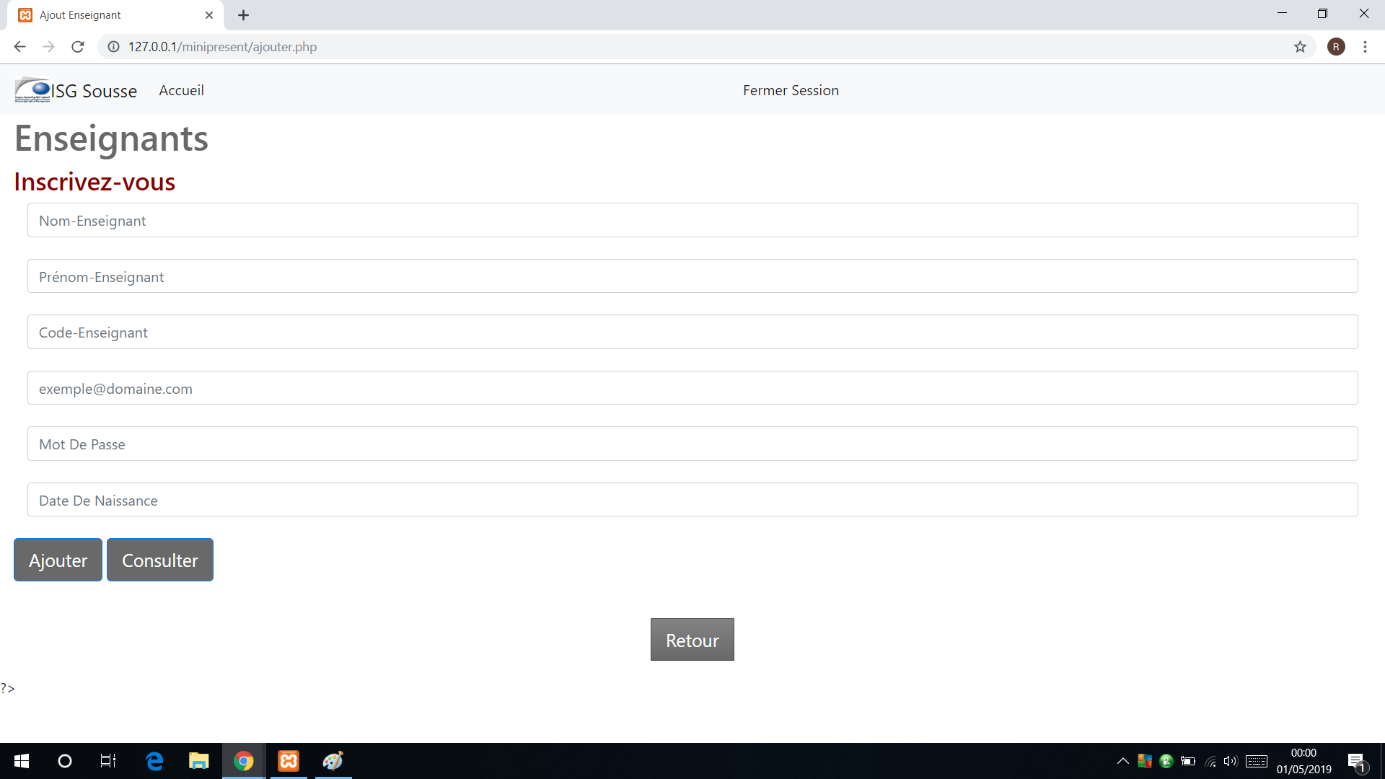
***Figure 2 :*** présente l’interface Page d’accueil de l’application. Cette interface permet de choisir de gestion d’enseignant ou de matière ou d’affecter une matière à un enseignant.



**Figure 2 : Page d’accueil**

**Figure 3 :**

***Figure 4 :*** présente le formulaire d’ajout d’un enseignant. L’administrateur remplit tous les champs puis enregistre pour que l’enseignant soit ajouté directement à la base de données.



**Figure 4 : Ajout d’un enseignant**

***Figure 5 :*** présente l’interface de suppression d’un enseignant. L’administrateur remplit les champs de l’enseignant qu’il désir supprimer puis supprimer pour que l’enseignant soit supprimé directement à la base de données.

**Figure 5 : suppression d’un enseignant**

**Figure 6 :**

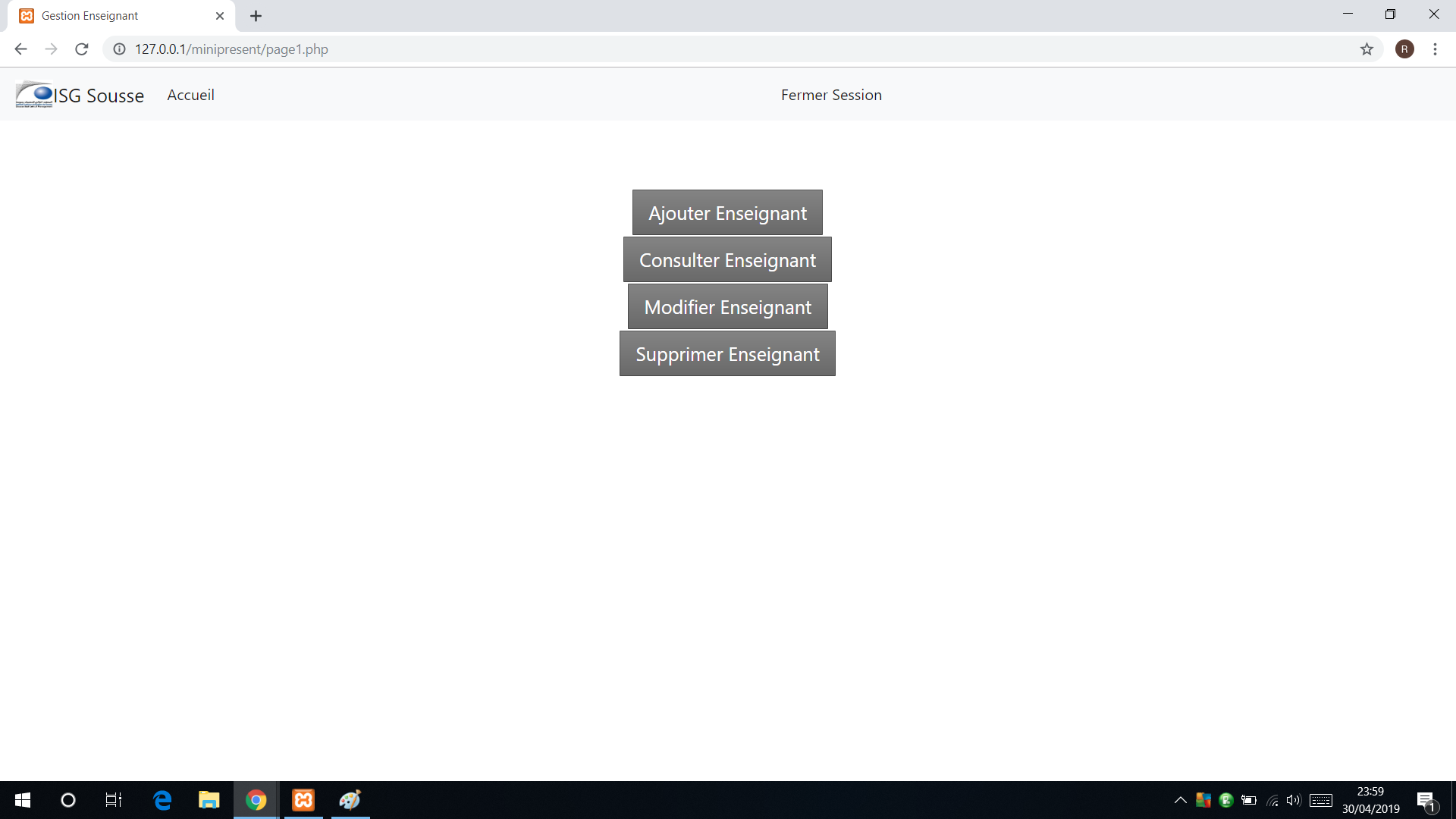
***Figure :*** présente le formulaire de modification d’un enseignant. L’administrateur modifie les champs qu’il désir puis enregistre pour que l’enseignant soit modifié directement à la base de données.

***Figure 7 :*** présente l’interface avec le formulaire d’ajout d’une Matière. L’administrateur remplit tous les champs puis enregistre pour que la matière soit ajoutée directement à la base de données.

**Figure 7 : Ajout d’une Matière**

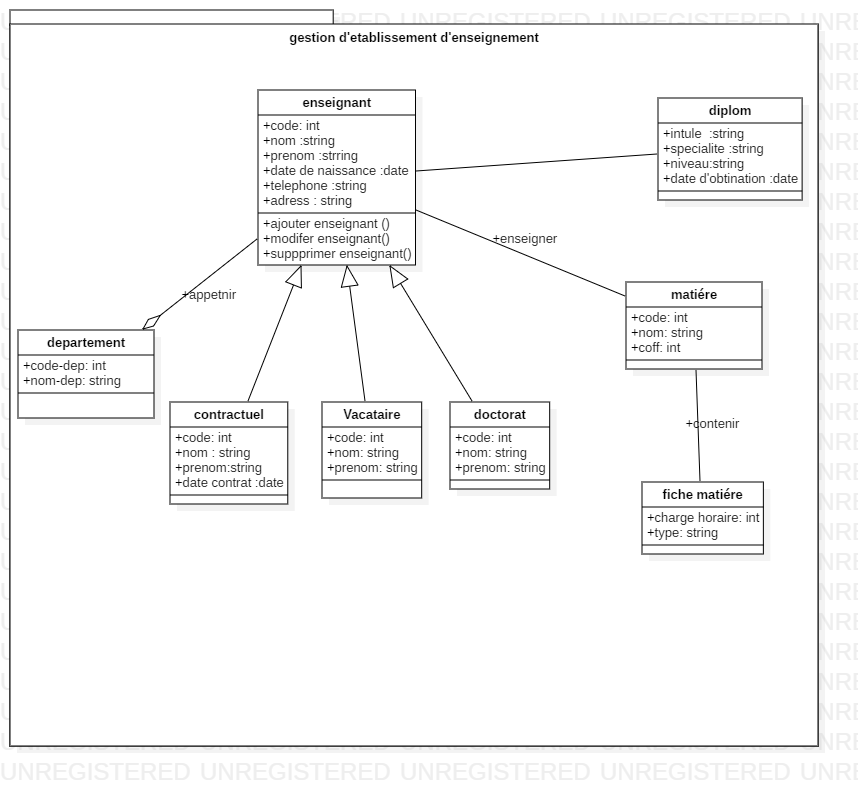
***Figure 8 :*** présente l’interface de suppression d’une Matière. L’administrateur remplit les champs de la matière qu’il désir supprimer puis supprimer pour que la matière soit supprimée directement à la base de données.

**Figure 8 : Suppression d’une Matière**



## ***Conception :***

### **Diagramme de classe :**



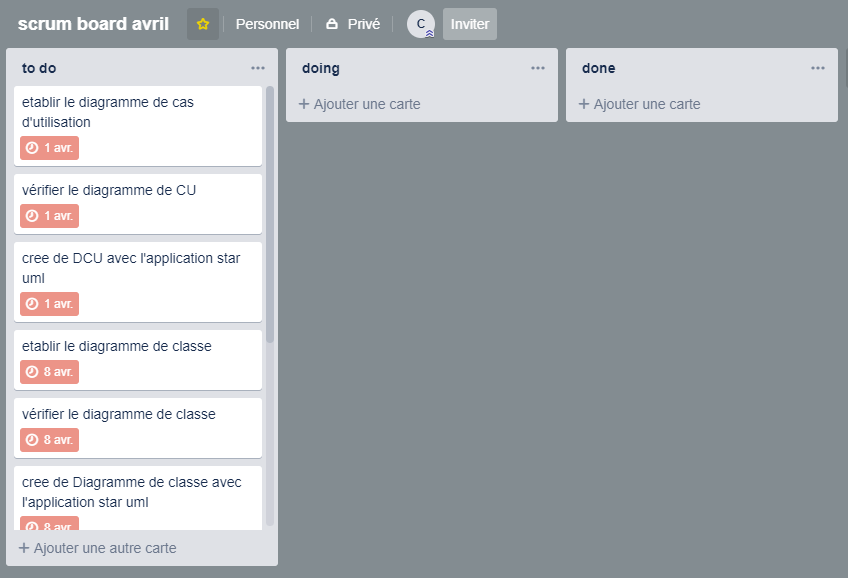
## ***Scrum :***

### **Sprint backlog :**

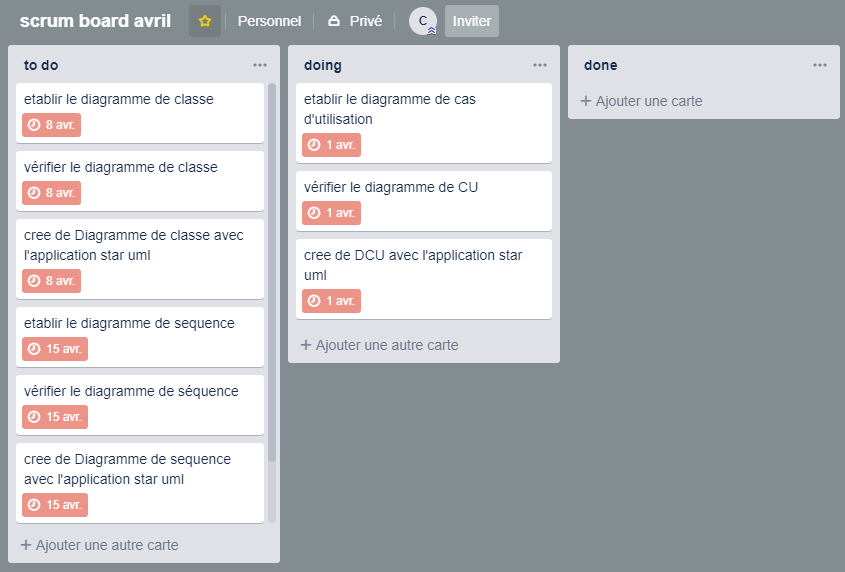


### **Scrum board :**

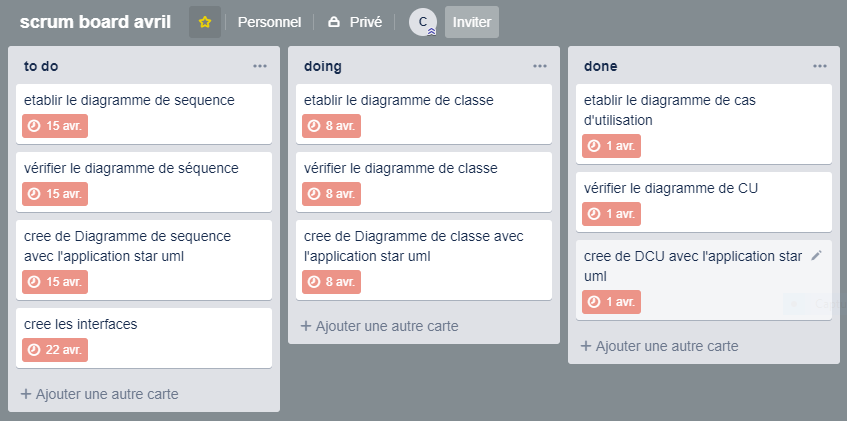
**Le 1 Avril :**



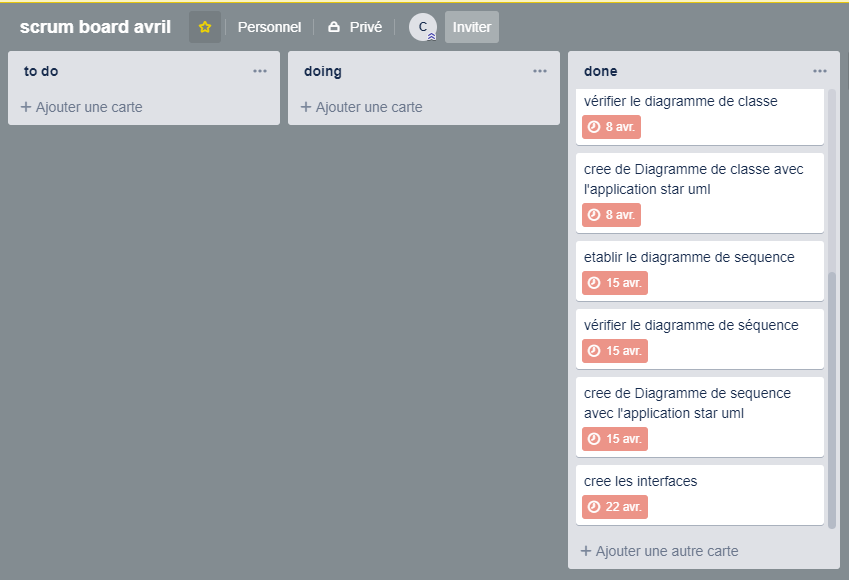
**Le 8 Avril :**

****

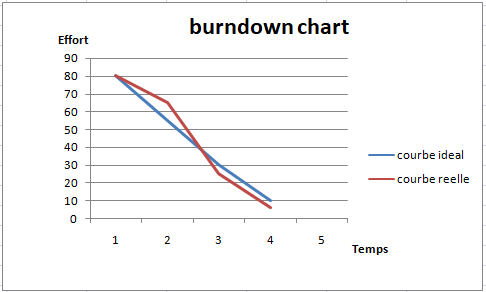
**Le 15 Avril :**

****

**Le 22 Avril :**

****

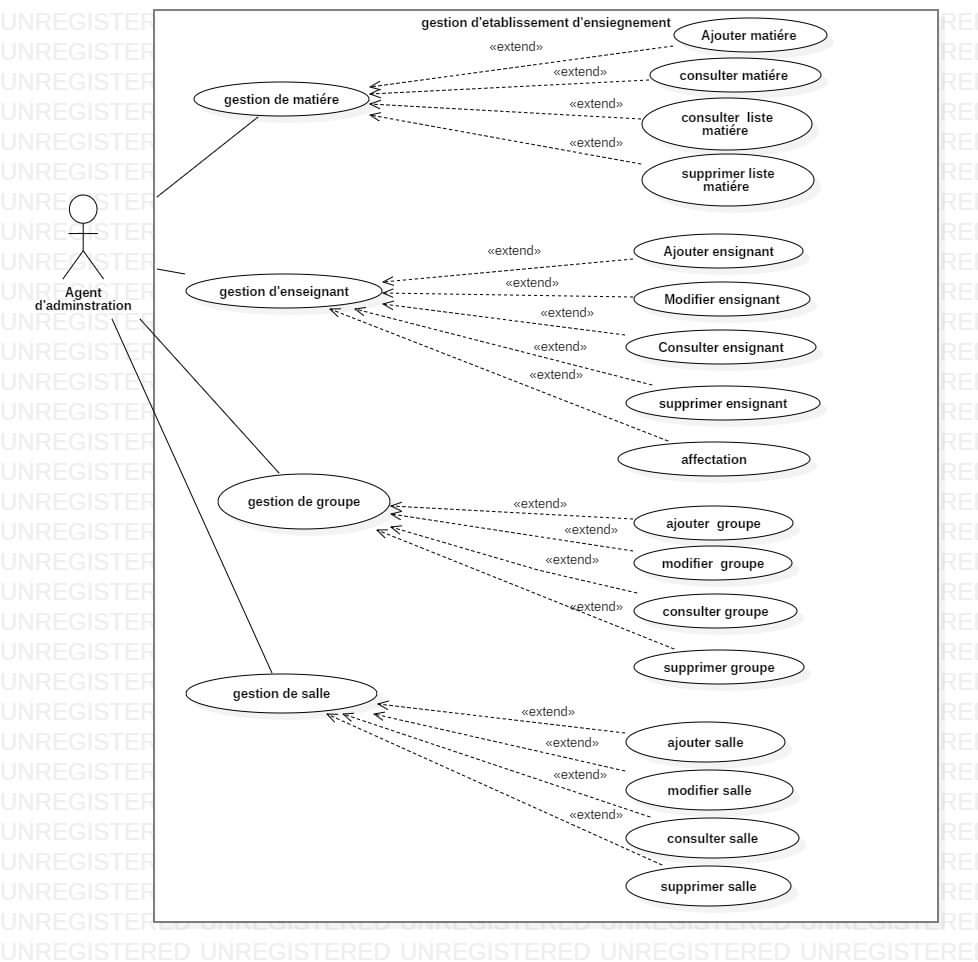
### **Burndown chart :**



# ***Sprint 2 :***

## ***Spécification des besoins :***

### **Diagramme de cas d’utilisation :**

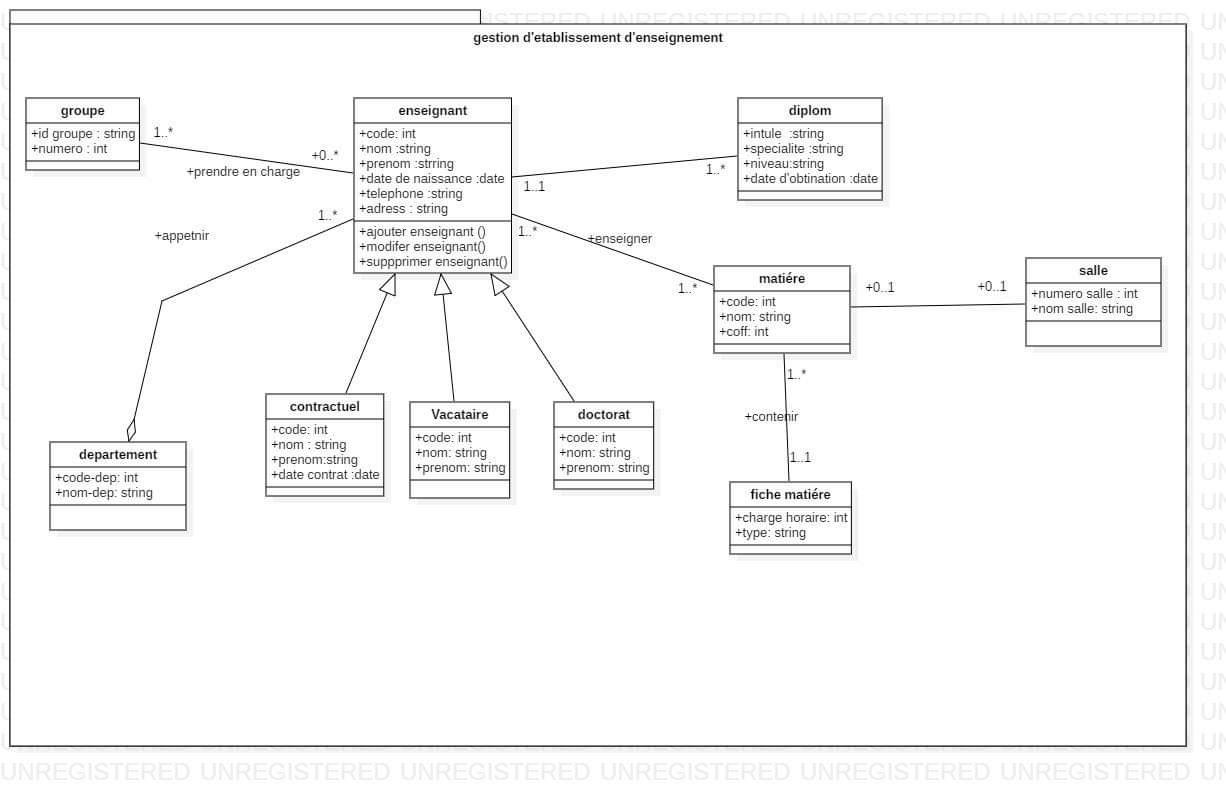


### **Diagramme de séquence système :**

### **Marquette d’interfaces :**

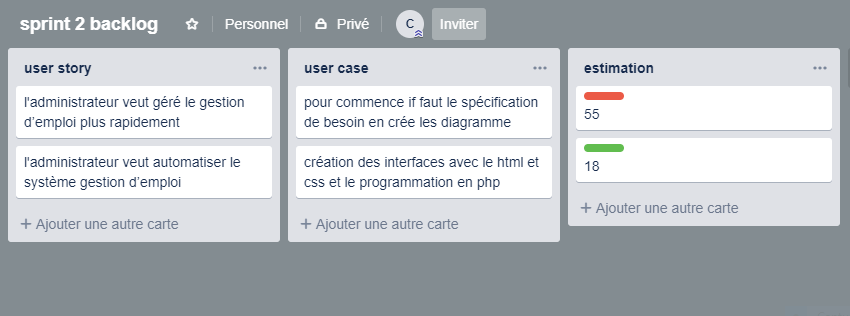
## ***Conception :***

### **Diagramme de classe :**



## ***Scrum :***

### **Sprint backlog :**



### **Scrum board :**

**Le 28 Avril :**

****

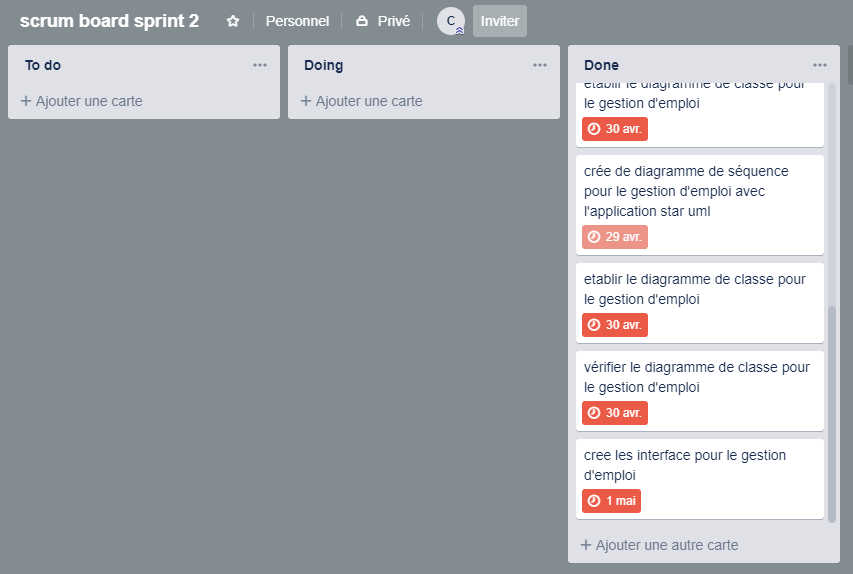
**Le 29 Avril :**

****

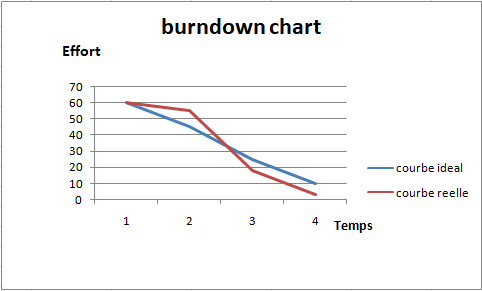
**Le 30 Avril :**

****

**Le premier Mai :**

****

### **Burndown chart :**



# ***Réalisation :***

Pour notre projet, ce chapitre sera consacré pour la présentation des langages et outils de programmation utilisés pour la réalisation de notre application. Ils sont globalement constitués de deux composants : le langage de programmation et la base de données.

## ***Langage de programmation :***

***PHP***

******

PHP ou HyperTextPreprocessor est un langage de script extrêmement puissant et destiné pour le développement des applications web. PHP est l’un des langages de programmation les plus populaire. Le point fort de ce langage c’est qu’il est portable et simple à utiliser.

***HTML / CSS***

******

Le langage HTML et le CSS (feuilles de style) sont des langages de base en informatique. Ils sont utilisés pour la description et le design de nos différentes interfaces.

## ***Méthodes et Logiciels :***

***XAMPP***

******

XAMPP est un ensemble de logiciels permettant de mettre en place facilement un serveur Web, un serveur FTP et un serveur de messagerie électronique. Il s'agit d'une distribution de logiciels libres (X Apache MySQL Perl PHP) offrant une bonne souplesse d'utilisation, réputée pour son installation simple et rapide.

***TRELLO***

******

TRELLO est un outil de gestion de projet en ligne, lancé en septembre 2011, et inspiré par la méthode **Kanban** de **Toyota**. Il est basé sur une organisation des projets en planches listant des cartes, chacune représente des taches. Les cartes sont assignables à des utilisateurs et sont mobiles d’une planche à l’autre, traduisant leur avancement. La version de base est gratuite, tandis qu’une formule payante permet d’obtenir des services supplémentaires.

***MOZILLA FIREFOX ET GOOGLE CHROME***

******

Comme navigateurs, nous avons utilisé MOZILLA FIREFOX et GOOGLE CHROME.

# ***Conclusion :***

A la lumière de ce que précède en peut conclure que notre projet gestion d’emplois de temps peut être bénéfique pour les administrateurs et facilite leurs taches.

D’où lors de réalisation de projet on a essayé de couvrir les besoins des salles car toujours on souffre de manque de salle même lorsque on a des heures et des enseignants disponibles. Par ailleurs ce projet était pour nous une occasion pour acquérir des atouts et des connaissances techniques sur un outil important dans web existants sur le marché, ainsi que l’utiliser pour réaliser un site web dynamique. On a pu partager ce travail entre nous Grâce au méthode SCRUM on fait une collaboration entre les membres de notre groupe.