



RAPPORT DE PROJET TECHNOLOGIE WEB

Application web de traitement de la vie estudiantine dans l'Ensias

Réalisé par :

SAOUDI MEHDI
TENNIA YOUSSEF
YOUNESS MOUAD
ZANATI ZANATI

Sous la direction de :
EL HAMLAOUI MAHMOUD

2019/2020

Remerciement

Nous tenons, avant tout, à adresser nos plus vifs remerciements à Monsieur Mahmoud EL HAMLAOUI, notre cher encadrant de ce projet, qui nous a aidés à escamoter beaucoup de problèmes.

Enfin, nos remerciements vont à tous les enseignants de l'École nationale supérieure d'informatique et d'analyse des systèmes, pour leurs aides et leur professionnalisme.

Table des figures

1.1	Modèle en cascade [4]	12
1.2	Planning	13
2.1	Diagramme de cas d'utilisation	17
2.2	Diagramme de séquence d'authentification	21
2.3	Diagramme de séquence de déclaration de problème	22
2.4	Diagramme de séquence de résolution de problème	23
2.5	Diagramme de séquence de visualisation d'archive de l'élève	24
2.6	Diagramme de séquence de visualisation de profil	25
2.7	Diagramme de séquence de visualisation de l'archive de tous les étudiants	26
2.8	Diagramme de séquence de visualisation de la liste des étudiants	27
2.9	Diagramme d'états transitions	28
3.1	Diagramme de classe	30
4.1	Connexion Clients/Serveur [7]	35
4.2	Architecture MVC [7]	36
4.3	Page d'accueil	37
4.4	Fenêtre de login	38
4.5	Interface de l'admin	39
4.6	Interface de l'élève	40
4.7	Page de déclaration de problèmes	41
4.8	Liste des problèmes de tous les élèves	42
4.9	Liste des problèmes d'un élève	43
4.10	Profil de l'élève	44
4.11	Liste de tous les problèmes	45
4.12	Liste des problèmes urgent	46
4.13	Page d'archive	47
4.14	Liste des problèmes de l'année en cours	48
4.15	Liste des étudiants	49
4.16	Qr code des étudiants	50
4.17	Distribution des problèmes de la comité bâtiments	51
4.18	Distribution des problèmes de la comité divers	52
4.19	Distribution des problèmes selon les comités	53
4.20	Statistiques globales	54
4.21	Page contact us	55

4.22 Page about us	56
------------------------------	----

Liste des tableaux

2.1	Description du cas d'utilisation <Gérer le profil >	18
2.2	Description du cas d'utilisation < Publier un problème >	19
2.3	Description du cas d'utilisation < Gérer les problèmes >	20
2.4	Description du cas d'utilisation <Postuler un emploi>	20
3.1	Table eleve	31
3.2	Table probleme	31
3.3	Table comite	31
3.4	Table boite	32
3.5	Table aimer	32

Sommaire

Remerciement	2
Introduction	8
1 Présentation de la problématique liée à la résolution des problèmes	9
1.1 Introduction	10
1.2 Présentation du contexte du site web Ensias Solvedy	10
1.3 Analyse des besoins	10
1.3.1 Identification des acteurs	11
1.3.2 Besoins fonctionnels	11
1.4 Processus de développement	12
1.4.1 Le modèle de développement choisi	12
1.4.2 Planning prévisionnel	13
1.5 Conclusion	14
2 Analyse et conception	15
2.1 Introduction	16
2.2 Langage de conception	16
2.2.1 Présentation d'UML	16
2.2.2 Avantages d'UML	16
2.3 Modélisation et spécification Fonctionnelles	16
2.3.1 Diagrammes Cas d'utilisation	16
2.3.2 Côté élève	18
2.3.3 Côté comité	19
2.4 Conception	21
3 Création de la base de données	29
3.1 Introduction	30
3.2 Diagramme de classe	30
3.3 Tables de la base de données	31
3.4 Conclusion	32
4 Réalisation	33
4.1 Introduction	34
4.2 Environnement Logiciel	34
4.3 Technologies utilisées	34
4.4 Architecture de l'application	35

4.4.1	Client / Serveur	35
4.4.2	Architecture MVC	36
4.5	Interfaces de l'application	37
4.6	Conclusion	57
Conclusion générale		58
Webographie		59

Introduction générale

De nos jours, l'optimisation du temps et la bonne gestion de nos problèmes joue un rôle très important dans notre vie quotidienne afin d'escamoter beaucoup de problèmes.

Après avoir été élu par les étudiants de l'ENSIAS, les membres de différents comités qui représentent les différentes parties de l'école travaillent pour produire des stratégies extrêmement importantes pour accomplir leur mission envers les étudiants. Pour cela, leur bonne gestion des problèmes est un moyen efficace pour progresser dans les différentes situations.

Dans ce contexte, nous proposons de présenter une application web nommée «Ensiyas Solvely» qui aide les différentes comités d'une part à bien gérer les problèmes et les résoudre, et les élèves d'une autre part afin de chercher des solutions dans les plus brefs délais pour leur problèmes proposer.

Le plan de ce rapport est composé de quatre chapitres : Le premier chapitre va présenter le contexte du projet, l'analyse des besoins et décrit le processus de développement de l'application. Le deuxième chapitre est consacré à la présentation des outils utilisés pour la modélisation, la conception et la présentation des spécificités fonctionnelles pour les deux cotés Comité et Élève.

Le troisième chapitre est réservé à la création de la base de données.

Le dernier chapitre présente les spécifications des environnements matériels et logiciels utilisées au cours du développement et aussi présente les différentes interfaces ainsi que les problèmes de programmation rencontrés.

Chapitre 1

Présentation de la problématique liée à la résolution des problèmes

1.1 Introduction

Aujourd’hui il y a plus de 900 étudiants à l’ENSIAS. En effet, la résolution de leurs problèmes est un devoir indispensable pour les différents comités. Les étudiants désirent depuis toujours que leurs demandes soient traiter dans les plus brefs délais et dans les meilleures conditions. Pour cela, les membres des comités doivent savoir manipuler ses demandes, les traiter d’une manière convenable afin de satisfaire les besoins des différents élèves. Ce projet est un site web « Ensias Solvely » qui sert à fournir un service d’optimisation dans la résolution des problèmes. Dans ce chapitre, nous commencerons par la présentation de la problématique du sujet du projet « Ensias Solvely », puis nous définissons les acteurs qui interagissent avec le site web, ensuite nous citons les besoins fonctionnels.

1.2 Présentation du contexte du site web Ensias Solvely

Le site web « Ensias Solvely », est un simple outil de partage des problèmes des étudiants de l’ENSIAS pour les révéler aux différents comités afin de trouver des solutions optimales. « Ensias Solvely » permet ainsi l’optimisation du temps des étudiants pour bénéficier d’un service de haute qualité et en utilisant des moyens bien développés.

Ce dernier permet à tous les élèves de gérer leurs comptes et d’assurer leur authentification :

- Si l’utilisateur est un élève qui n’appartient à aucune comité, il peut gérer son profile, déclarer ses problèmes, visualiser son archive, suivre les problèmes en cours et contacter les responsables des comités.
- Si l’utilisateur est un élève qui appartient à un comité, il peut ainsi déclarer ses problèmes, gérer son profile, visualiser son archive, suivre les problèmes en cours de traitement afin de les résoudre, découvrir l’ensemble des statistiques et afficher la liste des différents étudiants.

1.3 Analyse des besoins

Cette partie sera consacrée à la présentation des besoins fonctionnels de notre projet. L’analyse fonctionnelle est une démarche qui consiste à rechercher et à caractériser les fonctions offertes par un produit pour satisfaire les besoins de son utilisateur.

1.3.1 Identification des acteurs

Un acteur est une personne externe qui interagit avec le système étudié. Dans notre cas, nous trouvons les acteurs suivants :

L'élève : c'est un utilisateur qui aura la possibilité de publier ses demandes afin qu'elles soient résolues.

Le membre de comité : c'est l'utilisateur qui gère ses demandes dans les plus brefs délais, il va utiliser le site web pour chercher des problèmes et répondre aux différents élèves.

1.3.2 Besoins fonctionnels

Dans cette partie du rapport, nous présentons les services que notre site web fournira aux différents utilisateurs. Ces besoins se regroupent dans les diagrammes des cas d'utilisation.

< EnsiasSolvely > permet à l'élève de :

- S'authentifier : Se connecter par son id et un mot de passe.
- Publier un problème : Saisir les informations relatives au problème rencontré en déterminant les détails de sa demande.
- Visualiser l'ensemble des problèmes : Voir les différents problèmes et donner leur avis sur chaque problème selon l'importance de ce dernier.
- Visualiser son archive : Voir l'ensemble des problèmes déjà traiter.
- Contacter les comités : Afin de donner des suggestions ou des réclamations.
- Gérer le compte : Modifier les paramètres de son compte.

L'application permet au membre de comité de :

- S'authentifier : Se connecter par son id et un mot de passe.
- Publier un problème : Saisir les informations relatives au problème rencontré en déterminant les détails de sa demande.
- Visualiser l'ensemble des problèmes : Voir les différents problèmes et donner leur avis sur chaque problème selon l'importance de ce dernier.
- Visualiser l'archive : Voir l'archive de problèmes des différents étudiants.
- Visualiser les statistiques : Voir les statistiques des différents comités afin de déterminer le niveau de progression.
- Visualiser la liste des étudiants : Voir la liste des informations des différents

étudiants, ainsi le QR code de chacun d'eux.

- Gérer le compte : Modifier les paramètres de son compte.

1.4 Processus de développement

Avant de commencer à travailler sur un projet, il faut choisir un processus de développement, c'est-à-dire un ensemble d'activités successives, organisées en vue de la production d'un logiciel.

1.4.1 Le modèle de développement choisi

Pour notre projet, nous avons choisi le modèle en cascade.

Modèle en cascade :

Le modèle en cascade est une version répandue du modèle de gestion du cycle de développement des systèmes et des applications. Souvent considéré comme l'approche classique du développement, ce modèle décrit un cycle linéaire et séquentiel. Son alternative la plus connue aujourd'hui est la méthodologie Agile.

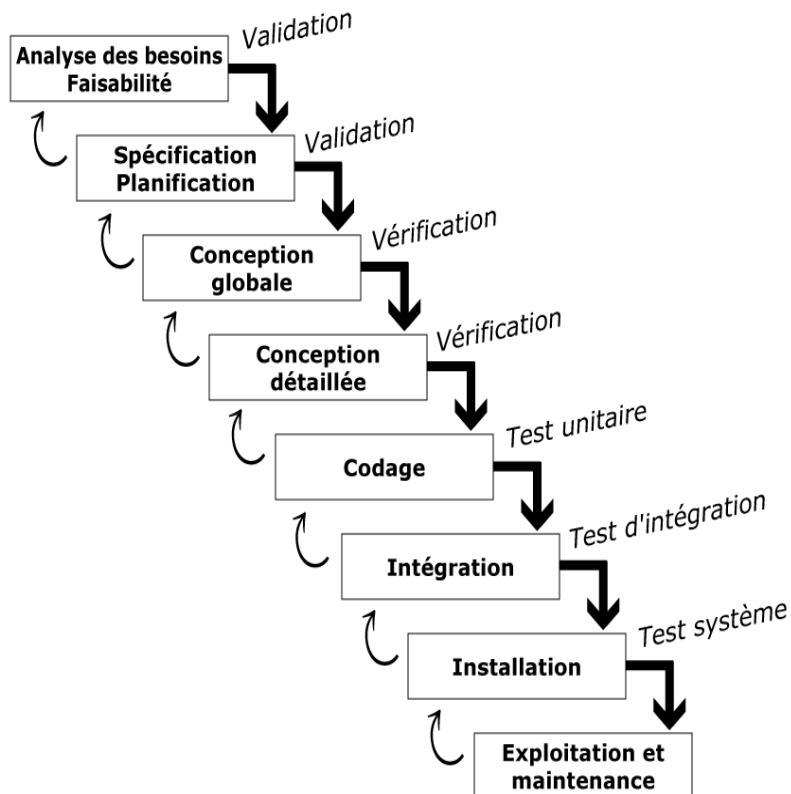


FIGURE 1.1 – Modèle en cascade [4]

Nous avons choisi ce modèle pour sa simplicité et sa précision de tâche, ainsi qu'il ne nécessite pas une grande expérience pour développer un projet.

1.4.2 Planning prévisionnel

Notre travail a débuté par une analyse et une conception du projet choisi. Cette phase a été suivie d'une étude théorique des outils de programmation avant d'aborder la phase de réalisation et la rédaction du rapport.

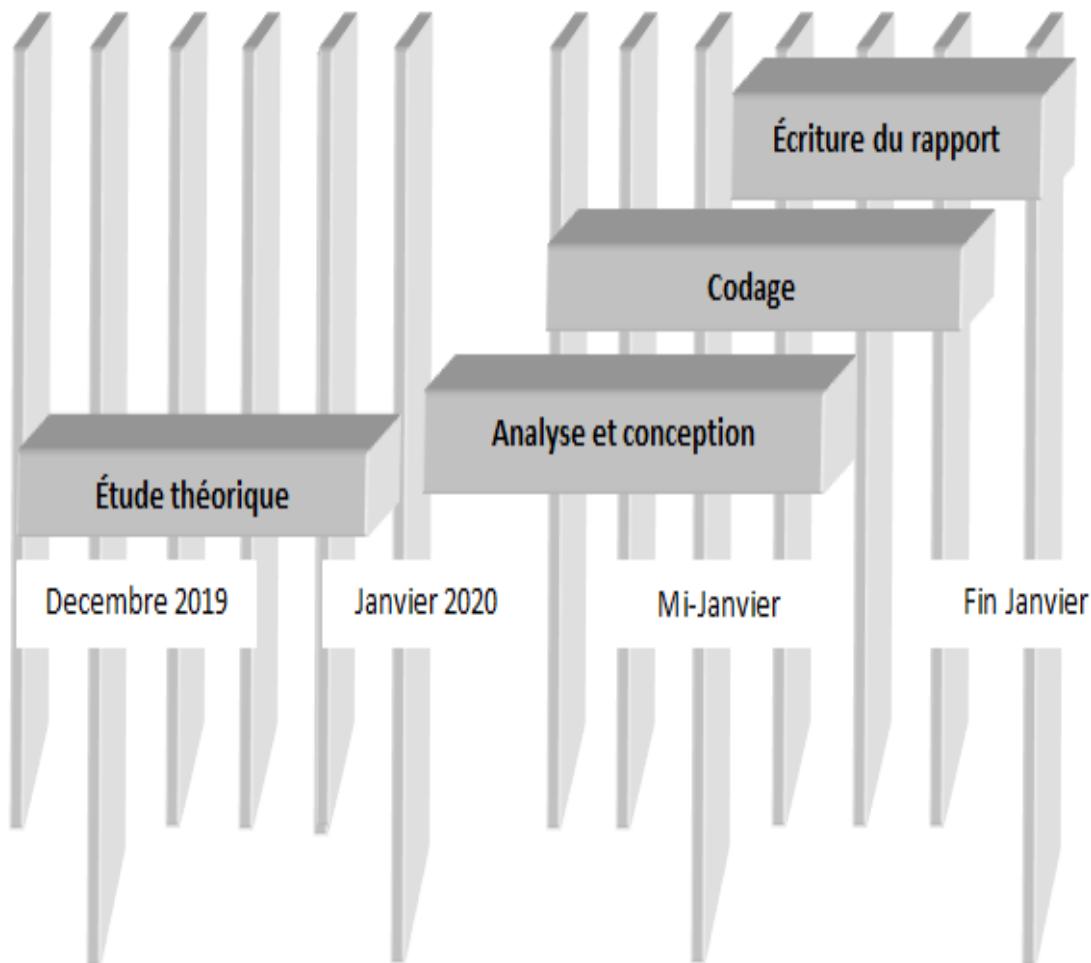


FIGURE 1.2 – Planning

- Étude théorique : L'étape de l'étude des différentes besoins, la rédaction du cahier de charges et le choix des langages et technologies du développement utilisées pour la réalisation du projet.
- Analyse et conception : L'étape de la réalisation des diagrammes et de schémas de conception utiles pour notre projet.
- Codage : L'étape de la création de la base de données et du développement de notre site web.

1.5 Conclusion

Dans ce premier chapitre, nous avons essayé de donner une vision générale sur notre projet. Nous avons défini les acteurs qui interagissent avec le site web, les besoins fonctionnels, et enfin le processus de développement suivi tout au long du projet.

Chapitre 2

Analyse et conception

2.1 Introduction

Avant de programmer le site web et se lancer dans l'écriture du code, il faut d'abord organiser les idées et les documenter, puis organiser la réalisation en définissant les modules et les étapes de la réalisation. Cette démarche s'appelle la modélisation, son produit est un module.

2.2 Langage de conception

Dans le cadre de notre projet nous avons utilisé le langage UML (Unified Modeling Language) [5] pour la modélisation des différents diagrammes.

2.2.1 Présentation d'UML

UML « Unified Modeling Language » : est un langage visuel constitué d'un ensemble des schémas, appelés diagrammes, qui donnent chacun une vision différente du projet à traiter. UML nous fournit donc des diagrammes pour représenter le logiciel à développer : son fonctionnement, sa mise en route, les actions susceptibles d'être effectuées par le logiciel, etc. Notre choix s'est basé sur les points forts de ce langage. UML est conçu pour s'adapter à n'importe quel langage de programmation orientée objet (POO), et présente plusieurs modèles (Diagrammes) pour une modélisation bien précise.

2.2.2 Avantages d'UML

- Universel.
- Adopté par les grandes entreprises.
- Adopté par plusieurs processus de développement
- Limite les risques d'erreur.
- N'est pas limité au domaine informatique.

2.3 Modélisation et spécification Fonctionnelles

2.3.1 Diagrammes Cas d'utilisation

Le diagramme de cas d'utilisation a pour but de donner une vision globale sur les interfaces de la future application. C'est le premier diagramme UML constitué d'un

ensemble d'acteurs qui agit sur des cas d'utilisation, le comportement d'une application du point de vue utilisateur.

Ci-dessous, nous présentons le diagramme de cas d'utilisation pour la compréhension du fonctionnement de l'application.

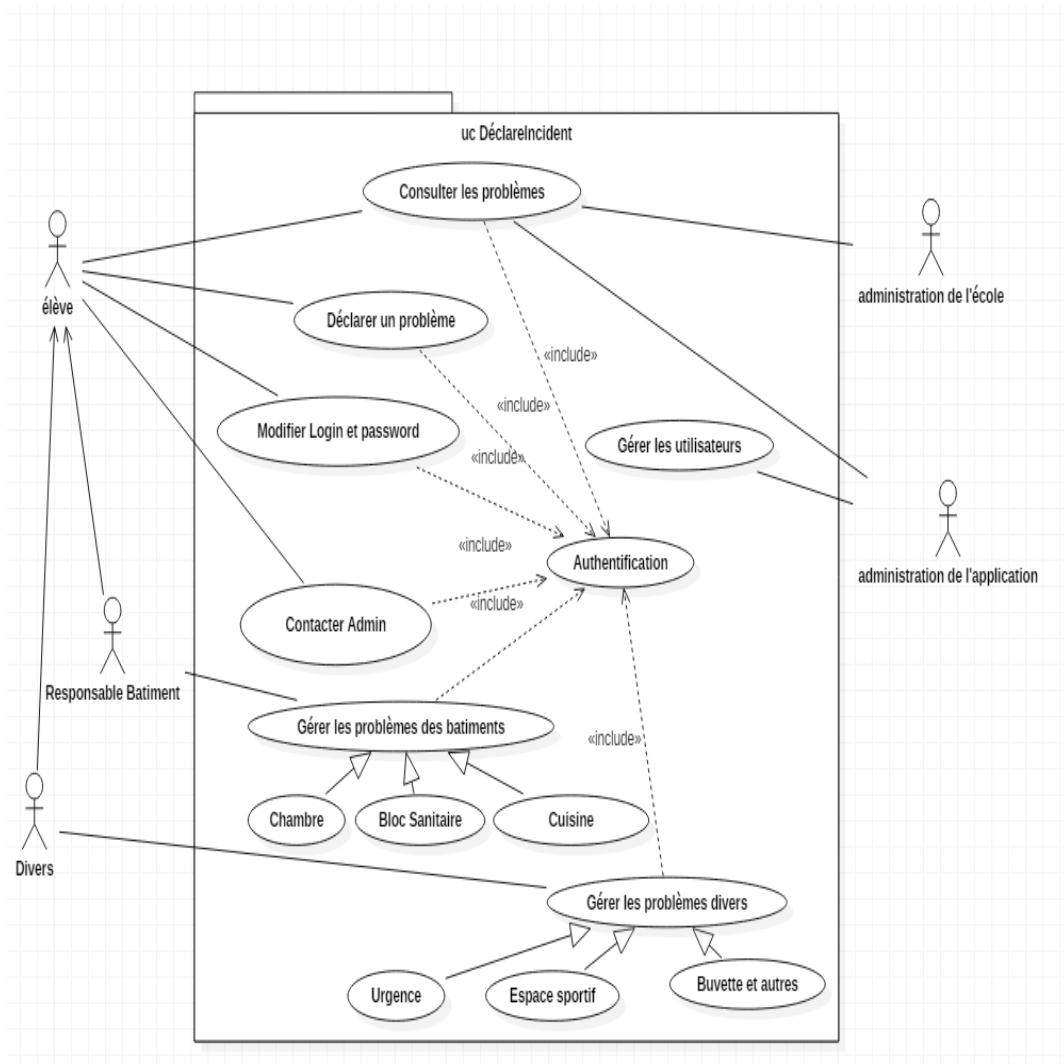


FIGURE 2.1 – Diagramme de cas d'utilisation

2.3.2 Côté élève

Le diagramme de cas d'utilisation montre l'interaction entre l'élève et le système. On distingue donc trois services dans ce diagramme :

- Gérer le Compte.
- Publier le problème.
- Visualiser les problèmes.
- Contacter les membres de comités.

Il faut que l'élève soit déjà authentifié pour profiter de ces services.

Description du cas d'utilisation <Gérer le compte>

L'élève doit initialement se connecter au site web à travers son email et son mot de passe. Puis, il clique sur le bouton profil pour voir l'ensemble des informations à remplir. Le tableau ci-dessous décrit la démarche de la gestion de compte par l'élève dans le site web.

SOMMAIRE	
Titre :	Gestion du profil
But :	Gérer l'ensemble d'informations et la possibilité de les modifier
Résumé :	L'élève clique sur profil, l'action se déclenche et un formulaire s'affiche.
Acteur :	Élève
Pré conditions	Post conditions
-Élève est authentifié	
Scénario nominal	
1-L'élève se connecte au site web à travers son email et un mot de passe. 2-L'élève clique sur profil et un formulaire s'affiche.	
Enchainement alternatif	
-L'élève n'a pas validé son choix. -Le site web affiche un message de confirmation.	

TABLE 2.1 – Description du cas d'utilisation <Gérer le profil >

Description du cas d'utilisation < Publier un problème >

L'élève doit initialement désigner son problème afin de le publier. Le tableau ci-dessous décrit la démarche de la publication du problème.

SOMMAIRE	
Titre :	publier problème
But :	Publication des différentes problèmes afin d'être résolus.
Résumé :	L'élève clique sur « problems », l'action se déclenche et l'élève reçoit un formulaire pour désigner son problème.
Acteur :	Élève
Pré conditions	Post conditions
-Élève est authentifié	
Scénario nominal	
1-L'élève clique sur <problems>. 2-L'élève remplit le formulaire.	
Enchainement alternatif	
-L'élève n'a pas rempli son choix. -Le site web affiche un message d'erreur.	

TABLE 2.2 – Description du cas d'utilisation < Publier un problème >

2.3.3 Côté comité

Le diagramme de cas d'utilisation montre l'interaction entre le membre de comité et le système. On distingue donc des différents services dans ce diagramme :

- Gérer le Compte.
- Visualiser les statistiques.
- Gérer les problèmes.
- Visualiser l'archive.
- Visualiser les problèmes en cours.
- Visualiser la liste des étudiants.
- Publier un problème.

Description du cas d'utilisation <Gérer les problèmes >

Le membre de comité doit initialement se connecter au site web à travers son email et son mot de passe. Puis, il sélectionne les problèmes à résoudre pour enfin répondre aux besoins de l'élève.

SOMMAIRE	
Titre :	Gérer les problèmes
But :	Gestion des problèmes des étudiants.
Résumé :	Le membre clique sur <Problems>, résout le problème et affecte un statut pour chaque problème.
Acteur :	Membre de comité
Pré conditions	Post conditions
-Élève est authentifié	
Scénario nominal	
1-Le membre se connecte au site web à travers son email et un mot de passe. 2-Le membre sélectionne les problèmes à résoudre. 3-Le membre affecte un statut pour chaque problème.	
Enchainement alternatif	
-Le membre n'a pas sélectionné un problème. -Le site web affiche un message d'erreur.	

TABLE 2.3 – Description du cas d'utilisation < Gérer les problèmes >

Description du cas d'utilisation <Visualiser statistiques>

Le membre doit initialement se connecter au site web à travers son email et son mot de passe. Puis, il clique sur <Statistics> et quatre boutons s'affichent. Le tableau ci-dessous décrit la démarche de la visualisation des statistiques.

SOMMAIRE	
Titre :	Postuler à un emploi
But :	Envoyer la candidature de recrutement au recruteur qui a publié l'offre.
Résumé :	Le candidat clique sur le bouton postuler et sa candidature est envoyée directement au recruteur.
Acteur :	Candidat
Pré conditions	Post conditions
-Le candidat est authentifié -L'emploi doit être déjà publié par un recruteur.	
Scénario nominal	
1-Le candidat se connecte à travers son email et un mot de passe. 2-Le candidat choisit l'offre convenable. 3-Le candidat clique sur le bouton postuler et sa candidature est envoyée au recruteur.	
Enchainement alternatif	
-L'offre d'emploi n'existe plus. -Le site web affiche un message d'erreur.	

TABLE 2.4 – Description du cas d'utilisation <Postuler un emploi>

2.4 Conception

Diagrammes de séquence

Les diagrammes de séquence peuvent servir à illustrer les cas d'utilisations décrits précédemment. Ils permettent de représenter la succession chronologique des opérations réalisées par un acteur et qui font passer d'un objet à un autre pour représenter un scénario. Initialement, l'utilisateur s'inscrit en tant que recruteur pour bénéficier des différents services disponibles dans le site web.

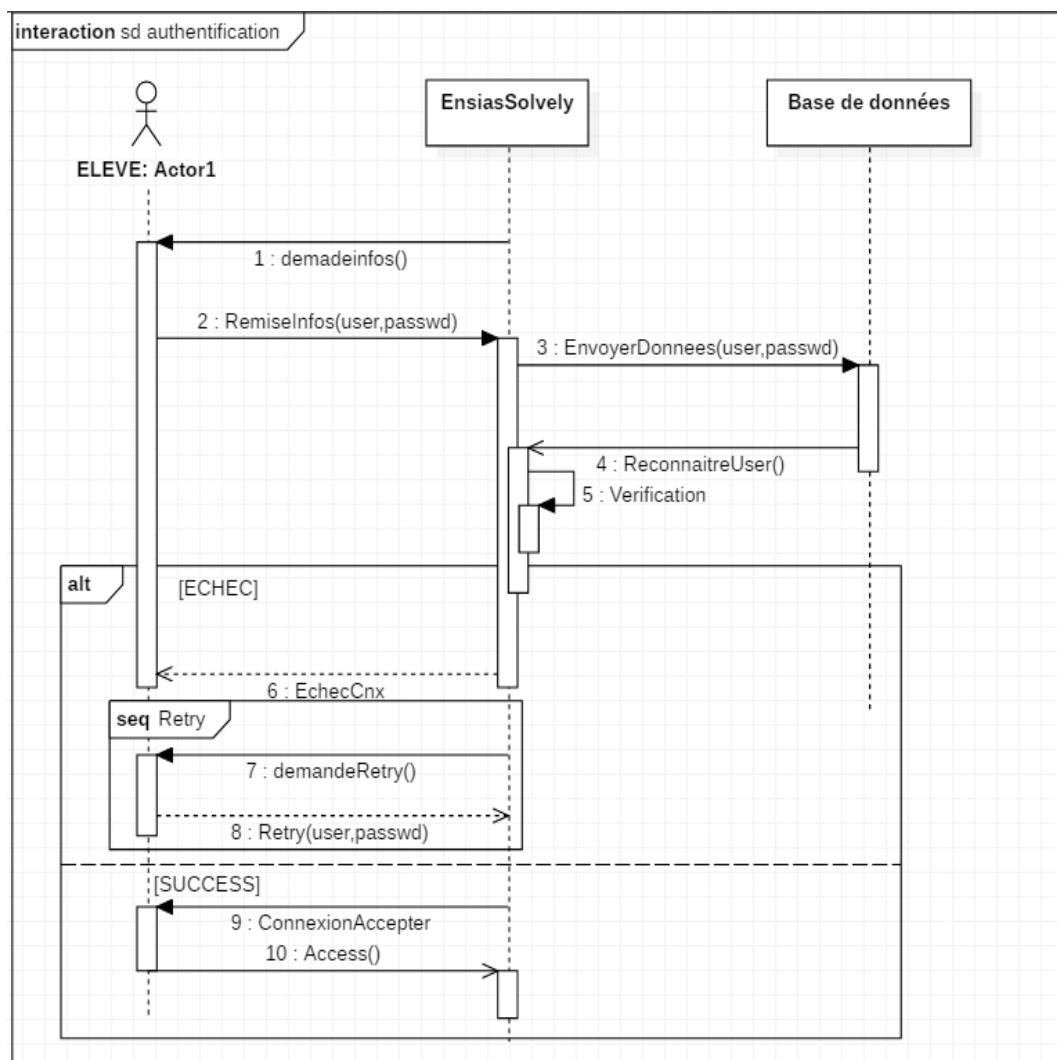


FIGURE 2.2 – Diagramme de séquence d'authentification

L'élève peut ainsi déclarer le problème rencontré la situation se résume comme suit.

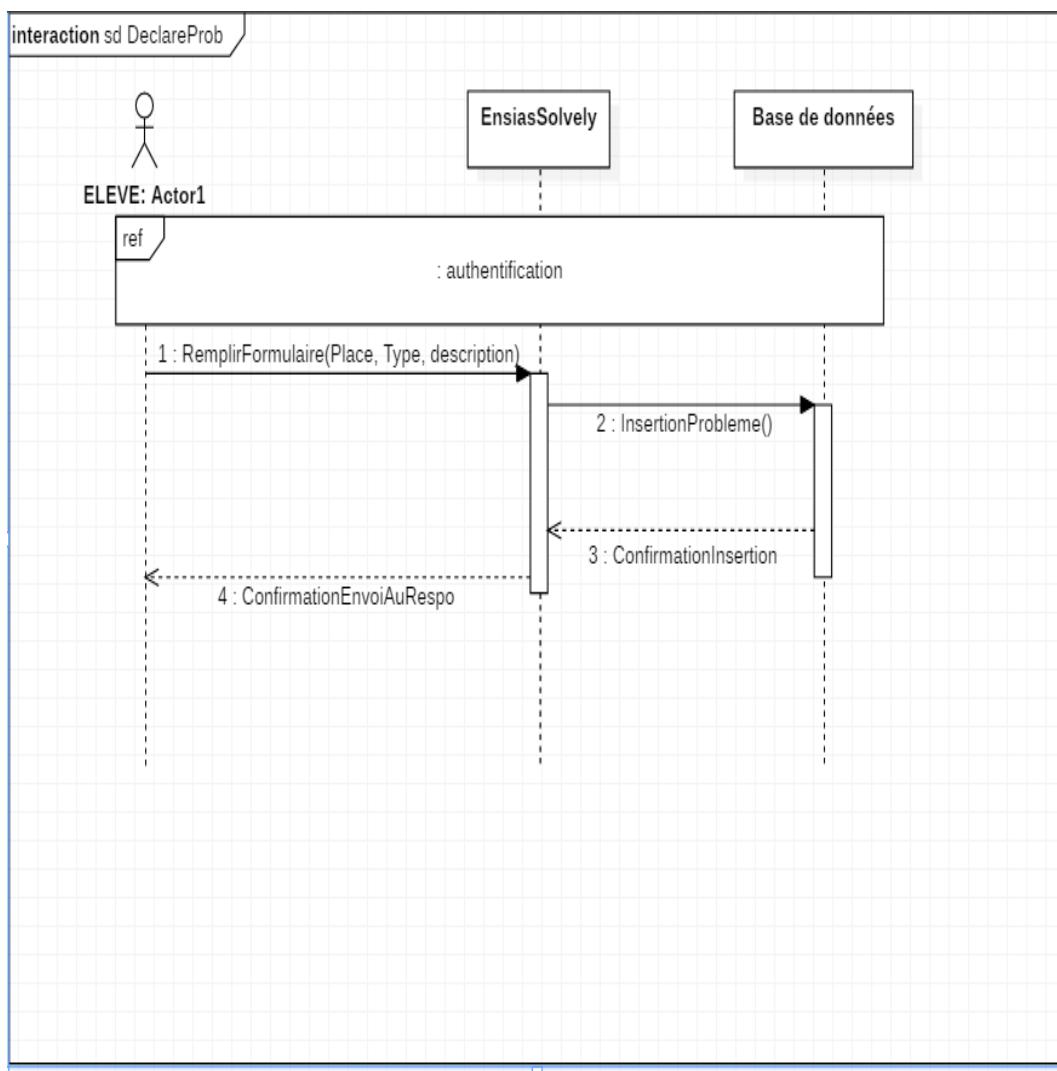


FIGURE 2.3 – Diagramme de séquence de déclaration de problème

Le membre de comité doit régler dans les plus brefs délais le problème rencontré par l'étudiant.

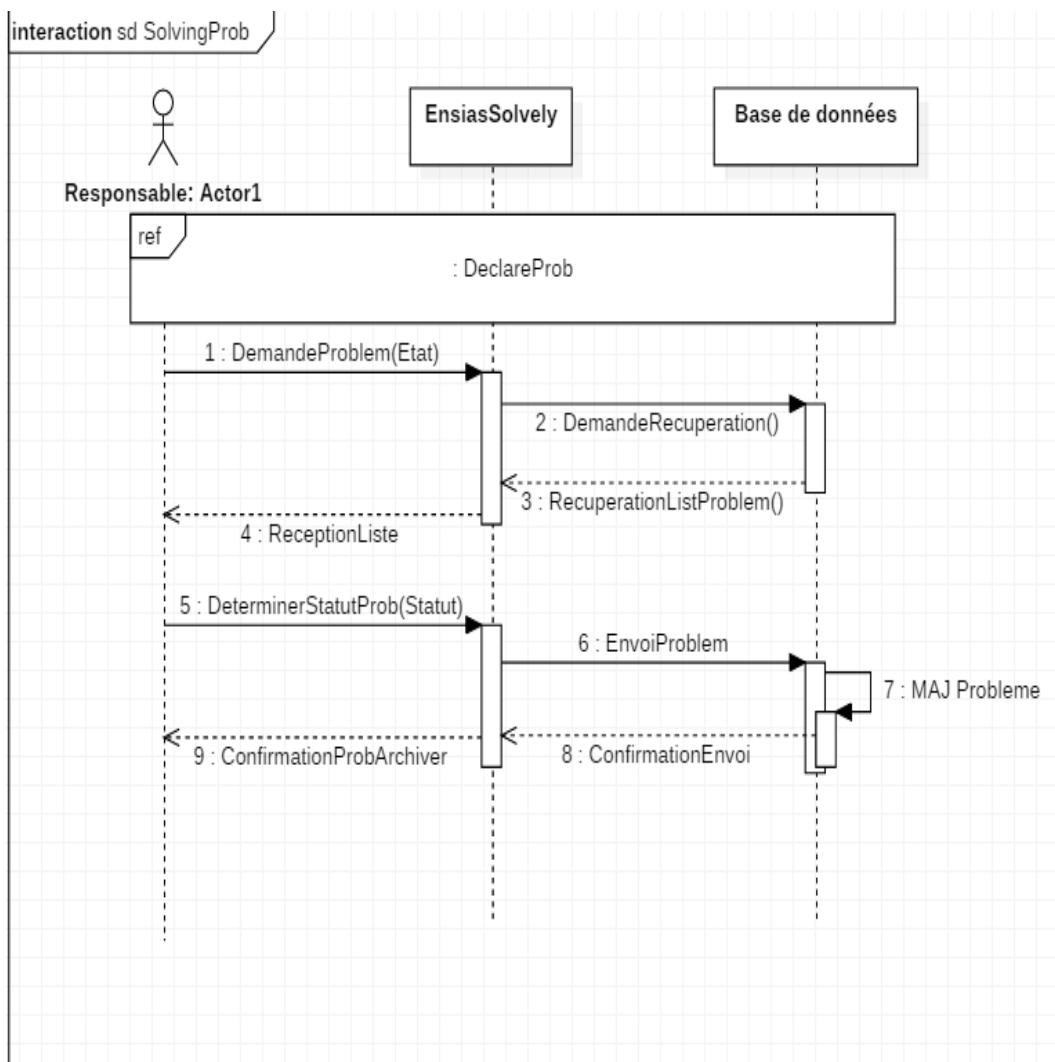


FIGURE 2.4 – Diagramme de séquence de résolution de problème

Puis, l'élève peut visualiser son archive.

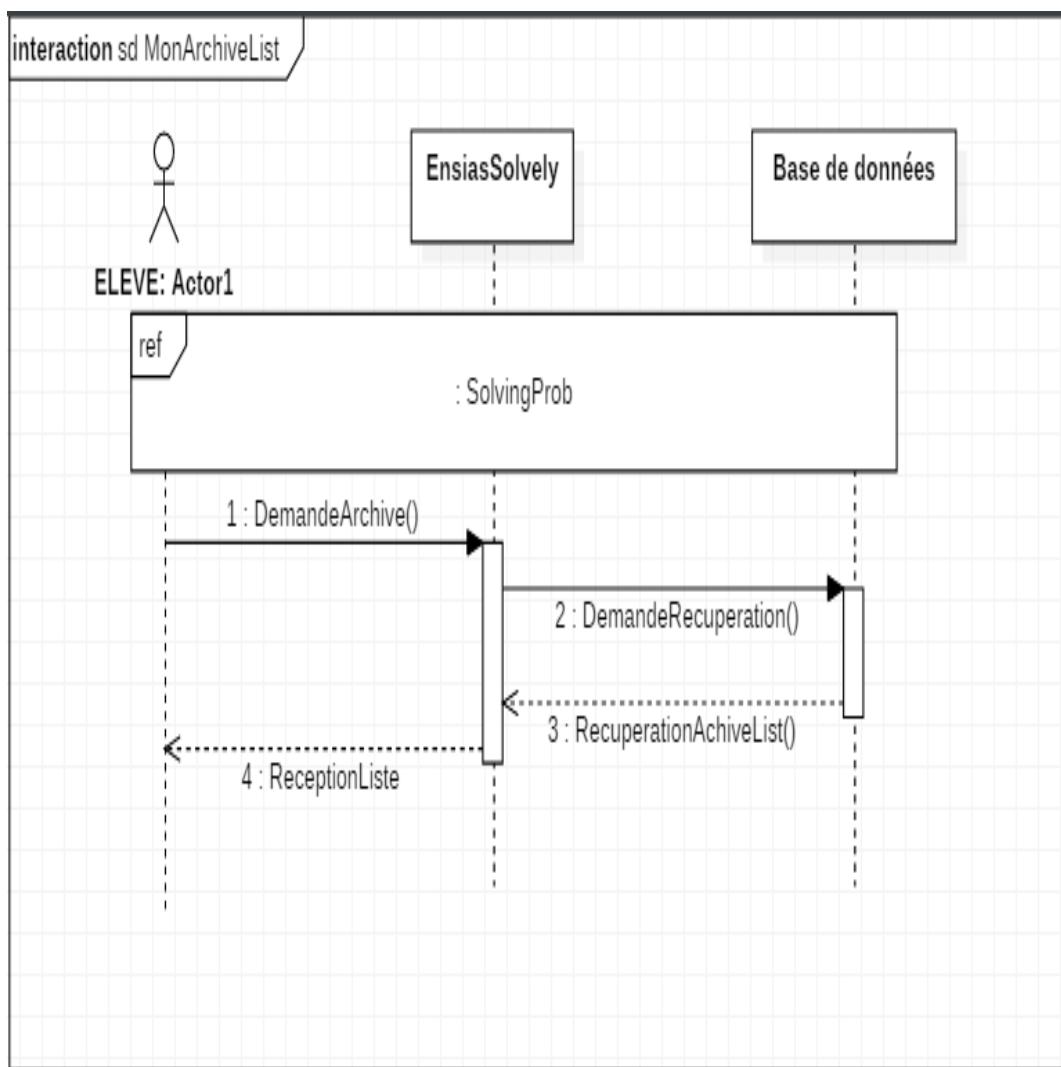


FIGURE 2.5 – Diagramme de séquence de visualisation d’archive de l’élève

L'élève peut visualiser clairement son profil, ainsi qu'il a la possibilité de modifier quelques informations.

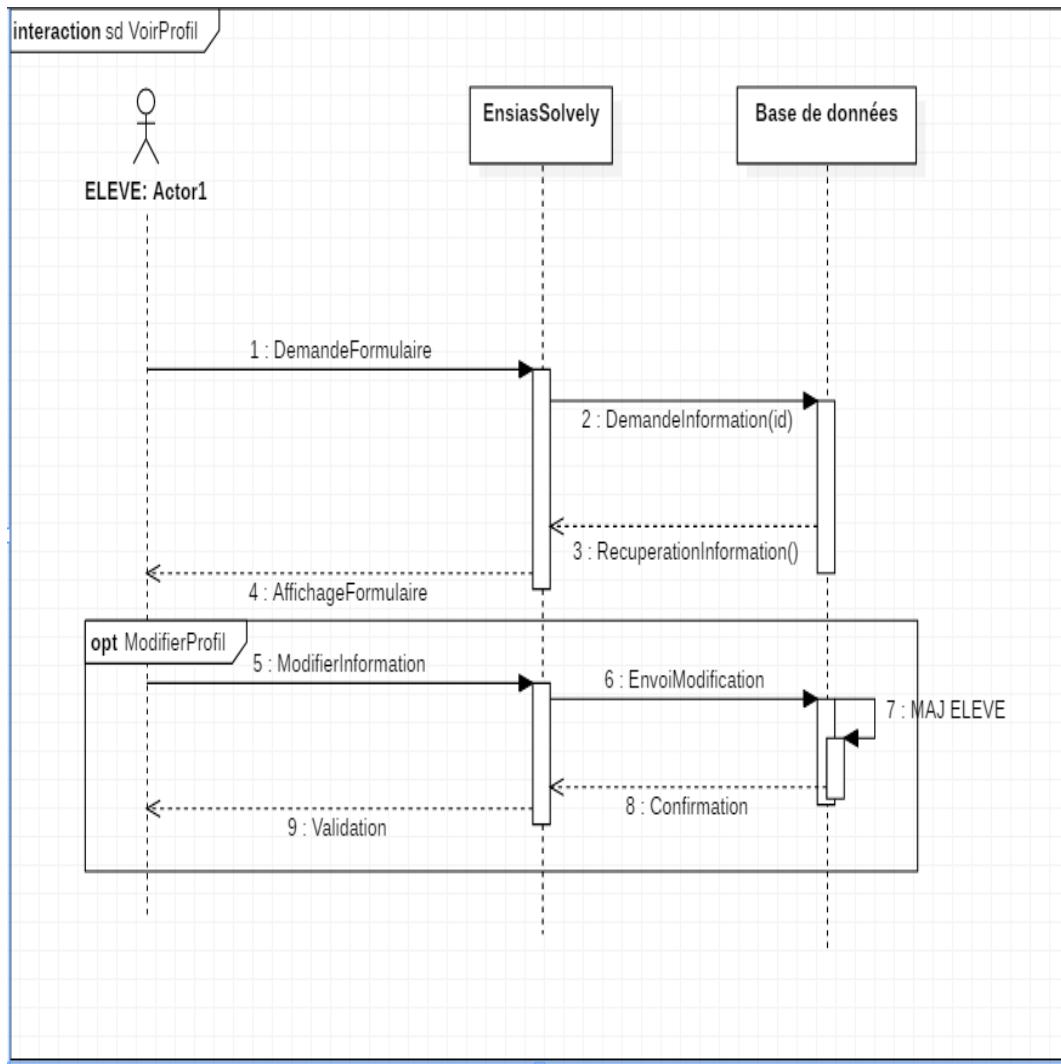


FIGURE 2.6 – Diagramme de séquence de visualisation de profil

Le membre de comité peut ainsi voir l'archive de l'ensemble des étudiants .

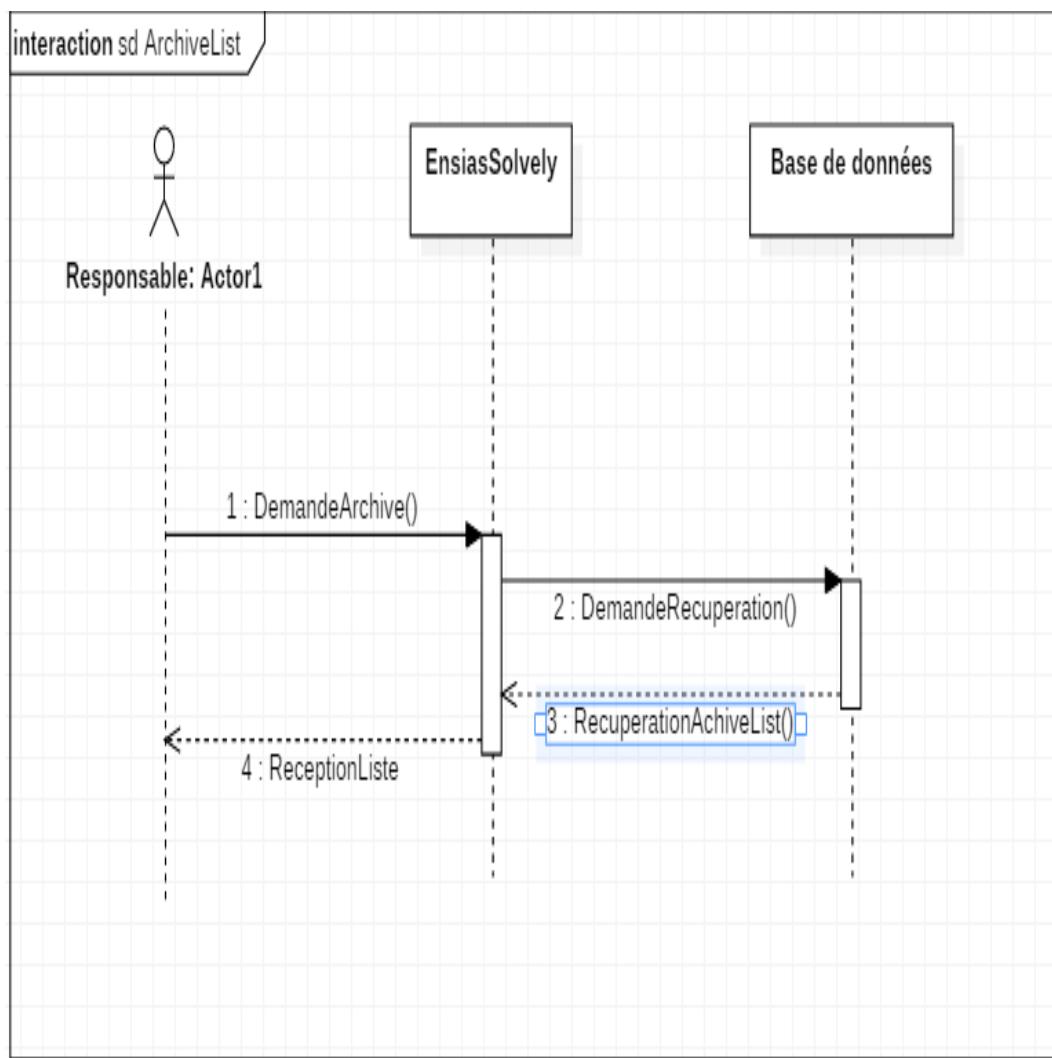


FIGURE 2.7 – Diagramme de séquence de visualisation de l'archive de tous les étudiants

Le membre de comité peut ainsi visualiser la liste des différents étudiant.

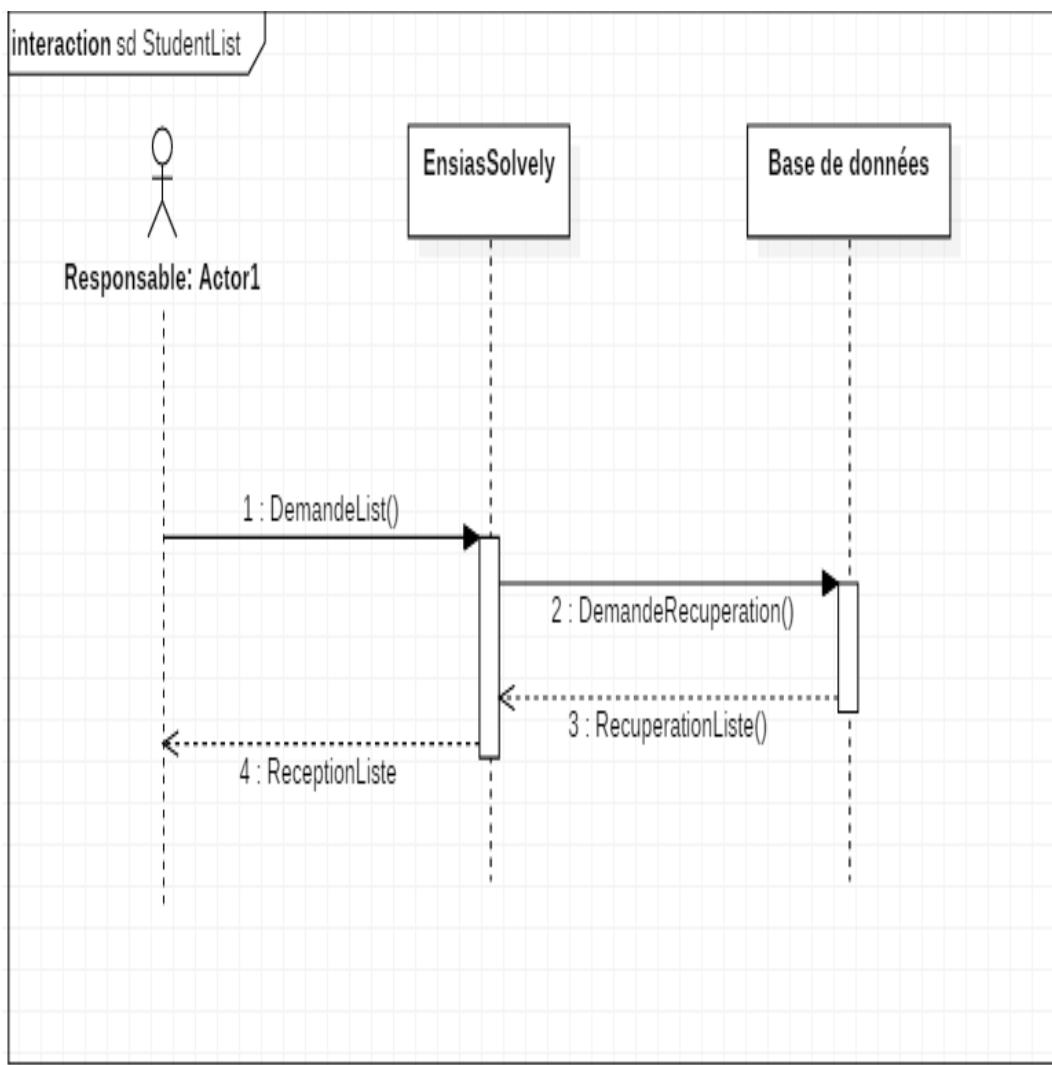


FIGURE 2.8 – Diagramme de séquence de visualisation de la liste des étudiants

Diagramme d'états-transition

L'élève peut passer par des différents statuts lors de son existence à l'école, cela se résume dans le diagramme suivant :

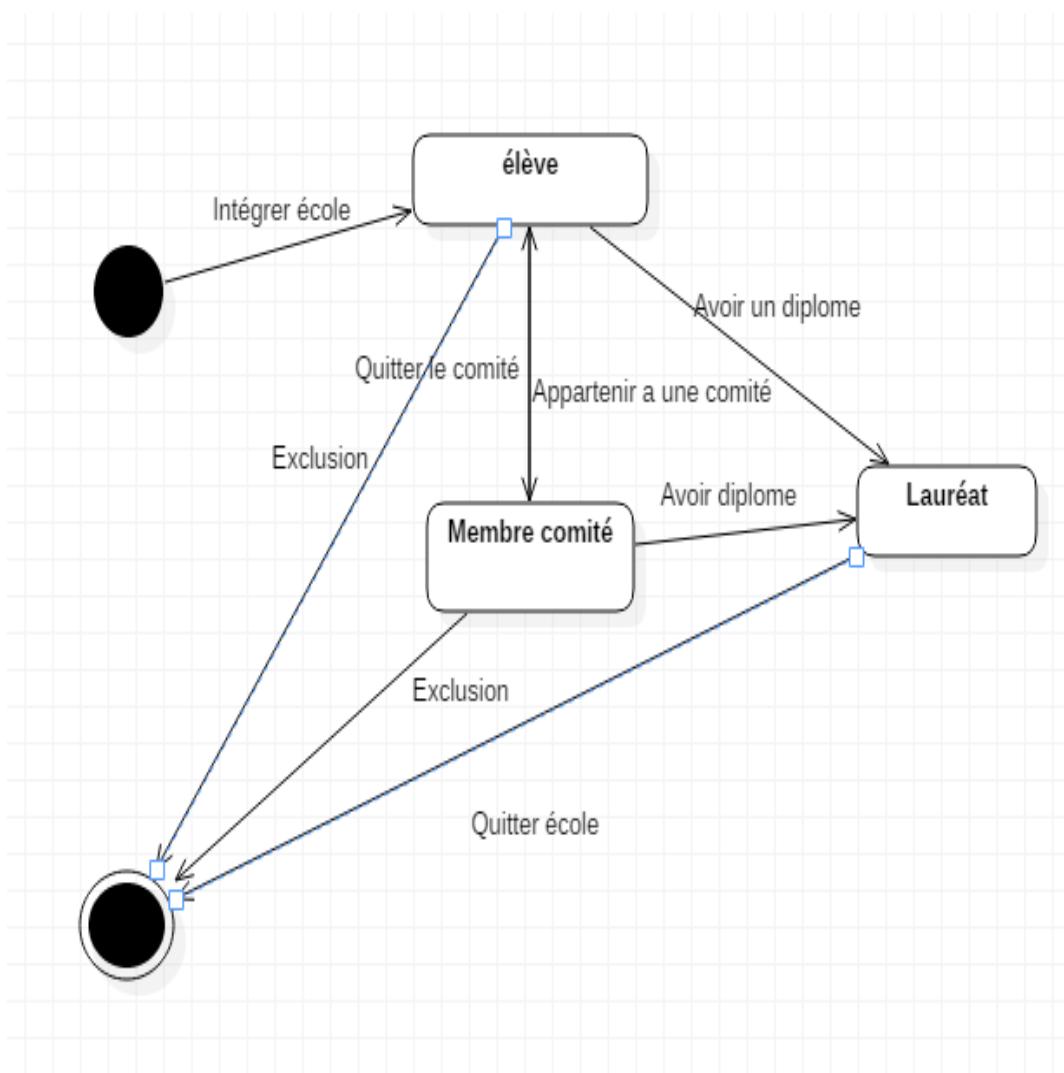


FIGURE 2.9 – Diagramme d'états transitions

Chapitre 3

Création de la base de données

3.1 Introduction

Dans ce chapitre, nous allons décrire les étapes de création de la base de données.

3.2 Diagramme de classe

Le traitement de la base de données représente un grand défi pour la plupart des site-web, c'est pour cela que nous avons pris beaucoup de temps afin d'optimiser notre travail et d'extraire les tables nécessaires. Pour cela, nous avons établis le diagramme de classe afin de faciliter notre travail.

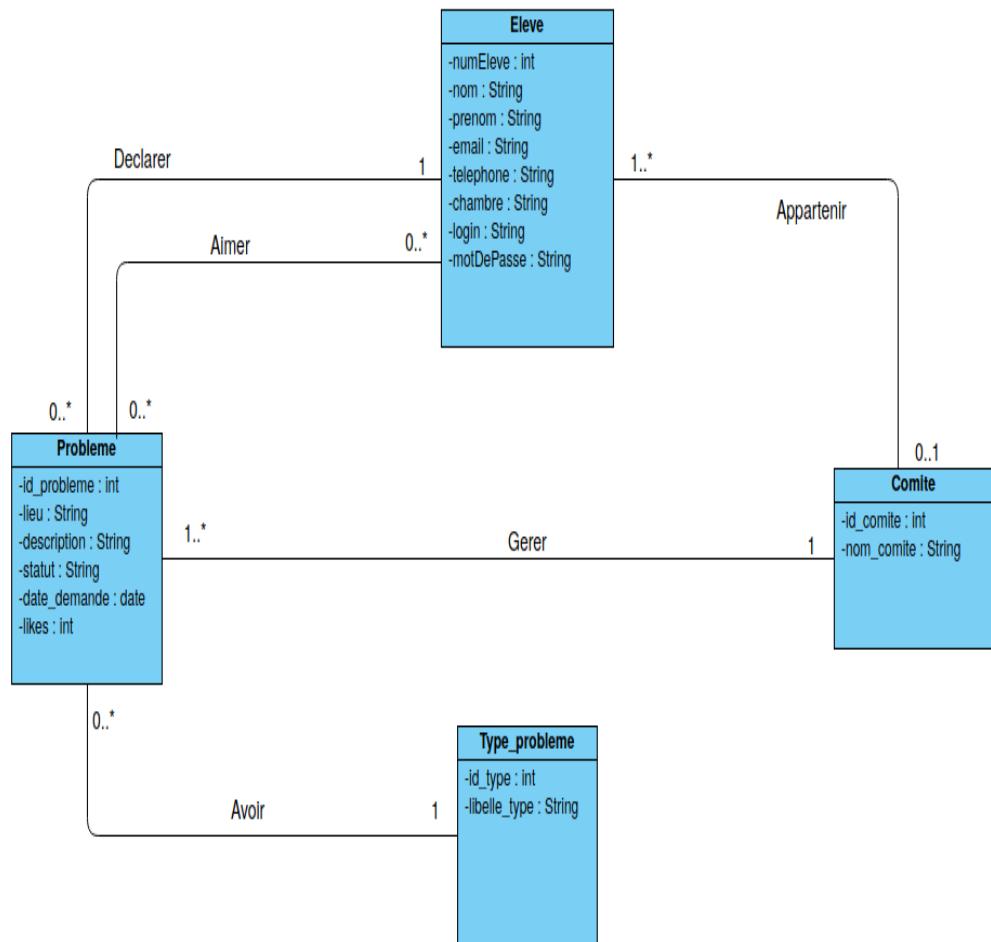


FIGURE 3.1 – Diagramme de classe

3.3 Tables de la base de données

eleve

Colonne	Type	Null	Valeur par défaut	Commentaire
numEleve(primaire)	int(11)	Non		
nom	varchar(50)	Oui	Null	
prenom	varchar(50)	Oui	Null	
email	varchar(50)	Oui	Null	
telephone	varchar(50)	Oui	Null	
chambre	varchar(50)	Oui	Null	
login	varchar(50)	Oui	Null	
motDePasse	varchar(100)	Oui	Null	
id_comite	int(11)	Oui	Null	

TABLE 3.1 – Table eleve

probleme

Colonne	Type	Null	Valeur par défaut	Commentaire
id_probleme(primaire)	int(11)	Non		
lieu	varchar(100)	Oui	Null	
description	varchar(300)	Oui	Null	
date_demande	date	Oui	Null	
statut	varchar(50)	Oui	Null	
id_comite	int(11)	Non		
login	varchar(50)	Non		
motDePasse	varchar(100)	Non		
numEleve	int(11)	Non		
id_type	int(11)	Non		
likes	int(11)	Non	0	

TABLE 3.2 – Table probleme

comite

Colonne	Type	Null	Valeur par défaut	Commentaire
id_comite(primaire)	int(11)	Non		
nom_comite	varchar(100)	Oui	Null	

TABLE 3.3 – Table comite

boite

Colonne	Type	Null	Valeur par défaut	Commentaire
numboite(primaire)	int(11)	Non		
message	varchar(300)	Oui	Null	
numEleve	int(11)	Oui	Null	

TABLE 3.4 – Table boite

aimer

Colonne	Type	Null	Valeur par défaut	Commentaire
id_probleme	int(11)	Non		
numEleve	int(11)	Non		

TABLE 3.5 – Table aimer

3.4 Conclusion

La bonne gestion de la base de données est le plus grand défi que rencontre chaque programmeur, c'est pour cela qu'il faut la manipuler avec prudence afin de bien gérer la totalité du travail et pour passer à la partie de la réalisation d'un site web dynamique.

Chapitre 4

Réalisation

4.1 Introduction

Cette partie constitue le dernier volet de ce rapport. Après avoir terminé la phase de spécification et conception, nous présentons l'environnement matériel et logiciel utilisés pour le développement de notre site web.

4.2 Environnement Logiciel

- Eclipse : Eclipse est l'Environnement de Développement Intégré (ou IDE) le plus largement utilisé pour la programmation Java, très performant, il est de plus gratuit et open source.
- WampServer : WampServer est une plate-forme de développement Web sous Windows pour des applications Web dynamiques à l'aide du serveur Apache2, du langage de scripts PHP et d'une base de données MySQL. Il possède également PHPMyAdmin pour gérer plus facilement les bases de données.
- StarUML : C'est un logiciel de modélisation UML utilisé dans la conception de diagrammes et de figures.
- Apache Tomcat : C'est un conteneur web libre de servlets et JSP Java EE. Issu du projet Jakarta, c'est un des nombreux projets de l'Apache Software Foundation. Il implémente les spécifications des servlets et des JSP, est paramétrable par des fichiers XML et des propriétés, et inclut des outils pour la configuration et la gestion. Il comporte également un serveur HTTP.
- Jasper report : JasperReports est un outil de reporting open source, offert sous forme d'une bibliothèque qui peut être embarquée dans tous types d'applications Java.

4.3 Technologies utilisées

- JAVA : Java est un langage de programmation orienté objet, développé par Sun Microsystems et destiné à fonctionner dans une machine virtuelle, il permet de créer des logiciels compatibles avec des nombreux systèmes d'exploitation.
- XML : C'est un langage nécessaire pour développer une application Android. XML n'est pas un langage de programmation mais c'est un langage informatique de balisage générique. Il sert essentiellement à stocker/transférer des données de type texte Unicode structurées en champs arborescents.
- JEE : Cette plate-forme J2EE est une suite robuste de services middleware qui

simplifie la tâche des développeurs d'applications côté serveur. Elle repose sur les technologies existantes de la plate-forme J2SE afin de faciliter la création d'applications.

4.4 Architecture de l'application

4.4.1 Client / Serveur

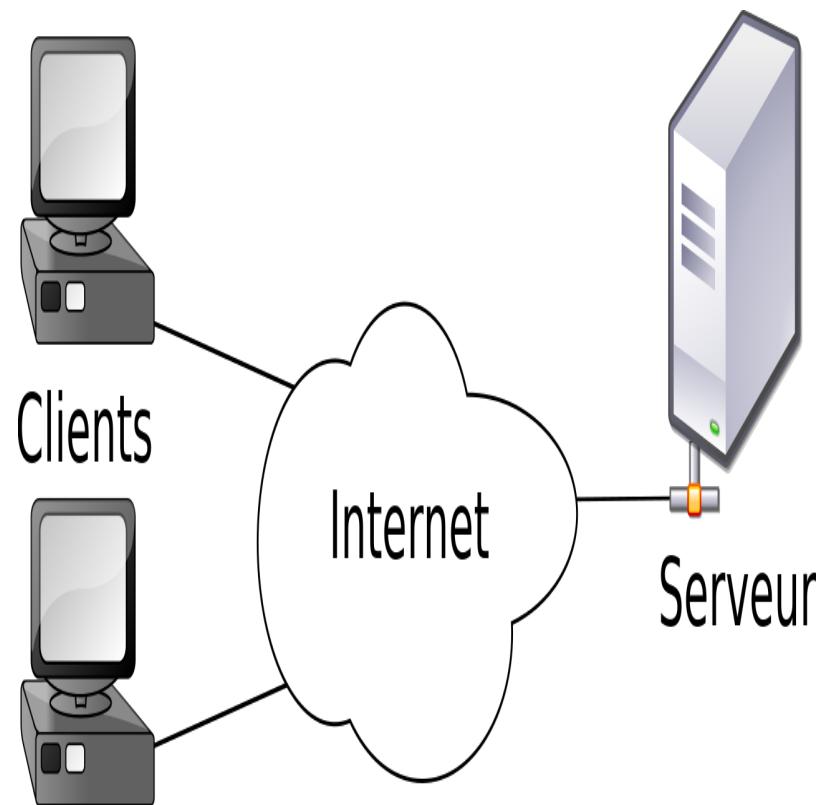


FIGURE 4.1 – Connexion Clients/Serveur [7]

Dans l'architecture à trois niveaux, les applications au niveau serveur sont délocalisées, c'est- à-dire que chaque serveur est spécialisé dans une tâche (serveur web/ serveur de base de données par exemple). Il permet :

- Une plus grande flexibilité et souplesse.
- Une sécurité accrue car la sécurité peut être définie indépendamment pour chaque service, et à chaque niveau.
- Une meilleure performance, étant donné le partage des tâches entre les différents serveurs.

Protocole de communication :

Dans notre projet, nous avons utilisé le protocole HTTP, afin de communiquer les données entre la partie client et le serveur web. Le HTTP est un protocole qui définit la communication entre un serveur et un client. Nous utilisons la méthode Post pour envoyer des données au programme situé à une URL spécifiée.

4.4.2 Architecture MVC

Le modèle MVC décrit une manière d'architecturer une application informatique en la décomposant en trois sous-parties :

- La partie Modèle.
- La partie Vue.
- La partie Contrôleur.

Voici à la figure suivante le schéma représentant l'architecture MVC :

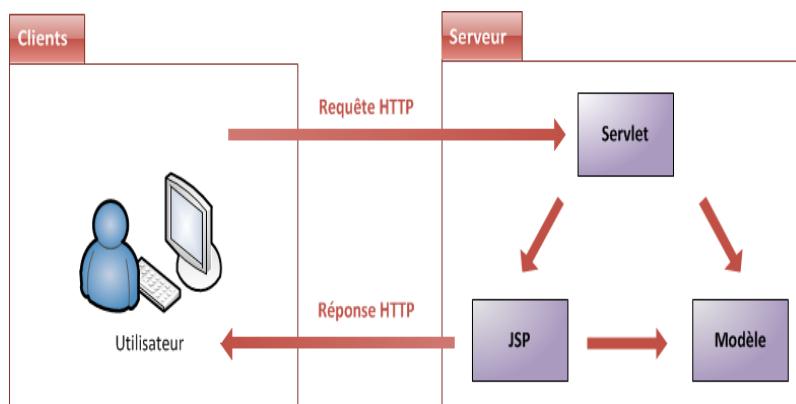


FIGURE 4.2 – Architecture MVC [7]

Nous avons donc codé notre site en mettant en place ce modèle :

- création des beans qui enregistrent les données saisies et validées.
- création des objets comportant les méthodes de récupération/conversion/validation des contenus des champs des formulaires du site web.
- les servlets n'interviennent plus directement sur les données des requêtes, mais aiguillent simplement les requêtes entrantes.
- Les JSP sont adaptées au modèle MVC.

4.5 Interfaces de l'application

Dans cette partie on décrit les différents interfaces de notre application web.

La page d'accueil est comme suit :

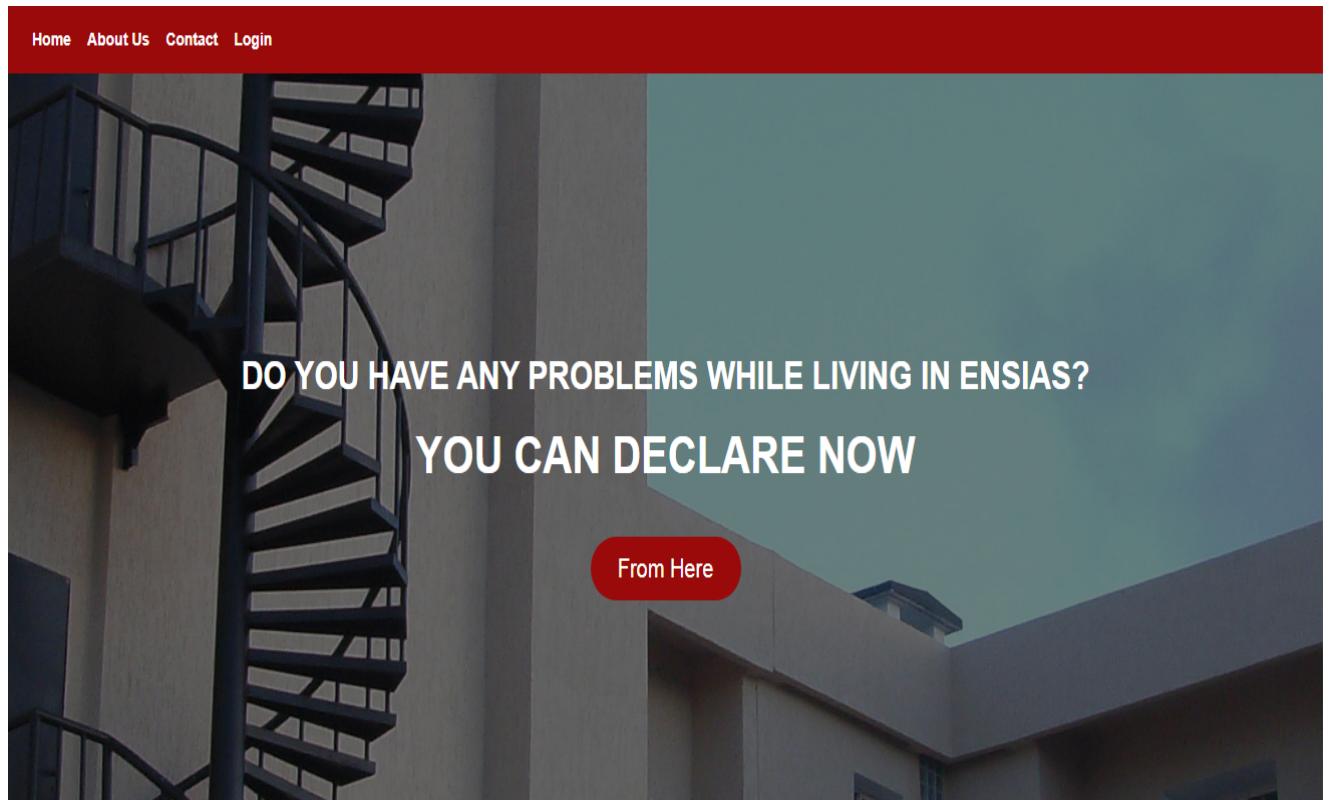


FIGURE 4.3 – Page d'accueil

L'utilisateur peut donc se connecter en cliquant sur 'Login' ou bien 'From Here'.

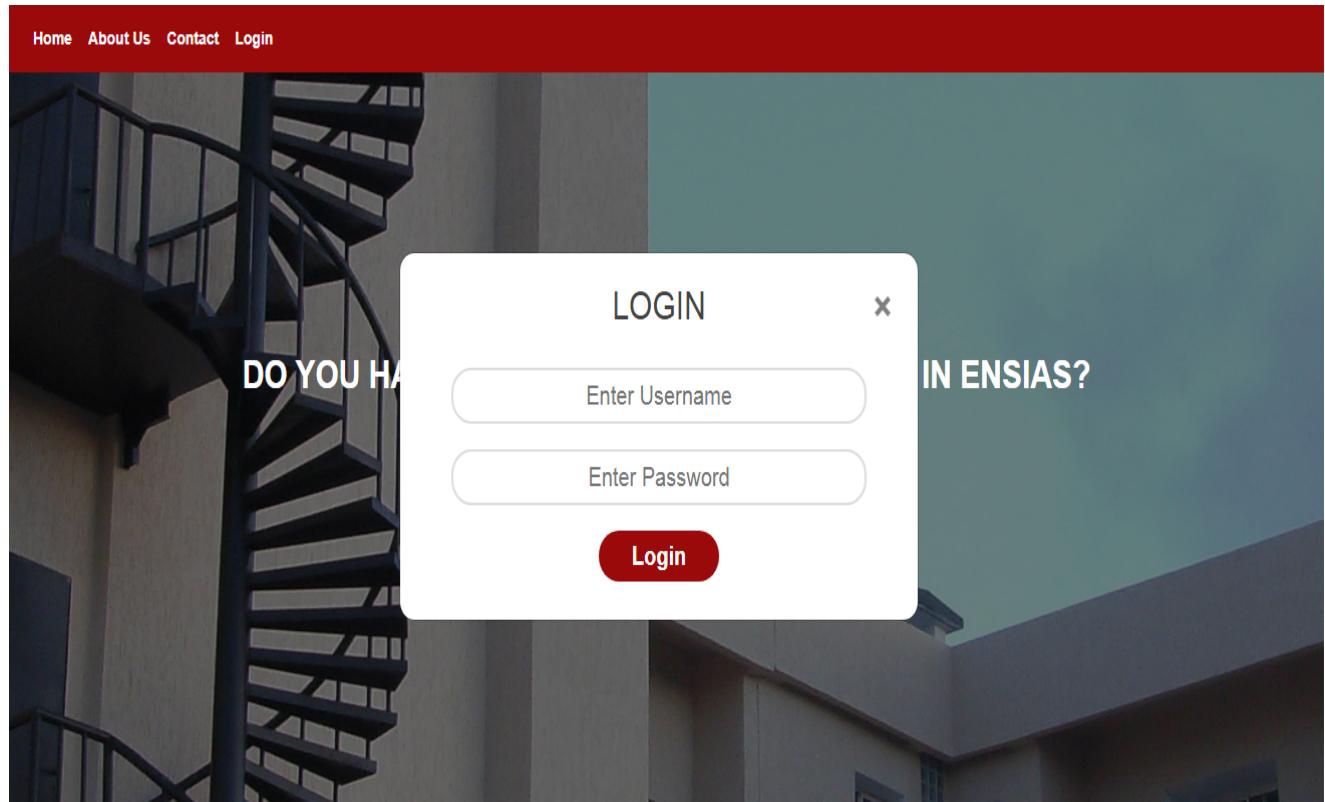


FIGURE 4.4 – Fenêtre de login

Si l'utilisateur est membre d'une comité, alors il se connecte en mode admin .

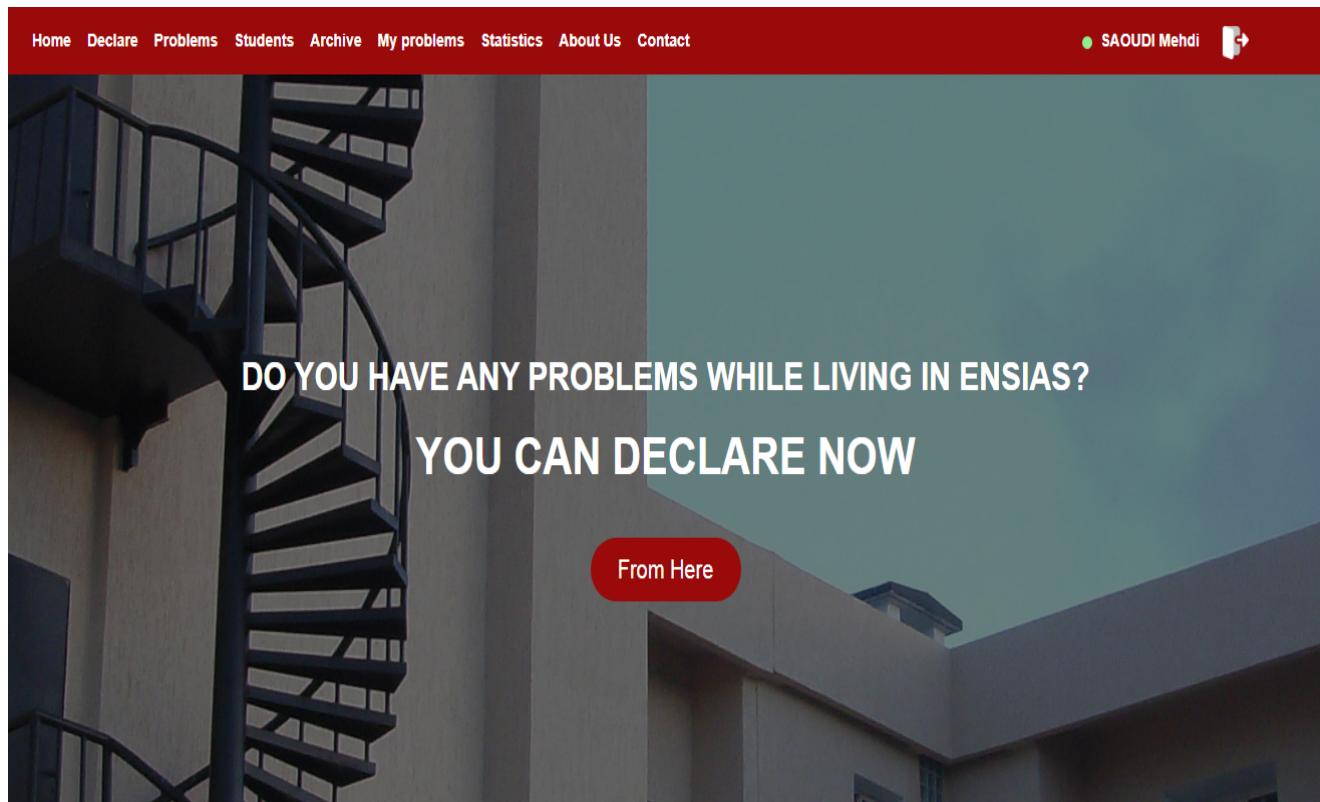


FIGURE 4.5 – Interface de l'admin

Un élève peut déclarer un problème en cliquant sur 'Declare' ou bien 'From Here'.

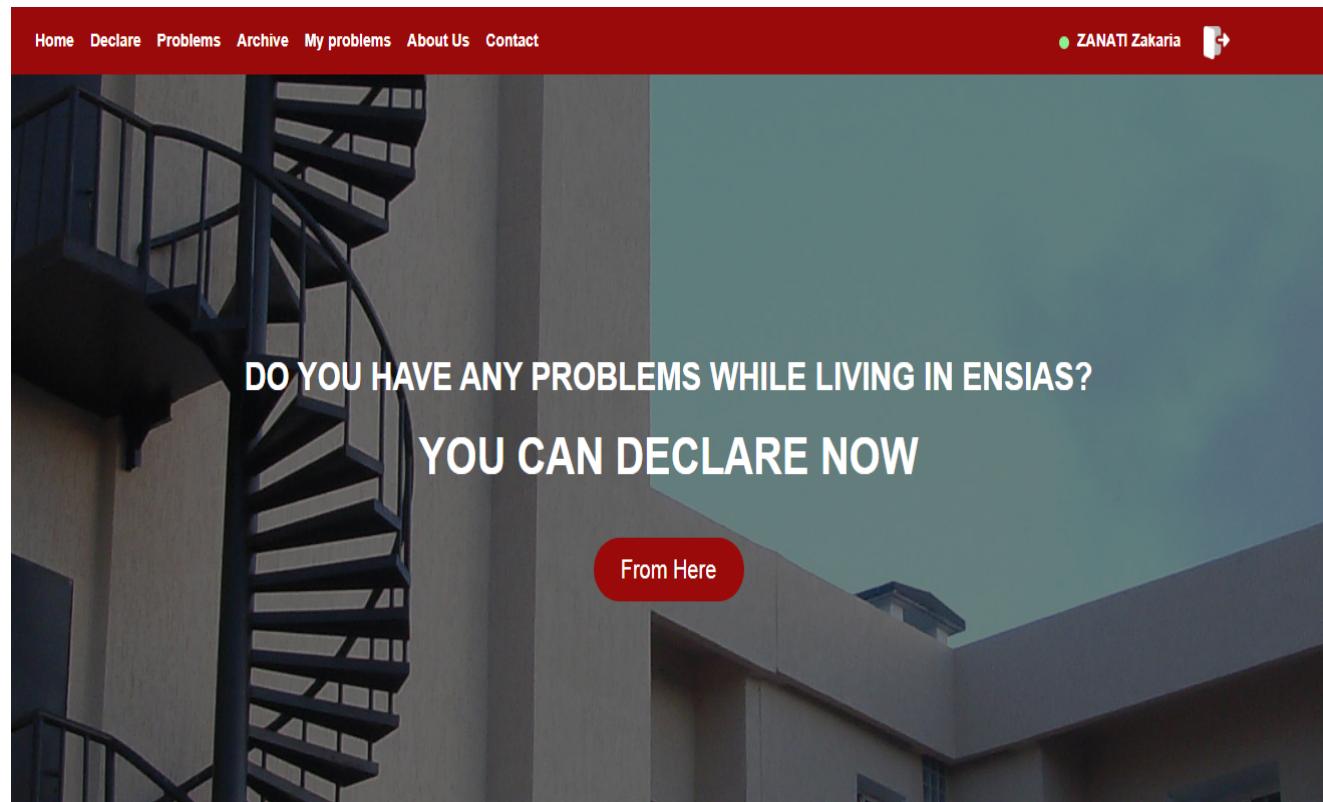


FIGURE 4.6 – Interface de l'élève

Il saisit en suite les détail de son problème puis il l'envoie.

The screenshot shows a web application interface for reporting a problem. At the top, there is a red header bar with navigation links: Home, Declare, Problems, Archive, My problems, About Us, and Contact. On the right side of the header, there is a user profile icon with the name "ZANATI Zakaria" and a small arrow icon. Below the header, the main content area has a title "Declare a problem". The form consists of several input fields arranged in a grid:

Student number	6
Last name	ZANATI
First name	Zakaria
Place	(empty)
Type	Kitchen
Description	(empty)
Date demande	07/02/2020

At the bottom of the form is a red "Send" button.

FIGURE 4.7 – Page de déclaration de problèmes

Tous les élèves ont le droit de consulter la liste de tous les problèmes déclaré. Un élève peut cliquer sur 'j'aime' s'il a le même problème qu'un autre élève qui l'a déjà déclaré, plus un problème a des "j'aime" plus il monte en haut et devient plus prioritaire. Avec cette fonction on évitera d'avoir plusieurs problèmes avec la même description.

Id problem	Place	Date demande	type	Last name	First name	Description	Statut
36	Chambre B54	2020-02-02	CHAMBRE	ZANATI	Zakaria	Fuite d'eau	Actif 

FIGURE 4.8 – Liste des problèmes de tous les élèves

Home Declare Problems Archive My problems About Us Contact

ZANATI Zakaria

My problems

Id problem	Place	Date demande	type	Last name	First name	Description	Statut
38	Terrain de foot	2020-02-02	ESPACE SPORTIF	ZANATI	Zakaria	Terrain de foot occupé depuis plus de 3 heures	Not Solved
36	Chambre B54	2020-02-02	CHAMBRE	ZANATI	Zakaria	Fuite d'eau	Actif

FIGURE 4.9 – Liste des problèmes d'un élève

Chaque élève a le droit de modifier ses informations personnelles.

Student number	6
Last name	ZANATI
First name	Zakaria
Email	zakariazanati@gmail.com
Phone	656852145
Room	C50
Committe id	3
Committe name	Elève
Login	Zakaria
Password 

Apply

FIGURE 4.10 – Profil de l'élève

Un admin peut changer l'état d'un problème en 'solved' ou bien 'not solved'.

Id problem	Place	Date demande	type	Last name	First name	Description	Statut
36	Chambre 854	2020-02-02	CHAMBRE	ZANATI	Zakaria	Fuite d'eau	Actif

FIGURE 4.11 – Liste de tous les problèmes

Il peut consulter les problèmes urgent en cliquant sur 'type'.

Home Declare Problems Students Archive My problems Statistics About Us Contact

SAOUDI Mehdi Print

List of problems

Id problem Solved apply

Id problem	Place	Date demande	Urgent	Last name	First name	Description	Statut
41	B56	2020-02-02	URGENCE	YOUNESS	Mouad	Maladie herpes grave	Actif 1

FIGURE 4.12 – Liste des problèmes urgent

Il peut aussi consulter l'archive de tous les problèmes déjà déclaré.

The screenshot shows a web application interface with a red header bar containing navigation links: Home, Declare, Problems, Archive, My problems, About Us, and Contact. To the right of these links are icons for ZANATI Zakaria and a user profile. A button labeled 'Statistics' is also visible. The main content area has a title 'Problems of the year :'. Below it is a table with the following data:

Id problem	Place	Date demande	type	Last name	First name	Description	Committe	Statut
40	Grand amphi	2020-02-02	BUVETTES ET AUTRES	SAOUDI	Mehdi	Problème d'éclairage	divers	Solved
39	Terrain de basketball	2020-02-02	ESPACE SPORTIF	TENNIA	Youssef	Pas de panneau depuis 2 mois	divers	Solved
38	Terrain de foot	2020-02-02	ESPACE SPORTIF	ZANATI	Zakaria	Terrain de foot occupé depuis plus de 3 heures	divers	Not Solved
37	Cuisine Bâtiment A	2020-02-02	CUISINE	TENNIA	Youssef	Problème d'hygiène	bâtiment	Solved

FIGURE 4.13 – Page d'archive

Il peut récupérer la liste des problèmes et la liste globale des étudiants.



Ensias Solvely

02:30 février 07, 2020

Liste globale des problèmes selon l'année : 2020

N°Problème	Lieu	Date	Statut	N°Eleve
36	Chambre B54	02/02/2020	Actif	6
37	Cuisine Batiment	02/02/2020	Solved	7
38	Terrain de foot	02/02/2020	Not Solved	6
39	Terrain de	02/02/2020	Solved	7
40	Grand amphi	02/02/2020	Solved	5

FIGURE 4.14 – Liste des problèmes de l'année en cours



Liste globale des étudiants

N°	Nom	Prenom	Email	Téléphone	N°Chambre
5	SAOUDI	Mehdi	mehdisaoudi270@gmail.com	0618880918	B56
6	ZANATI	Zakaria	zakariazanati@gmail.com	0656852145	C50
7	TENNIA	Youssef	tenniayoussef@gmail.com	0623541236	C42
8	YOUNESS	Mouad	younessmouad@gmail.com	0632541269	C58

FIGURE 4.15 – Liste des étudiants

Chaque étudiant a un code qr généré automatiquement par l'application web pour s'authentifier.



Ensias Solvely

02:32 février 07, 2020

Identification des étudiants à l'aide d'un code QR

N° Eleve 5

Nom SAOUDI

Prénom Mehdi



N° Eleve 6

Nom ZANATI

Prénom Zakaria



FIGURE 4.16 – Qr code des étudiants

Quelques statistiques sur la distribution et la résolution des problèmes.

Distrubtion des problèmes relatifs à la comité batiment

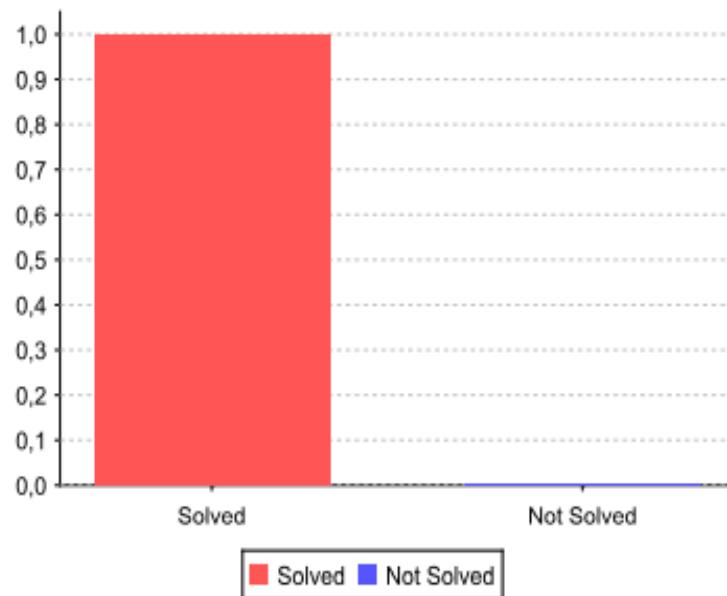


FIGURE 4.17 – Distribution des problèmes de la comité bâtiments

Distrubtion des problèmes relatifs à la comité divers

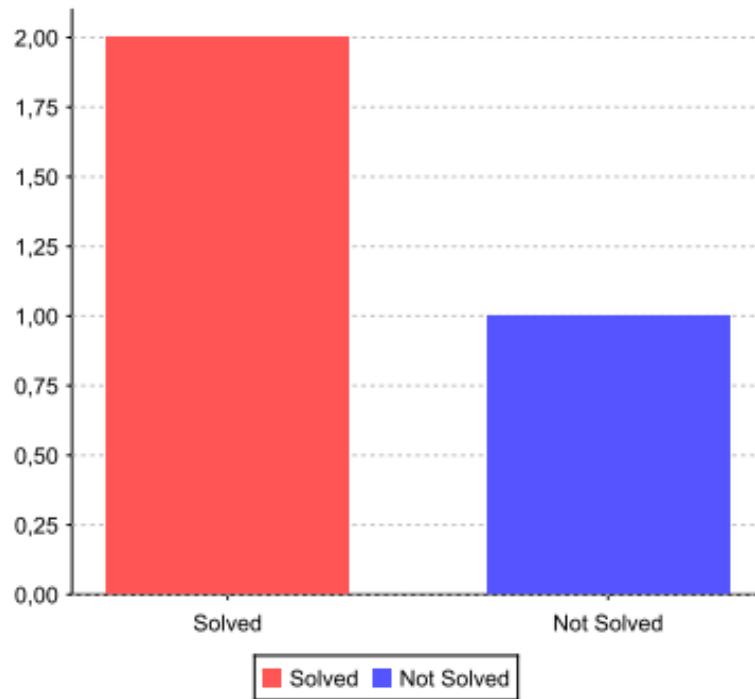


FIGURE 4.18 – Distribution des problèmes de la comité divers

Distrubtion des problèmes selon les comités

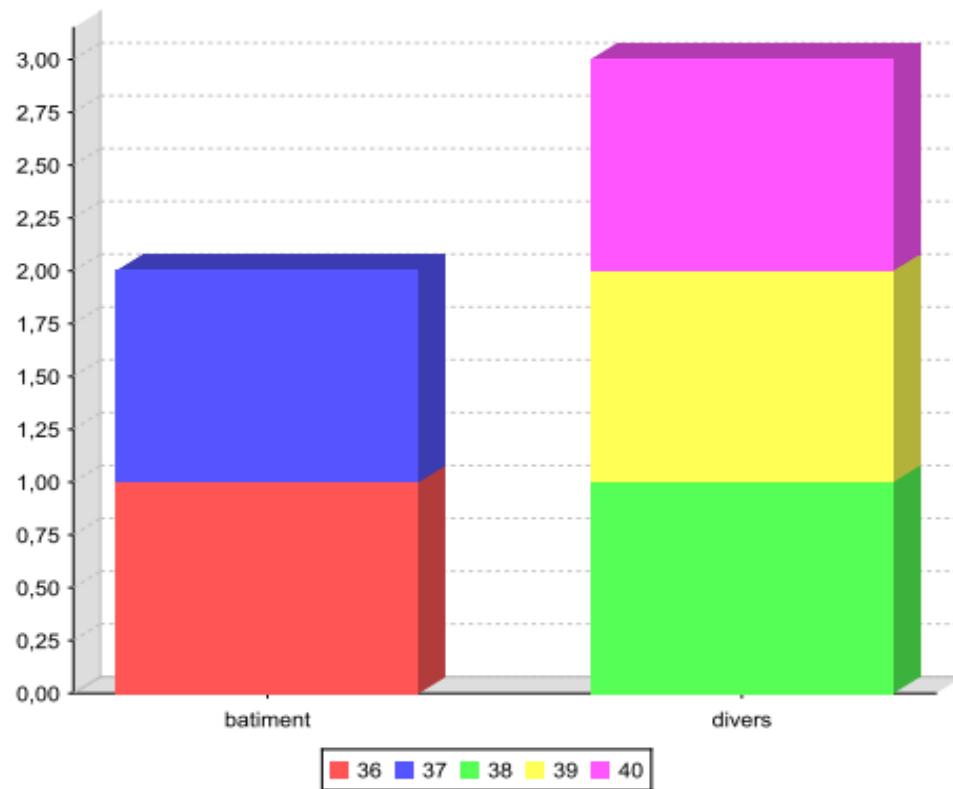


FIGURE 4.19 – Distribution des problèmes selon les comités

Statistique de résolution de problèmes selon l'année : 2020

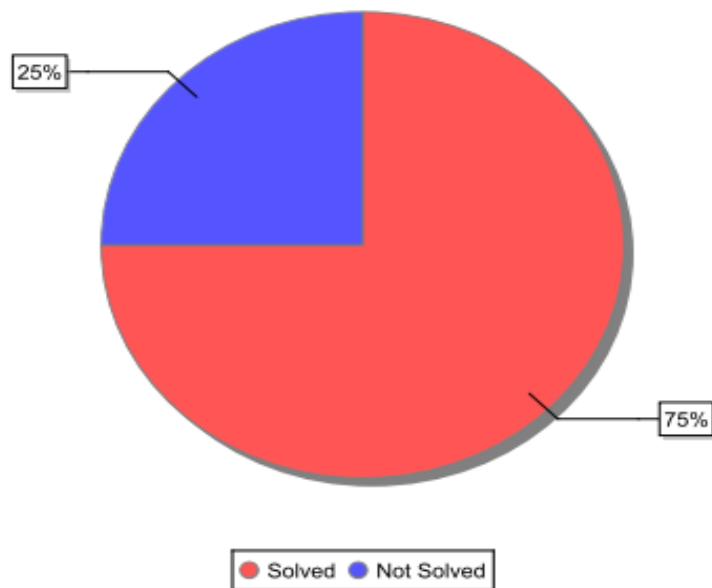


FIGURE 4.20 – Statistiques globales

Contact Us

FIGURE 4.21 – Page contact us

The screenshot shows a website page for 'EnsiasSolvey'. At the top, there is a red navigation bar with links: Home, Declare, Problems, Archive, My problems, About Us, and Contact. To the right of the navigation bar is a user profile icon with the name 'ZANATI Zakaria' and a small green dot. Below the navigation bar is a large, dark photograph of a modern building's exterior, featuring a curved balcony and a staircase. Overlaid on the right side of the photo is a teal-colored rectangular area containing the word 'ABOUT US' in white capital letters. A thin red horizontal line separates this section from the main text area. The main text area contains a paragraph describing the website's purpose: 'EnsiasSolvey is a web-site dedicated to engineering students of the National School of Computer Science and Systems Analysis. It allows to declare the problems encountered in the student residence of ENSIAS. Students can also see the progress of the solutions offered by the responsible committees.'

FIGURE 4.22 – Page about us

4.6 Conclusion

Dans la première partie de ce chapitre, nous avons présenté l'environnement matériels et logiciels sur lesquels nous nous sommes basés pour réaliser ce travail. Puis nous avons présenté les technologies utilisées. Par la suite, nous avons clôturé ce chapitre par une représentation des interfaces du site <Ensias Solveley> qui montrent les services offerts par notre application. S'il y'a un point qu'il fallait réussir tout de même c'est la partie «Comité» pour que le responsable puisse administrer le site Internet sans aucun problème.

Conclusion générale

Tout au long de ce rapport, nous avons présenté les différentes étapes de la réalisation de l'application du projet. Pour le développement de ce projet nous avons utilisé le langage UML, ce qui a permis de mener correctement la tâche d'analyse des besoins à l'aide du diagramme de cas d'utilisation et la tâche de conception, ainsi les scénarios sont aussi détaillés afin d'expliquer toutes les tâches faites puisque nous travaillons avec la technologie J2EE.

Ce projet nous a permis de développer nos compétences techniques, d'approfondir nos connaissances théoriques et pratiques, de stimuler un esprit d'initiative et de créativité, et notamment dans le domaine de développement des applications web. Il nous a donné la méthode pour assurer un travail de groupe, comment compter sur soi pour résoudre les problèmes au cas où ils se présentent, comment être professionnels dans notre travail, comment être attentifs aux indications de notre encadrant, comment être bien organisés pour accomplir dans les meilleurs délais, et meilleures conditions les tâches qui nous sont confiées. Ce projet nous a donné l'occasion de faire le lien entre les connaissances académiques, notamment en JAVA, Base de Données et le monde professionnel.

Notre application peut être aisément améliorée grâce à son aspect ouvert. Dans notre application, nous avons travaillé dans un réseau local. Pour la mettre en ligne nous avons seulement besoin de l'héberger sur un serveur.

Webographie

- [1] <https://openclassrooms.com/fr/courses/626954-creez-votre-application-web-avec-java-ee>
- [2] <https://docs.oracle.com/javaee/7/index.html>
- [3] <https://stackoverflow.com/>
- [4] <https://www.supinfo.com/articles/single/3099-waterfall-model-modele-cascade>
- [5] <http://www.uml-sysml.org/documentation>
- [6] <https://www.w3schools.com/>
- [7] <https://www.supinfo.com/articles/single/2519-architecture-client-serveur>