

---

# Projet de Plateformes de Développement : Conception et développement d'une application web de gestion du problèmes d'internat

---

*Génie Logiciel*

*Réalisé par :*

MOUNJI SAID  
OUAHMANE FATIMA EZZAHRA  
RHOUCHE OUSSAMA  
Toubil MOUAD

*Encadré par :*

Mr. Mahmoud EL HAMLAOUI



## Remerciements

Nous tenons à exprimer notre sincère gratitude à notre encadrant, Pr Mahmoud EL HAMLAOUI, pour avoir accepté de superviser nos travaux et pour son soutien constant durant cette période. Sa confiance dans notre travail et son temps pour nous ont beaucoup aidés.

Nous tenons également à remercier tous ceux qui ont contribué à la réussite de ce projet. Leur soutien, leur expertise et leur collaboration ont été essentiels à la réalisation de ce travail.

Nous remercions chacun d'entre eux et leur adressons nos remerciements les plus sincères.



# Table des figures

2.1	Diagramme de cas d'utilisation . . . . .	5
2.2	Diagramme de classe . . . . .	6
2.3	Diagramme de sequence . . . . .	7
2.4	Diagramme d'activité . . . . .	8
2.5	Diagramme de transition . . . . .	9
3.1	Logo du TS . . . . .	11
3.2	Logo de Java . . . . .	11
3.3	Logo de CSS . . . . .	12
3.4	Logo de MongoDB . . . . .	12
3.5	Logo du LaTeX . . . . .	12
3.6	Logo de Node.JS . . . . .	13
3.7	Logo de SpringBoot . . . . .	13
3.8	Logo de Angular . . . . .	13
3.9	Logo de Bootstrap . . . . .	13
3.10	Logo du VScode . . . . .	14
3.11	Logo d'IntelliJ . . . . .	14
3.12	Logo de draw.io . . . . .	14
3.13	Logo de git . . . . .	15
3.14	Logo de gitub . . . . .	15
3.15	Logo de Postman . . . . .	15
3.16	Logo de Docker . . . . .	16
3.17	Logo de Jenkins . . . . .	16
3.18	Register Page . . . . .	17
3.19	Registration Succeded! . . . . .	17
3.20	Page d'authentification . . . . .	18
3.21	Authentifié avec succès . . . . .	18
3.22	Affichage profile . . . . .	19
3.23	modification d'un profil . . . . .	19
3.24	Tableau de profiles . . . . .	20
3.25	page modification user . . . . .	20
3.26	confirmation de la suppression . . . . .	21
3.27	suppression du User . . . . .	21
3.28	Tableau de problème - Worker . . . . .	21
3.29	Ajout d'un problème . . . . .	22
3.30	Tableau de problèmes - Student . . . . .	22

# Table des matières

<b>1</b>	<b>Présentation du cadre général du projet</b>	<b>2</b>
1.1	Introduction . . . . .	2
1.2	Présentation de la mission du Projet . . . . .	2
1.2.1	Problématique . . . . .	2
1.2.2	Objectif . . . . .	2
1.3	Conclusion . . . . .	2
<b>2</b>	<b>Analyse et conception</b>	<b>3</b>
2.1	Introduction . . . . .	3
2.2	Analyse . . . . .	3
2.2.1	Cahier de charge . . . . .	3
2.2.1.1	Sujet : . . . . .	3
2.2.1.2	Fonctionnalités demandées : . . . . .	3
2.2.2	Besoins fonctionnels . . . . .	3
2.2.3	Besoins non fonctionnels . . . . .	4
2.3	Conception . . . . .	4
2.3.1	Diagramme de cas d'utilisation . . . . .	4
2.3.2	Diagramme de classe . . . . .	6
2.4	diagramme de sequence . . . . .	7
2.5	diagramme d'activité . . . . .	8
2.6	diagramme de transition . . . . .	9
2.7	Conclusion . . . . .	10
<b>3</b>	<b>Réalisation de l'application web</b>	<b>11</b>
3.1	Introduction . . . . .	11
3.2	Environnement de travail et outils . . . . .	11
3.2.1	Langages . . . . .	11
3.2.1.1	TS : Type Script . . . . .	11
3.2.1.2	Java . . . . .	11
3.2.1.3	CSS . . . . .	12
3.2.1.4	MongoDB . . . . .	12
3.2.1.5	LaTeX . . . . .	12
3.2.2	Frameworks . . . . .	12
3.2.2.1	Node.JS . . . . .	12
3.2.2.2	SpringBoot . . . . .	13
3.2.2.3	Angular . . . . .	13
3.2.2.4	Bootstrap . . . . .	13
3.2.3	Interfaces . . . . .	13
3.2.3.1	VScode . . . . .	13
3.2.3.2	IntelliJ . . . . .	14
3.2.3.3	draw.io . . . . .	14
3.2.3.4	git . . . . .	15
3.2.3.5	github . . . . .	15
3.2.3.6	postman . . . . .	15
3.2.4	Docker . . . . .	16
3.2.5	Jenkins . . . . .	16
3.3	Developpement . . . . .	17
3.3.1	Fonctionnalités globales . . . . .	17
3.3.1.1	Register Page . . . . .	17

3.3.1.2	Authentication . . . . .	17
3.3.2	Fonctionnalités . . . . .	18
3.3.2.1	Profile . . . . .	19
3.3.2.2	Fonctionnalités - Admin . . . . .	20
3.3.2.2.1	Accès aux profils . . . . .	20
3.3.2.2.2	Modifier profile . . . . .	20
3.3.2.3	Fonctionnalités - Worker . . . . .	21
3.3.2.4	Fonctionnalités - User . . . . .	22

# INTRODUCTION

Un système de gestion d'internat basé sur le Web est conçu pour stocker et gérer les informations relatives aux élèves et aux divers problèmes qu'ils rencontrent dans les internats. Les applications de gestion d'internat permettent à différents groupes de personnes, tels que les élèves et le personnel, d'accéder aux informations pertinentes et de les traiter de manière efficace et sécurisée. Cette collaboration a pour but d'améliorer la communication entre les différents acteurs impliqués dans la vie étudiante et de faciliter la gestion des problèmes et imprévus.

Un système de gestion d'internat basé sur le Web peut aider à augmenter l'efficacité et la productivité de l'école. Étant donné que l'application Web est facilement accessible via un navigateur, les utilisateurs n'ont pas besoin d'installer ou de mettre à jour le logiciel. De plus, les élèves et les responsables des internats peuvent accéder aux informations nécessaires à partir de différents endroits, ce qui facilite la coordination des activités et des ressources. Enfin, la collaboration entre les différents acteurs de l'internat est favorisée grâce à la disponibilité d'outils d'information et de communication en temps réel.

# Chapitre 1

## Présentation du cadre général du projet

### 1.1 Introduction

Dans ce chapitre, nous avons exposé le contexte général de notre projet, nous avons également décrit d'une manière générale le contexte du travail à effectuer pour situer l'environnement et les objectifs de notre projet de plateformes de développement «Conception et développement d'une application web de gestion des problèmes d'internat».

### 1.2 Présentation de la mission du Projet

#### 1.2.1 Problématique

La gestion d'un internat présente de nombreux défis pour les administrateurs et les élèves. Certains de ces problèmes incluent la gestion des chambres, le traitement des demandes spéciales et des plaintes des étudiants, et le traitement de divers problèmes imprévus tels que des problèmes de robinet, de portes, etc. Une application Web de gestion d'embarquement peut aider à ces tâches en fournissant une plate-forme centralisée pour gérer toutes ces tâches, permettant une communication efficace avec les étudiants et le personnel, et fournissant des outils pour une gestion plus efficace et automatisée.

#### 1.2.2 Objectif

Le but de notre application web est de fournir une solution complète et efficace pour la gestion des différentes problématiques des internats. Notre application aide à la communication, à la gestion du temps en fournissant une plate-forme centralisée pour gérer divers problèmes que les étudiants peuvent avoir dans leurs chambres (portes, éviers, électricité, etc.), la communication avec le personnel de l'internat et les demandes des résidents, coordonner et suivre les problèmes. En facilitant la gestion quotidienne des internats, notre application vise à améliorer l'expérience globale des résidents et du personnel des internats.

### 1.3 Conclusion

Pour conclure, le système de gestion d'internat basé sur le web est un système entièrement fonctionnel qui correspond à la fois aux besoins des élèves et du personnel et aide à faire la gestion des différents problèmes d'une façon facile et transparente. c'est un moyen d'améliorer le taux des problèmes réglés et d'accroître la performance, la productivité et l'efficacité.

# Chapitre 2

## Analyse et conception

### 2.1 Introduction

Il est à rappeler que l'objectif du projet consiste à développer une application web de gestion d'internat. Ce chapitre permet de faire une analyse théorique. En effet, cette conception est cruciale afin de comprendre la totalité des principes et les coder en se basant sur le cahier des charges fournit.

### 2.2 Analyse

#### 2.2.1 Cahier de charge

##### 2.2.1.1 Sujet :

Réalisation d'une application collaborative permettant la gestion du projets au sein de l'internat.

##### 2.2.1.2 Fonctionnalités demandées :

- Fonctionnalités globales : ( Gestion utilisateurs )
  - Authentification
  - Profil utilisateur : Admin , Worker , Student
- Fonctionnalités particulières :
  - Admin :
    - Gérer les utilisateurs (Modifier, ajouter et supprimer)
  - Worker :
    - Voir les problèmes
    - Gérer les problèmes (gérer la situation ...)
  - Student :
    - Voir les problèmes (voir l'état et l'avancement de chacun)
    - Ajouter un problème (détaillé)

#### 2.2.2 Besoins fonctionnels

Notre application permettra de gérer les problème où les élèves peuvent principalement accéder à leurs problèmes et en ajouter d'autres s'il existent. On aura trois profiles d'utilisateurs : Admin, Worker et Student.

Les besoins fonctionnels seront comme suit :

- L'administrateur a pour rôle de créer, supprimer et modifier les comptes des utilisateurs.
- Le Worker a pour rôle de mettre à jour la situation des différents problèmes existants



- Le Student peut de même ajouter un problème et consulter ses problèmes déjà postulés

### 2.2.3 Besoins non fonctionnels

Les besoins non fonctionnels seront comme suit :

- La simplicité : l'utilisation de l'application ne doit pas être trop compliquée.
- La sécurité : le système doit être sécurisé.
- L'ergonomie de l'interface : Les interfaces doivent être conviviales.
- La performance : la gestion du flux des données et l'optimisation du temps de réponse.

## 2.3 Conception

### 2.3.1 Diagramme de cas d'utilisation

En langage UML, les diagrammes de cas d'utilisation modélisent le comportement d'un système et permettent de capturer les exigences du système. Ces diagrammes décrivent les fonctions générales et la portée d'un système. Autrement dit, les cas d'utilisation et les acteurs dans les diagrammes de cas d'utilisation décrivent ce que le système fait et comment les acteurs l'utilisent, mais ne montrent pas comment le système fonctionne en interne.

Le diagramme de cas d'utilisation ci-dessous décrit les cas d'utilisation que permet l'application web à l'utilisateur.

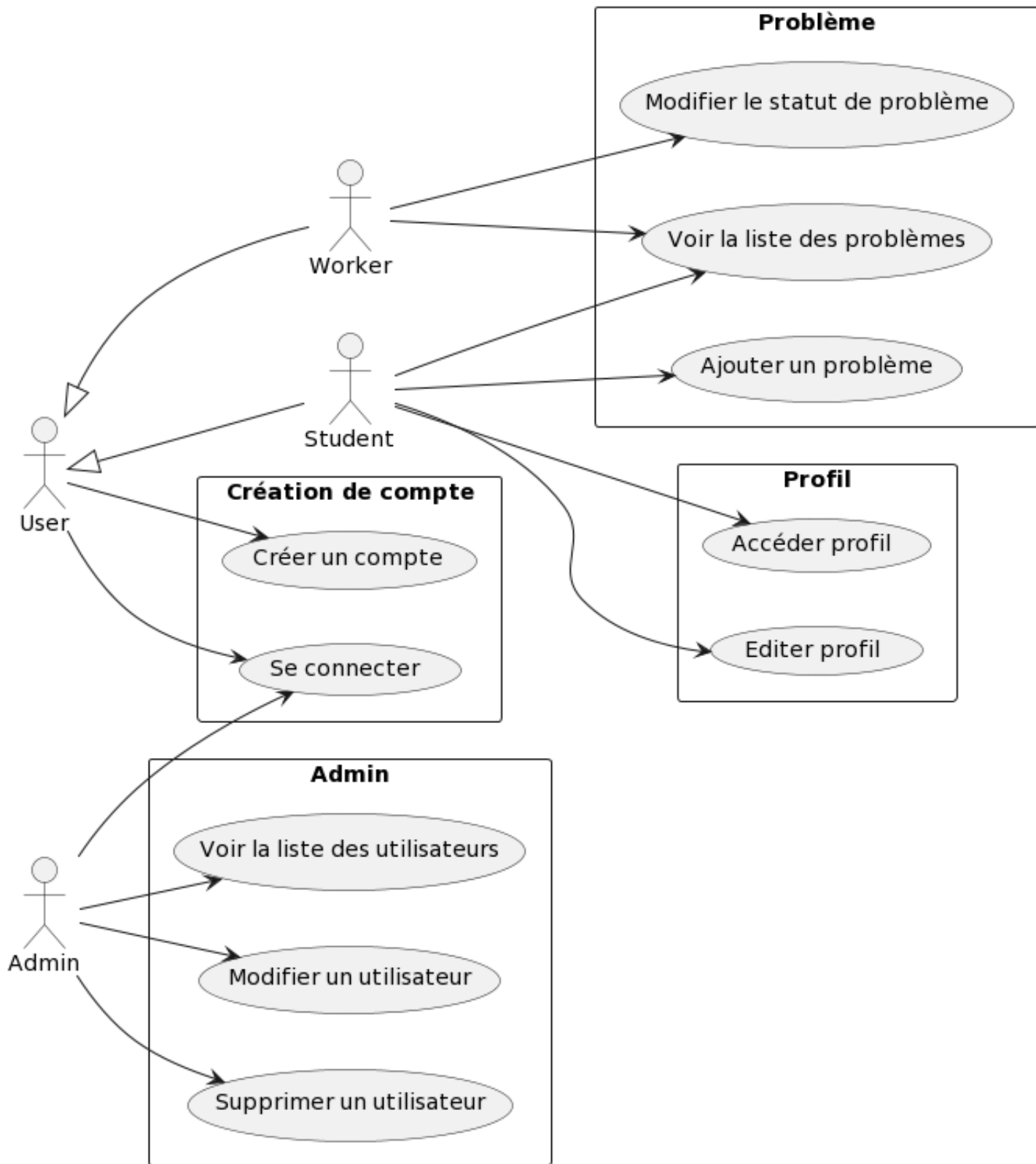


FIGURE 2.1 – Diagramme de cas d'utilisation

### 2.3.2 Diagramme de classe

Le diagramme de classe fait partie de la partie statique d'UML car il fait abstraction des aspects temporels et dynamiques. Une classe décrit les responsabilités, le comportement et le type d'un ensemble d'objets. Les éléments de cet ensemble sont les instances de la classe.

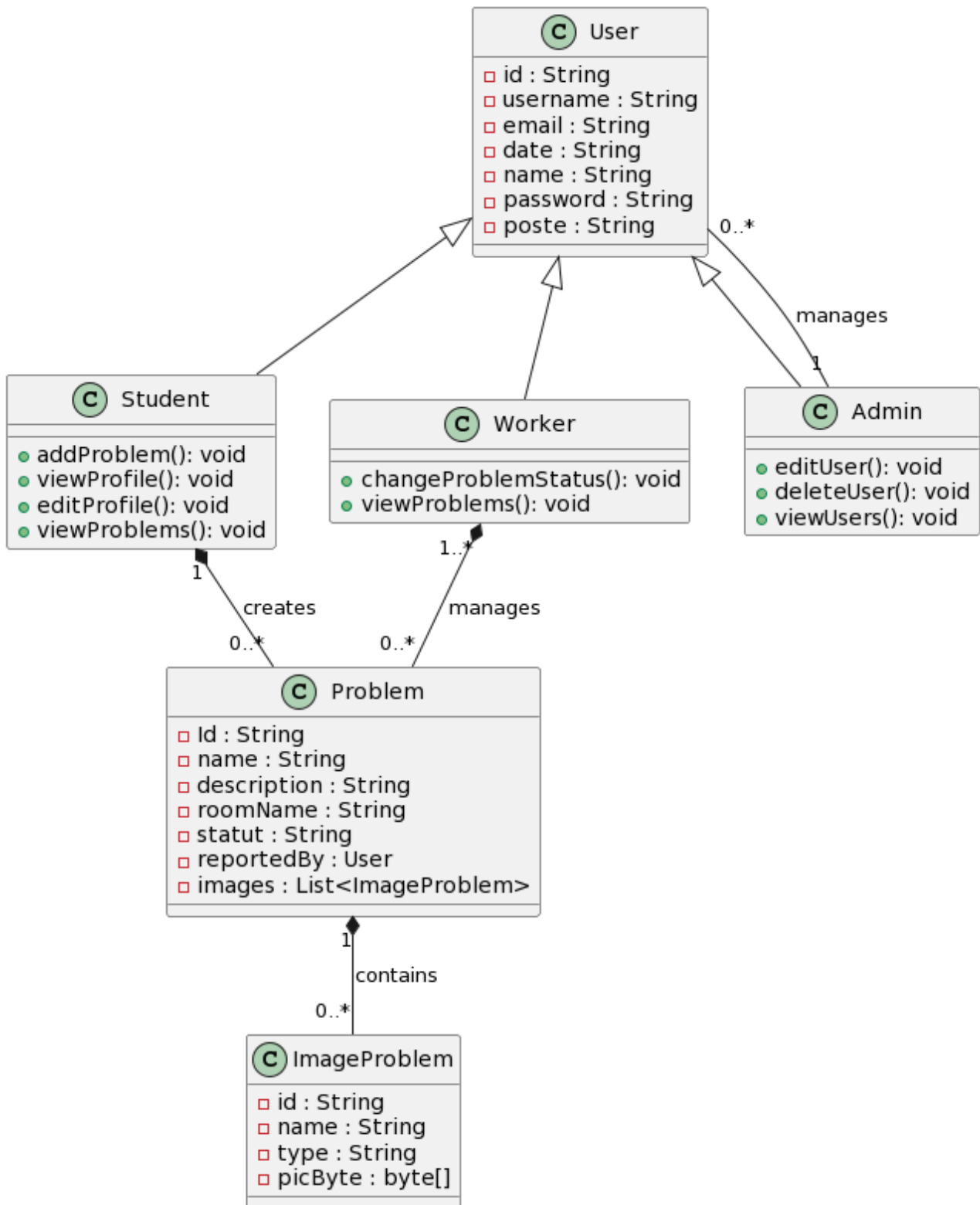


FIGURE 2.2 – Diagramme de classe

## 2.4 diagramme de sequence

Le diagramme de séquence est la représentation graphique des différents interactions entre les acteurs et le système selon un ordre chronologique dans la formulation UML.

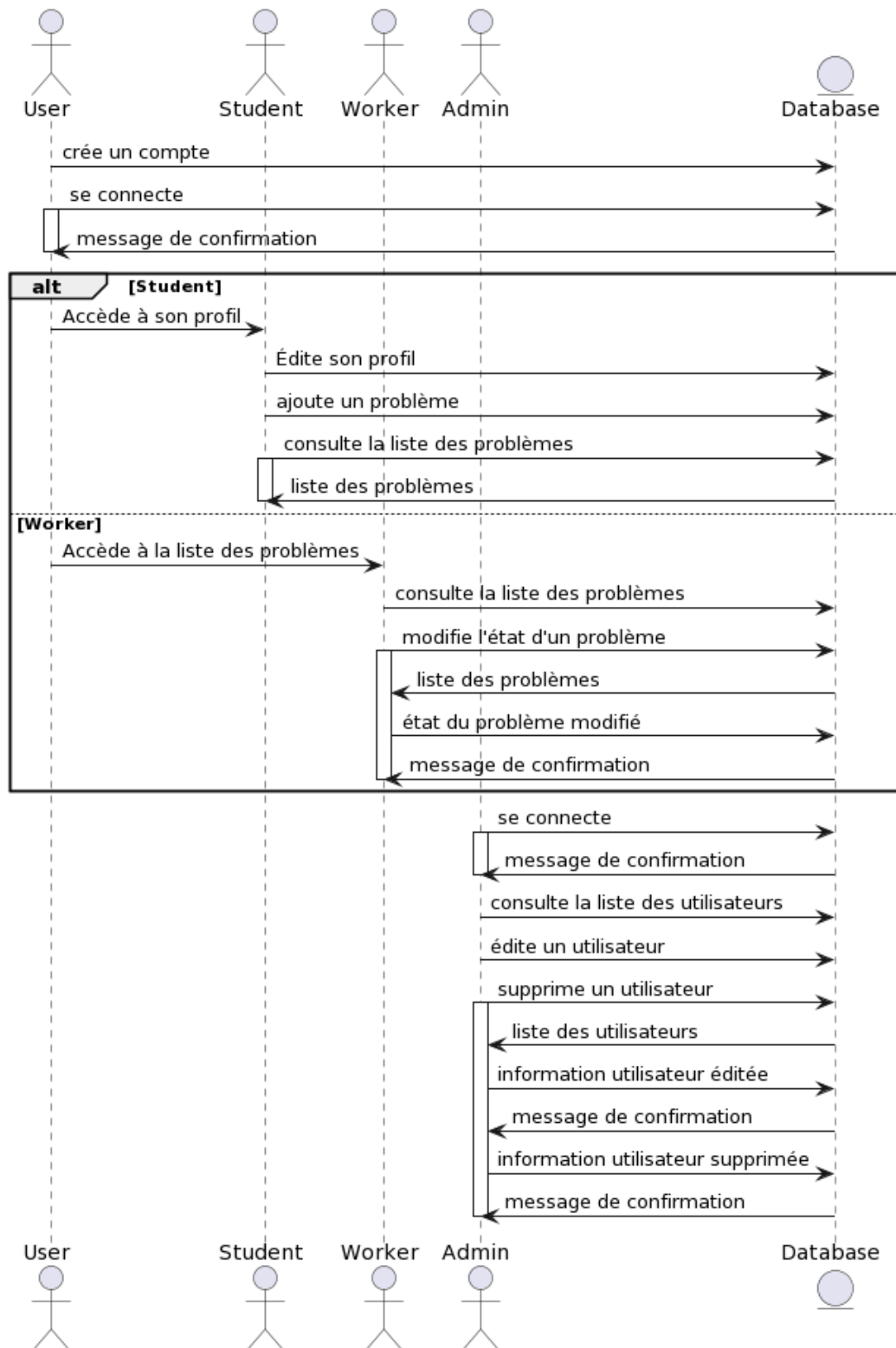


FIGURE 2.3 – Diagramme de sequence

## 2.5 diagramme d'activité

Dans le langage UML, un diagramme d'activité fournit une vue fondamentale du comportement d'un système en décrivant la séquence d'actions d'un processus donné.

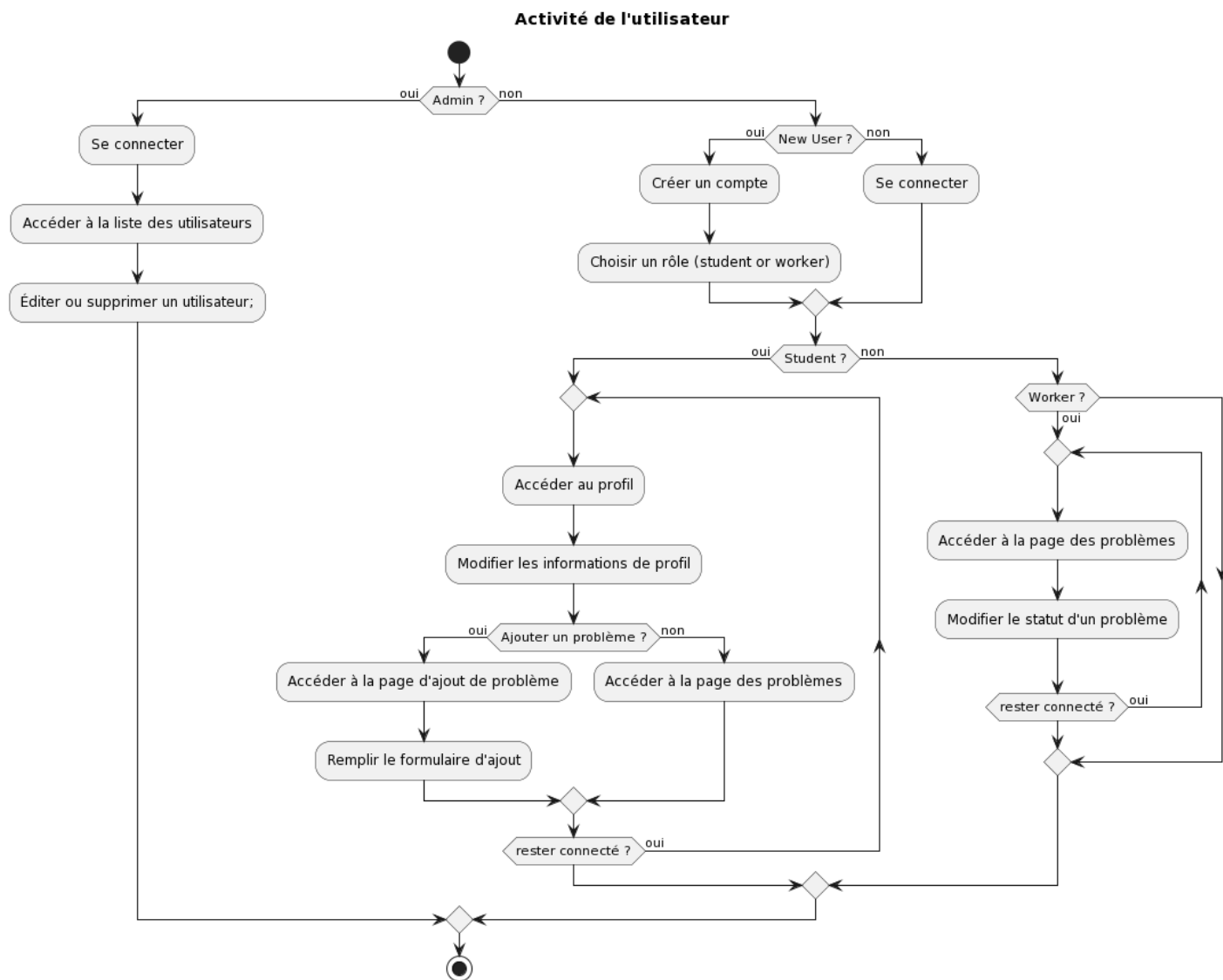


FIGURE 2.4 – Diagramme d'activité

## 2.6 diagramme de transition

Les diagrammes d'états-transitions d'UML décrivent le comportement interne d'un objet à l'aide d'un automate à états finis. Ils présentent les séquences possibles d'états et d'actions qu'une instance de classe peut traiter au cours de son cycle de vie en réaction à des événements discrets (de type signaux, invocations de méthode).

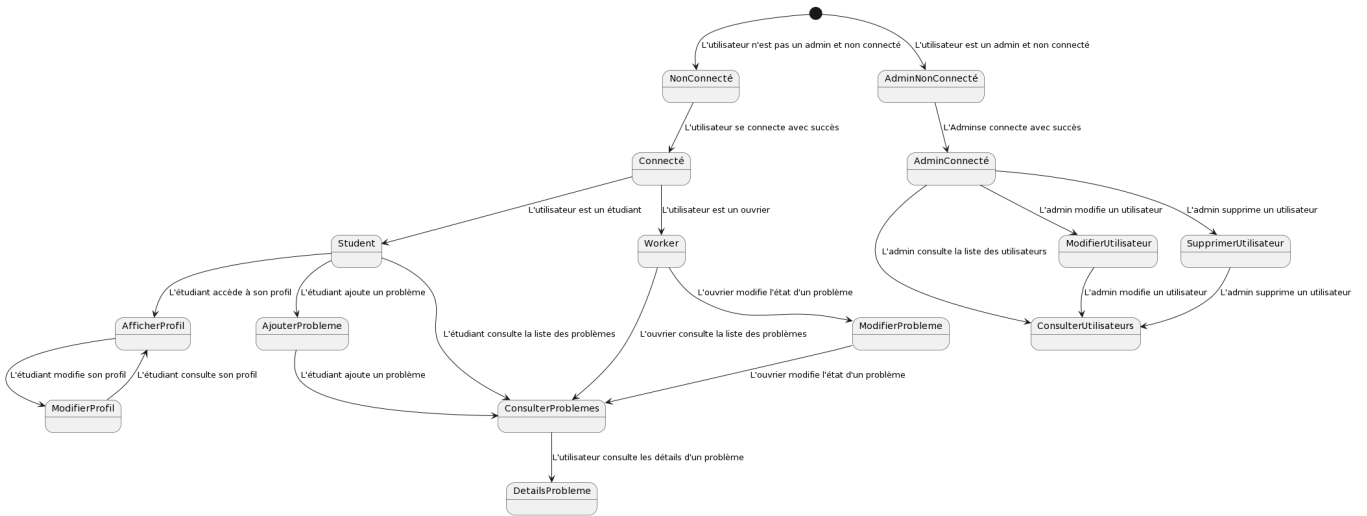


FIGURE 2.5 – Diagramme de transition

## 2.7 Conclusion

La conception du projet a été un point très important dans le processus de travail vu qu'elle permet d'aborder la structure de l'application et de mettre au point les objectifs à atteindre.. Principalement l'application permet aux utilisateurs de se connecter à leur compte tout en ajustant les fonctionnalités selon leur poste.

## Chapitre 3

# Réalisation de l'application web

### 3.1 Introduction

Ce chapitre est consacré à la présentation de l'environnement logiciel utilisé pour le développement de la solution proposée, avec une explication éventuelle aux choix techniques relatifs aux langages de programmation et des outils utilisés. Puis nous allons valider les attentes du projet à travers le développement de l'application.

### 3.2 Environnement de travail et outils

#### 3.2.1 Langages

##### 3.2.1.1 TS : Type Script

TypeScript is a statically-typed programming language developed by Microsoft. It is a superset of JavaScript, which means that any valid JavaScript code is also valid TypeScript code. TypeScript extends JavaScript by adding static typing, which allows for better tooling, code quality, and improved maintainability.



FIGURE 3.1 – Logo du TS

##### 3.2.1.2 Java

Java est un langage multiplateforme de programmation orienté objet de haut niveau créé par Sun Microsystems



FIGURE 3.2 – Logo de Java



### 3.2.1.3 CSS

Le CSS pour Cascading Style Sheets, est un langage informatique utilisé sur Internet pour la mise en forme de fichiers et de pages HTML. On le traduit en français par feuilles de style en cascade.



FIGURE 3.3 – Logo de CSS

### 3.2.1.4 MongoDB

est un système de gestion de base de données orienté documents, répartitionnable sur un nombre quelconque d'ordinateurs et ne nécessitant pas de schéma prédéfini des données. Il est écrit en C++.



FIGURE 3.4 – Logo de MongoDB

### 3.2.1.5 LaTeX

Les documents professionnels nécessitent un logiciel de traitement et Latex est l'outil parfait pour cette tâche. Il était donc inévitable de travailler avec ce dernier.



FIGURE 3.5 – Logo du LaTeX

## 3.2.2 Frameworks

Un framework est, comme son nom l'indique en anglais, un “cadre de travail”. L'objectif d'un framework est généralement de simplifier le travail des développeurs informatiques, en leur offrant une architecture “prête à l'emploi” et qui leur permette de ne pas repartir de zéro à chaque nouveau projet.[5]

### 3.2.2.1 Node.JS

Node.js est un environnement d'exécution single-thread, open-source et multi-plateforme permettant de créer des applications rapides et évolutives côté serveur et en réseau. .



FIGURE 3.6 – Logo de Node.JS

### 3.2.2.2 SpringBoot

Spring Boot est un framework qui facilite le développement d'applications fondées sur Spring en offrant des outils permettant d'obtenir une application packagée en JAR, totalement autonome.



FIGURE 3.7 – Logo de SpringBoot

### 3.2.2.3 Angular

Angular est un framework pour clients, open source, basé sur TypeScript et codirigé par l'équipe du projet « Angular » chez Google ainsi que par une communauté de particuliers et de sociétés



FIGURE 3.8 – Logo de Angular

### 3.2.2.4 Bootstrap

Bootstrap est une collection d'outils utiles à la création du design (graphisme, animation et interactions avec la page dans le navigateur, etc.) de sites et d'applications web. C'est un ensemble qui contient des codes HTML et CSS, des formulaires, boutons, outils de navigation et autres éléments interactifs, ainsi que des extensions JavaScript en option.



FIGURE 3.9 – Logo de Bootstrap

## 3.2.3 Interfaces

### 3.2.3.1 VScode

Visual Studio Code : est un éditeur de code extensible développé par Microsoft pour Windows, Linux et mac OS. Les fonctionnalités incluent la prise en charge du débogage, la mise en évidence de la syntaxe, la

complétion intelligente du code, les snippets, la refactorisation du code et Git intégré. Les utilisateurs peuvent modifier le thème, les raccourcis clavier, les préférences et installer des extensions qui ajoutent des fonctionnalités supplémentaires. Le code source de Visual Studio Code provient du projet logiciel libre et open source VSCoDe de Microsoft publié sous la licence MIT permissive, mais les binaires compilés sont des logiciels gratuits pour toute utilisation. Dans le Stack Overflow 2019 Developer Survey, Visual Studio Code a été classé comme l'outil d'environnement de développement le plus populaire.



FIGURE 3.10 – Logo du VScode

### 3.2.3.2 IntelliJ

IntelliJ IDEA également appelé « IntelliJ », « IDEA » ou « IDJ » est un environnement de développement intégré (en anglais Integrated Development Environment - IDE) destiné au développement de logiciels informatiques reposant sur la technologie Java.



FIGURE 3.11 – Logo d'IntelliJ

### 3.2.3.3 draw.io

draw.io est l'un des outils de création de diagrammes les plus riches en fonctionnalités du marché et un concurrent direct de solutions professionnelles de bureau comme Visio. Ses nombreuses barres d'outils, réunissant des centaines d'éléments visuels et de fonctions, permettent de créer des diagrammes pour toutes les applications imaginables.



FIGURE 3.12 – Logo de draw.io

#### 3.2.3.4 git

Git est de loin le système de contrôle de version le plus largement utilisé aujourd'hui. Git est un projet open source avancé, qui est activement maintenu. À l'origine, il a été développé en 2005 par Linus Torvalds, le créateur bien connu du noyau du système d'exploitation Linux. De plus en plus de projets logiciels reposent sur Git pour le contrôle de version, y compris des projets commerciaux et en open source.



FIGURE 3.13 – Logo de git

#### 3.2.3.5 github

GitHub est considéré comme un outil essentiel pour les ingénieurs logiciels, et sa popularité est inégalée. Il accueille actuellement plus de 25 millions d'utilisateurs. C'est un nombre considérable de professionnels qui se tournent vers GitHub pour améliorer le flux de travail et la collaboration.



FIGURE 3.14 – Logo de gitub

#### 3.2.3.6 postman

Postman est officiellement présentée comme une plateforme API pour la création et l'utilisation d'API. D'une manière générale, Postman est une plateforme qui permet de simplifier chaque étape du cycle de vie des API et de rationaliser la collaboration, afin de créer, plus facilement et plus rapidement, de meilleures API.



FIGURE 3.15 – Logo de Postman

### 3.2.4 Docker

Le logiciel Docker est une technologie de conteneurisation qui permet la création et l'utilisation de conteneurs Linux®.



FIGURE 3.16 – Logo de Docker

### 3.2.5 Jenkins

Jenkins is an open-source automation server that facilitates the continuous integration and delivery (CI/CD) of software projects. It is designed to automate the building, testing, and deployment of applications, enabling teams to deliver software more rapidly and reliably..



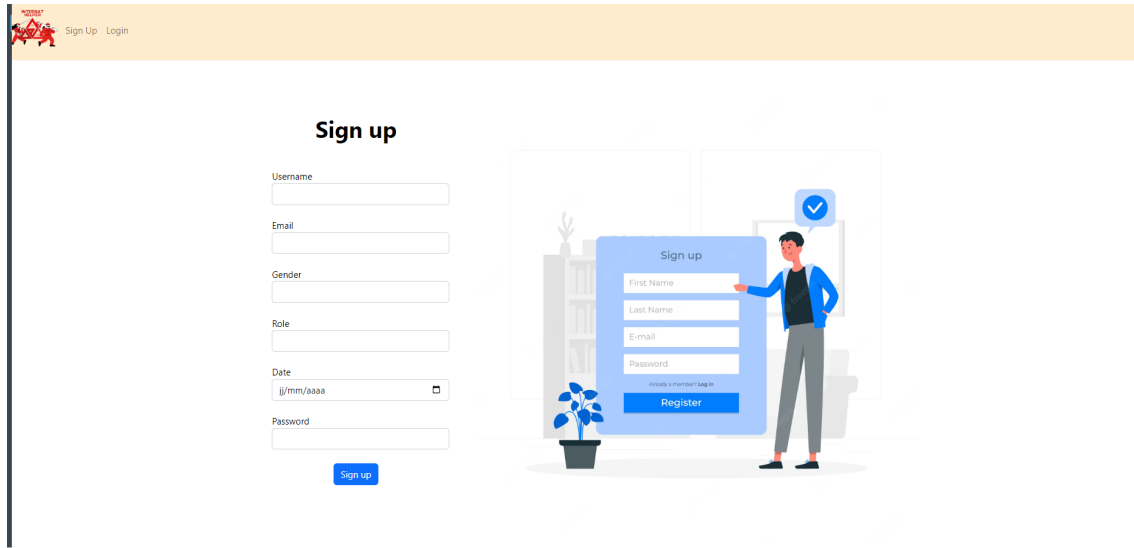
FIGURE 3.17 – Logo de Jenkins

## 3.3 Developpement

### 3.3.1 Fonctionnalités globales

#### 3.3.1.1 Register Page

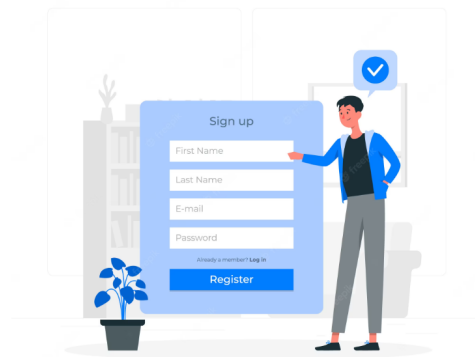
Depuis la page d'accueil, l'utilisateur peut se connecter en entrant l'email et le mot de passe. ou bien créer un nouveau compte !



The screenshot shows a web application interface for a registration page. At the top, there is a navigation bar with a logo and links for 'Sign Up' and 'Login'. The main content area is titled 'Sign up' and contains a form with the following fields: Username, Email, Gender, Role, Date (with a date picker set to 'jj/mm/aaaa'), and Password. A 'Sign up' button is located below the form. To the right of the form is an illustration of a man in a blue jacket and grey pants standing next to a large 'Sign up' form that mirrors the one on the left. The illustration also includes a potted plant and a checkmark icon. The entire page is enclosed in a dark border.

FIGURE 3.18 – Register Page

#### Sign up



Your registration is successfull

FIGURE 3.19 – Registration Succeeded !

#### 3.3.1.2 Authentification

Depuis la page d'accueil, l'utilisateur peut se connecter en entrant l'email et le mot de passe, ou bien passer au procédure de renitialisation du mot de passe.

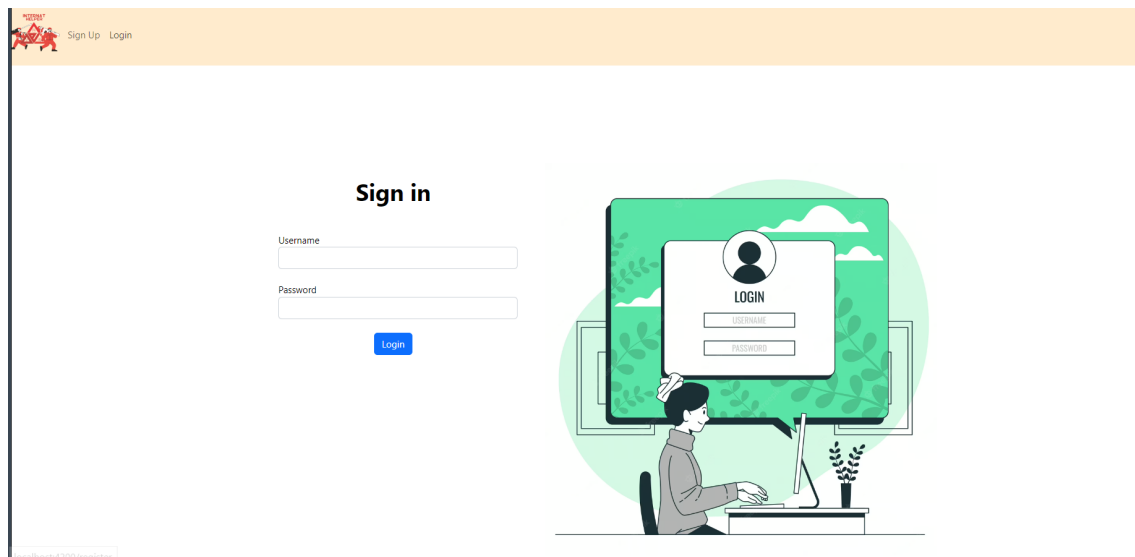


FIGURE 3.20 – Page d'authentification

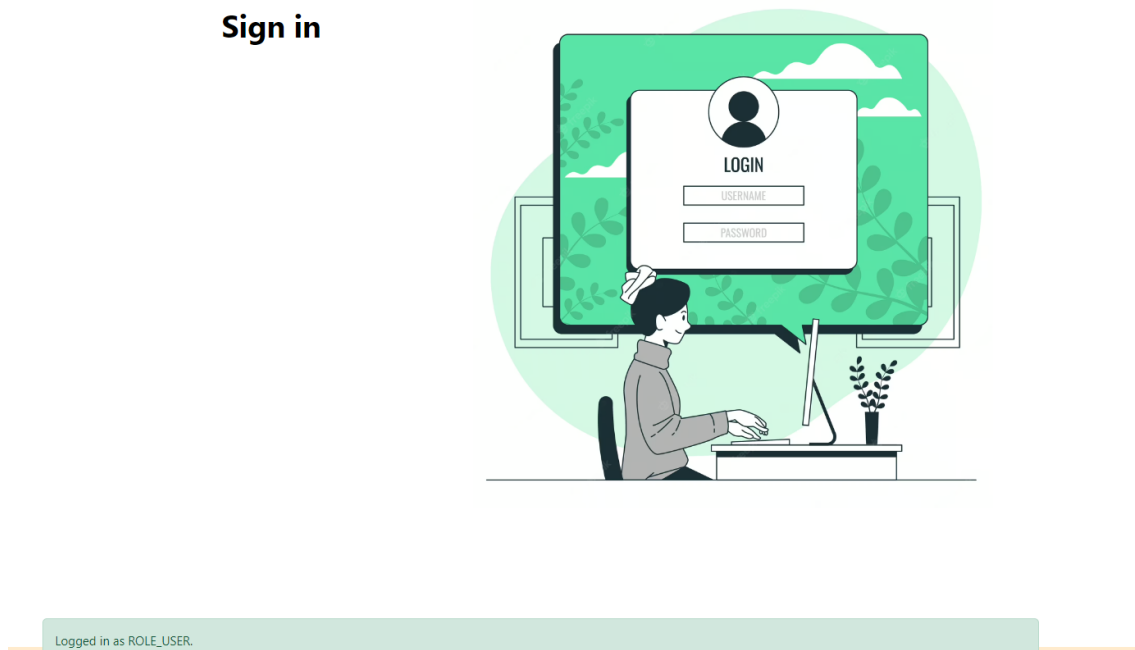


FIGURE 3.21 – Authentifié avec succès

### 3.3.2 Fonctionnalités

### 3.3.2.1 Profile

chaque utilisateur peut accéder à son profile

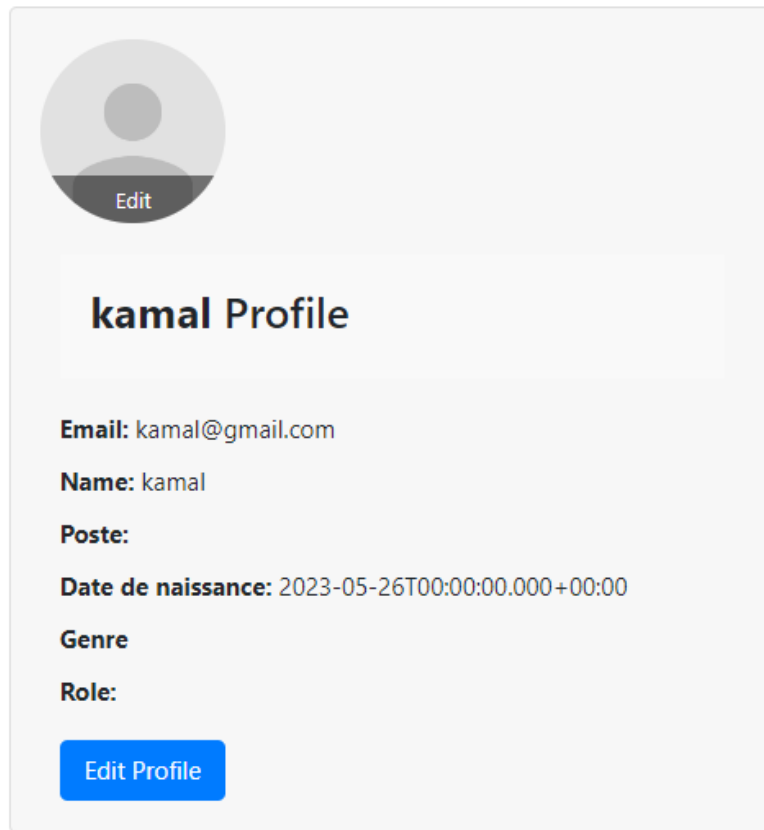


FIGURE 3.22 – Affichage profile

il peut même le modifier

A screenshot of a user profile modification form. At the top is the title 'kamal Profile' and a square profile picture placeholder. Below are input fields for 'Email:' (kamal@gmail.com), 'Name:' (kamal), 'Gender:' (a dropdown menu), 'Poste:' (a dropdown menu), and 'Date de naissance:' (a date picker showing jj/mm/aaaa). At the bottom is a blue button labeled 'Click to Edit'.

FIGURE 3.23 – modification d'un profil



### 3.3.2.2 Fonctionnalités - Admin

L'Admin est le responsable sur la gestion des utilisateurs.

#### 3.3.2.2.1 Accès aux profiles L'Admin à une vue globale sur les differents comptes existants.

Users Table

[+ Create](#)

1	HichamAzzroual	26	Hicham@gmail.com	ROLE_WORKER	Plumber			
2	ZakariaZiani	23	Ziko@gmail.com	ROLE_WORKER	Electrician			
3	SaidMounji	23	mwanj@gmail.com	ROLE_ADMIN	4			
4	OussamaRhouch	23	Ouss_rhouch@gmail.com	ROLE_ADMIN	Admin			
5	HossameTorch	23	hos-hoss@gmail.com	ROLE_USER	User			
6	MouadToubil	23	pogchampou@gmail.com	ROLE_USER	User			
7	MounjSaid	-1	saidmounjisam@technocolabs.com	ROLE_USER				
8	oussama	21	oussamarhouch@gmail.com	ROLE_USER				

Items per page: 10 1 - 8 of 8 < > >>

FIGURE 3.24 – Tableau de profiles

#### 3.3.2.2.2 Modifier profile Il peut notemment modifier un profile.

Modifier L'Utilisateur :

Gender :

☒ Male ☐ Female

Roles

5/22/1996

FIGURE 3.25 – page modification user

ou bien supprimer un profile.

1	HichamAzzeroual	26	Hicham@gmail.com	ROLE_WORKER	Plombier			
2	ZakariaZiani	23	Ziko@gmail.com	ROLE_WORKER	Electricien			
3	SaidMounji	23	mwanj@gmail.com		d			
4	OussamaRhouch	23	Ouss_rhouch@gmail.com	ROLE_ADMIN	Admin			
5	HossameTorchi	23	hos-hoss@gmail.com	ROLE_USER	User			
6	MouadToubil	23	pogchampou@gmail.com	ROLE_USER	User			
7	MounjiSaid	-1	saidmounjisam@technocolabs.com	ROLE_USER				
8	oussama	21	oussamarhouch@gmail.com	ROLE_USER				

FIGURE 3.26 – confirmation de la suppression

↑								
1	HichamAzzeroual	26	Hicham@gmail.com	ROLE_WORKER	Plombier			
2	SaidMounji	23	mwanj@gmail.com	ROLE_ADMIN	d			
3	OussamaRhouch	23	Ouss_rhouch@gmail.com	ROLE_ADMIN	Admin			
4	HossameTorchi	23	hos-hoss@gmail.com	ROLE_USER	User			
5	MouadToubil	23	pogchampou@gmail.com	ROLE_USER	User			
6	MounjiSaid	-1	saidmounjisam@technocolabs.com	ROLE_USER				
7	oussama	21	oussamarhouch@gmail.com	ROLE_USER				

Items per page: 10 1 - 7 of 7

FIGURE 3.27 – suppression du User

### 3.3.2.3 Fonctionnalités - Worker

Un compte du type personnel à le droit de voir différents problèmes existants et même en modifier les situations.

#### Problems Table

#	Problem Name	Problem Description ↑	Problem Room Number	Problem reported By	Images	Situation
1	poignet	jai un probleme poignet	D45	Unknown		Complété
2	robinet	problem de robinet	E56	Unknown		
3	Toilette	La chasse ne marche plus...	C57	Unknown		

Items per page: 3 4 - 6 of 12

FIGURE 3.28 – Tableau de problème - Worker

3.3.2.4 Fonctionnalités - User

pour un étudiant, ce dernier à le droit de voir les problèmes existants et aussi d'en ajouter un nouveau problème si necessaire

Problem Name

Problem Description

Problem Room

ada@mail.com

Add Problem

Clear

browse a file

FIGURE 3.29 – Ajout d'un problème




7	la porte	jai un porbleme dans la porte tout le temps	D45	Unknown		en cours
8	chambre	il ya une mauvaise odeur a cause de lavabo	D45	Unknown		completé
9	probleme de porte	ma porte ne s'ouvre pas	B28	Unknown		en cours

FIGURE 3.30 – Tableau de problèmes - Student

# CONCLUSION

En conclusion , Ce projet s'est révélé très enrichissant dans la mesure où il a consisté en une approche concrète en ce qui concerne le full-stack development. En effet, les techniques utilisées dans le projet seront des aspects essentiels de n'importe quel projet sous ce thème.