



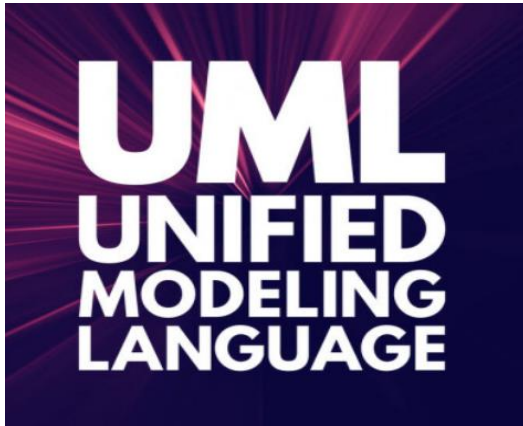
## Encadré par :

- Prof. Nidal Lamghari

# TABLE DES MATIÈRES

Introduction	3
Méthode Minimale	4
Modelisation	5
1. Diagramme de cas d'utilisation	5
2. Modèle du domaine	7
3. Diagrammes de séquence	8
Etudiant	9
Responsable cellule de stages	10
Jury	14
4. Les maquettes	15
5. Diagrammes de classes participantes	22
6. Diagrammes d'interaction	24
Etudiant	25
Responsable cellule de stages	26
Jury	29
7. Diagrammes de classes de conception	30
8. Diagrammes d'activité	31
Etudiant	32
Responsable cellule de stages	33
Jury	36
Conclusion	37

# INTRODUCTION



Le langage UML (Unified Modeling Language, ou langage de modélisation unifié) a été pensé pour être un langage de modélisation visuelle commun, et riche sémantiquement et syntaxiquement. Il est destiné à l'architecture, la conception et la mise en œuvre de systèmes logiciels complexes par leur structure aussi bien que leur comportement. L'UML a des applications qui vont au-delà du développement

logiciel, notamment pour les flux de processus dans l'industrie.

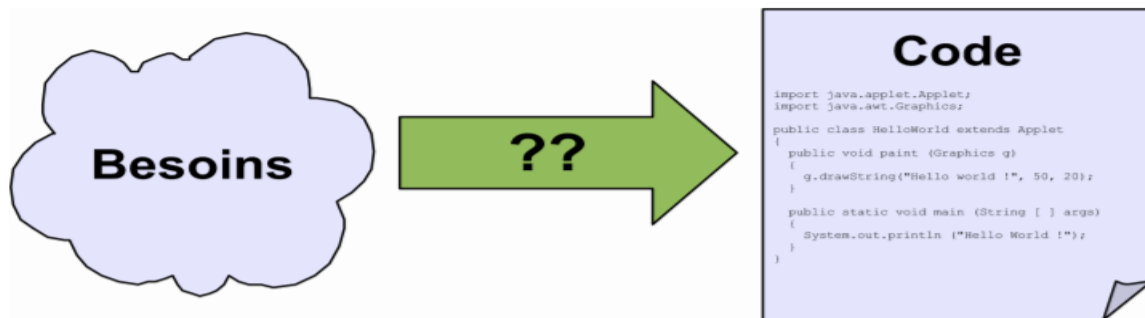
Il ressemble aux plans utilisés dans d'autres domaines et se compose de différents types de diagrammes. Dans l'ensemble, les diagrammes UML décrivent la limite, la structure et le comportement du système et des objets qui s'y trouvent.

L'UML n'est pas un langage de programmation, mais il existe des outils qui peuvent être utilisés pour générer du code en plusieurs langages à partir de diagrammes UML. L'UML a une relation directe avec l'analyse et la conception orientées objet.

Dans ce projet, on essaye de mettre en œuvre la méthode minimale de l'UML en passant par des étapes pour aboutir à la fin au code de l'application. On a travaillé par cette méthode pour garantir une bonne conception après une étape d'analyse des fonctionnalités et de besoins.

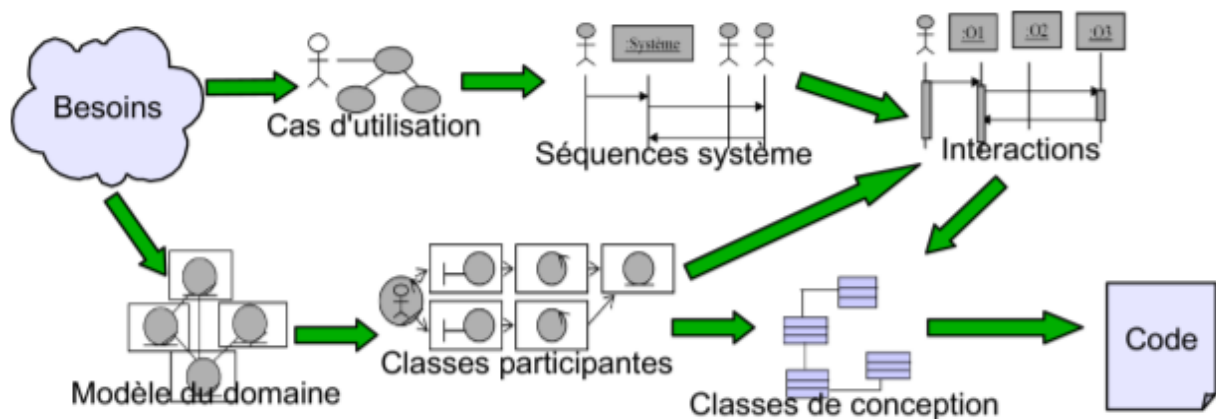
# METHODE MINIMALE

## Petit aperçu



La problématique que pose la mise en œuvre d'UML est simple : comment passer de l'expression des besoins au code de l'application ? Pour ça on a fait recours à la méthode minimale qui est un ensemble d'étapes partiellement ordonnées, qui concourent à l'obtention d'un système logiciel ou à l'évolution d'un système existant. Ces étapes ont pour objectif produire des logiciels de qualité (qui répondent aux besoins de leurs utilisateurs) dans des temps et des coûts prévisibles.

Cette méthode s'appuie sur la règle de 80% 20% : « Résoudre 80% des problèmes avec 20% d'UML. »



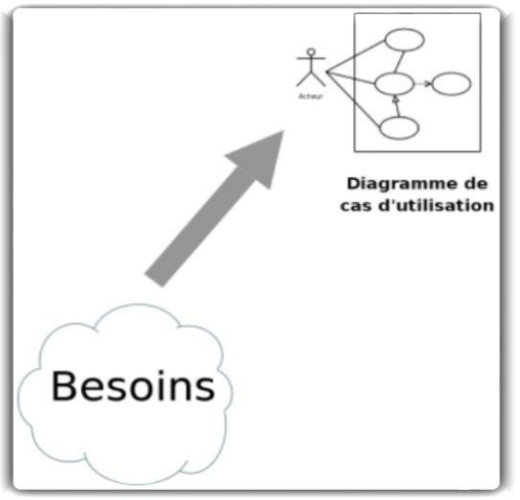
Comme mentionné ci-dessus, on procède avec les diagrammes usuels :

- Diagramme de cas d'utilisation.
- Diagramme de séquence.
- Modèle du domaine.
- Diagramme des classes participantes.
- Diagramme d'interaction.
- Diagramme de classe de conception.

# MODELISATION

## 1. Diagramme de cas d'utilisation

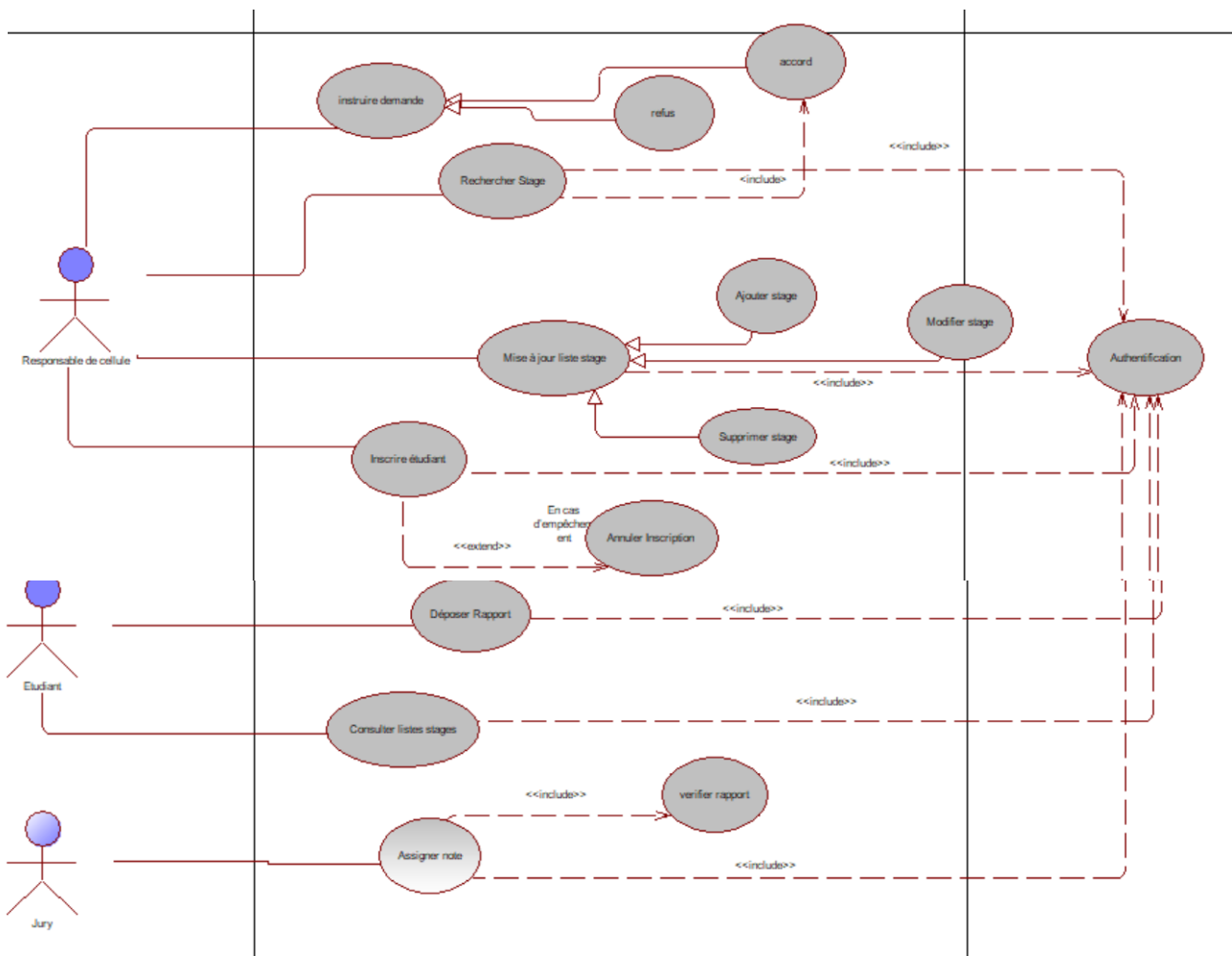
---



Dans un premier temps, on crée les cas d'utilisation pour identifier et modéliser les besoins des utilisateurs. Ces besoins sont déterminés à partir des informations recueillies lors des rencontres entre informaticiens et utilisateurs. Il faut impérativement proscrire toute considération de réalisation lors de cette étape.

Durant cette étape, on doit déterminer **les limites du système, identifier les acteurs et recenser les cas d'utilisation**. Si l'application est complexe, on peut organiser les cas d'utilisation en paquetages. On définit les besoins comme suit :

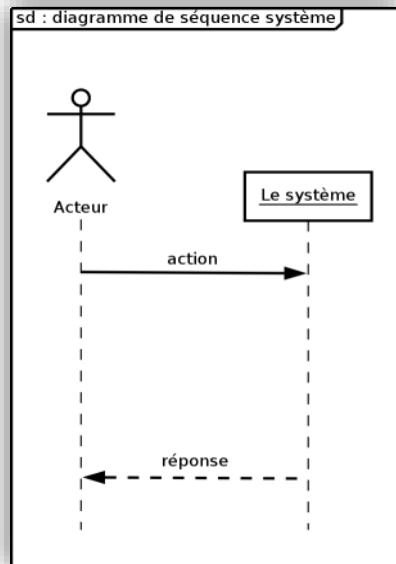
- Identifier les limites du système
- Identifier les acteurs
- Identifier les cas d'utilisation
- Structurer les cas d'utilisation en package
- Ajouter les relations entre cas d'utilisation
- Classer les cas d'utilisation par ordre d'importance





### 3. Diagrammes de séquence

---

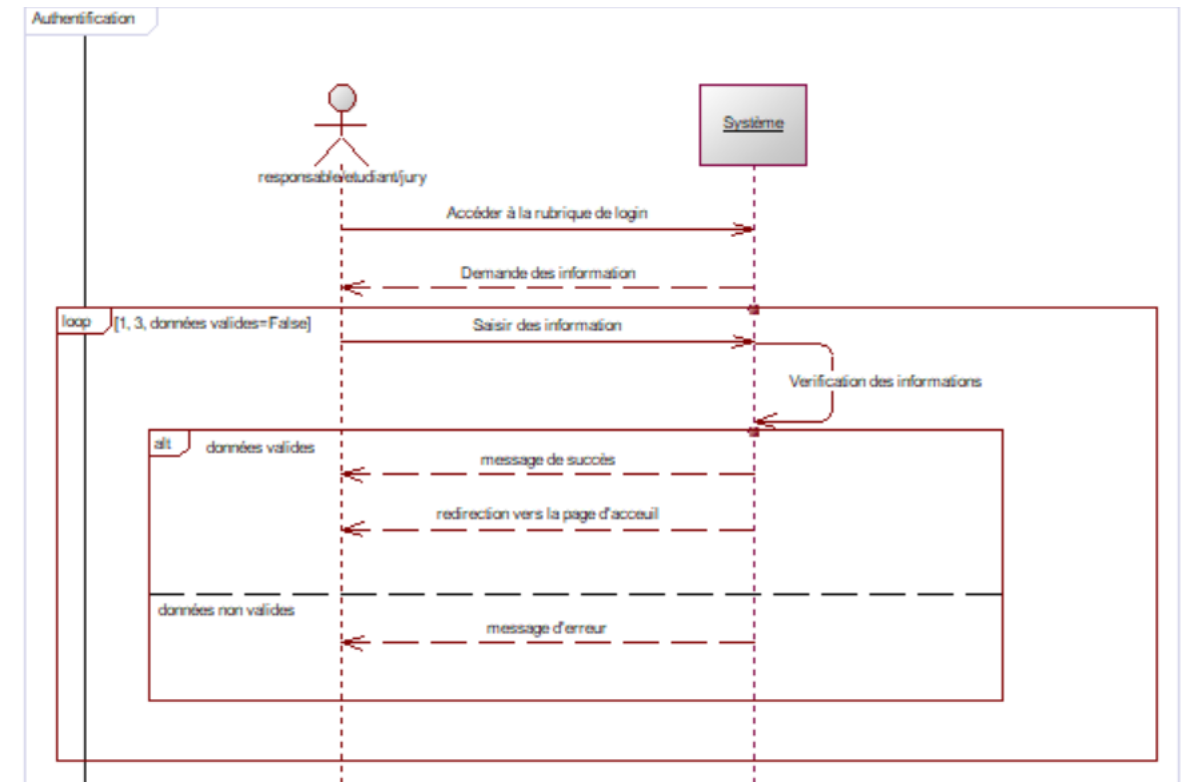


On cherche à détailler la description des besoins par la description textuelle des cas d'utilisation et la production de diagrammes de séquence système illustrant cette description textuelle. Cette étape amène souvent à mettre à jour le diagramme de cas d'utilisation puisque nous sommes toujours dans la spécification des besoins. Les scénarios de la description textuelle des cas d'utilisation peuvent être vus comme des instances de cas d'utilisation et sont illustrés par des diagrammes de séquence système. Il faut, au minimum, représenter le scénario nominal de chacun des cas d'utilisation par un diagramme de séquence qui rend compte de l'interaction

entre l'acteur, ou les acteurs, et le système. **Le système est ici considéré comme un tout** et est représenté par une ligne de vie. Chaque acteur est également associé à une ligne de vie. Les diagrammes de séquence sont parfois très simples mais ils seront enrichis par la suite.

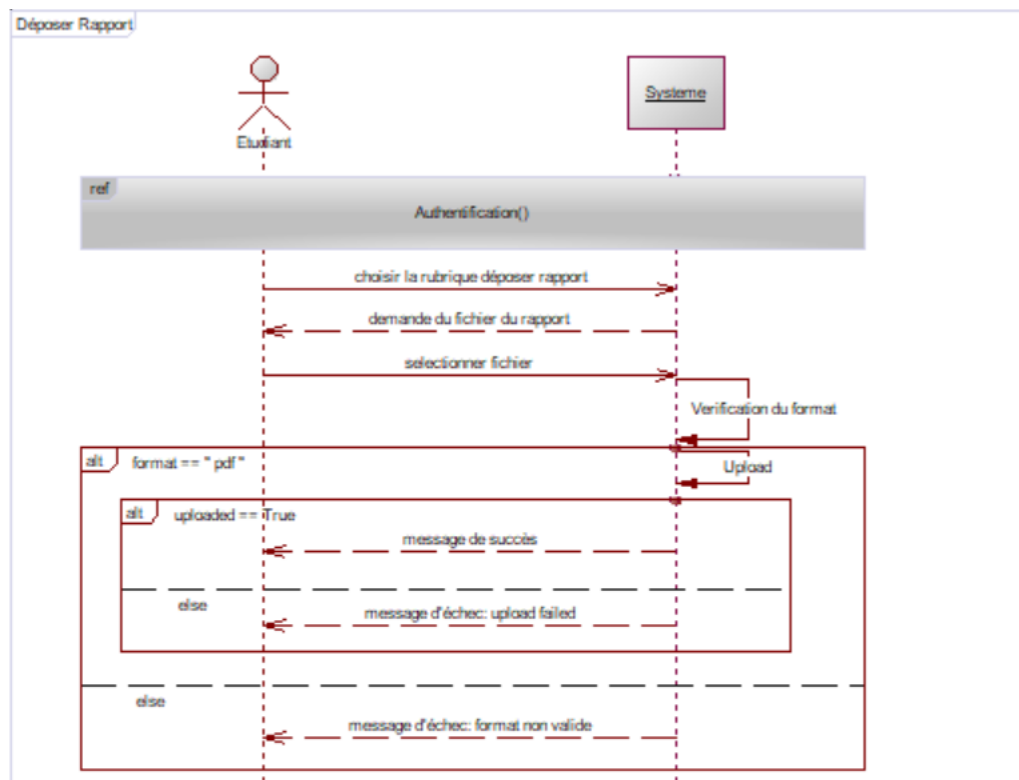


## Authentification :

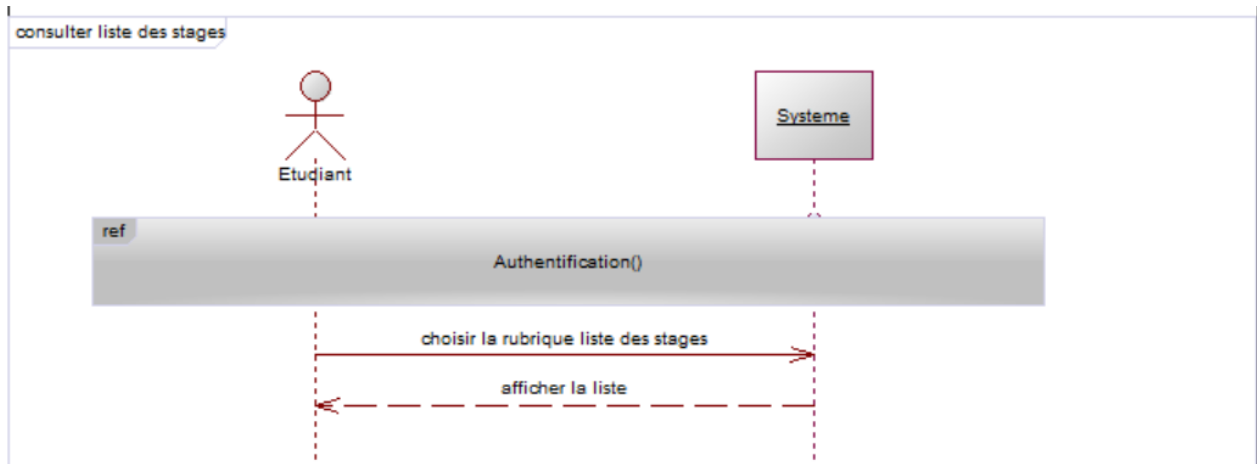


## Etudiant

### Déposer rapport :

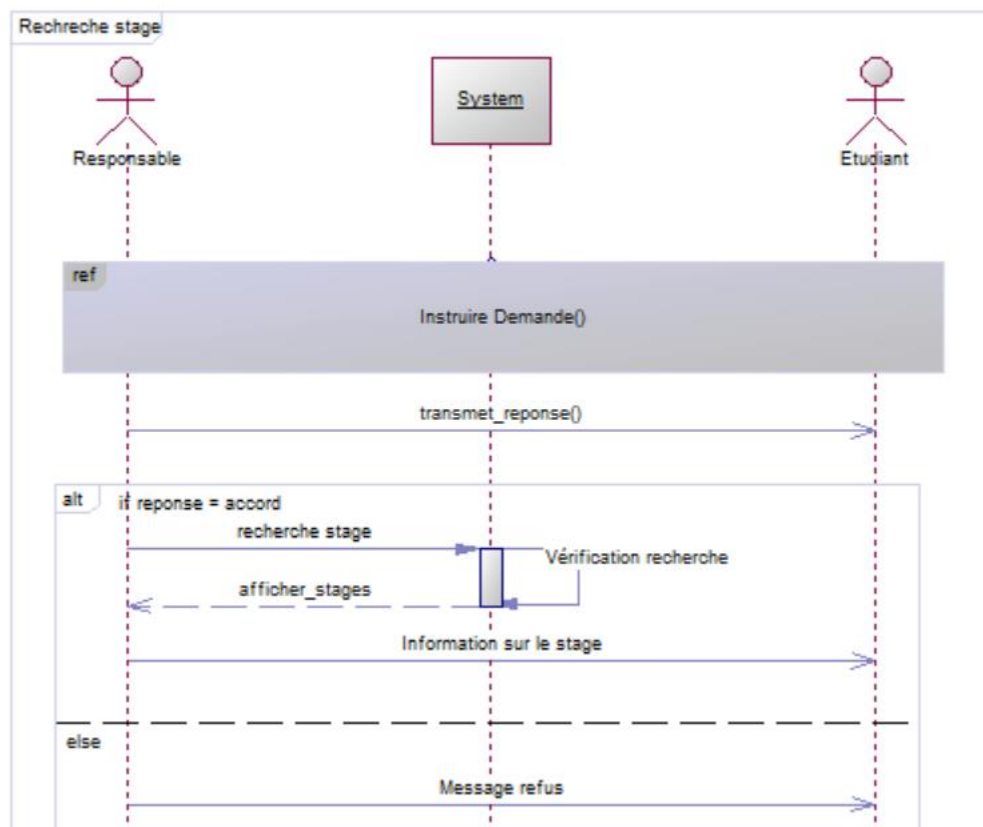


Consulter liste stages :

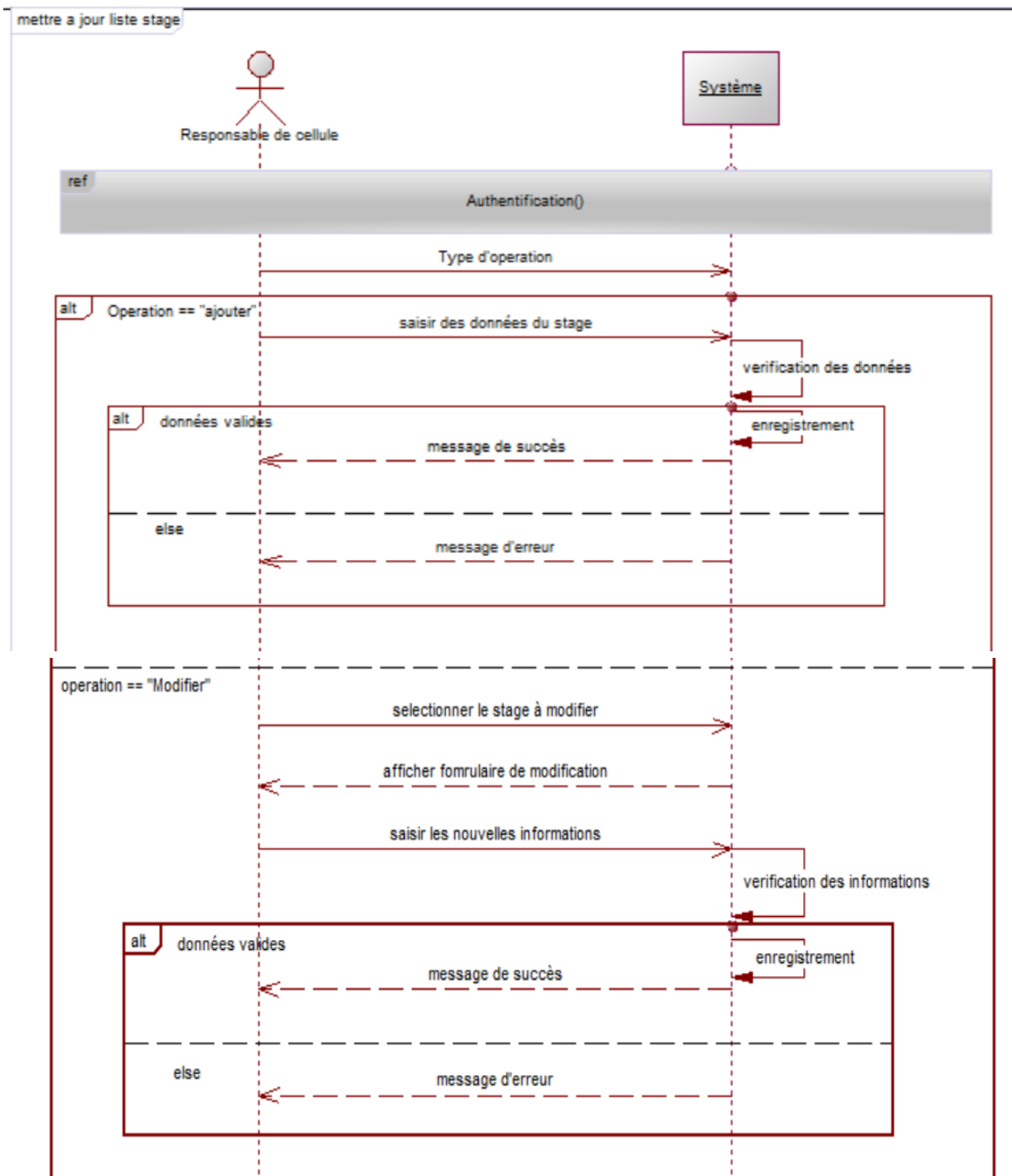


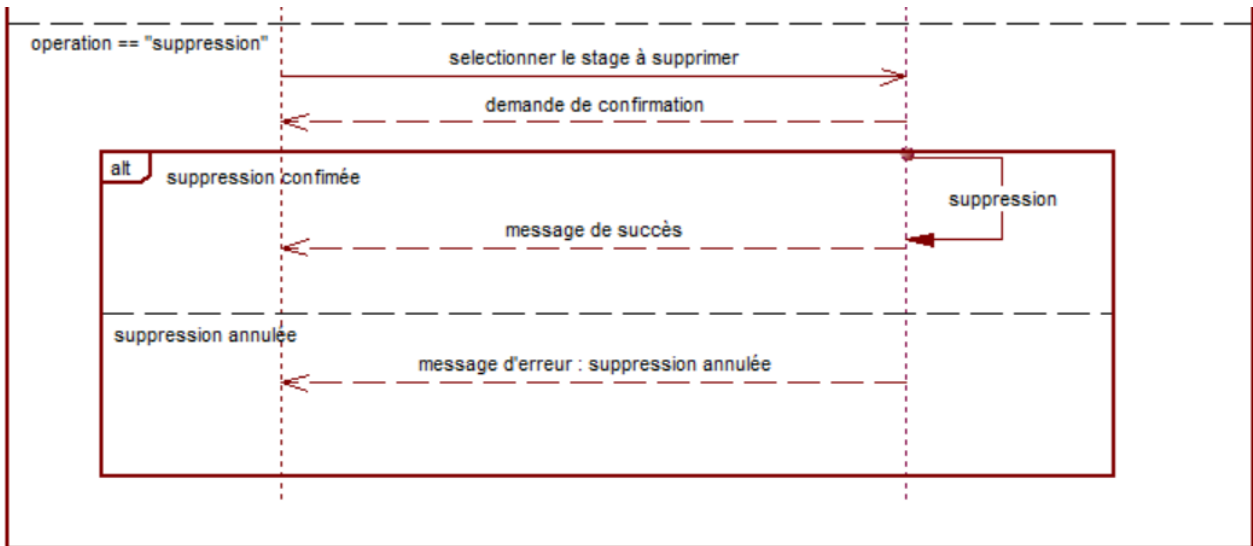
## Responsable cellule de stages

Chercher stage :

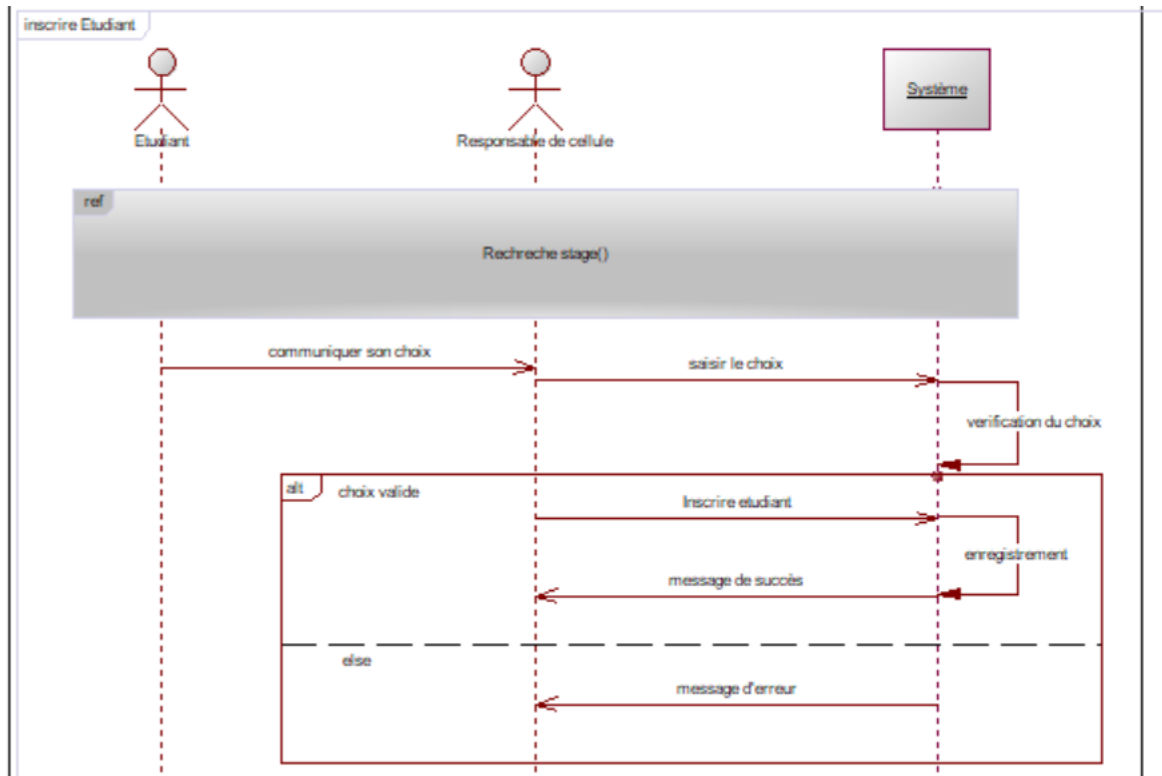


## Mettre à jour :

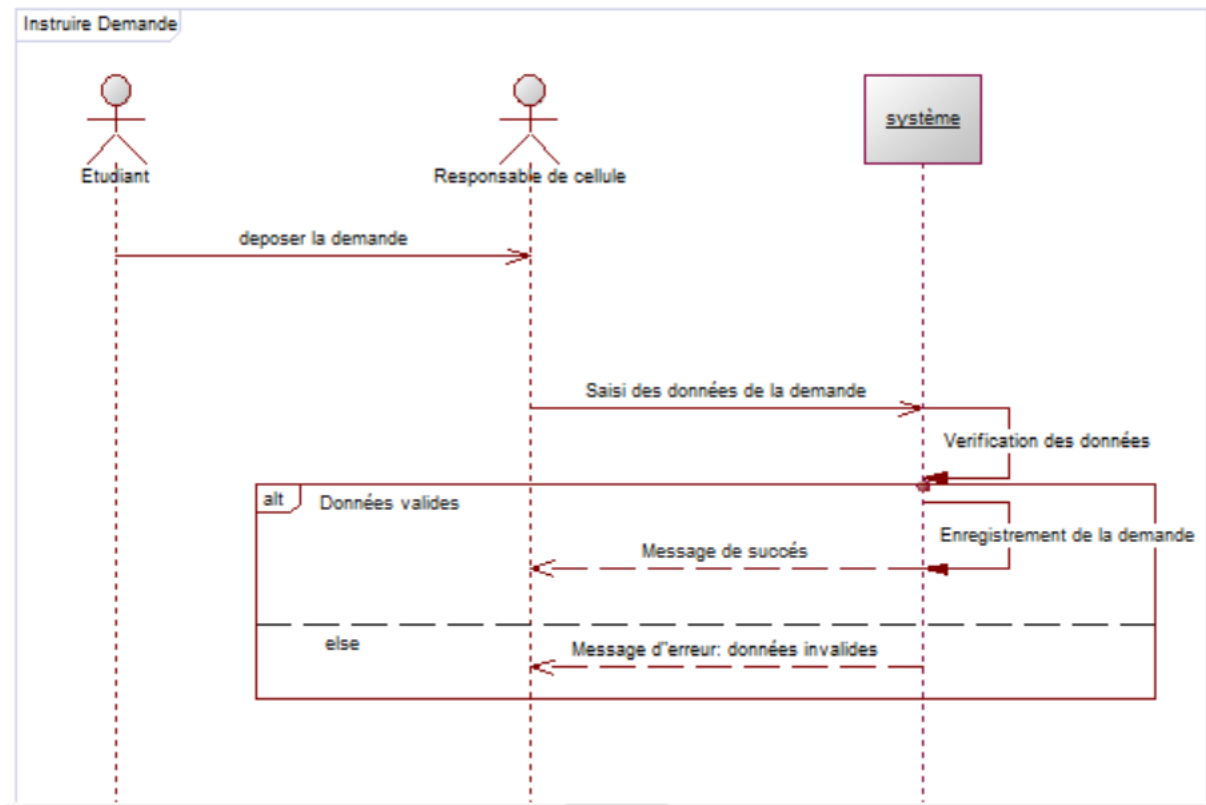




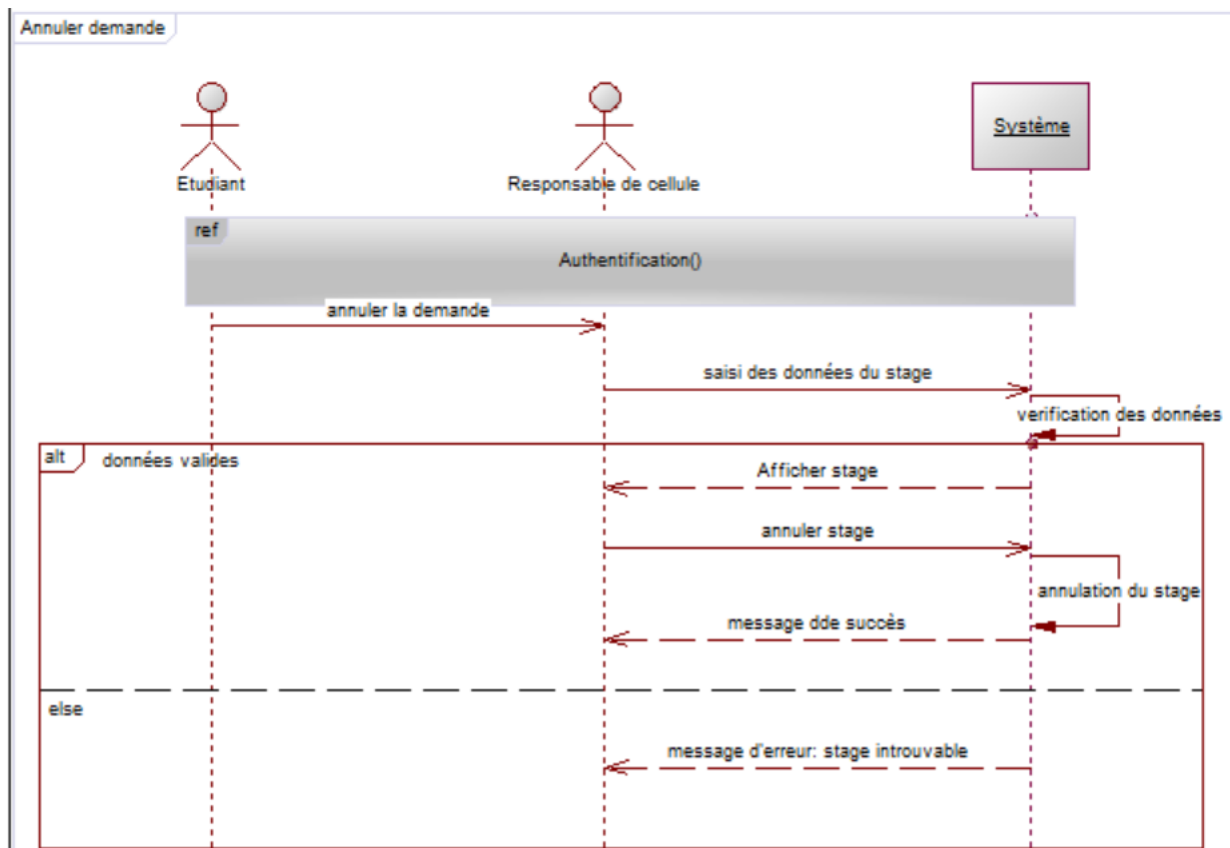
### Inscrire Etudiant :



### Instruire demande :

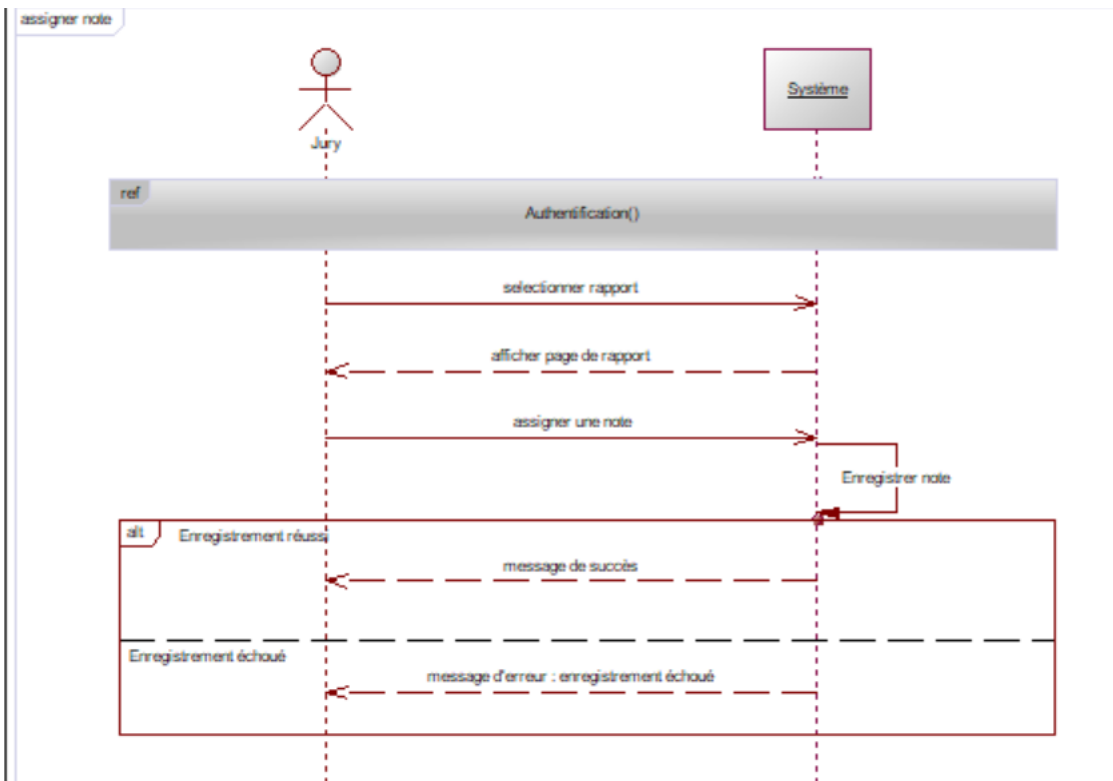


### Annuler inscription :



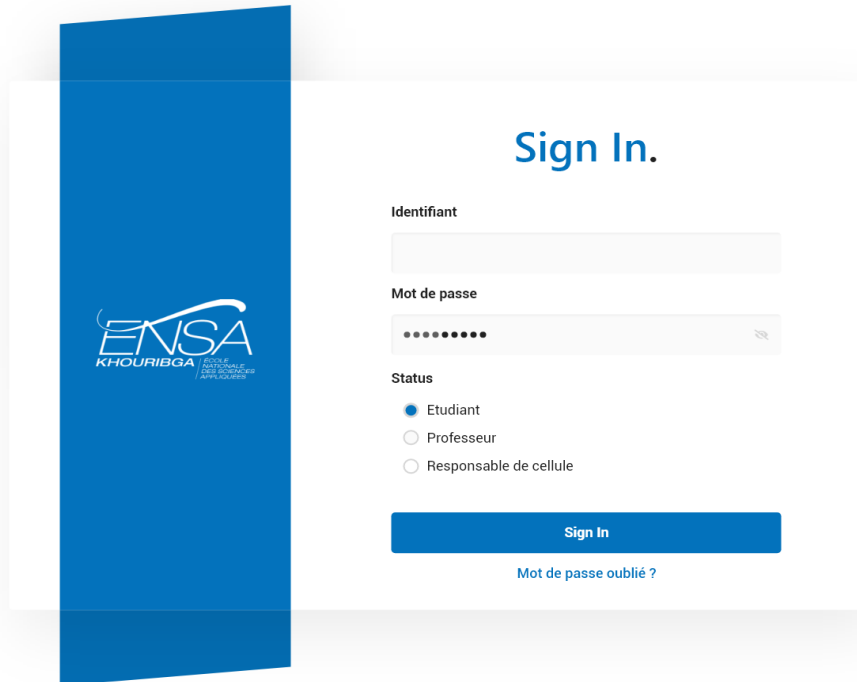
## Jury

Assigner note stage :



## 4. Les maquettes

# Login



**Sign In.**

Identifiant

Mot de passe

Status

☒ Etudiant

☐ Professeur

☐ Responsable de cellule

**Sign In**

[Mot de passe oublié ?](#)

# Mot de passe oublié



## Mot de passe oublié ?

Veuillez saisir votre Email et identifiant, vous allez recevoir un nouveau Code

Email

Identifiant

Valider

# Espace Etudiant

Hind Fakhradine · · ·

Déposer Rapport

Consulter la liste des stages

### Informations Personnelles

Nom : Fakhradine	Lieu de Naissance : Khouribga
Prenom : Hind	Date de Naissance : 25 Juin 2000
Niveau : 2ème année Cycle d'ingénieur Génie Informatique	Email Universitaire : hind.fakhradine@usms.ac.ma
CNE : *****	Nationalité : Marocaine



# Déposer Rapport

Hind Fakhradine · · ·

[Consulter la liste des stages](#)[Accueil](#)

## Informations Personnelles

Nom : Fakhradine

Prenom : Hind

Niveau : 2ème année Cycle d'ingénieur  
Génie Informatique

CNE : \*\*\*\*\*

Lieu de Naissance : Khouribga

Date de Naissance : 25 Juin 2000

Email Universitaire : hind.fakhradine@usms.ac.ma

Nationalité : Marocaine

## Déposer un rapport

Nom de rapport \* : \_\_\_\_\_

[Joindre un fichier \\*](#)[Valider](#)

# Consulter la liste des stages

Hind Fakhradine · · ·

[Déposer Rapport](#)[Accueil](#)

## Informations Personnelles

Nom : Fakhradine

Prenom : Hind

Niveau : 2ème année Cycle d'ingénieur  
Génie Informatique

CNE : \*\*\*\*\*

Lieu de Naissance : Khouribga

Date de Naissance : 25 Juin 2000

Email Universitaire : hind.fakhradine@usms.ac.ma

Nationalité : Marocaine

## Liste des stages

Entrez un stage



Code	Sujet du stage	Organisme d'accueil	Etat	Plus d'informations
001	Système RPA	MedTech Groupe	✓	<a href="#">Info</a>
002	Application Mobile	Deloitte	✗	<a href="#">Info</a>
003	Recherche Opératio...	Capgemini	✓	<a href="#">Info</a>
004	Application Web - React	Orange BS	✗	<a href="#">Info</a>
005	Business Analysis	Société Générale	✓	<a href="#">Info</a>
006	Single-page Web APP	Oracle	✓	<a href="#">Info</a>

# Espace Professeur

John Doe

## Informations Personnelles

Nom : Doe

Prenom : John

Grade : Professeur de l'Enseignement Supérieur (PES)

Code : \*\*\*\*\*

Email Universitaire : john.doe@usms.ma

Nationalité : Marocaine

## Liste des stages

Entrez un stage



Code	Sujet du stage	Organisme d'accueil	Etudiant	Note	Action
001	Système RPA	MedTech Groupe	Hind Fakhra...	17/20	--
002	Application Mobile	Deloitte	Safaa Bidd...	16/20	--
003	Recherche Opératio...	Capgemini	Selma Ennasi...	--	Noter
004	Application Web - React	Orange BS	Afraa Zeroual	--	Noter
005	Business Analysis	Société Générale	Sanaa Semmat	16/20	--
006	Single-page Web APP	Oracle	Soukaina Bentalia	--	Noter

# Assigner Note

John Doe

## Informations Personnelles

Nom : Doe

Prenom : John

Grade : Professeur de l'Enseignement Supérieur (PES)

Code : \*\*\*\*\*

Email Universitaire : john.doe@usms.ma

Nationalité : Marocaine

## Assigner une note

Code du rapport : Rapport.0021

Titre du rapport : Rapport de stage - Système RPA

Organisme d'accueil : MedTech Groupe

Etudiant : Hind Fakhradine

Note Finale : \_\_\_\_ / 20

Valider



Télécharger le rapport

Accueil

# Espace Responsable

Res-NL

Instruire une demande
Annuler une demande
Ajouter un stage
Modifier un stage
Ajouter un étudiant

Liste des stages

Entrez un stage

Code	Sujet du stage	Organisme d'accueil	Etat	Plus d'informations
001	Système RPA	MedTech Groupe	✓	Info
002	Application Mobile	Deloitte	✗	Info
003	Recherche Opératio...	Capgemini	✓	Info
004	Application Web - React	Orange BS	✗	Info
005	Business Analysis	Société Générale	✓	Info
006	Single-page Web APP	Oracle	✓	Info

# Instruire demande

Res-NL

Accueil
Ajouter un stage
Annuler Demande
Modifier un stage
Ajouter un étudiant

Instruire une demande

Nom de l'étudiant
This field is required

Prénom de l'étudiant
This field is required

Filière
This field is required

CNE
This field is required

Sujet du stage
This field is required

Etat de la demande
This field is required

Date de la demande
This field is required

Valider

# Annuler demande

Res-NL . . .

- Accueil
- Instruire une demande
- Ajouter un stage
- Modifier un stage
- Ajouter un étudiant

Liste des stages

Entrez un stage

Selectionnez les demandes à annuler

Code	Sujet du stage	Organisme d'accueil	Etudiant	Selectionner
001	Système RPA	MedTech Groupe	Hind Fakhra...	<input checked="" type="radio"/>
002	Application Mobile	Deloitte	Safaa Bidd...	<input checked="" type="radio"/>
003	Recherche Opératio...	Capgemini	Selma Ennasi...	<input type="radio"/>
004	Application Web - React	Orange BS	Afraa Zeroual	<input checked="" type="radio"/>
005	Business Analysis	Société Générale	Sanaa Semmat	<input type="radio"/>
006	Single-page Web APP	Oracle	Soukaina Bentalia	<input type="radio"/>

Annuler

# Ajouter Stage

Res-NL . . .

- Accueil
- Instruire une demande
- Annuler Demande
- Modifier un stage
- Ajouter un étudiant

Ajouter Stage

Sujet du stage

This field is required

Organisme d'accueil

This field is required

Professeur 1 = Professeur 2 = Professeur 3 =

This field is required

Etudiant Concerné

This field is required

Valider

# Inscrire Etudiant

Res-NL

Accueil  
Instruire une demande  
Annuler Demande  
Modifier un stage  
Ajouter un stage

### Ajouter Etudiant

Nom de l'étudiant

This field is required

Filière

This field is required

Lieu de Naissance

This field is required

Email Universitaire

This field is required

Prénom de l'étudiant

This field is required

CNE

This field is required

Date de Naissance

This field is required

Nationalité

This field is required

Valider

# Modifier Stage

Res-NL

Accueil  
Instruire une demande  
Annuler Demande  
Ajouter un stage  
Ajouter un étudiant

### Modifier Stage

Système RPA

This field is required

Professeur 1

Professeur 2

Professeur 3

This field is required

MedTech Groupe

This field is required

Hind Fakhradine

This field is required

☒ Disponible
☐ Non Disponible

Valider

## 5. Diagrammes de classes participantes

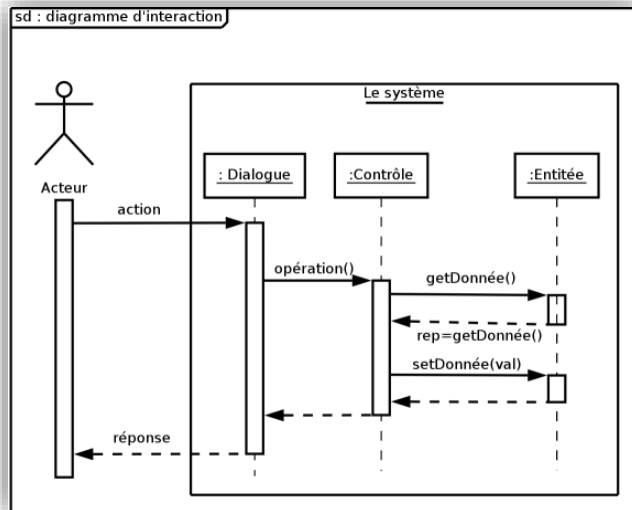
Le Diagramme des Classes Participantes est un diagramme de classes décrivant toutes les classes d'analyse. Le DCP est une version enrichie du modèle du domaine, auquel on adjoint les classes d'interaction et de contrôle. A ce point du développement, seules les classes de dialogue ont des opérations, qui correspondent aux opérations système, c'est à dire aux messages échangés avec les acteurs, que seules les classes de dialogues sont habilitées à intercepter ou à émettre.

Architecture en couches :

- Les dialogues ne peuvent être reliés qu'aux contrôles ou à d'autres dialogues (en général, associations unidirectionnelles).
- Les classes métier ne peuvent être reliées qu'aux contrôles ou à d'autres classes métier.
- Les contrôles peuvent être associés à tous les types de classes.



## 6. Diagrammes d'interaction



Il faut attribuer précisément les responsabilités de comportement, dégagées par le diagramme de séquence système, aux classes d'analyse du diagramme de classes participantes. Les résultats de cette réflexion sont présentés sous la forme de diagrammes d'interaction UML. Inversement, l'élaboration de ces diagrammes facilite grandement la réflexion.

Parallèlement, une première ébauche de la vue statique de conception, c'est-à-dire du diagramme de classes de

conception, est construite et complétée. Durant cette phase, l'ébauche du diagramme de classes de conception reste indépendante des choix technologiques qui seront faits ultérieurement.

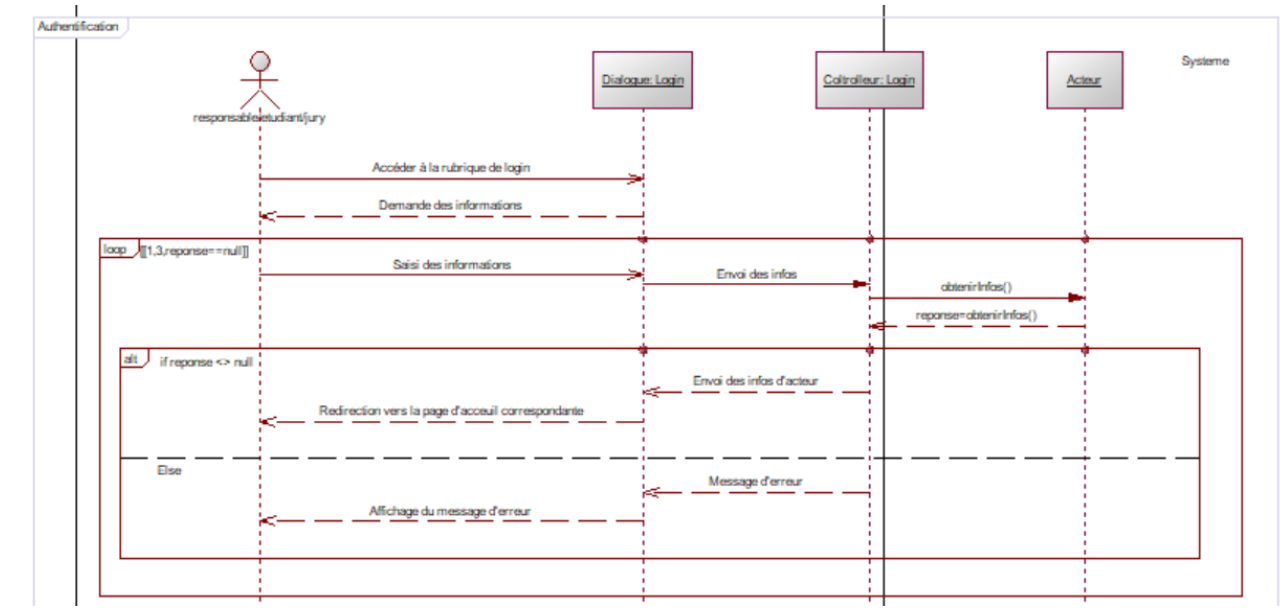
Pour chaque service ou fonction, il faut décider quelle est la classe qui va le contenir. Les diagrammes d'interactions sont particulièrement utiles au concepteur pour représenter graphiquement ces décisions d'allocations des responsabilités. Chaque diagramme va représenter un ensemble d'objets de classes différentes collaborant dans le cadre d'un scénario d'exécution du système.

Ici le système, vu comme **une boîte noire**, on le remplace par un ensemble d'objets en collaboration. Ces objets sont des instances des trois types de classes d'analyse du diagramme de classes participantes, à savoir **des dialogues, des contrôles et des entités**. Les diagrammes de séquences élaborés dans cette section doivent donc toujours respecter les règles édictées. Ces règles doivent cependant être transposées, car, pour que deux objets puissent interagir directement, il faut que :

- Les classes dont ils sont issus soient en association dans le diagramme de classes participante
- L'interaction respecte la navigabilité de l'association en question

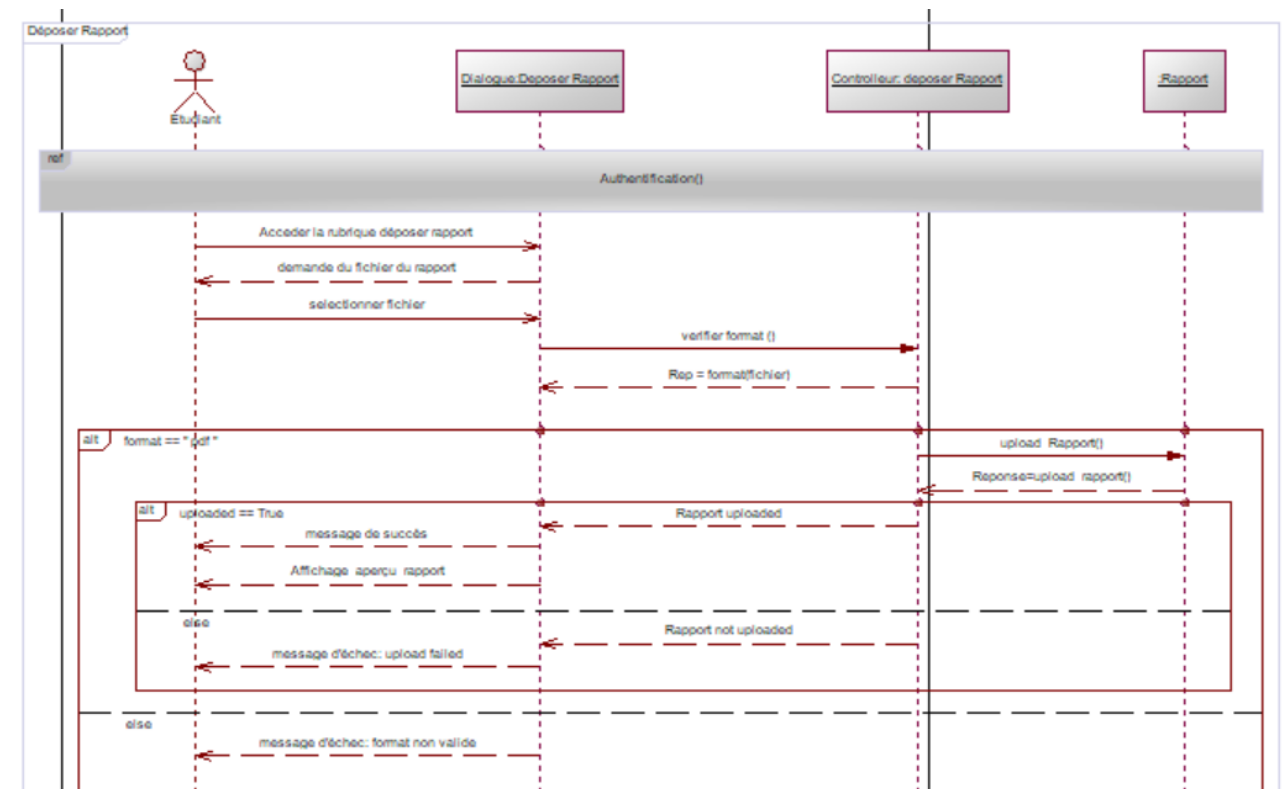


## Authentification :



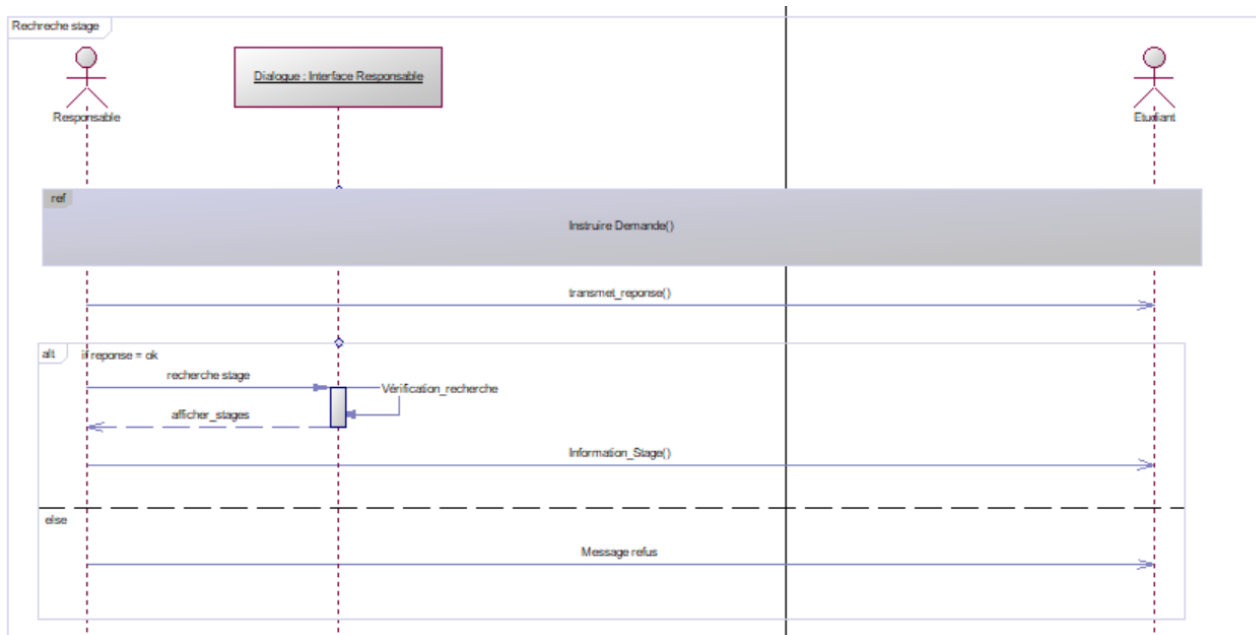
## Etudiant

### Déposer rapport :

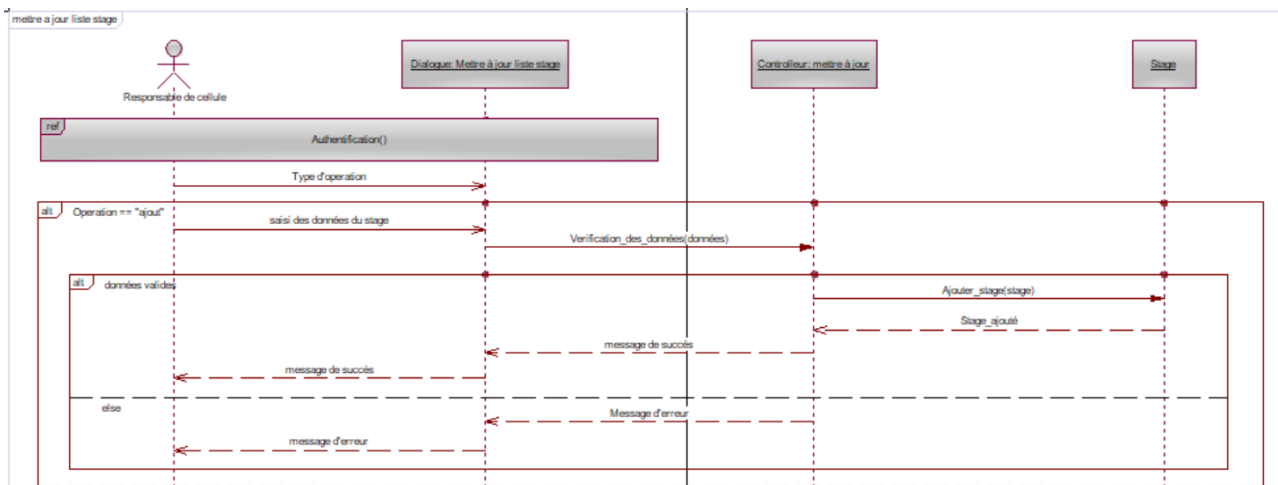


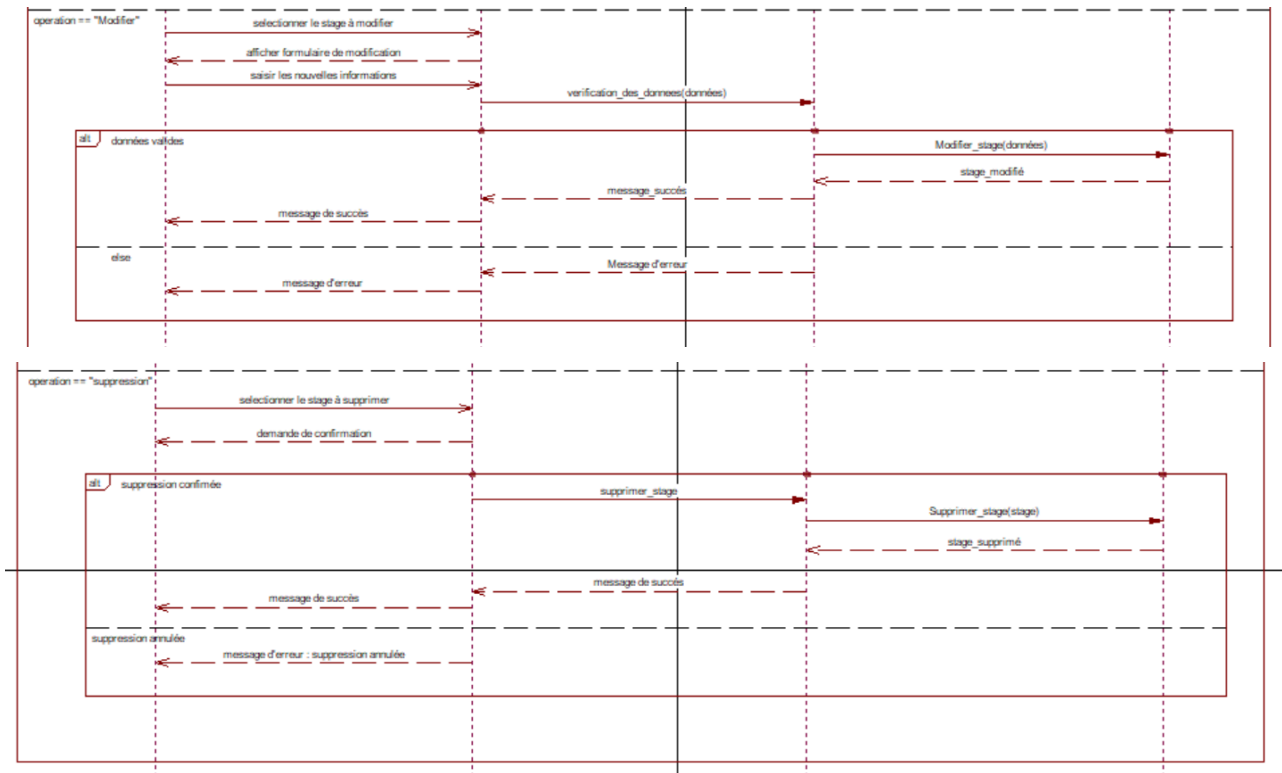
## Responsable cellule de stages

*Chercher stage:*

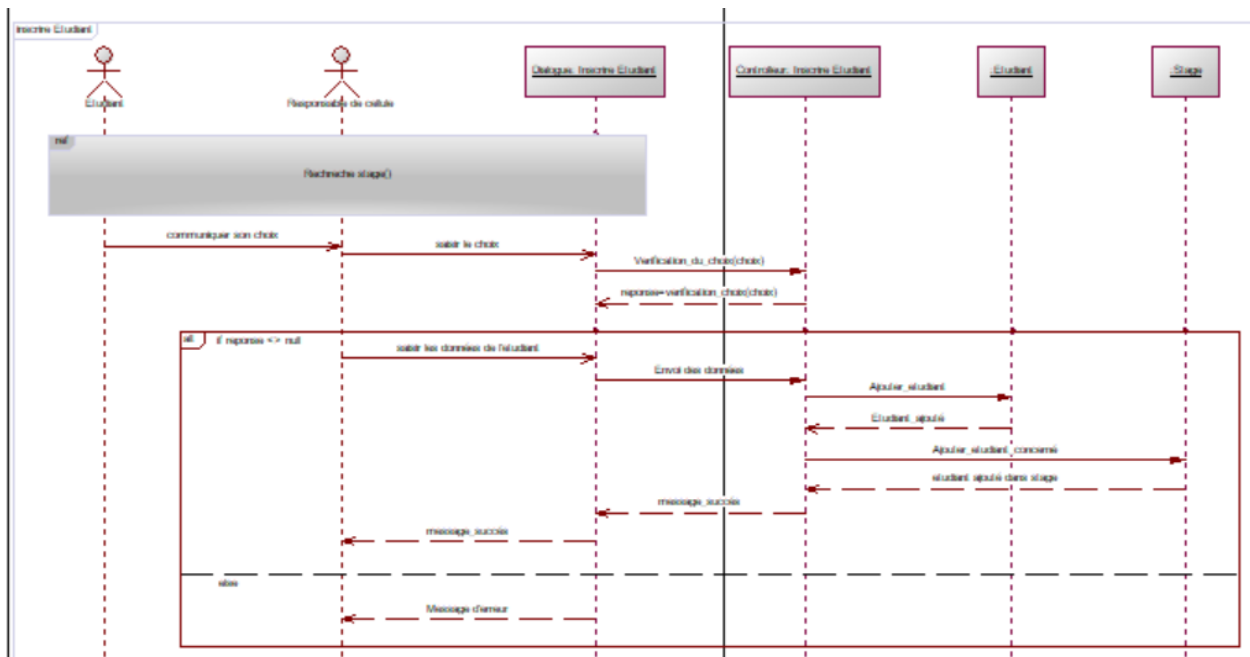


*Mettre à jour :*

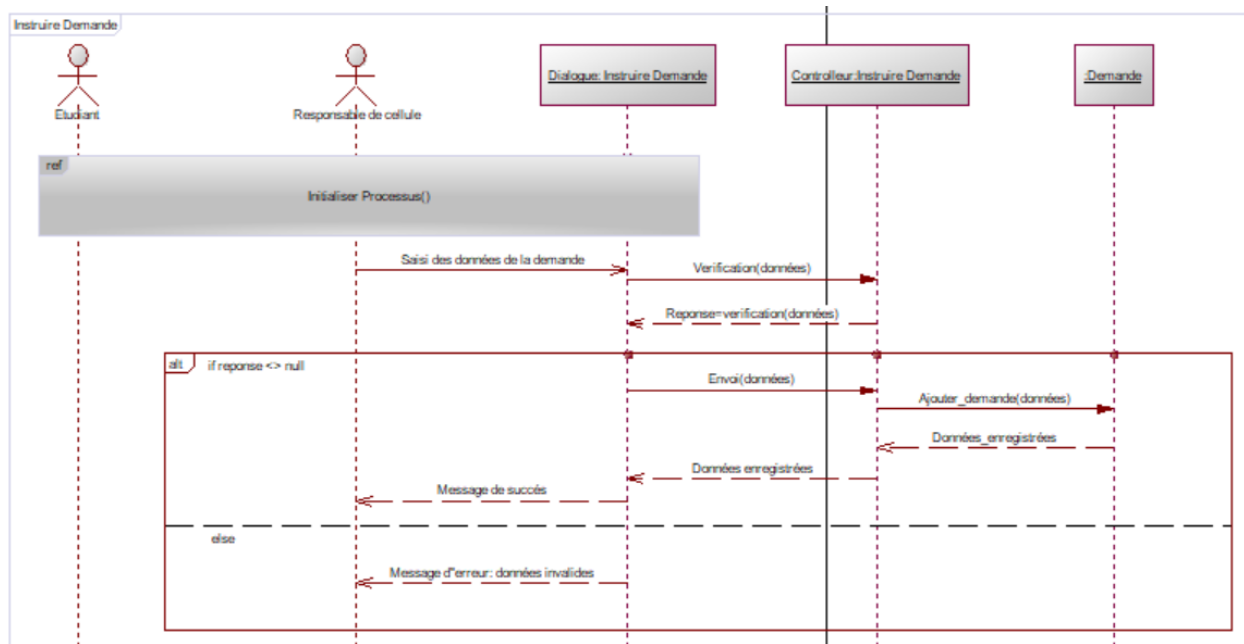




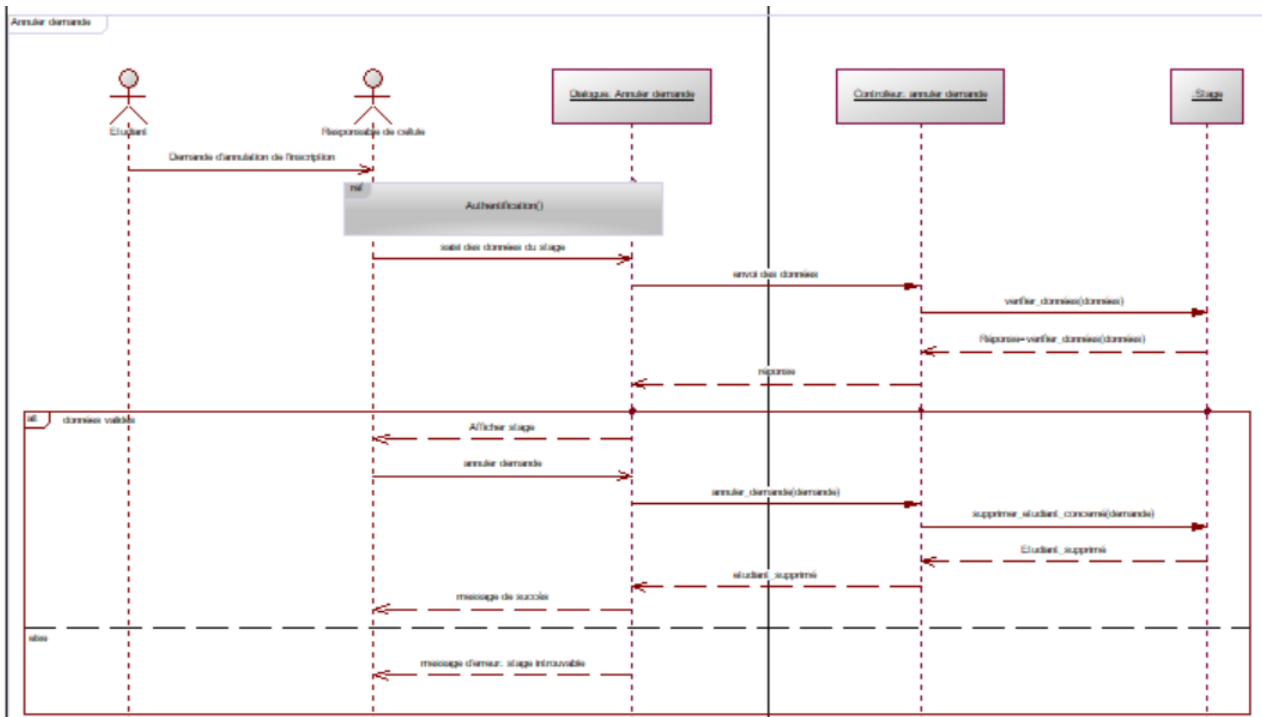
### Inscrire Etudiant :



### Instruire demande :

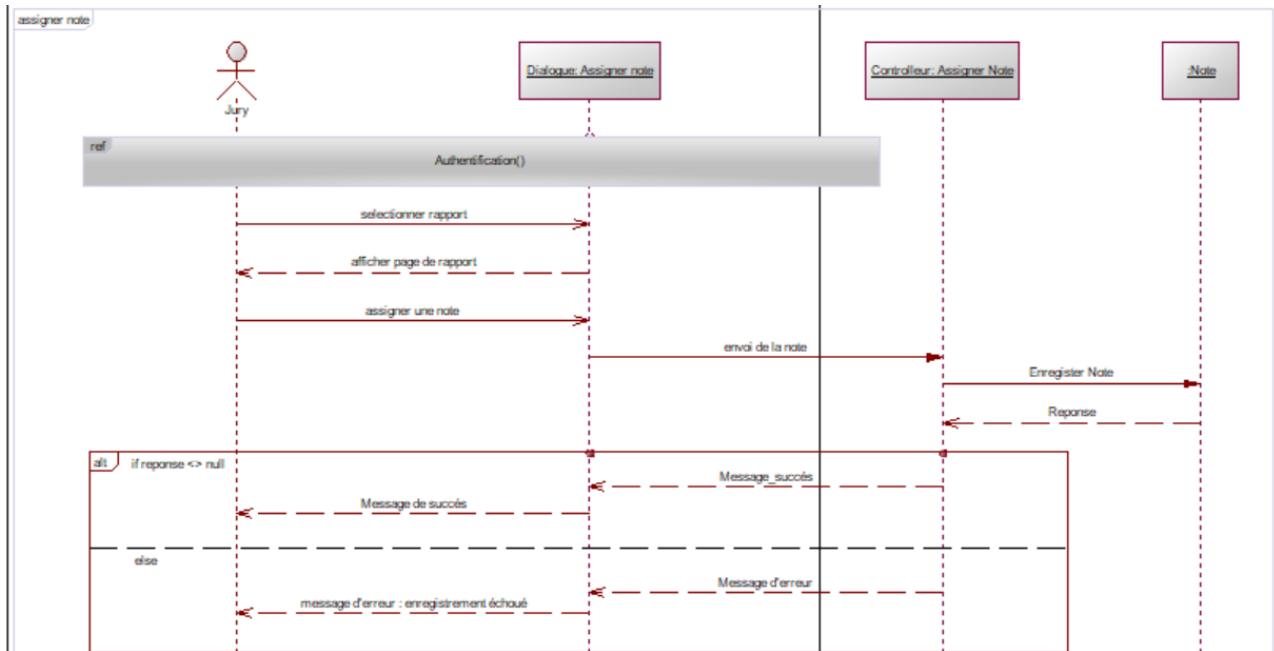


### Annuler inscription :



## Jury

Assigner note stage :

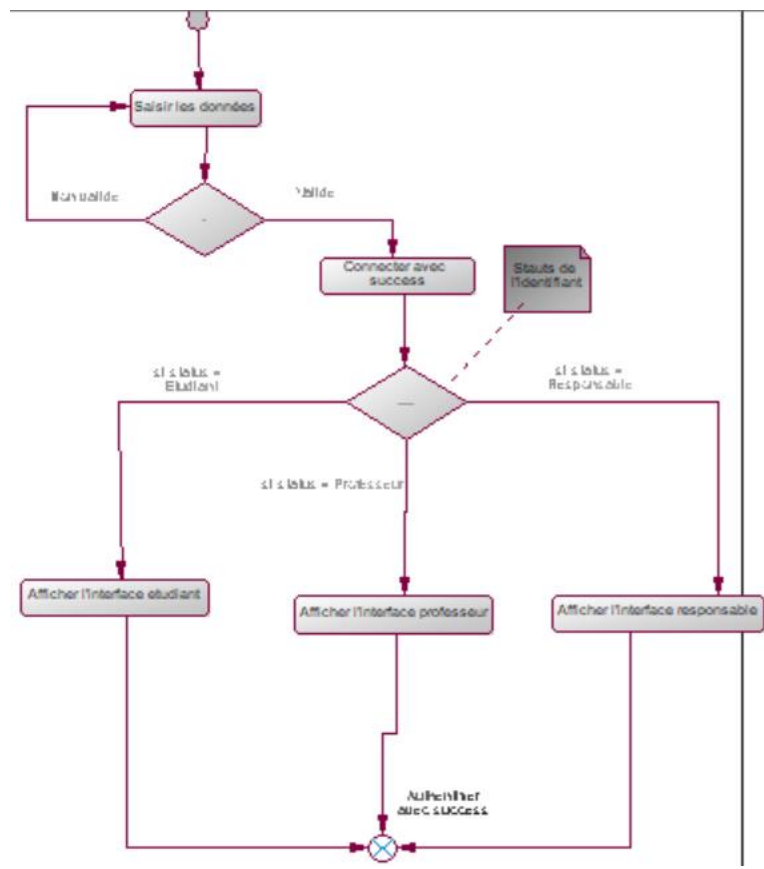




## 8. Diagrammes d'activité

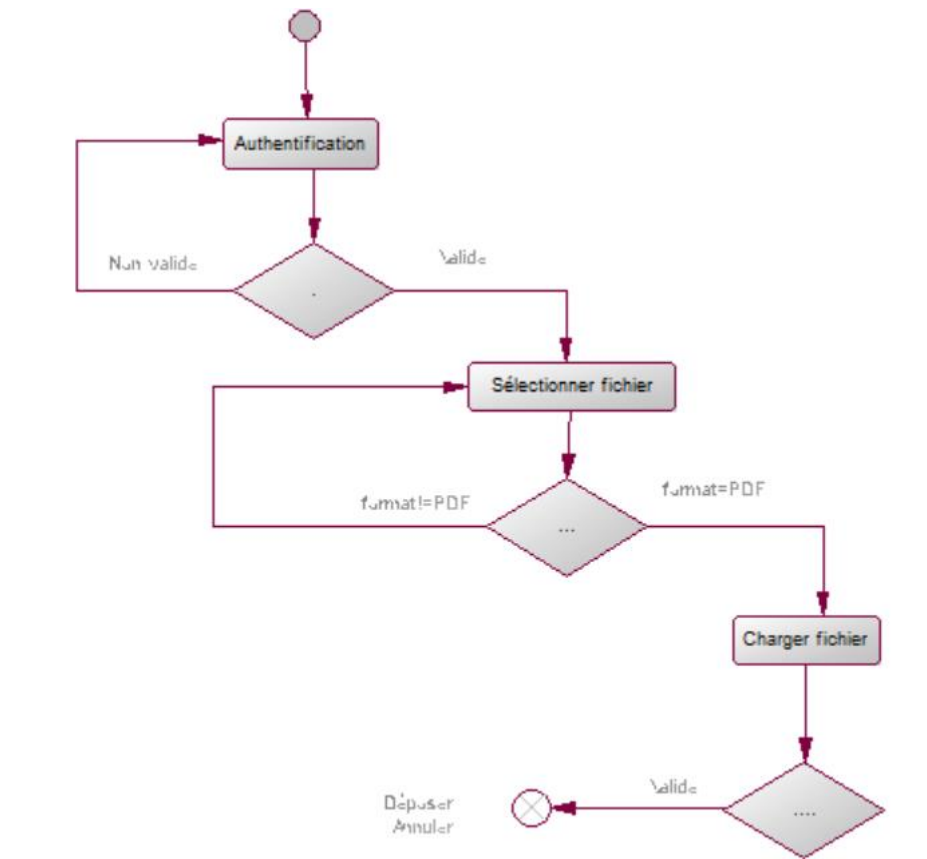
Pour enrichir notre recherche dans le projet nous avons ajouté le diagramme d'activité. Les diagrammes d'activités permettent de mettre l'accent sur les traitements. Ils sont donc particulièrement adaptés à la modélisation du cheminement de flots de contrôle et de flots de données. Ils permettent ainsi de représenter graphiquement le comportement d'une méthode ou le déroulement d'un cas d'utilisation.

### Authentification :

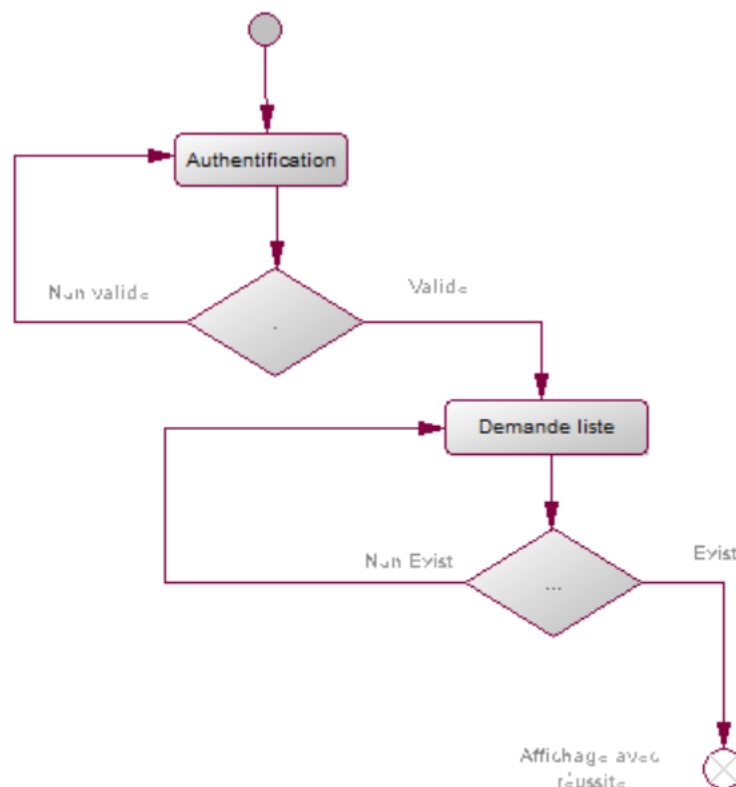


## Etudiant

Déposer rapport :



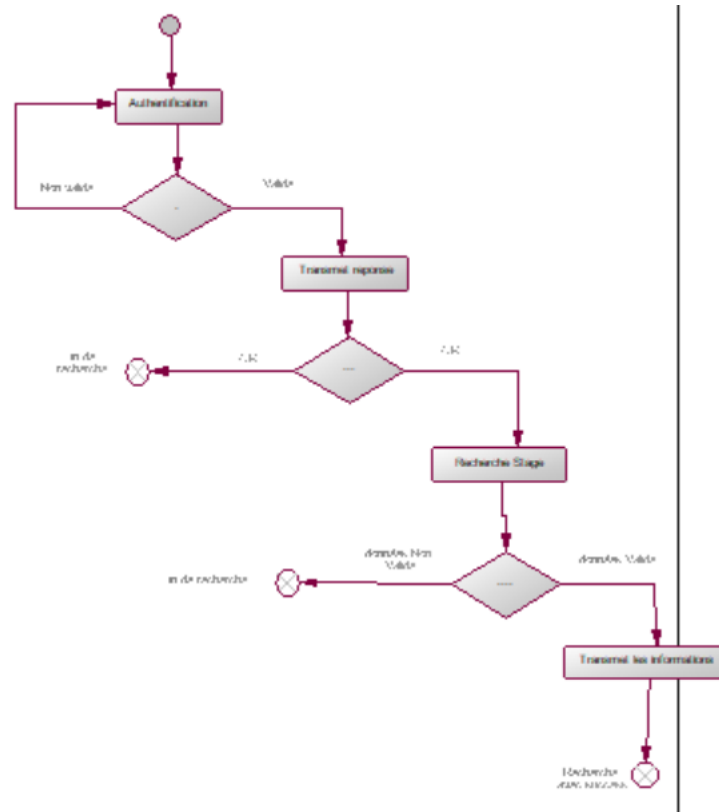
Consulter liste :



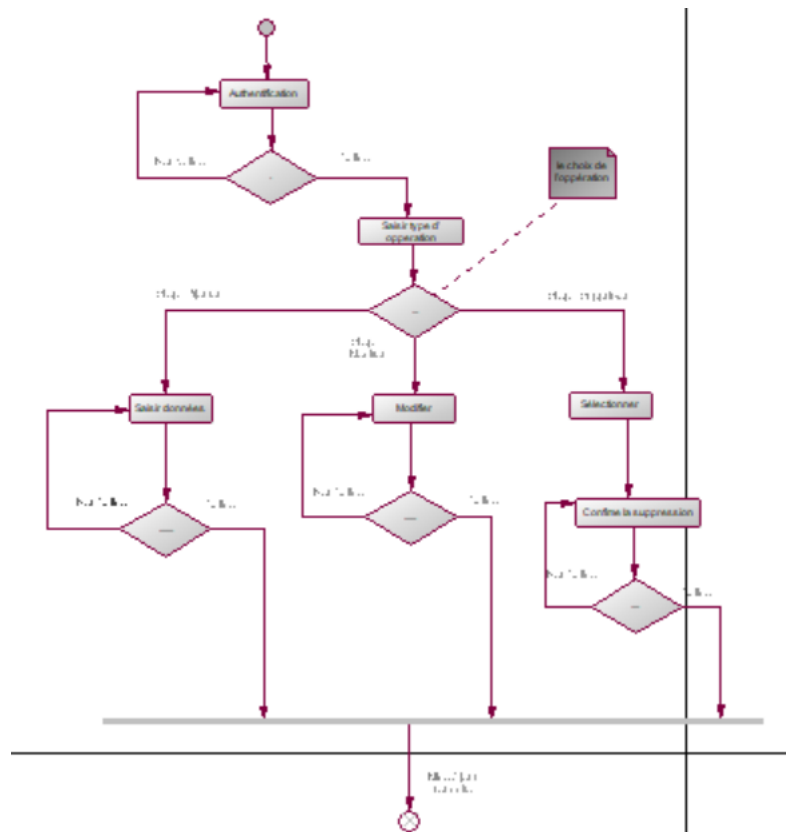


## Responsable cellule de stages

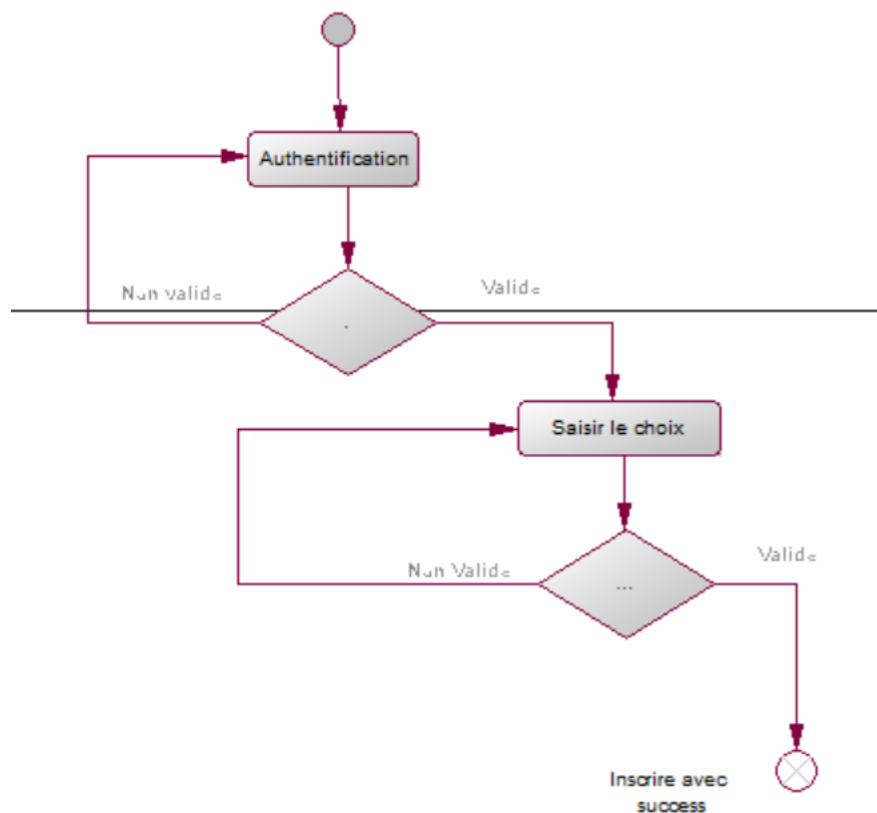
Chercher stage:



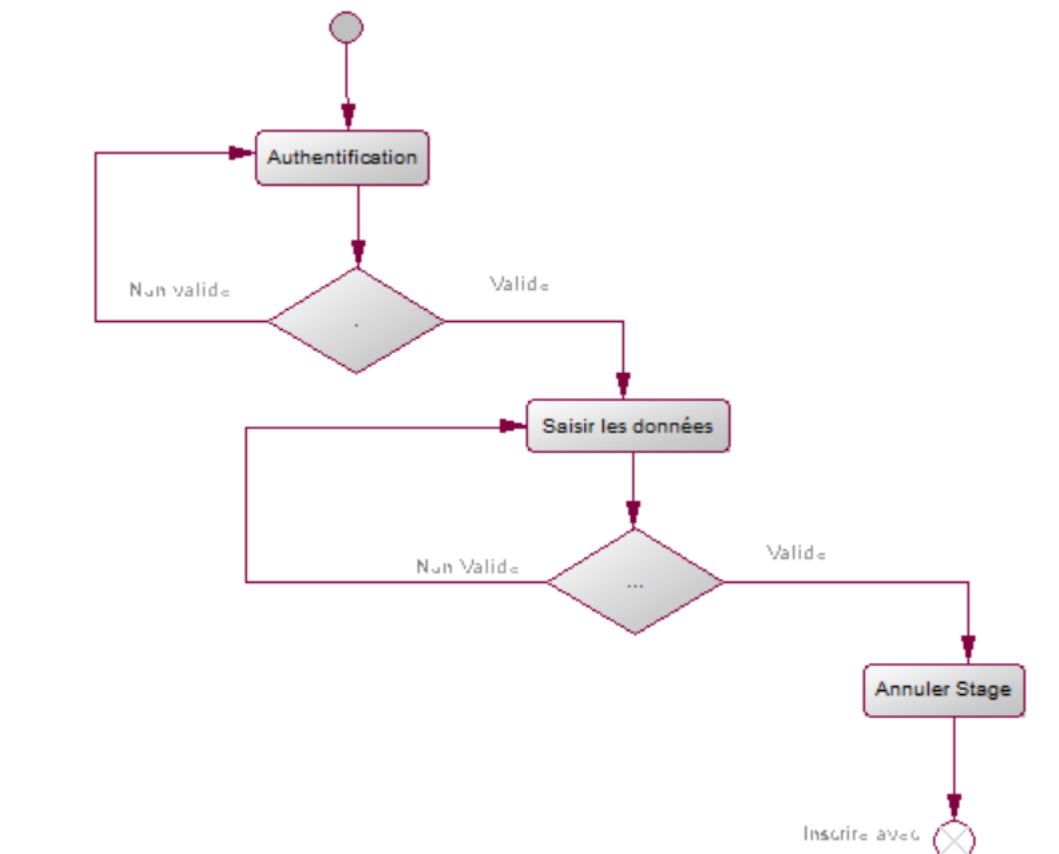
Mettre à jour :



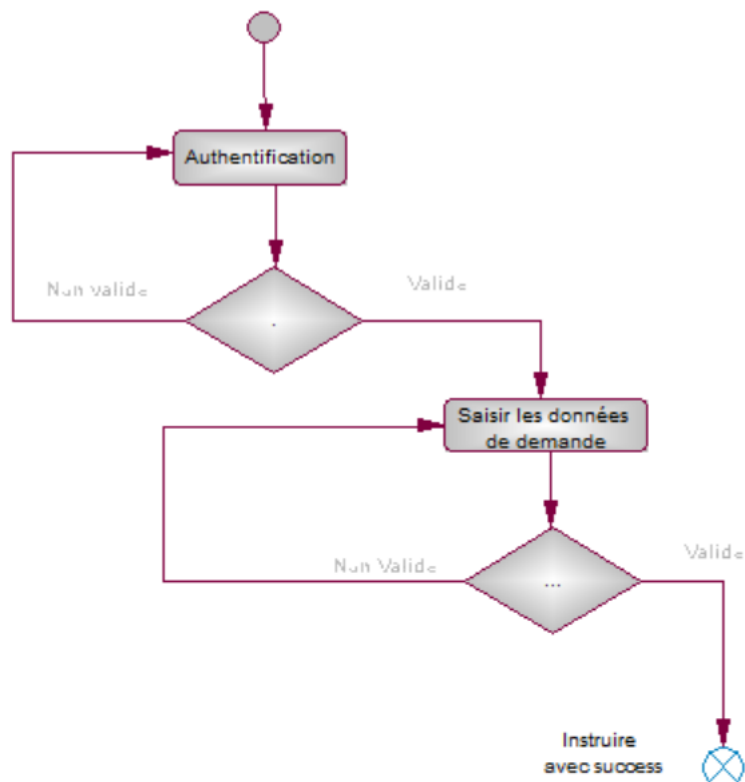
Inscrire Etudiant :



Annuler inscription :

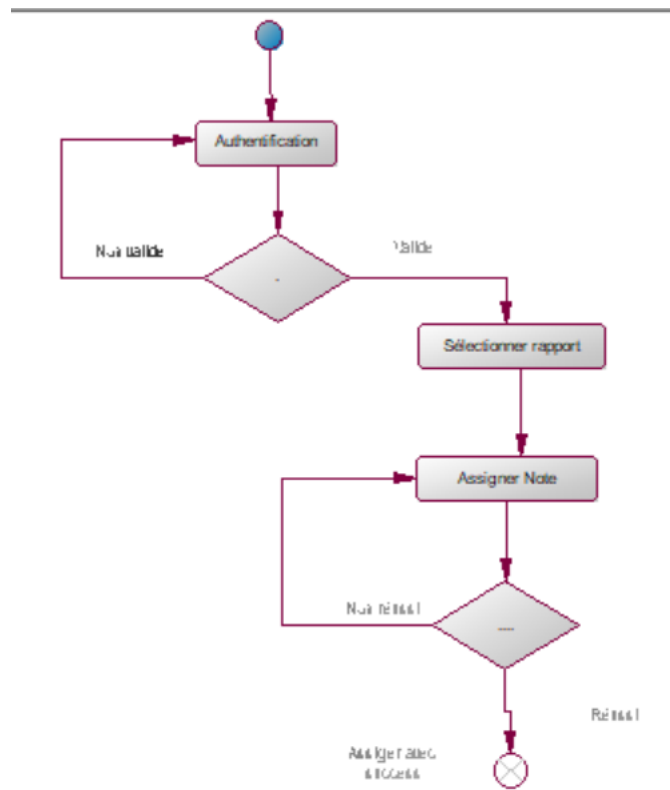


Instruire demande:



## Jury

Assigner note stage :



# CONCLUSION

Ce projet nous a permis de bien maîtriser les notions de UML et principalement la Modélisation Orienté Objet. C'est la première fois qu'on modélise du haut niveau jusqu'au bas niveau un processus d'application. On a bien compris l'intérêt de la POO et le fait de penser « en POO » ; mettre en place seulement les classes nécessaires et identifier les besoins fonctionnels, en se basant sur la méthode de minimale, et savoir attribuer précisément les responsabilités de comportement. On a rencontré quelques problèmes dans la distinction entre les classes de dialogue et de contrôle et dans la considération des classes d'entités comme tables de base de données. Mais on a pu les surmonter grâce aux diagrammes d'interaction qui sont un peu plus détaillés.

En guise de conclusion, nous avons bien mis les choses acquises, concernant l'UML, en place. On a réussi à modéliser une application de processus de stage liant plusieurs parties.

Finalement, nous tenons aussi à remercier Prof Nidal Lamghari pour tous ces efforts fournis durant ce semestre, et de cette opportunité d'appliquer tous ce qu'on sait à propos des méthodes d'analyse et de conception.