République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Université M'Hamed Bougara - Boumerdès



Faculté des Sciences
Département d'Informatique

Domaine : Mathématiques et Informatique
Filière : Informatique
Spécialité : système informatique

Mémoire de fin d'études en vue de l'obtention du diplôme de Licence

Thème

Conception et réalisation d'une plateforme web pour la gestion des stages et stagiaires

Présenté par :
Roufaida Aouissi
Meriem Aoudache
Khaoula Khelifi

Stage pratique réalisé à : Division Explortion de Sontrach à Boumerdès
Soutenu le : ../../2025

Mme AKROUR Djouher	Examinateur
Mr MOKRANI Hocine	Encadreur
Mme OURAGH Naouel	Promoteur

Dédicace

À notre équipe, pour notre persévérance, notre travail acharné et notre engagement tout au long de ce projet.

À nos familles, pour leur soutien et leur encouragement constants.

À tous ceux qui nous ont aidées de près ou de loin.

Ensemble, nous avons surmonté les défis et accompli ce travail avec passion et détermination.*Nous dédions ce mémoire :*

À notre équipe, pour notre persévérance, notre travail acharné et notre engagement tout au long de ce projet.

À nos familles, pour leur soutien et leur encouragement constants.

À tous ceux qui nous ont aidées de près ou de loin.

Ensemble, nous avons surmonté les défis et accompli ce travail avec passion et détermination.

Roufaida Aouissi
Meriem Aoudache
Khaoula Khelifi

Table des matières

Liste des figures **4**

Liste des tableaux **5**

Introduction Générale **6**

1 Présentation de l'organisme d'accueil **7**
1.1 Introduction . . . 7
1.2 Présentation de Sonatrach . . . 7
1.3 Présentation de la division Exploration . . . 8
1.4 Problématique du système . . . 8
1.5 Objectifs du système . . . 9
1.6 Conclusion . . . 9

2 Étude de l'existant **10**
2.1 Introduction . . . 10
2.2 Vocabulaire de l'existant . . . 10
2.2.1 Champs d'étude . . . 10
2.2.2 L'étude du flux d'information . . . 10
2.2.3 Étude des postes de travail . . . 11
2.2.4 Étude des documents . . . 12
2.3 Conclusion . . . 15

3 Analyse des besoins et conception **16**
3.1 Introduction . . . 16
3.2 Présentation de la méthode utilisée . . . 16
3.3 Analyse et spécification des besoins . . . 16
3.3.1 Besoins fonctionnels . . . 16
3.3.2 Besoins non fonctionnels . . . 17
3.4 Diagramme de cas d'utilisation . . . 17
3.4.1 Identification des acteurs . . . 17
3.4.2 Diagramme de cas d'utilisation général du système . . . 18
3.4.3 Description des cas d'utilisation . . . 20
3.5 Diagramme de séquence . . . 20
3.5.1 authentification . . . 21
3.5.2 Ajouter . . . 22
3.5.3 Modifier . . . 23
3.5.4 Consulter . . . 24
3.5.5 Supprimer . . . 25
3.6 diagramme de classe . . . 26
3.6.1 Élaboration d'un diagramme de classes . . . 26
3.6.2 Règles de gestion . . . 27
3.7 Dictionnaire de données . . . 27
3.7.1 Transformation du modèle Objet au modèle relationnel . . . 28
3.7.2 Modèle relationnel . . . 28
3.8 Conclusion . . . 29

4 Implémentation **30**
4.1 Introduction . . . 30
4.2 MVC . . . 30
4.3 Diagramme de déploiement . . . 30
4.4 Outils retenus pour le développement de l'application . . . 30
4.4.1 Environnement hardware . . . 31
4.4.2 Environnement software . . . 31
4.5 Présentation des interfaces graphiques . . . 32
4.5.1 Page de Login . . . 32
4.5.2 Page Liste des sujets . . . 32
4.5.3 Page de Modification . . . 33
4.5.4 Page de Rendez-vous . . . 33
4.6 Conclusion . . . 33

Conclusion Générale **34**

Bibliographie **35**

Table des figures

1.1 Logo de Sonatrach 7
1.2 Organigramme général de Sonatrach 8
1.3 Organigramme de la division Exploration 8

2.1 Diagramme de flux des données 11
2.2 Les différentes interfaces du système 12

3.1 Diagramme de cas d'utilisation général du système 19
3.2 Diagramme de séquence : Authentification 21
3.3 Diagramme de séquence :Ajouter 22
3.4 Diagramme de séquence : Modifier 23
3.5 Diagramme de séquence :Consulter 24
3.6 Diagramme de séquence :Supprimer 25
3.7 Diagramme de classes du système 27

4.1 Diagramme de déploiement de l'application 30
4.2 page de login 32
4.3 Page de liste des sujets 33
4.4 Page de modification 33
4.5 Page de Rendez-vou 33

Liste des tableaux

2.1 Éléments du diagramme de flux d'information 10
2.2 Composants des documents utilisés 12
2.3 Liste des documents utilisés 12
2.4 Fiche d'identification de sujet de PFE 13
2.5 Éléments d'information de la fiche d'identification de sujet de PFE 13
2.6 Convention de stage 13
2.7 Éléments d'information de la convention de stage 14
2.8 Recommandation pour un stage pratique 14
2.9 Éléments d'information de la recommandation de stage 14
2.10 Autorisation d'accès au siège 15
2.11 Éléments d'information de l'autorisation d'accès 15

3.1 Symboles du diagramme de cas d'utilisation 17
3.2 Description des cas d'utilisation 20
3.3 Les éléments d'un diagramme de séquence 21
3.4 Description de diagramme de séquence d'Authentification 22
3.5 Description de diagramme de séquence d' Ajouter 23
3.6 Description de diagramme de séquence de Modifier 24
3.7 Description de diagramme de séquence de Consulter 25
3.8 Description de diagramme de séquence de Supprimer 26
3.9 Dictionnaire de données 28

Introduction Générale

Avec la transformation numérique actuelle, les organisations sont de plus en plus confrontées à la nécessité de gérer efficacement leurs informations et leurs activités. En effet, la digitalisation des processus apparaît aujourd'hui comme un levier essentiel pour améliorer la performance, la fiabilité et la rapidité des services proposés.

Dans les méthodes traditionnelles, la gestion des stages et des stagiaires repose souvent sur des procédures manuelles, ce qui engendre plusieurs difficultés, notamment la perte ou la redondance de données, les erreurs de traitement, ainsi qu'un manque de coordination entre les différents intervenants. Ces limites rendent la gestion des stages complexe, chronophage et peu efficace, ce qui nuit à la qualité de l'accompagnement des stagiaires ainsi qu'à la transparence administrative.

Dans ce cadre, notre projet s'inscrit dans une démarche de modernisation et d'optimisation à travers la conception et la réalisation d'une plateforme web dédiée à la gestion des stages et des stagiaires. Cette solution vise à centraliser les informations relatives aux stagiaires, aux encadreurs et aux tutelles, à automatiser les différentes tâches administratives, à faciliter le suivi des stagiaires tout au long de leur parcours, et à améliorer la communication entre les acteurs concernés, à savoir les stagiaires, les encadreurs et l'administration.

Ce mémoire est structuré en quatre chapitres complémentaires. Le premier chapitre est consacré à la présentation de l'organisme d'accueil. Le deuxième chapitre traite de l'étude de l'existant ainsi que de la spécification détaillée des besoins. Le troisième chapitre est dédié à l'analyse et à la conception du système, tandis que le quatrième et dernier chapitre présente la réalisation de l'application, ainsi que les outils et technologies utilisés.

Chapitre 1

Présentation de l'organisme d'accueil

1.1 Introduction

Ce chapitre présente l'organisme d'accueil Sonatrach– Division Exploration, ses services principaux et annexes, ainsi que le contexte dans lequel le projet de fin d'études a été réalisé. Il décrit également les services d'exploitation, leurs organigrammes, et permet de comprendre la problématique qui a motivé le développement du site web de gestion des stages et des stagiaires. Enfin, ce chapitre définit les objectifs poursuivis par le projet.

1.2 Présentation de Sonatrach

Sonatrach, créée le 31 décembre 1963, est la compagnie nationale algérienne d'hydrocarbures et constitue la première entreprise en Afrique [1]. Elle intervient sur l'ensemble de la chaîne de valeur du secteur énergétique, notamment l'exploration, la production, le transport, la liquéfaction, le raffinage, la pétrochimie ainsi que la commercialisation.

Sur le plan économique, Sonatrach joue un rôle stratégique en assurant la sécurité énergétique de l'Algérie et en contribuant à une part importante des revenus fiscaux du pays. À l'échelle internationale, elle occupe une position majeure en tant que fournisseur de gaz, notamment vers l'Italie et l'Espagne.

Par ailleurs, le groupe s'appuie sur plusieurs filiales importantes, telles que Naftal, ENTP et Hyproc SC, qui participent activement à ses différentes activités.

En raison de son importance et de son influence, Sonatrach est souvent surnommée la « major africaine ».

Son logo est présenté dans la figure suivante.



FIGURE 1.1 – Logo de Sonatrach

Sonatrach dispose de plusieurs services clés qui garantissent le bon fonctionnement et la coordination efficace au sein de l'entreprise [3]. Ces services sont les suivants :

— **Formation Professionnelle :** Développement des compétences pour les carrières dans le secteur du pétrole et du gaz.
— **Recherche et Développement :** Expertise technique et recherche dans le secteur de l'énergie.

— **Formation Continue :** Programmes adaptés aux professionnels du secteur des hydrocarbures.
— **Accueil International :** Formation des apprenants venant principalement d'Afrique.

Organigramme de Sonatrach L'organigramme de Sonatrach montre les principales divisions de l'entreprise : Exploration, Production, Laboratoires, ainsi que les services de support et d'administration. Il permet de comprendre la répartition des responsabilités et la coordination des activités, y compris la gestion des stagiaires comme illustré dans la Figure 2.

Dans le cadre de notre projet, cet organigramme a été utilisé comme référence afin d'identifier les différents services concernés par la gestion des stages et de mieux structurer les modules développés dans l'application. Les parties mises en évidence en couleur (jaune) représentent les sections prises en charge et implémentées dans notre système.

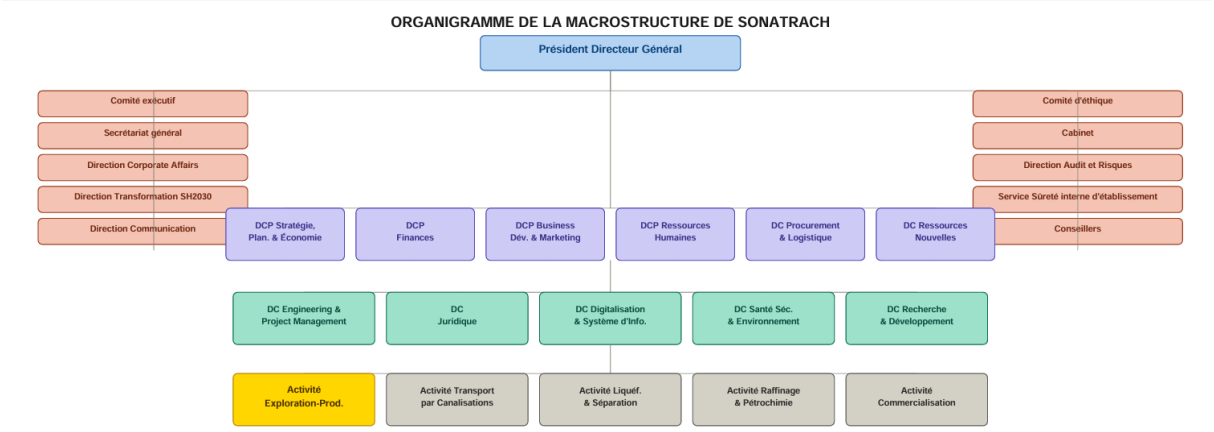


FIGURE 1.2 – Organigramme général de Sonatrach

1.3 Présentation de la division Exploration

La division Exploration de Sonatrach est responsable de la recherche et de la découverte de nouveaux gisements d'hydrocarbures. Elle réalise des études géologiques, effectue des opérations de forage et évalue les ressources disponibles. Son objectif principal est d'assurer le renouvellement des réserves et de soutenir la production énergétique nationale [3]. **Organigramme de la division Exploration**

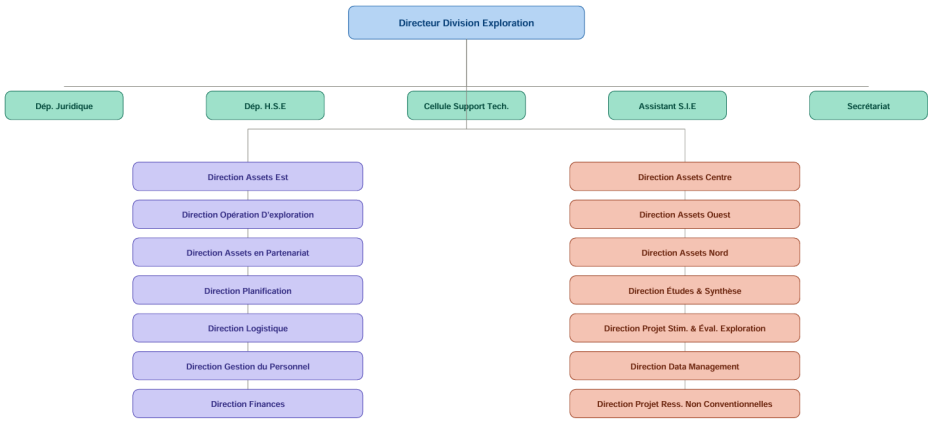


FIGURE 1.3 – Organigramme de la division Exploration

1.4 Problématique du système

La gestion des stagiaires au sein de l'organisation présente plusieurs difficultés qui affectent le suivi des stages et l'organisation du travail. L'identification de ces problèmes est essentielle pour comprendre

les limites du système actuel. Les principaux problèmes sont :

- Suivi manuel des dates de début et de fin des stages.
- Difficulté à connaître le nombre exact de stagiaires actifs.
- Difficulté d'associer chaque stagiaire à son encadrant.
- Risque de chevauchement des sujets entre les groupes (DI/FAC).

1.5 Objectifs du système

Afin d'améliorer la gestion des stagiaires, il est nécessaire de mettre en place un système efficace et fiable. Les objectifs principaux sont :

- Assurer un suivi automatique des dates de début et de fin des stages.
- Fournir en temps réel le nombre de stagiaires actifs.
- Faciliter l'association entre stagiaires et encadrants.
- Éviter le chevauchement des sujets entre les groupes et assurer une répartition correcte en permettant à chaque encadrant de connaître les sujets proposés par les autres encadrants.

1.6 Conclusion

En résumé, ce chapitre a permis de présenter l'organisme d'accueil **Sonatrach – Division Exploration**, ses services principaux et d'exploitation, ainsi que les organigrammes correspondants. Il a également mis en évidence les besoins et les problèmes qui ont motivé le développement du site web de gestion des stages et des stagiaires. Cette présentation constitue une base solide pour l'étude de l'existant et l'analyse des besoins dans les chapitres suivants.

Chapitre 2

Étude de l'existant

2.1 Introduction

Dans le but de bien appréhender ce projet, il a été nécessaire d'étudier un ensemble de notions fondamentales liées à la gestion des stages et des stagiaires. Ainsi, ce chapitre est consacré à la présentation des systèmes d'information, des principes des processus métiers, ainsi que des outils et technologies intervenant dans la mise en œuvre d'une plateforme web dédiée à ce domaine.

2.2 Vocabulaire de l'existant

2.2.1 Champs d'étude

Il s'agit de l'ensemble des postes de travail au sein desquels sont exécutées les tâches du système à étudier [5]. Dans notre cas, nous avons retenu un échantillon relevant de la direction « Développement web » de la division Exploration.

2.2.2 L'étude du flux d'information

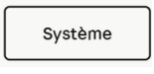
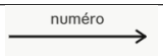
Flux d'information : Le flux d'information est une information échangée entre les acteurs d'un système afin d'assurer la communication et le bon fonctionnement [6].

Diagramme des flux (DF) : Le diagramme des flux est une représentation graphique qui illustre les différents flux d'information au sein d'une organisation lors d'une activité donnée [6].

Acteur : Entité capable d'émettre ou de recevoir un flux d'information, interne ou externe au système [6].

Le tableau 2.1 montre les différents éléments du diagramme de flux d'information.

TABLE 2.1 – Éléments du diagramme de flux d'information

Élément	Symbole	Description
Entité externe		Source ou destination des informations échangées avec le système.
Entité interne		Élément interne du système participant au traitement et à la circulation des informations.
Flux de données		ReprésCapture d'écran 2026-05-26 005244 (1).pngente le déplacement des informations entre les éléments du diagramme.
Stock de données		Référentiel où les informations sont conservées.

La figure 2.1 présente le diagramme de flux d'information du système de gestion des stages et des stagiaires.

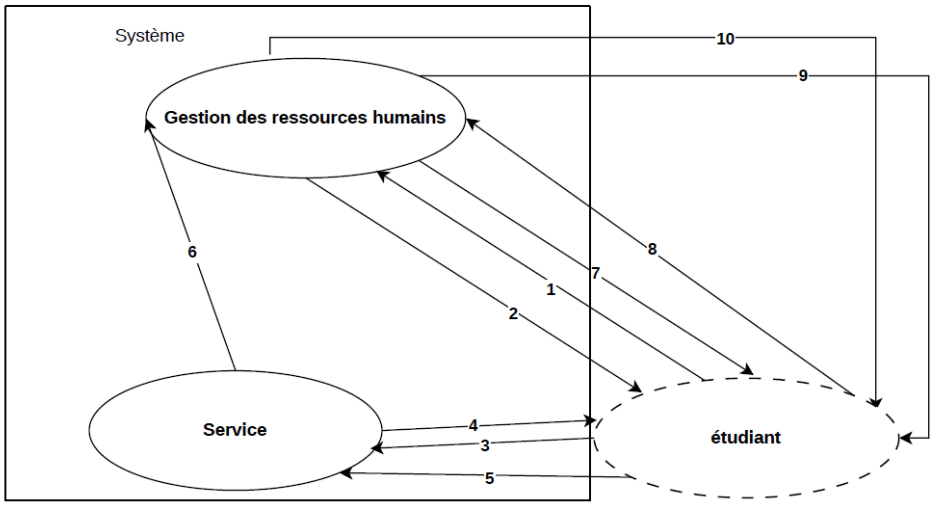


FIGURE 2.1 – Diagramme de flux des données

Les flèches représentent les flux d'information suivants :

1. Demande de stage
2. Orientation de l'étudiant vers un service spécialisé selon sa spécialité
3. Demande de sujet de stage
4. Proposition de sujets
5. Dépôt du sujet validé selon les exigences de l'université
6. Validation du thème
7. Délivrance de la convention accompagnée des dossiers requis
8. Remise de la convention signée conformément aux exigences de l'université
9. Convention signée validée par le GRH
10. Fourniture de l'autorisation d'accès

2.2.3 Étude des postes de travail

En organisation du travail, un poste de travail désigne un ensemble de tâches et de responsabilités attribuées à un acteur, disposant des ressources nécessaires pour les accomplir dans un cadre organisationnel donné.

Dans cette étude, nous nous intéressons aux postes de travail internes impliqués dans la gestion des stages, à savoir le service des ressources humaines (GRH) et le service spécialisé.

- **GRH (Gestion des Ressources Humaines) :** Le service GRH est chargé de la gestion administrative des stages au sein de l'organisation. Il assure la coordination entre les différents acteurs et veille au respect des procédures. Il est chargé de :
  - Délivrer la convention accompagnée des dossiers requis
  - Valider la convention signée
  - Fournir l'autorisation d'accès
  - Orienter l'étudiant vers un service spécialisé selon sa spécialité
- **Service spécialisé :** Le service spécialisé assure l'encadrement technique des stagiaires et propose des sujets adaptés à leur domaine d'étude. Il est chargé de :
  - Proposer des sujets de stage
  - Valider le thème de stage

2.2.4 Étude des documents

Cette étude concerne l'analyse des documents utilisés au sein de la structure d'étude. Le but de cette étude est de faciliter la collecte des informations manipulées dans chaque service et d'obtenir un référentiel de données utilisé lors de l'analyse des besoins et de la conception.

A. Les composants des divers documents utilisés :

TABLE 2.2 – Composants des documents utilisés

Élément	**Description**
Code, désignation et rôle	Identification et objectif du document
Nature	Interne ou externe
Fréquence	Fréquence d'établissement du document
Nombre d'exemplaires	Nombre de copies du document
Provenance	Source du document
Destination	Destinataire du document
Archivage	Lieu d'archivage du document
Désignation	Nom de l'élément d'information
Type	A : alphabétique, N : numérique, AN : alphanumérique, D : date
Nature des données	Élémentaire ou calculée
Taille	Taille de l'information
Observation	Informations complémentaires

B. Liste des divers documents utilisés :

TABLE 2.3 – Liste des documents utilisés

N°	**Document**
1	Recommandation pour un stage pratique
2	Fiche d'identification de sujet de projet de fin d'études
3	Convention de stage
4	Autorisation d'accès au siège

Les figures suivantes présentent quelques exemples des documents utilisés dans le système de gestion des stages et des stagiaires.

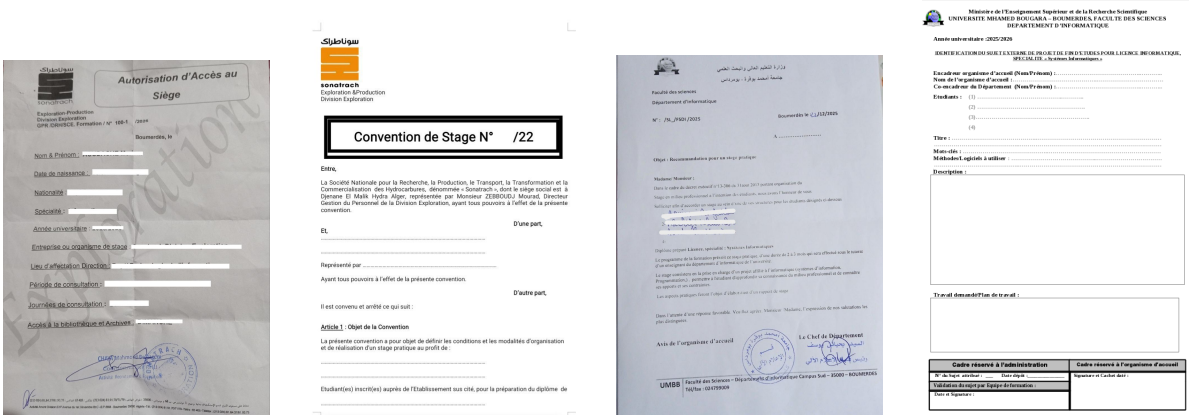


FIGURE 2.2 – Les différentes interfaces du système

Dans la gestion des stages et des stagiaires, nous trouvons les documents suivants :

• **Fiche d'identification de sujet de PFE**

TABLE 2.4 – Fiche d'identification de sujet de PFE

Élément	**Description**
Code	C.F.I.P
Désignation	Canevas de fiche d'identification de sujet de PFE pour licence informatique
Rôle	Servir de modèle type pour formaliser les propositions de sujets de PFE
Nature	Externe
Fréquence	À chaque début de projet de fin d'études
En-tête	Logo, identification du département et de la spécialité
Nombre d'exemplaires	2
Provenance	Département informatique
Destinataire	Organisme d'accueil
Archivage	Département + Organisme d'accueil

TABLE 2.5 – Éléments d'information de la fiche d'identification de sujet de PFE

Désignation	**Type**	**Taille**	**Nature**	**Observation**
Nom et prénom de l'encadreur organisme	A	50	E	
Nom de l'organisme d'accueil	A	50	E	
Nom et prénom du co-encadreur département	A	50	E	
Nom et prénom de l'étudiant (1,2,3)	A	50 chacun	E	
Titre du sujet	AN	200	E	Conception d'une plateforme web pour la gestion des stages et stagiaires
Mots-clés	AN	100	E	Framework, Bootstrap, Laravel
Méthodologie / Logiciels	AN	200	E	HTML, CSS, JS, PHP, MySQL, UML
Modules à développer	AN	300	E	Sujets, thèmes, stages, utilisateurs, édition, dashboard
Travail demandé / Plan	AN	300	E	Cas d'utilisation, séquence, conception BDD, développement
N° du sujet attribué	AN	15	E	À compléter
Date de dépôt	D	10	E	À compléter
Validation par équipe de formation	A	50	E	
Signature et cachet (organisme)	A	50	E	
Signature et cachet (département)	A	50	E	

• **Convention de stage**

TABLE 2.6 – Convention de stage

Élément	**Description**
Code	C.D.S
Désignation	Convention de stage
Rôle	Définir les conditions et modalités d'organisation d'un stage pratique entre Sonatrach et l'organisme de formation
Nature	Externe (Sonatrach et organisme de formation)
Fréquence	À chaque début de stage
En-tête	Logo et nom de l'entreprise
Nombre d'exemplaires	3 (selon l'article 11)
Provenance	Sonatrach (Division Exploration)
Destinataire	Organisme de formation
Archivage	Sonatrach et organisme de formation

TABLE 2.7 – Éléments d'information de la convention de stage

Désignation	**Type**	**Taille**	**Nature**	**Observation**
Nom et prénom du représentant de Sonatrach	A	50	E	Directeur Gestion du Personnel
Nom et prénom du représentant de l'organisme	A	50	E	À compléter
Diplôme préparé	AN	100	E	Objet du stage
Lieu du stage	AN	100	E	Boumerdès – Division Exploration
Date de début de stage	D	10	E	À compléter
Date de fin de stage	D	10	E	À compléter
Nom de l'encadreur (Sonatrach)	A	50	E	Désigné par l'entreprise
Thème du stage	AN	200	E	Objectif pédagogique
Obligations du stagiaire	AN	500	E	Texte descriptif
Obligations de l'organisme de formation	AN	500	E	Texte descriptif
Obligations de Sonatrach	AN	500	E	Délivrance attestation, etc.
Date de signature	D	10	E	Lieu et date
Signature du représentant Sonatrach	A	50	E	M. ZEBBOUDJ
Signature du représentant organisme	A	50	E	À compléter

• **Recommandation pour un stage pratique**

TABLE 2.8 – Recommandation pour un stage pratique

Élément	**Description**
Code	R.S.P
Désignation	Recommandation pour un stage pratique
Rôle	Communiquer des directives officielles concernant un stage dans le cadre d'un décret exécutif
Nature	Externe (Sonatrach et organisme de formation)
Fréquence	À chaque recommandation de stage
En-tête	Logo, émetteur, numéro de référence, date et lieu d'émission
Nombre d'exemplaires	1
Provenance	Département d'Informatique
Destinataire	Organisme d'accueil
Archivage	Organisme d'accueil + Département d'Informatique

TABLE 2.9 – Éléments d'information de la recommandation de stage

Désignation	**Type**	**Taille**	**Nature**	**Observation**
Numéro de document	AN	15	E	
Date du document	D	10	E	
Objet	AN	50	E	Recommandation pour un stage pratique
Destinataire	A	50	E	Madame/Monsieur
Référence réglementaire	AN	50	E	Décret exécutif n° 13-306 du 31 août 2013
Contexte	AN	100	E	
Nom et prénom de l'étudiant (1,2,3)	A	50 chacun	E	
Signature et cachet (chef département)	A	50	E	
Signature et cachet (organisme d'accueil)	A	50	E	
Département	A	50	E	Faculté des Sciences – Département d'Informatique
Numéro de téléphone	N	10	E	0247799099

• **Autorisation d'accès au siège**

TABLE 2.10 – Autorisation d'accès au siège

Élément	**Description**
Code	A.A.S
Désignation	Autorisation d'accès au siège
Rôle	Autoriser un stagiaire à accéder aux locaux de Sonatrach pour une période déterminée
Nature	Interne
Fréquence	À chaque début de stage ou période de consultation
En-tête	Logo et nom de l'entreprise (Exploration-Production / Division Exploration)
Nombre d'exemplaires	1
Provenance	Division Exploration (Service Formation)
Destinataire	Stagiaire
Archivage	DRH + Stagiaire

TABLE 2.11 – Éléments d'information de l'autorisation d'accès

Désignation	**Type**	**Taille**	**Nature**	**Observation**
Numéro de document	AN	20	E	GPR/DRH/SCE. Formation / N° 100-2 /2026
Date d'émission	D	10	E	Boumerdès, le (à compléter)
Nom et prénom du stagiaire	A	50	E	À compléter
Date de naissance	D	10	E	À compléter
Nationalité	A	20	E	
Spécialité	A	50	E	Informatique
Année universitaire	AN	10	E	2025/2026
Entreprise ou organisme de stage	A	50	E	Sonatrach Division Exploration
Lieu d'affectation (Direction)	A	50	E	Département Technologie de l'information
Période de consultation	D	20	E	du 01/03/2026 au 31/05/2026
Journées de consultation	A	15	E	Dimanche
Accès à la bibliothèque et archives	A	15	E	Dimanche

2.3 Conclusion

Cette section a permis de réaliser une étude préliminaire du projet afin de mieux saisir les enjeux de la gestion des stages et des stagiaires, ainsi que les principales contraintes liées au développement. La partie suivante sera consacrée à l'analyse des besoins et à la conception de l'application, en présentant la démarche méthodologique adoptée pour modéliser le système.

Chapitre 3

Analyse des besoins et conception

3.1 Introduction

Tout projet informatique nécessite une phase d'analyse des besoins suivie d'une étape de conception. Dans ce chapitre, nous présentons l'étude des besoins et la conception de notre application de gestion des stages et des stagiaires, afin de définir les fonctionnalités du système et les interactions des utilisateurs.

3.2 Présentation de la méthode utilisée

Afin de mener à bien notre projet et de répondre efficacement aux besoins des utilisateurs, il est indispensable d'adopter une méthode de travail permettant de structurer l'analyse et la conception du système.

Dans ce contexte, nous avons choisi d'utiliser le langage de modélisation UML (Unified Modeling Language), qui constitue un outil standard pour représenter les différents aspects d'un système informatique. UML permet de décrire de manière claire et graphique les fonctionnalités du système ainsi que les interactions entre ses acteurs [7].

Par ailleurs, pour organiser le processus de développement, nous nous sommes appuyés sur la méthode RUP (Rational Unified Process), qui est une approche itérative et incrémentale facilitant la gestion du cycle de vie du projet [8].

Le processus RUP se décompose en quatre phases principales :

— **Inception :** Identification des besoins et étude de faisabilité du projet.
— **Élaboration :** Analyse détaillée des besoins et conception de l'architecture du système.
— **Construction :** Développement et test des différentes fonctionnalités du système.
— **Transition :** Déploiement du système et validation finale auprès des utilisateurs.

3.3 Analyse et spécification des besoins

Dans cette section, nous allons présenter les différents besoins de notre application de gestion des stages et des stagiaires, en distinguant les besoins fonctionnels et les besoins non fonctionnels.

3.3.1 Besoins fonctionnels

Les besoins fonctionnels représentent les services que le système doit fournir aux utilisateurs. Dans notre cas, le système doit permettre :

— La gestion des stages : permet d'organiser et de planifier les différentes périodes de stage au sein de l'organisme.
— La gestion des stagiaires : consiste à enregistrer et suivre les informations tout au long de leur stage.
— La gestion des documents (fiche, convention, autorisation...) : assure la création, le stockage et la gestion des différents documents administratifs liés aux stages.
— L'attribution des sujets de stage : permet d'affecter à chaque stagiaire un sujet de stage selon son domaine et sa spécialité.

— Le suivi des stages : permet de contrôler l'évolution du stage et d'évaluer le travail réalisé par les stagiaires.

3.3.2 Besoins non fonctionnels

Les besoins non fonctionnels définissent les contraintes et les exigences de qualité du système. Ils concernent notamment :

— La sécurité des données : garantit la protection des informations contre tout accès non autorisé et assure leur confidentialité et leur intégrité.
— La performance du système : assure un temps de réponse rapide et une exécution efficace des différentes fonctionnalités de la plateforme.
— La facilité d'utilisation (ergonomie) : permet aux utilisateurs d'interagir grâce à une interface claire, intuitive et facile à manipuler.
— La fiabilité : garantit le bon fonctionnement du système de manière continue, avec un minimum d'erreurs ou de pannes.

3.4 Diagramme de cas d'utilisation

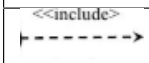
Le diagramme de cas d'utilisation est une représentation graphique appartenant au langage UML, utilisée pour illustrer les différentes fonctionnalités d'un système selon la vision de l'utilisateur. Il permet de montrer comment les acteurs interagissent avec le système afin de satisfaire leurs besoins [9].

Un cas d'utilisation sert à :

— Donner une vision claire et simplifiée du fonctionnement global du système.
— Présenter les différentes étapes suivies lors de l'exécution d'une fonctionnalité.
— Mettre en évidence les liens existants entre les cas d'utilisation (comme include, extend ou généralisation).
— Aider à identifier les contraintes et les exigences nécessaires pour la réalisation du système.

Le tableau 3.1 présente les symboles du diagramme de cas d'utilisation.

TABLE 3.1 – Symboles du diagramme de cas d'utilisation

Symbole	Nom	Description
	Acteur	Représente une entité externe qui interagit avec le système
	Cas d'utilisation	Représente une fonctionnalité du système
	Association	Représente la relation entre un acteur et un cas d'utilisation
	Inclusion	Un cas inclut un autre
	Extension	Comportement optionnel ajouté
	Généralisation	Relation d'héritage

3.4.1 Identification des acteurs

L'identification des acteurs constitue une étape importante dans l'élaboration du diagramme de cas d'utilisation. Un acteur représente toute entité externe qui interagit avec le système.

D'après le diagramme de cas d'utilisation présenté, nous avons identifié les acteurs suivants :

— **GRH (Gestion des Ressources Humaines) :** Il est responsable de la gestion administrative des stages, notamment la validation des conventions, la gestion des stagiaires et la délivrance des autorisations.
— **Encadreur :** Il intervient dans le suivi des stagiaires, la proposition des sujets et la validation des thèmes de stage.

3.4.2 Diagramme de cas d'utilisation général du système

La figure 3.1 présente le diagramme de cas d'utilisation général du système. Il permet d'identifier les différentes interactions entre le système et les acteurs, ainsi que les fonctionnalités offertes par le système.

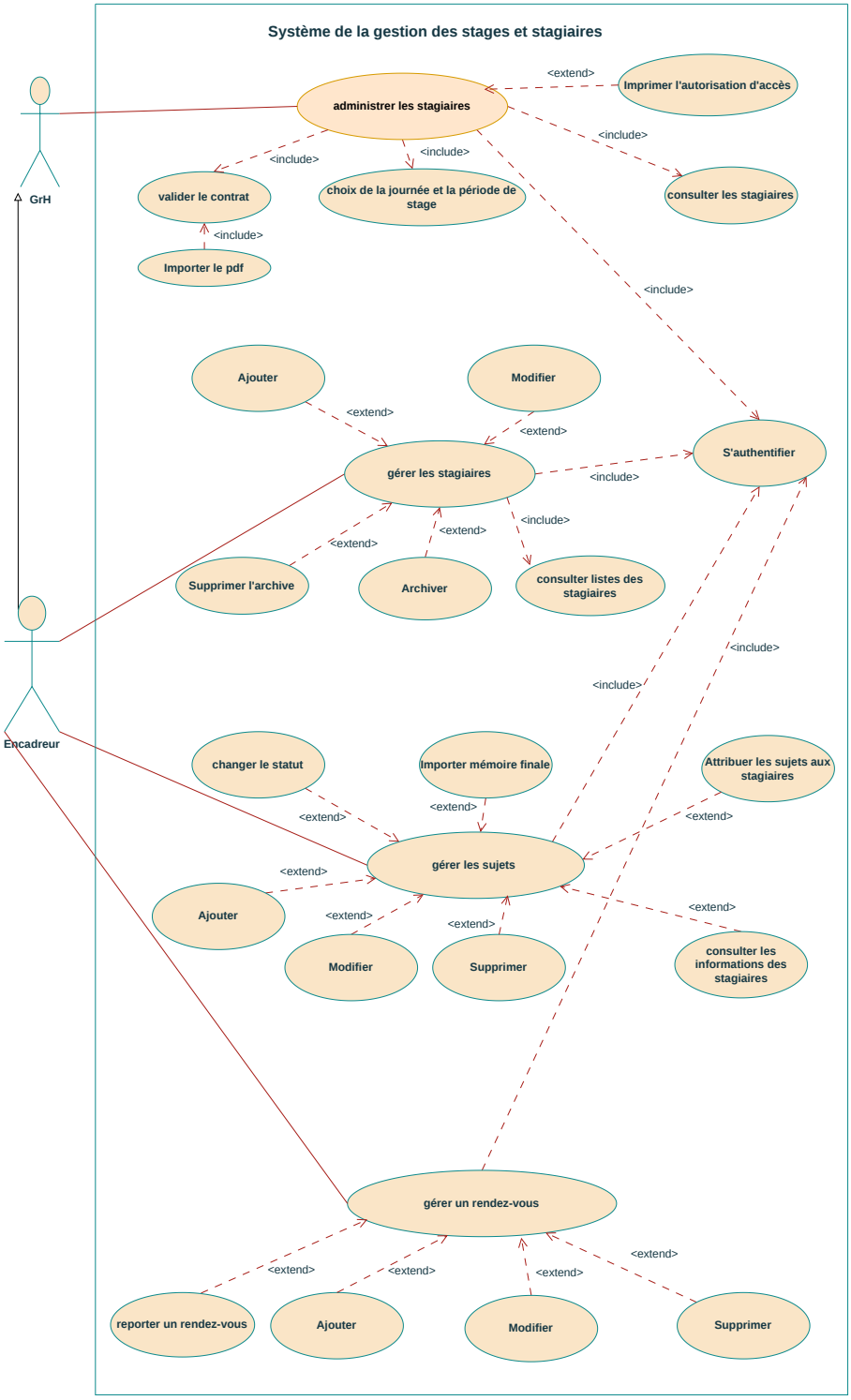


FIGURE 3.1 – Diagramme de cas d'utilisation général du système

3.4.3 Description des cas d'utilisation

Ce tableau présente les tâches des différents acteurs.

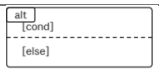
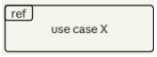
TABLE 3.2 – Description des cas d'utilisation

Acteur	**Cas d'utilisation**	**Description**
GRH	administrer les stagiaires	Permet au GRH de gérer l'ensemble des informations liées aux stagiaires.
	valider le contrat	Permet au GRH de vérifier et valider le contrat de stage soumis par le stagiaire.
	choix de la journée et de la période de stage	Permet de définir la période et les jours de stage du stagiaire.
	consulter les stagiaires	Permet au GRH de visualiser la liste des stagiaires enregistrés dans le système.
	importer le PDF	Permet d'importer un document PDF dans le système.
	imprimer l'autorisation d'accès	Permet de générer et imprimer une autorisation d'accès pour le stagiaire.
Encadreur	ajouter stagiaire	Permet d'ajouter un nouveau stagiaire dans le système.
	modifier stagiaire	Permet de modifier les informations d'un stagiaire existant.
	archiver stagiaire	Permet d'archiver un stagiaire.
	supprimer archive	Permet de supprimer définitivement un stagiaire archivé.
	consulter liste des stagiaires	Permet d'afficher la liste complète des stagiaires avec leurs informations.
	ajouter sujet	Permet d'ajouter un nouveau sujet de stage.
	modifier sujet	Permet de modifier les informations d'un sujet existant.
	supprimer sujet	Permet de supprimer un sujet de stage.
	changer le statut	Permet de modifier l'état du sujet (accepté, refusé, en attente...).
	importer mémoire final	Permet d'ajouter le mémoire final du stagiaire dans le système.
	attribuer sujet aux stagiaires	Permet d'affecter un ou plusieurs stagiaires à un sujet de stage.
	consulter informations stagiaires	Permet à l'encadreur de voir les détails des stagiaires.
	ajouter rendez-vous	Permet de planifier un nouveau rendez-vous.
	modifier rendez-vous	Permet de modifier les détails d'un rendez-vous existant.
	supprimer rendez-vous	Permet d'annuler un rendez-vous.
	reporter rendez-vous	Permet de déplacer un rendez-vous à une autre date.
Système	s'authentifier	Permet à l'utilisateur (GRH ou encadreur) de se connecter au système en saisissant ses identifiants afin d'accéder aux différentes fonctionnalités.

3.5 Diagramme de séquence

Le diagramme de séquence est un diagramme dynamique du langage UML permettant de représenter les interactions entre les différents acteurs et objets d'un système selon un ordre chronologique. Il met en évidence les messages échangés entre les participants d'un scénario, ainsi que les différentes structures de contrôle pouvant intervenir durant l'exécution du système.

TABLE 3.3 – Les éléments d'un diagramme de séquence

Élément	Symbole	Description
Ligne de vie (Lifeline)		Représente un acteur ou un objet participant à l'interaction.
Messages		Représentent les échanges de communication entre les objets du système.
Flèche continue		Indique l'envoi d'un message ou d'une requête.
Flèche en pointillés		Représente une réponse ou un message de retour.
Fragment conditionnel (alt)		Permet de représenter des scénarios conditionnels.
Référence (ref)		Permet de faire référence à un autre diagramme afin d'éviter la redondance.
Fragment optionnel (opt)		Représente un comportement optionnel exécuté sous une condition donnée.

Ainsi, le diagramme de séquence facilite la compréhension du comportement dynamique du système et des interactions entre ses composants. Dans cette partie, nous présentons les différents diagrammes de séquences détaillés correspondant aux principaux scénarios du système de gestion des stages et des stagiaires.

3.5.1 authentification

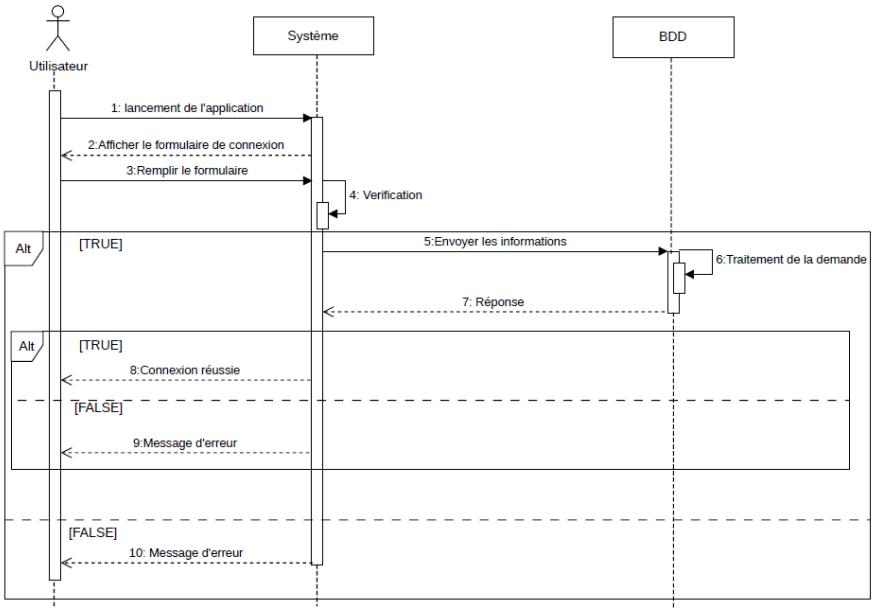


FIGURE 3.2 – Diagramme de séquence : Authentification

Ce tableau présente la description textuelle du cas d'utilisation « Authentification ». Il décrit les différentes étapes permettant à l'utilisateur de s'authentifier afin d'accéder au système selon son rôle.

Cas d'utilisation	Authentification
Acteurs	GRH, Encadreur
Objectif	Sécuriser l'accès au système après l'identification de l'utilisateur.
Pré-condition	Chaque utilisateur doit posséder un nom d'utilisateur et un mot de passe.
Scénario nominal	1. L'utilisateur clique sur le bouton de connexion. 2. Le système affiche le formulaire d'authentification. 3. L'utilisateur saisit ses informations. 4. Le système vérifie les informations saisies. 5. Le système envoie les informations à la base de données. 6. La base de données traite la demande. 7. La base de données renvoie le résultat au système. 8. Le système ouvre la session de l'utilisateur.
Scénario alternatif	A1 :2. Si le format des informations est incorrect 10.le système affiche un message d'erreur. A2 : 4.Si les informations de l'utilisateur sont incorrectes 9.le système affiche un message d'erreur.

TABLE 3.4 – Description de diagramme de séquence d'Authentification

3.5.2 Ajouter

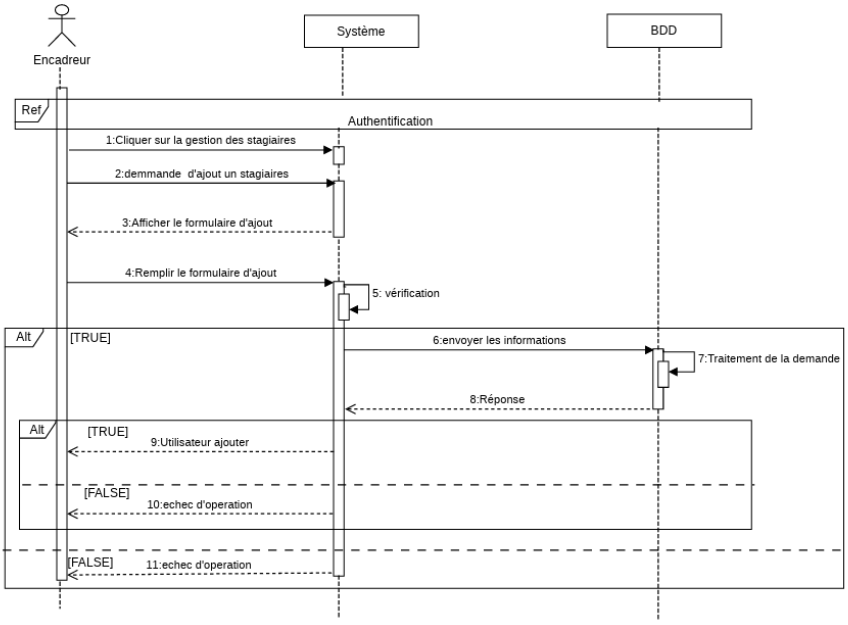


FIGURE 3.3 – Diagramme de séquence :Ajouter

Ce tableau présente la description textuelle du cas d'utilisation « Ajouter ». Il décrit les différentes étapes permettant à l'utilisateur d'effectuer une opération d'ajout. Ainsi, toutes les actions telles que « Ajouter un stagiaire », « Ajouter un sujet », « Ajouter un rendez-vous » s'adaptent à ce cas d'utilisation en utilisant les données et informations appropriées.

Cas d'utilisation	Ajouter un stagiaire
Acteur	Encadreur
Objectif	Permet à l'encadreur d'ajouter un stagiaire au système.
Pré-condition	L'encadreur doit être authentifié et avoir accès au formulaire d'ajout.
Scénario nominal	1. L'encadreur clique sur la gestion des stagiaires. 2. L'encadreur demande l'ajout d'un stagiaire. 3. Le système affiche le formulaire d'ajout. 4. L'encadreur saisit les informations du stagiaire. 5. Le système vérifie les informations saisies. 6. Le système envoie les informations à la base de données. 7. La base de données traite la demande. 8. La base de données renvoie le résultat au système. 9. Le système ajoute le stagiaire.
Scénario alternatif	A1 :5. Si les informations du formulaire sont incorrectes 10.le système affiche un message d'échec de l'opération. A2 : 7.Si le traitement de la base de données échoue 11.le système affiche un message d'échec de l'opération.

TABLE 3.5 – Description de diagramme de séquence d' Ajouter

3.5.3 Modifier

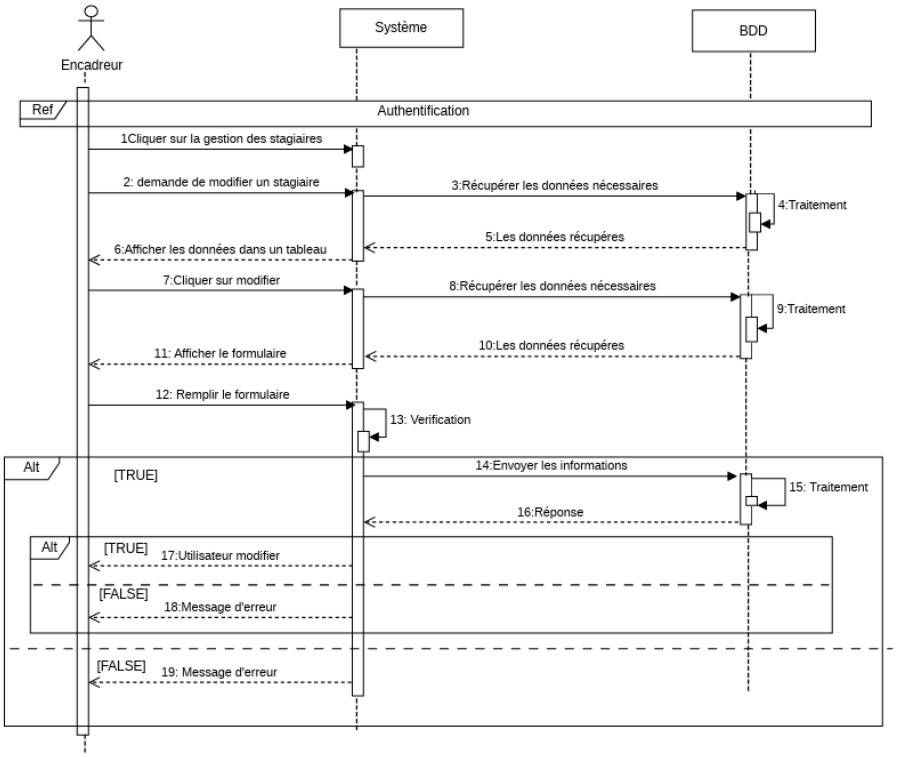


FIGURE 3.4 – Diagramme de séquence : Modifier

Ce tableau présente de façon globale la tâche « Modification ». Ainsi, toutes les actions telles que « Modifier un sujet», « Modifier un stagiaire »ou encore « Modifier un rendez-vous » s'adaptent à ce cas d'utilisation en utilisant les données et informations appropriées.

Cas d'utilisation	Modifier un stagiaire
Acteur	Encadreur
Objectif	Permett à l'encadreur de modifier les informations d'un stagiaire dans le système.
Pré-condition	L'encadreur doit être authentifié et un stagiaire doit être sélectionné.
Scénario nominal	1. L'encadreur clique sur la gestion des stagiaires. 2. L'encadreur demande la modification d'un stagiaire. 3. Le système récupère les données depuis la base de données. 4. La base de données traite la demande. 5. La base de données renvoie les données au système. 6. Le système affiche les données dans un tableau. 7. L'encadreur clique sur « Modifier ». 8. Le système récupère les données depuis la base de données. 9. La base de données traite la demande. 10. La base de données renvoie les données au système. 11. Le système affiche le formulaire. 12. L'encadreur saisit les informations. 13. Le système vérifie les informations. 14. Le système envoie les informations à la base de données. 15. La base de données traite la requête. 16. La base de données renvoie le résultat au système. 17. Le système affiche un message de succès.
Scénario alternatif	A1 :10. Si les informations incorrectes 19.le système affiche un message d'erreur. A2 : 13.Si les informations étaient incorrectes 18. le système affiche un message d'erreur.

TABLE 3.6 – Description de diagramme de séquence de Modifier

3.5.4 Consulter

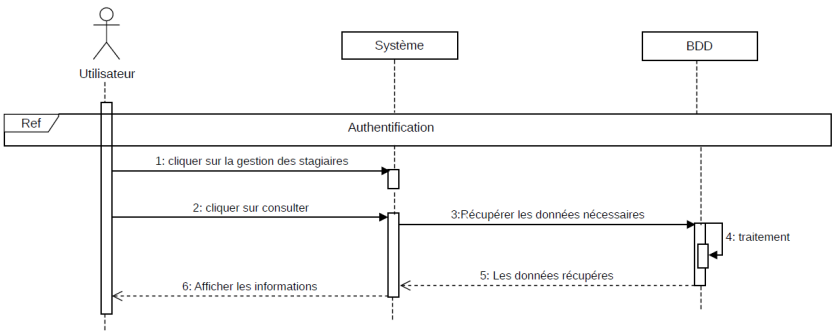


FIGURE 3.5 – Diagramme de séquence :Consulter

Ce tableau présente de façon globale la tâche « Consultation ». Ainsi, toutes les actions telles que « Consulter un stagiaire » s'adaptent aux données et informations appropriées

Cas d'utilisation	Consulter les stagiaires
Acteurs	Encadreur, GRH
Objectif	Consulter les informations des stagiaires enregistrés.
Pré-condition	L'utilisateur doit être authentifié pour consulter les informations.
Scénario nominal	1. L'utilisateur clique sur la gestion des stagiaires. 2. L'utilisateur clique sur « Consulter ». 3. Le système récupère les données depuis la base de données. 4. La base de données traite la requête. 5. La base de données renvoie les données au système. 6. Le système affiche les informations.

TABLE 3.7 – Description de diagramme de séquence de Consulter

3.5.5 Supprimer

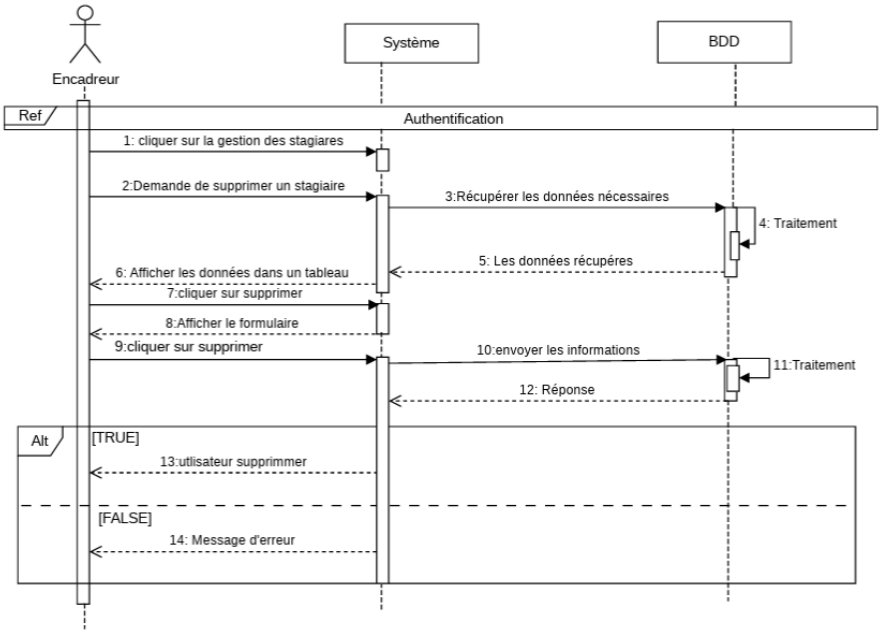


FIGURE 3.6 – Diagramme de séquence :Supprimer

Ce tableau présente de façon globale la tâche « Suppression ». Ainsi, toutes les actions telles que « Supprimer un stagiaire », « Supprimer un sujet »ou encore « Supprimer un rendez-vous » s'adaptent à ce cas d'utilisation en utilisant les données et informations appropriées.

Cas d'utilisation	Supprimer un stagiaire
Objectif	Permet à l'encadreur de supprimer un stagiaire du système.
Pré-condition	L'encadreur doit être authentifié et un stagiaire doit être sélectionné.
Scénario nominal	1. L'encadreur clique sur la gestion des stagiaires. 2. L'encadreur demande la suppression d'un stagiaire. 3. Le système récupère les données depuis la base de données. 4. La base de données traite la requête. 5. La base de données renvoie les données au système. 6. Le système affiche les données dans un tableau. 7. L'encadreur clique sur « Supprimer ». 8. Le système affiche le formulaire de suppression. 9. L'encadreur confirme la suppression. 10. Le système envoie les informations à la base de données. 11. La base de données traite la requête. 12. La base de données renvoie le résultat au système. 13. Le système affiche un message de succès.
Scénario alternatif	A1 :12. Si les informations sont incorrectes 14. le système affiche un message d'erreur.

TABLE 3.8 – Description de diagramme de séquence de Supprimer

3.6 diagramme de classe

Le diagramme de classe est un diagramme structurel du langage UML permettant de représenter la structure statique d'un système orienté objet. Il décrit les différentes classes du système, leurs attributs, leurs opérations ainsi que les relations existantes entre elles, telles que les associations, les héritages et les dépendances.

Chaque classe est généralement composée des éléments suivants :

— **Nom :** représente le concept modélisé par la classe.
— **Attributs :** définissent les données associées à la classe. Chaque attribut possède un nom, un type et une visibilité.
— **Opérations (méthodes) :** représentent les différentes actions ou traitements que peut effectuer la classe.

Le diagramme de classe constitue ainsi une étape essentielle dans la conception d'un système, car il permet de visualiser l'organisation et les relations entre les différentes entités du systéme.

3.6.1 Élaboration d'un diagramme de classes

Une démarche couramment utilisée pour élaborer un diagramme de classes consiste à suivre plusieurs étapes :

— **Identifier les classes du domaine étudié :** cette étape se réalise généralement en collaboration avec un expert du domaine. Les classes représentent souvent des concepts ou des objets importants du système.
— **Identifier les associations entre les classes :** les associations représentent les relations existantes entre les différentes classes du système, comme « gère », « contient », « possède » ou « dépend de ».
— **Identifier les attributs des classes :** les attributs décrivent les propriétés de chaque classe, telles que le nom, la date, le numéro ou le statut.
— **Organiser et simplifier le modèle :** cette étape consiste à éliminer les classes redondantes, utiliser l'héritage lorsque cela est nécessaire et améliorer la structure générale du modèle.
— **Vérifier et affiner le diagramme :** le diagramme est progressivement amélioré afin d'obtenir un modèle cohérent et adapté aux besoins du système.

3.6.2 Règles de gestion

— Un utilisateur possède un seul compte.
— Un compte appartient à un seul utilisateur.
— Un utilisateur peut se connecter et se déconnecter du système.
— Un encadreur est un utilisateur.
— Un GRH est un utilisateur.
— Un encadreur peut gérer plusieurs stagiaires.
— Un encadreur peut gérer plusieurs sujets.
— Un encadreur peut gérer plusieurs rendez-vous.
— Un encadreur peut proposer un ou plusieurs thèmes.
— Un thème est proposé par un seul encadreur.
— Un mémoire final traite un seul thème.
— Un thème est traité dans un seul mémoire final.
— Un mémoire final peut importer un contrat.
— Le GRH peut gérer plusieurs stagiaires.
— Le GRH peut valider plusieurs contrats.
— Un contrat peut contenir plusieurs documents.
— Un document appartient à un seul contrat.
— Un document peut être affiché par le système.
— Un compte permet de vérifier le login et de changer le mot de passe.

La figure 3.7 représente le diagramme de classes de notre système. Elle montre la structure statique interne de l'application à travers les différentes classes, leurs attributs, leurs méthodes ainsi que les relations existantes entre elles.

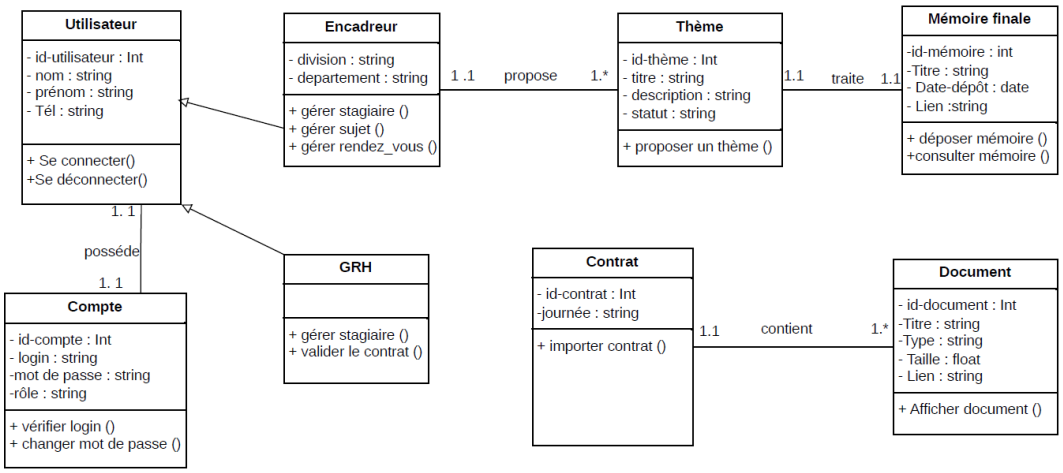


FIGURE 3.7 – Diagramme de classes du système

3.7 Dictionnaire de données

Le dictionnaire de données présente la définition, le type (texte, numérique, date, etc.) ainsi que le format (nombre de caractères, nombre de décimales, etc.) de l'ensemble des données gérées dans la base de données.

L'organisation de ces données est représentée sous forme de schémas appelés modèles de données. L'objectif du dictionnaire de données est de permettre à toute personne découvrant la base de données de pouvoir la reconstruire dans n'importe quel Système de Gestion de Base de Données (SGBD).

Les différents types de données sont présentés comme suit :

Table 3.9 – Dictionnaire de données

Entité	Attribut	Type	Description
Utilisateur	id_utilisateur	int	Identifiant de l'utilisateur
	nom	String	Nom de l'utilisateur
	prénom	String	Prénom de l'utilisateur
	tel	String	Numéro de téléphone de l'utilisateur
Compte	id_compte	int	Identifiant du compte
	login	String	Nom d'utilisateur
	mdp	String	Mot de passe
	rôle	String	Rôle de l'utilisateur
Encadreur	division	String	Division de l'encadreur
	département	String	Département de l'encadreur
Theme	id_thème	int	Identifiant du thème
	titre	String	Titre du thème
	description	String	Description du thème
	statut	String	État du thème
Contrat	id_contrat	int	Identifiant du contrat
	journée	String	Jour du stage
Memoire_finale	id_memoire	int	Identifiant du mémoire
	titre	String	Titre du mémoire
	date_dépôt	String	Date de dépôt
	lien	String	Lien du document
Document	id_document	int	Identifiant du document
	titre	String	Titre du document
	type	String	Type du document
	taille	Float	Taille du fichier
	lien	String	Lien du document

3.7.1 Transformation du modèle Objet au modèle relationnel

Pour le passage du modèle conceptuel de données UML vers le modèle relationnel, nous appliquons les règles suivantes :

- **Règle 1 :** Chaque classe devient une table. Les attributs de la classe deviennent les attributs de la table, et l'identifiant de la classe devient la clé primaire.
- **Règle 2 :** Chaque association de type 1–1 est représentée par l'ajout de la clé primaire d'une des tables comme clé étrangère dans l'autre table.
- **Règle 3 :** Chaque association de type 1–N est représentée par l'ajout de la clé primaire de la table du côté « 1 » comme clé étrangère dans la table du côté « N ».
- **Règle 4 :** Chaque association de type M–N est représentée par la création d'une nouvelle table contenant les clés primaires des deux tables associées. Ces clés deviennent des clés étrangères et forment ensemble la clé primaire de la nouvelle table. Les attributs de l'association peuvent également être ajoutés dans cette table si nécessaire.

3.7.2 Modèle relationnel

Dans cette étape, nous élaborons le modèle relationnel de données qui contient les tables suivantes :

- **Utilisateur**(id_utilisateur, nom, prénom, tel)
- **Compte**(id_compte, login, mot_de_passe, rôle, #id_utilisateur)
- **GRH**(#id_utilisateur)

— **Encadreur**(#id_utilisateur, division, département)
— **Thème**(id_thème, titre, description, statut, #id_utilisateur)
— **MémoireFinale**(id_mémoire, titre, date, lien, #id_thème)
— **Contrat**(id_contrat, journée)
— **Document**(id_document, titre, type, taille, lien, #id_contrat)

Remarque : la clé primaire est soulignée, tandis que la clé étrangère est précédée du symbole # en gras.

3.8 Conclusion

Dans ce chapitre, nous avons modélisé le système à l'aide du langage UML et du processus RUP, en présentant les fonctionnalités principales et les interactions entre les acteurs à travers différents diagrammes.

Chapitre 4

Implémentation

4.1 Introduction

Dans ce chapitre final, nous abordons la phase de réalisation de notre application, qui vient après l'étape de conception. Nous y décrivons le diagramme de déploiement ainsi que les différentes interfaces conçues à l'aide des outils déjà présentés.

4.2 MVC

MVC signifie Modèle-Vue-Contrôleur. C'est une architecture logicielle qui sépare une application en trois parties : le modèle pour les données, la vue pour l'affichage, et le contrôleur pour la logique qui relie les deux. Cette séparation améliore l'organisation du code et facilite la maintenance.

4.3 Diagramme de déploiement

Un diagramme de déploiement est une représentation statique utilisée pour décrire la distribution physique des ressources matérielles d'un système ainsi que la manière dont les composants sont répartis sur ces ressources. Il permet aussi de montrer les liens existants entre les nœuds et les composants. Ce diagramme est utilisé pour représenter l'architecture de notre application.

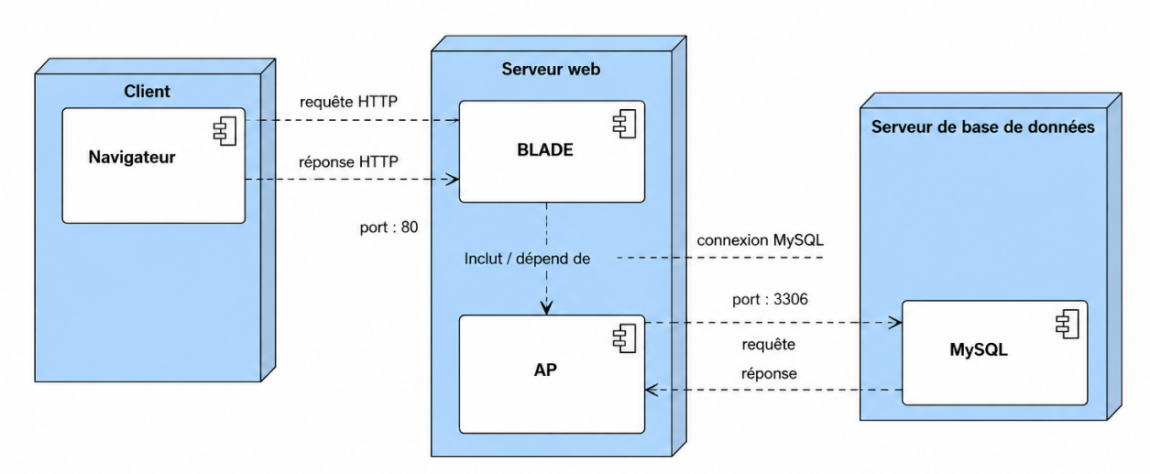


FIGURE 4.1 – Diagramme de déploiement de l'application

4.4 Outils retenus pour le développement de l'application

Dans cette section, nous présentons les différents outils utilisés pour le développement de notre application.

4.4.1 Environnement hardware

Pour développer cette application, nous avons utilisé une machine configurée comme suit : Le développement et les tests de l'application ont été réalisés sur un ordinateur possédant les caractéristiques suivantes :

- **Processeur :** Intel(R) Core(TM) i5-10210U CPU @ 1.60GHz (2.11 GHz)
- **Mémoire RAM :** 16 Go
- **Carte graphique :** Intel(R) UHD Graphics
- **Stockage :** 238 Go
- **Type du système :** Système d'exploitation 64 bits, processeur x64

4.4.2 Environnement software

Pour créer des applications web, on utilise deux types de langages : les langages côté serveur et les langages côté client. Les langages côté serveur s'occupent du traitement des données et de la base de données. Les langages côté client servent à afficher la page et à la rendre interactive. Dans notre projet, nous avons choisi un langage côté serveur pour gérer les données et créer une application dynamique.



PHP

PHP est un langage côté serveur utilisé pour développer des applications web dynamiques. Il gère le traitement des données, la logique de l'application et la communication avec la base de données. [10]



Laravel

Laravel est un framework PHP moderne qui facilite le développement d'applications web structurées. Il fournit des outils prêts à l'emploi comme l'authentication, le routing et la gestion de bases de données. [11]



MySQL

MySQL est un système de gestion de bases de données relationnelles (SGBDR). Il stocke les informations dans des tableaux organisés et permet de les consulter, modifier et supprimer via le langage SQL. [12]



Bootstrap

Bootstrap est un framework CSS/JS pour créer des interfaces web modernes et responsive. Il s'adapte automatiquement aux écrans mobiles, tablettes et ordinateurs sans écrire beaucoup de CSS. [13]

jQuery

jQuery est une bibliothèque JavaScript qui simplifie les interactions côté client. Elle permet de réaliser des animations, gérer les événements et manipuler le DOM avec moins de code. [14]



AJAX

AJAX (Asynchronous JavaScript and XML) permet d'échanger des données avec le serveur sans recharger toute la page. Cela rend l'application plus rapide et l'expérience utilisateur plus fluide. [15]



XAMPP

XAMPP est un environnement local de développement regroupant Apache (serveur web), MySQL (base de données) et PHP. Il permet de tester l'application sur son ordinateur sans hébergement externe. [16]

phpMyAdmin

phpMyAdmin est une interface web intuitive pour gérer les bases de données MySQL. Elle permet de créer des tableaux, ajouter des données, exécuter des requêtes SQL sans coder. [17]

Visual Studio Code

Visual Studio Code est un éditeur de code moderne, gratuit et léger. Il offre la coloration syntaxique, la complétion automatique, le debuggage et des extensions pour tous les langages. [18]

Composer

Composer est un gestionnaire de dépendances pour PHP. Il installe automatiquement les bibliothèques externes dont l'application a besoin et gère leurs versions pour éviter les conflits. [19]

Node.js

Node.js est un environnement d'exécution JavaScript côté serveur. Il est utilisé pour les outils frontend comme la compilation de Sass et le regroupement (bundling) du JavaScript. [20]

Git

Git est un système de gestion de versions qui enregistre chaque modification du code. Il permet de revenir à une version précédente et de travailler en équipe sans perdre le travail. [21]

4.5 Présentation des interfaces graphiques

Nous présentons ici, à travers un ensemble de captures d'écran, les différents scénarios d'exécution offrant un aperçu général du fonctionnement de notre site web. Ces interfaces illustrent les principales fonctionnalités du système ainsi que les différentes interactions possibles entre les utilisateurs et le site web.

4.5.1 Page de Login

nterface (Connexion) : Page d'authentification sécurisée permettant aux utilisateurs de se connecter à la plateforme en saisissant leurs identifiants et en sélectionnant leur rôle spécifique (Encadrant ou GRH).

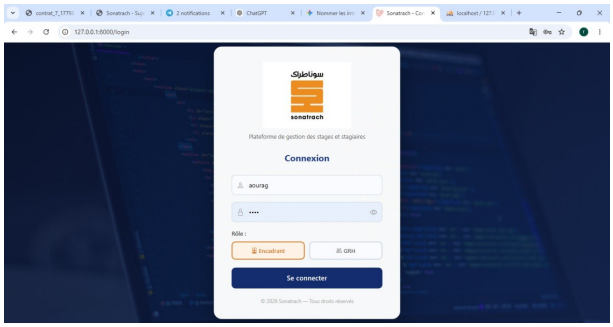


FIGURE 4.2 – page de login

4.5.2 Page Liste des sujets

Interface (Liste des sujets) : Espace centralisé affichant les thèmes de stage proposés, leurs statuts de disponibilité (attribué, ouvert, verrouillé) ainsi qu'une infobulle descriptive au survol pour faciliter leur gestion.

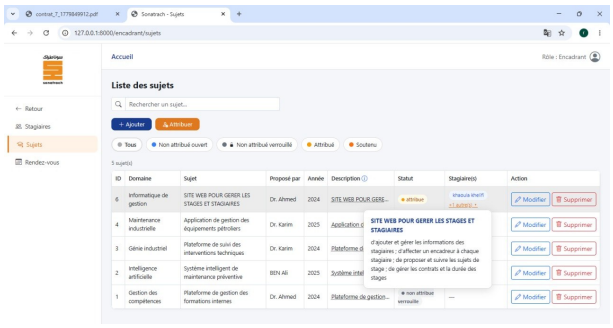


FIGURE 4.3 – Page de liste des sujets

4.5.3 Page de Modification

Interface de modification : Fenêtre modale permettant de mettre à jour les données personnelles, académiques et la période de stage du stagiaire, avec la possibilité d'attribuer ou de modifier le sujet choisi.

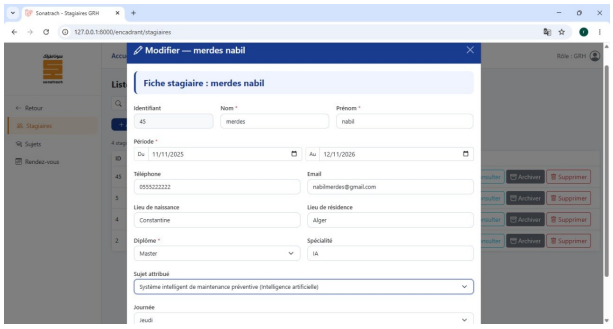


FIGURE 4.4 – Page de modification

4.5.4 Page de Rendez-vous

Interface (Rendez-vous) : Tableau de suivi de l'encadrant permettant de planifier, modifier ou supprimer des rendez-vous avec les stagiaires, tout en visualisant l'état d'avancement des rapports associés.

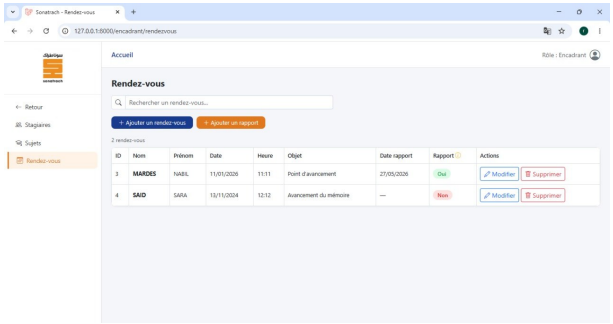


FIGURE 4.5 – Page de Rendez-vou

4.6 Conclusion

Ce chapitre a été consacré à la présentation des outils et des langages ayant servi à l'implémentation de notre application. Il a également permis de mettre en évidence le diagramme de déploiement ainsi que les différentes interfaces réalisées au niveau de la plateforme.

Conclusion Générale

Au terme de ce travail, notre problématique s'est articulée autour de l'amélioration de la gestion des stages et des stagiaires au sein d'un organisme d'accueil. L'objectif principal consistait à concevoir et à développer une application web permettant de centraliser les informations, de les organiser et de faciliter le suivi des stagiaires, tout en offrant une interface ergonomique aux différentes parties prenantes. Cette application vise à se substituer aux méthodes manuelles traditionnelles, souvent sources d'erreurs et de perte de temps, au profit d'une solution numérique fiable et performante.

Afin de mener ce projet à bien, une méthodologie structurée et cohérente a été adoptée, fondée sur le processus unifié RUP et la modélisation UML. Cette approche méthodologique a permis de suivre des étapes séquentielles et rigoureuses, englobant l'analyse des besoins, la conception des différents diagrammes — cas d'utilisation, séquence, classes et déploiement — puis la phase de développement et de tests. L'architecture client/serveur a été retenue comme cadre architectural garantissant une séparation nette entre la couche présentation et la couche logique applicative.

Sur le plan technique, ce projet a abouti à une application web fonctionnelle, développée à l'aide des technologies PHP, MySQL via phpMyAdmin et Apache, permettant une gestion complète des dossiers des stagiaires en termes d'ajout, de modification, de suppression et de consultation. Ce travail a contribué à approfondir les compétences techniques acquises dans les domaines du développement web, de la modélisation UML et de la gestion des bases de données relationnelles, ainsi qu'à développer les aptitudes méthodologiques et organisationnelles nécessaires à la conduite de projets réels.

En guise de perspective, et malgré les résultats satisfaisants obtenus, les pistes d'amélioration demeurent nombreuses. À court terme, l'application pourrait être enrichie par l'intégration d'un système de notifications, d'un module de gestion des documents officiels tels que les conventions de stage et les rapports, ainsi que d'un tableau de bord statistique interactif. À long terme, le système pourrait évoluer vers une plateforme plus complète, incluant une application mobile, un espace dédié aux établissements d'accueil, ainsi que l'intégration de techniques d'intelligence artificielle pour l'affectation automatisée des stagiaires. Ces perspectives constituent ensemble des pistes de recherche et de développement prometteuses pour de futurs travaux.

Bibliographie

[1] Sonatrach, *Site officiel de Sonatrach*.
Disponible sur : https ://www.sonatrach.com (consulté le 30 mars 2026).

[2] Institut Algérien du Pétrole (IAP), *Présentation et formation*.
Disponible sur : https ://www.iap.dz (consulté le 30 mars 2026).

[3] Sonatrach, *Rapport annuel*. Alger, Algérie, 2022.

[4] Ministère de l'Énergie, *Données du secteur énergétique en Algérie*, 2023.

[5] BENETTO, Enrico, *Analyse du cycle de vie – Réalisation de l'inventaire*. Techniques de l'Ingénieur, 10 octobre 2005.
Disponible sur : https ://www.techniques-ingenieur.fr (consulté le 06 avril 2026).

[6] Université de Mila, *Analyse du cycle de vie – Cours*.
Disponible sur : https ://elearning.univ-mila.dz (consulté le 06 avril 2026).

[7] OMG (Object Management Group), *Unified Modeling Language (UML) Specification*.
Disponible sur : https ://www.omg.org/spec/UML

[8] IBM, *Rational Unified Process (RUP)*.
Disponible sur : https ://www.ibm.com

[9] D. Gabay et J. Gabay, *UML 2 : Analyse et conception*. Paris : Dunod, 2008.

[10] PHP, *Documentation officielle PHP*.
Disponible sur : https ://www.php.net/manual/fr/index.php (consulté le 21 mai 2026).

[11] Laravel, *Documentation officielle Laravel*.
Disponible sur : https ://laravel.com/docs (consulté le 21 mai 2026).

[12] Oracle, *Documentation MySQL*.
Disponible sur : https ://dev.mysql.com/doc/ (consulté le 21 mai 2026).

[13] Bootstrap, *Documentation officielle Bootstrap*.
Disponible sur : https ://getbootstrap.com (consulté le 21 mai 2026).

[14] jQuery Foundation, *Documentation jQuery*.
Disponible sur : https ://jquery.com (consulté le 21 mai 2026).

[15] MDN Web Docs, *AJAX*.
Disponible sur : https ://developer.mozilla.org/fr/docs/Glossary/AJAX (consulté le 21 mai 2026).

[16] Apache Friends, *XAMPP*.
Disponible sur : https ://www.apachefriends.org/fr/index.html (consulté le 21 mai 2026).

[17] phpMyAdmin, *Documentation phpMyAdmin*.
Disponible sur : https ://docs.phpmyadmin.net/fr/latest/intro.html (consulté le 21 mai 2026).

[18] Microsoft, *Visual Studio Code Documentation*.
Disponible sur : https ://code.visualstudio.com/docs (consulté le 21 mai 2026).

[19] Composer, *Documentation Composer*.
Disponible sur : https ://getcomposer.org/doc/ (consulté le 21 mai 2026).

[20] Node.js, *Documentation Node.js*.
Disponible sur : https ://nodejs.org/en/docs (consulté le 21 mai 2026).

[21] GitHub, *Git Cheat Sheet*.
Disponible sur : https ://training.github.com/downloads/fr/github-git-cheat-sheet.pdf (consulté le 21 mai 2026).