

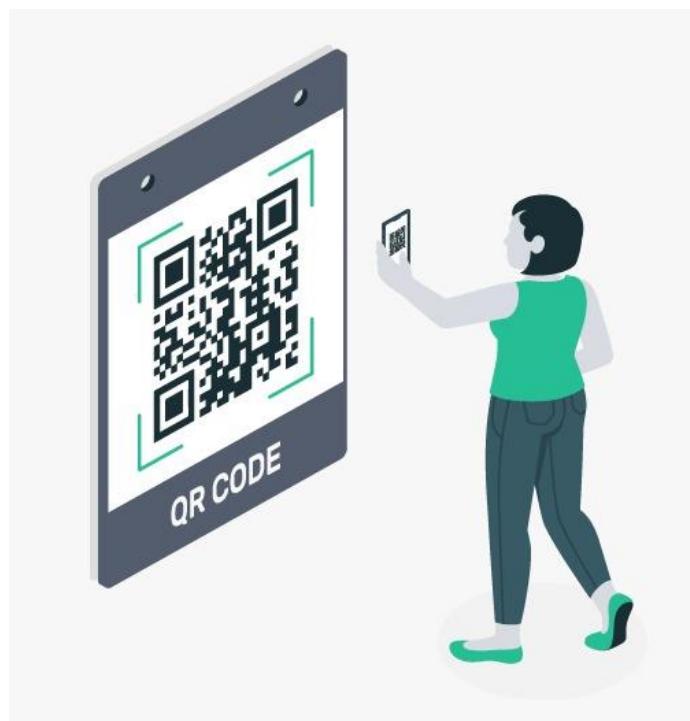
# *Université Hassan II.*

*Ecole supérieur de technologie Casablanca.*

*Département : Génie informatique.*

**Rapport de projet de fin d'études :**

***La réalisation d'une application de gestion d'absence  
basée sur QR CODE :***



*Réalisés par : -Moustarzik Oussama*

*-Radouane Soumiya*

*- Moussaid Nouhaila*

*- Mrhir Omaima*

*Encadré par : Madame Nadia Afifi*

**Année Universitaire :2022/2023**



## *Remerciements*

*Avant d'exposer comme-ci après le Rapport du projet de fin d'études, on tient à remercier Madame Nadia Afifi et Monsieur Hicham Belhedaoui nos encadrants de projet de fin d'études pour la proposition de ce sujet, leurs disponibilités, soutiens et précieuses aide lors de l'élaboration du projet.*

*Nous adressons aussi nos sincères remerciements à tous nos proches et amis qui nous ont toujours soutenu et encouragés au cours de la réalisation de ce projet ainsi qu'à nos profs qui ont contribué à la réussite de cette formidable année universitaire.*

*Finalement, Nous adressons aussi nos sincères remerciements à tout ce qui ont participé de près ou de loin à la réalisation de ce projet, aux personnes qui, malgré que leurs noms ne figurent pas dans ce document, étaient toujours prêtes à aider et à contribuer dans le bon déroulement de ce travail.*

## *Table des Matières*

### I. Table des matières

<b>II.</b>	<b><i>Introduction :</i></b>	<b>10</b>
<b>III.</b>	<b><i>Cadre du projet :</i></b>	<b>10</b>
<b>IV.</b>	<b><i>Présentation du cahier de charge :</i></b>	<b>10</b>
<b>IV.1</b>	<b><i>Problématique :</i></b>	<b>10</b>
<b>IV.2</b>	<b><i>La solution :</i></b>	<b>10</b>
<b>V.</b>	<b><i>CONCLUSION:</i></b>	<b>11</b>
<b>I.</b>	<b><i>Introduction</i></b>	<b>13</b>
<b>II.</b>	<b><i>Langage de conception</i></b>	<b>13</b>
<b>II.1</b>	<b><i>Langage UML</i></b>	<b>13</b>
<b>II.2</b>	<b><i>Utilisation de l'UML</i></b>	<b>14</b>
<b>III.</b>	<b><i>Les Diagrammes de conception</i></b>	<b>14</b>
<b>III.1</b>	<b><i>Modélisation comportementaux :</i></b>	<b>14</b>
<b>I.</b>	<b><i>III.1.1.....</i></b>	<b>Diagramme de cas d'utilisation</b>
	<b>14</b>	
<b>III.2</b>	<b><i>Modélisation dynamique .....</i></b>	<b>16</b>
<b>II.</b>	<b><i>III.2.1.....</i></b>	<b>Diagrammes de séquences</b>
	<b>16</b>	
<b>III.</b>	<b><i>III.2.2.....</i></b>	<b>Diagrammes de séquences de Security</b>
	<b>16</b>	

IV.	III.2.3.....	Diagrammes de séquences de parent
	17	
V.	III.2.4.....	Diagrammes de séquences d'enseignant
	18	
VI.	III.2.5.....	Diagrammes de séquences d'Administrateur
	19	
III.3	Modélisation statique .....	20
VII.	III.3.1.....	Diagramme de classes
	20	
VIII.	III.3.2.....	Diagramme de classe global
	20	
IX.	III.3.3.....	Modèle logique de données lié au diagramme de classe
	21	
III.4	Diagramme d'Activité.....	21
X.	III.4.1.....	Définition
	21	
XI.	III.4.2 Diagramme d'Activité .....	22
IV.	<b>Conclusion .....</b>	<b>22</b>
I.	<b>Introduction .....</b>	<b>24</b>
II.	<b>Environnement logiciel.....</b>	<b>24</b>
II.1	Structure de Django .....	27
III.	<b>Installation.....</b>	<b>29</b>
IV.	<b>QR_CODE.....</b>	<b>30</b>
V.	<b>Présentation de l'application .....</b>	<b>31</b>
V.1	les Espaces.....	34
XII.	V.1.1 .....	les interfaces de l'authentification
	34	
XIII.	V.1.2 .....	Interface Administrateur
	36	
XIV.	V.1.3 .....	interface Agent de Security
	41	
XV.	V.1.4 .....	Interface enseignant
	44	
XVI.	V.1.5 .....	Interface Parent
	46	
V.2	Contact interface .....	49
V.3	About us interface.....	50
VI.	<b>conclusion.....</b>	<b>51</b>

## Table de Figure

<b>FIGURE 1 : diagramme de cas d'utilisation :</b> .....	<b>15</b>
<b>FIGURE 2 : diagramme de séquence de Security :</b> .....	<b>16</b>
<b>FIGURE 3 : diagramme de séquence de parent:</b> .....	<b>17</b>
<b>FIGURE 4 : diagramme de séquence d'enseignant:</b> .....	<b>18</b>
<b>FIGURE 5 : diagramme de séquence d'Administrateur:</b> .....	<b>19</b>
<b>FIGURE 6 : diagramme de classes.....</b>	<b>20</b>
<b>FIGURE 7 : diagramme d'activité.....</b>	<b>22</b>
<b>Figure8 : Visual Studio Code.....</b>	<b>24</b>
<b>Figure 9 : Power Designer : .....</b>	<b>25</b>
<b>Figure 10 : HTML :.....</b>	<b>25</b>
<b>Figure 11 : CSS:.....</b>	<b>25</b>
<b>Figure 12 : JavaScript: .....</b>	<b>26</b>
<b>Figure 13: PostgreSQL:.....</b>	<b>26</b>
<b>Figure 14 : Bootstrap .....</b>	<b>26</b>
<b>Figure 15 : Django.....</b>	<b>27</b>
<b>Figure 16 : Architecture de Django .....</b>	<b>27</b>
<b>Figure 17 : QR CODE: .....</b>	<b>31</b>
<b>Figure 18 : Interface d'accueil : .....</b>	<b>32</b>
<b>Figure 19 : Interface d'accueil (Parent section) .....</b>	<b>33</b>
<b>Figure 20 : Interface d'accueil (Teachers section) .....</b>	<b>33</b>
<b>Figure 21 : Interface d'accueil (Security Agent section).....</b>	<b>34</b>
<b>Figure 22 : Interface d'accueil (Administration section).....</b>	<b>34</b>
<b>Figure 23 : Interface Log In(Parent) .....</b>	<b>35</b>
<b>Figure 24: Interface Log In(Teacher) .....</b>	<b>35</b>
<b>Figure 25 : Interface Log In(Agent) .....</b>	<b>36</b>
<b>Figure 26 : Interface Log In(Administration) .....</b>	<b>36</b>
<b>Figure 27 : Interface Administration (Dashboard) .....</b>	<b>37</b>
<b>Figure 28 : Interface Administration (boîte de messages).....</b>	<b>38</b>

<b>Figure 29 : Interface Administration (Générateur de QR code) :</b> .....	<b>39</b>
<b>Figure 30 : Interface Administration (Archive : classes) :</b> .....	<b>40</b>
<b>Figure 31 : Interface Administration (Archive : calendrier)</b> .....	<b>41</b>
<b>Figure 32 : Interface Administration (Archive : liste d'archive)</b> .....	<b>42</b>
<b>Figure 33: Interface Agent De Security (Accueil)</b> .....	<b>42</b>
<b>Figure 34 : Interface Agent De Security (Scan 1)</b> .....	<b>43</b>
<b>Figure 35 : Interface Agent De Security (Scan 2)</b> .....	<b>43</b>
<b>Figure 36 : Interface Agent De Security (Scan 3)</b> .....	<b>44</b>
<b>Figure 37 : Interface Agent De Security (Archive Scan)</b> .....	<b>44</b>
<b>Figure 38 : Interface Teacher(Accueil)</b> .....	<b>45</b>
<b>Figure 39 : Interface Teacher(Liste de presence)</b> .....	<b>46</b>
<b><i>Figure 40 : Interface Teacher(Contact)</i></b> .....	<b>47</b>
<b>Figure 41 : Interface Parent(Accueil) .....</b>	<b>48</b>
<b>Figure 42: Interface Parent (generateur)</b> .....	<b>48</b>
<b><i>Figure 43: Interface Parent (message)</i></b> .....	<b>49</b>
<b>Figure 44: Interface Parent (justification formulaire)</b> .....	<b>49</b>
<b>Figure 45: Interface Parent (contact formulaire)</b> .....	<b>49</b>
<b>Figure 46 : Formulaire de contact</b> .....	<b>50</b>
<b>Figure 47 : E-mail réponde à réclamation</b> .....	<b>50</b>
<b><i>Figure 48 : About us Interface</i></b> .....	<b>51</b>

## Introduction générale

La gestion des absences des élèves représente un défi majeur pour les établissements scolaires. Il est indispensable de suivre de manière précise les absences des élèves pour des raisons de sécurité et de suivi académique. Toutefois, la gestion manuelle de ces données peut être fastidieuse et source d'erreurs. Dans ce contexte, l'utilisation d'une application de gestion d'absence basée sur les codes QR peut s'avérer être une solution efficace. Cette application permet aux élèves de scanner un code QR lors de leur arrivée et de leur départ, permettant ainsi à l'établissement scolaire de suivre de manière précise et automatisée les absences et les présences de chaque élève.

Dans ce rapport, nous présenterons la conception et la réalisation d'une telle application pour la gestion des absences des élèves basée sur les codes QR. Nous expliquerons les choix technologiques et les étapes de développement nécessaires pour mettre en place cette application. Nous montrerons également comment cette application répond aux besoins spécifiques de l'établissement scolaire en matière de suivi des absences et de gestion académique. Nous discuterons également des avantages et des limites de l'utilisation de cette application, notamment en termes de sécurité et de confidentialité des données des élèves. Enfin, nous aborderons les perspectives d'avenir pour cette application, y compris les améliorations possibles et les possibilités d'extension à d'autres fonctionnalités pour la gestion des élèves dans un environnement scolaire.

## Chapitre 1

# Présentation Générale Du projet

## Chapitre 1

### ***II. Introduction :***

Dans ce chapitre, nous allons présenter et détailler le cahier de charge en traitant la problématique et proposant une solution adéquate au besoin.

### ***III. Cadre du projet :***

Ce projet entre dans le cadre de préparation de projet de fin d'études pour l'obtention de diplôme universitaire de technologie à l'école supérieure de technologie de Casablanca pendant une durée de 3 mois.

### ***IV. Présentation du cahier de charge :***

#### ***IV.1 Problématique :***

au niveau des études primaires, la majorité des écoles ont des problèmes au niveau de perte du temps lors des séances de cours et parmi les causes de ce problème, c'est lorsque les profs font l'absence d'une manière classique ce qui engendre une perte du temps ainsi qu'une déconcentration des élèves. Au

#### ***IV.2 La solution :***

Pour résoudre cette problématique on tient à informatiser cette tâche à travers la création d'une application web de gestion d'absence basée sur le QR Code

. A travers cette technologie l'école va marquer l'absence des élèves au moment de leurs entrées à l'école ou il trouve l'agent de sécurité qui va scanner le code QR imprimé dans le cartable

de chaque élève, le scan de chaque élève présent va introduire une liste présence qui va être consultée par son professeur à travers son espace pour la vérification de la liste.

Cette Solution peut informatiser aussi l'archivage des listes de présences au cas de besoin, ainsi qu'elle donne la possibilité des justifier l'absence à travers un espace dédié au parent de l'élève.

## V. CONCLUSION:

Dans ce chapitre, nous avons présenté le cadre du projet. En outre, nous avons exposé le cahier des charges afin de répondre aux besoins et à la problématique. Par la suite, nous avons proposé une solution qui sera en adéquation avec la problématique. Cela nous permettra alors d'aborder la prochaine étape, qui concerne la présentation de la phase de conception adoptée pour répondre à la problématique.

## Chapitre 2

# Conceptions et Modélisation Du Projet

## Chapitre 2

### I. Introduction

La conception est une étape essentielle dans la création de tout produit, service ou système. Elle consiste à planifier, à modéliser et à développer des idées en vue de produire quelque chose de nouveau, original et utile. En informatique, la conception est un processus qui permet de concevoir et de développer des logiciels en passant par des étapes d'affinage. Elle est essentielle pour garantir la qualité et la fiabilité du produit final, ainsi que pour répondre aux besoins et aux attentes des utilisateurs. Dans cet esprit, la conception joue un rôle crucial dans l'innovation, la compétitivité et le succès des entreprises et des organisations.

### II. Langage de conception

Pour élaborer cet application on doit établir une conception modeste pour attentera le but de notre projet pour cela on doit choisir un langage de conception adaptable avec notre besoins.

#### II.1 Langage UML

« UML » (en anglais *Unified Modeling Language* ou *langage de modélisation unifié*) est un langage de modélisation graphique à base de pictogrammes. Il est apparu dans le monde du génie logiciel, dans le cadre de la « conception orientée objet ».

Couramment utilisé dans les projets logiciels, il peut être appliqué à toutes sortes de systèmes ne se limitant pas au domaine informatique.

Les 14 diagrammes UML sont dépendants hiérarchiquement et se complètent, de façon à permettre la modélisation d'un projet tout au long de son cycle de vie. Ces diagrammes sont :

- Diagrammes structurels ou statiques:
  - Diagramme de classes
  - Diagramme d'objets
  - Diagramme de composants

## Chapitre 2

- Diagramme de déploiement
- Diagramme des paquetages
- Diagramme de structure composite
- Diagramme de profils

### ➤ Diagrammes comportementaux :

- Diagramme des cas d'utilisation
- Diagramme états-transitions
- Diagramme d'activité

### ➤ Diagrammes d'interaction ou dynamiques :

- Diagramme de séquence
- Diagramme de communication
- Diagramme global d'interaction
- Diagramme de temps

## II.2 Utilisation de l'UML

L'UML est un langage de modélisation normalisé et formel pour les systèmes orientés objet. Sa flexibilité, sa polyvalence et son indépendance vis-à-vis des langages de programmation en ont fait un langage universel. UML est principalement utilisé comme outil de communication, facilitant la représentation et la compréhension de solutions orientées objet. Sa notation graphique permet une expression visuelle claire des solutions, réduisant ainsi l'ambiguïté et les malentendus et facilitant la comparaison et l'évaluation des solutions.

## III. Les Diagrammes de conception

### III.1 Modélisation comportementaux :

#### III.1.1 Diagramme de cas d'utilisation

##### III.1.1.1 Définition

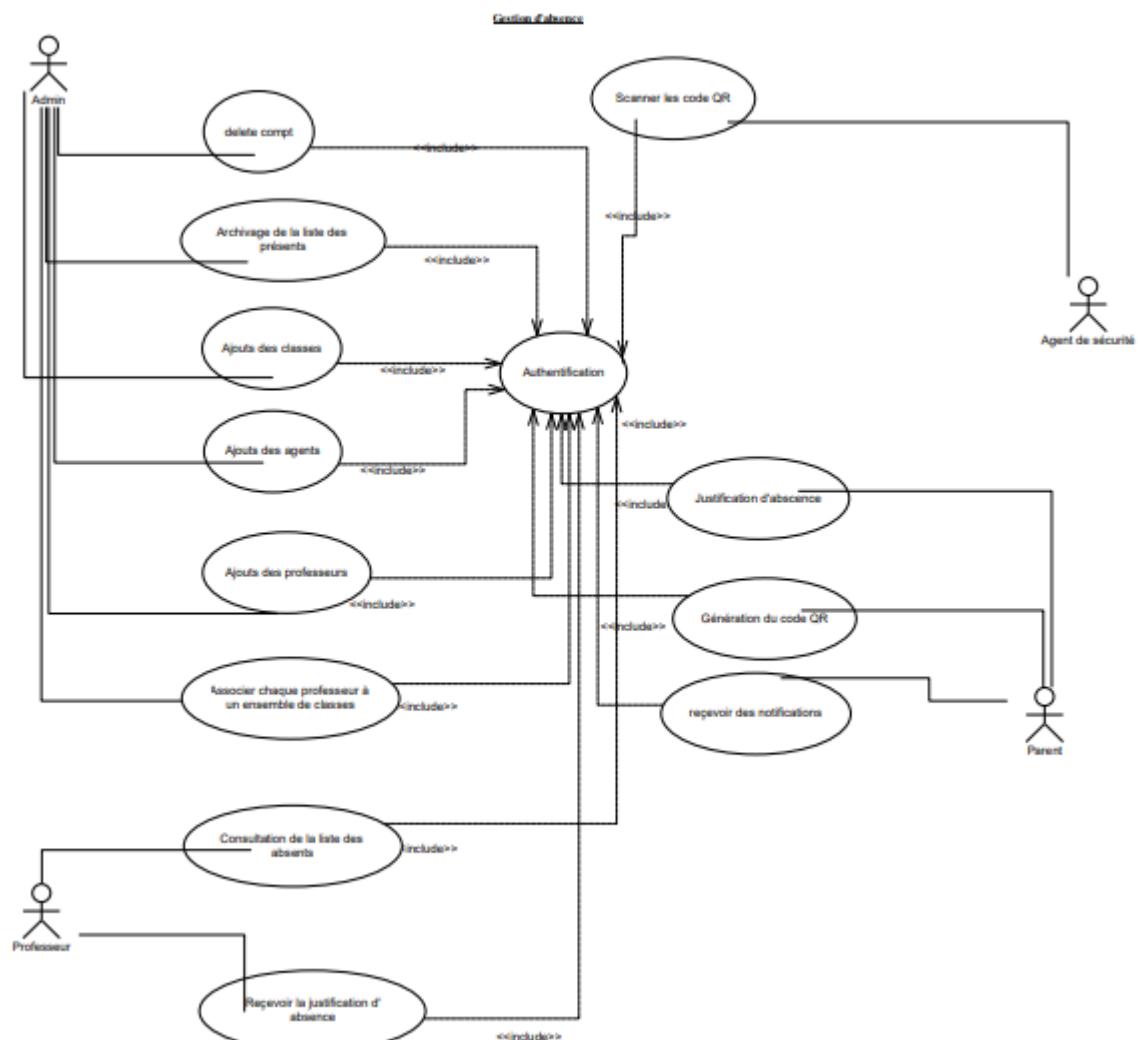
Un diagramme de cas d'utilisation utilisé pour représenter les interactions entre les acteurs et le système. Il décrit les différentes actions et fonctionnalités du système du point de

## Chapitre 2

vue de l'utilisateur. Il est généralement utilisé pour définir les exigences fonctionnelles d'un système et pour fournir une vue globale des fonctionnalités offertes par ce système. Le diagramme est constitué d'acteurs et de cas d'utilisation, reliés par des flèches qui représentent

les interactions entre l'utilisateur et le système. En somme, c'est un outil pour décrire les interactions entre les utilisateurs et le système de manière claire et concise.

### III.1.1.2 Diagramme de cas d'utilisation



**FIGURE 1 : diagramme de cas d'utilisation**

## Chapitre 2

### III.2 Modélisation dynamique

#### III.2.1 Diagrammes de séquences

Le diagramme de séquence représente la succession chronologique des opérations réalisées par un acteur. Il indique les objets que l'acteur va manipuler et les opérations qui font passer d'un objet à l'autre.

#### III.2.2 Diagrammes de séquences de Security

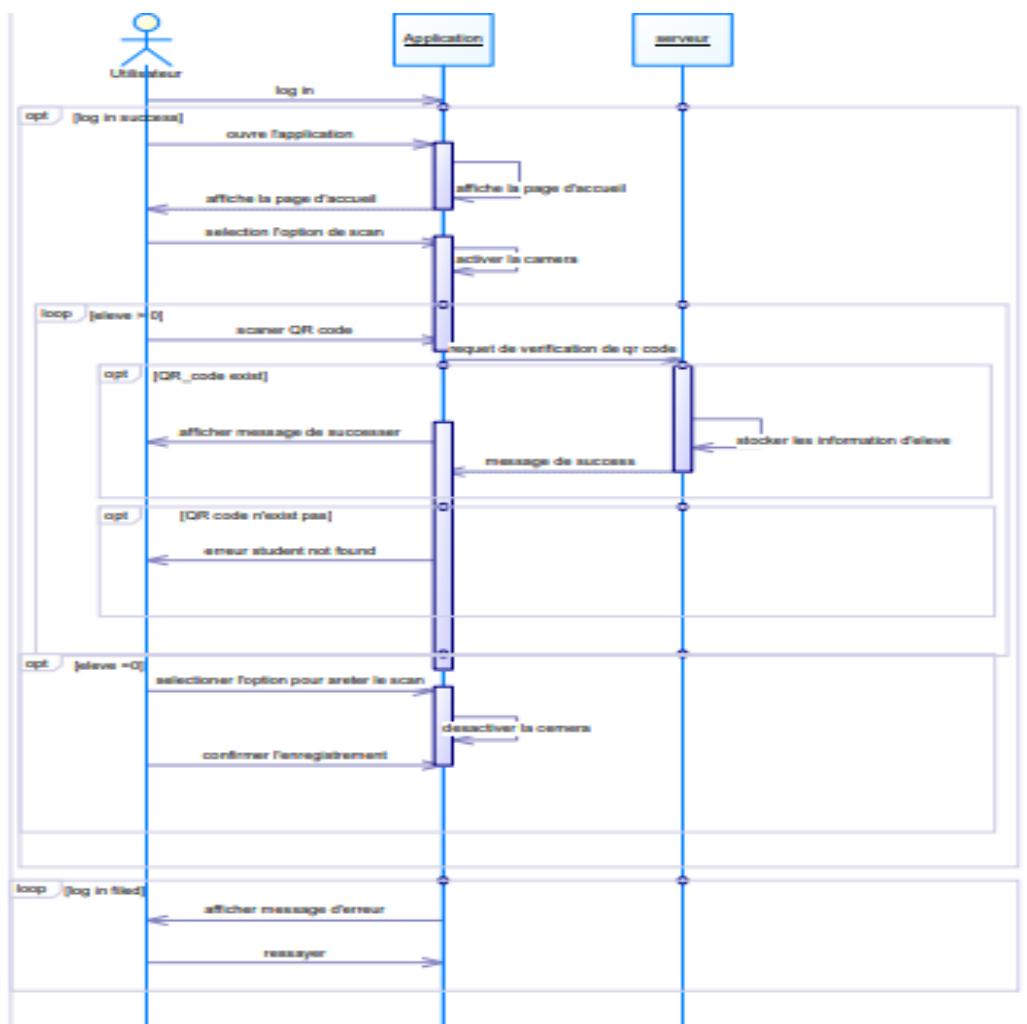
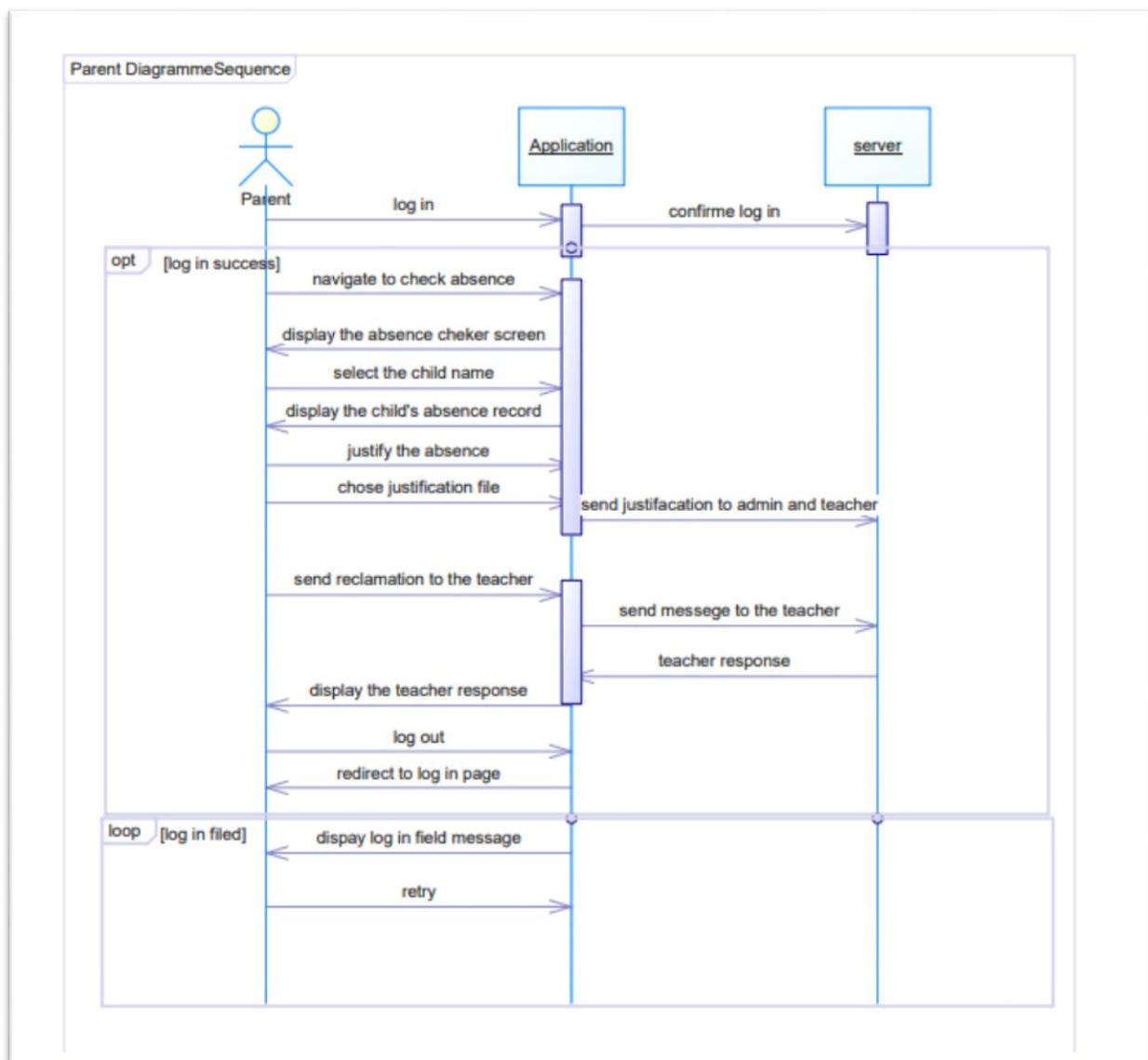


FIGURE 2 : diagramme de séquence de Security

Ce diagramme représente l'activité de l'agent de sécurité. Après s'être authentifié avec succès, l'agent peut ouvrir l'application de scan et commencer à scanner les QR codes.

des élèves. Si le QR code n'existe pas ou s'il y a un problème lors du scan, l'application affiche un message d'erreur. Sinon, l'élève est marqué présent dans la base de données. Si aucun élève n'est disponible pour être scanné, l'agent de sécurité peut cliquer sur l'option pour arrêter le scan. En cas de problème lors de l'authentification, un message d'erreur s'affiche et l'agent de sécurité doit retenter de saisir le nom d'utilisateur et le mot de passe corrects.

### III.2.3 Diagrammes de séquences de parent



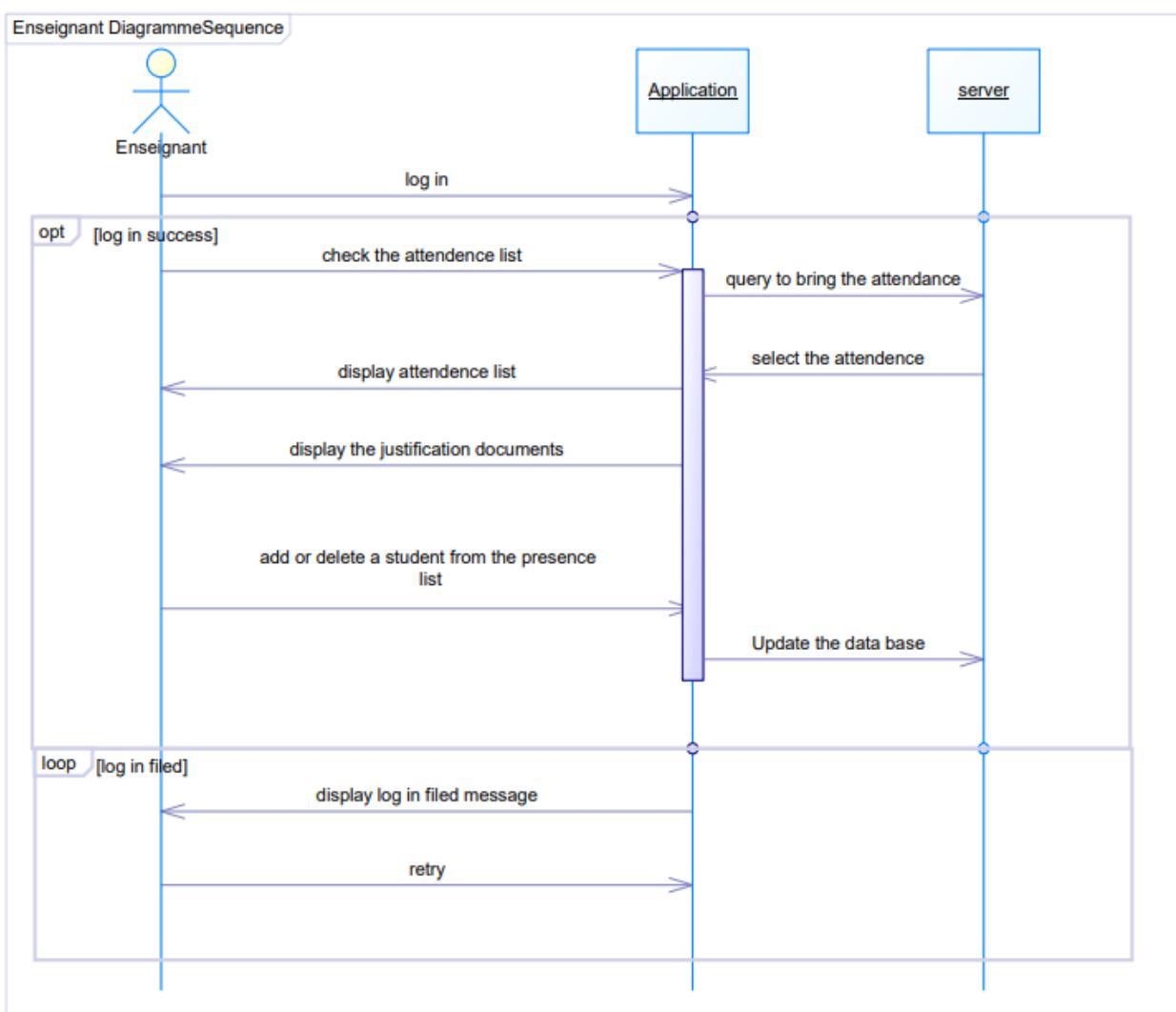
**FIGURE 3 : diagramme de séquence de parent**

- Ce diagramme représente l'activité d'un parent. Après s'être authentifié, il peut accéder à l'application pour vérifier l'absence de son enfant. L'application affiche ensuite

→ l'interface où se trouvent les enfants du parent. Le parent sélectionne l'enfant dont il souhaite consulter le statut d'absence, puis l'application affiche l'absence de l'enfant. Le parent peut justifier l'absence de son enfant et contacter les enseignants.

Si l'authentification échoue, un message d'erreur s'affiche. Le parent peut réessayer jusqu'à ce que le nom d'utilisateur et le mot de passe soient corrects.

### III.2.4 Diagrammes de séquences d'enseignant



*FIGURE 4 : diagramme de séquence d'enseignant*

→ Ce diagramme représente l'activité de l'enseignant. Après s'être authentifié, il peut accéder à l'application pour voir la liste des présences. Il peut également consulter

les justifications associées à chaque absence, s'il y en a. Si des changements surviennent dans la liste des présences, l'enseignant peut ajouter ou supprimer un élève.

Si un problème d'authentification survient, l'enseignant ne pourra pas accéder à l'application tant que le nom d'utilisateur et le mot de passe ne seront pas corrects.

### III.2.5 Diagrammes de séquences d'Administrateur

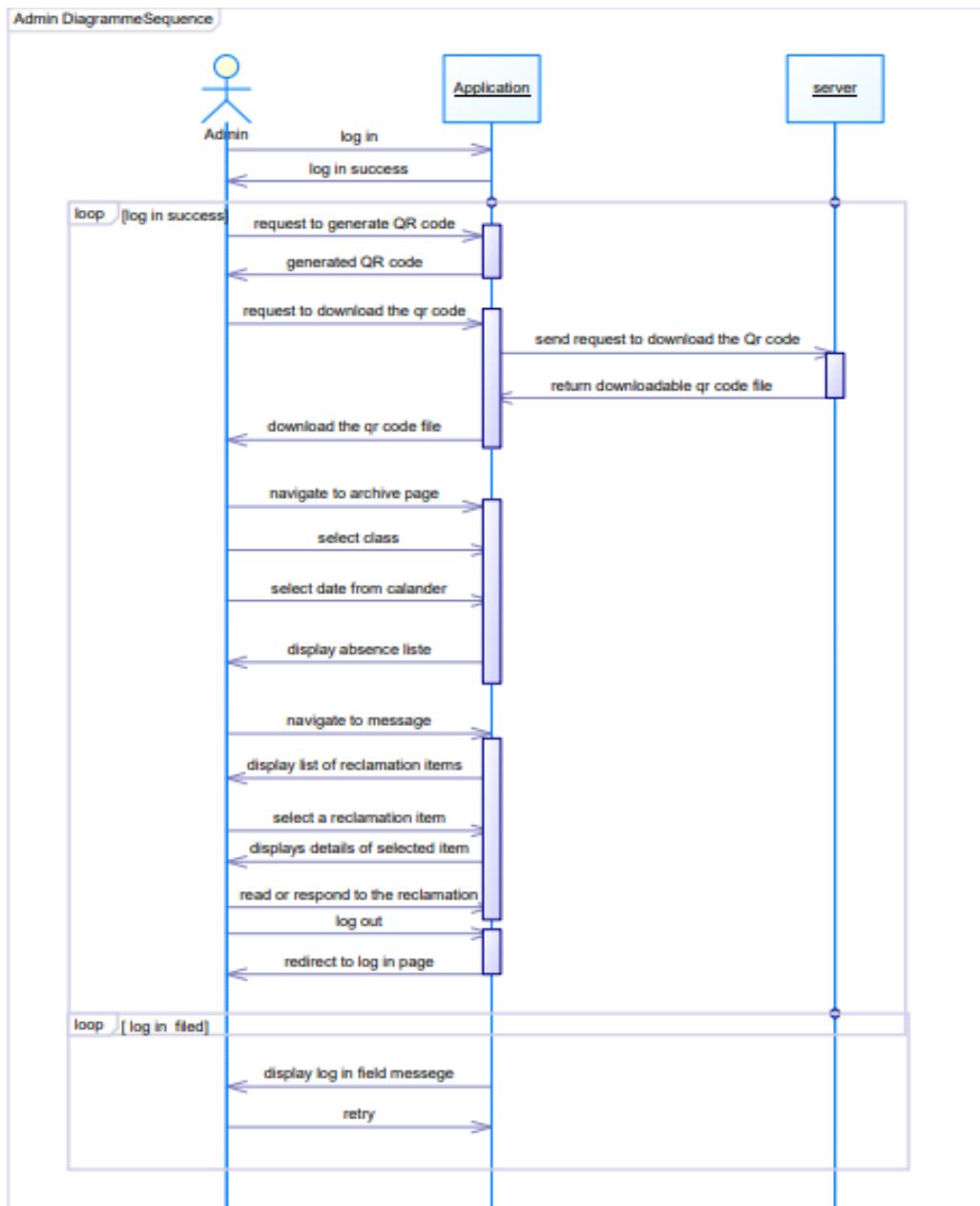


FIGURE 5 : diagramme de séquence d'Administrateur

→ Ce diagramme représente l'activité de l'administrateur. Après s'être authentifié, l'administrateur peut générer le code QR d'un élève en entrant ses informations. L'application génère ensuite une copie que l'administrateur peut imprimer

L'administrateur peut également consulter les archives d'absences de chaque classe. En outre, il peut accéder à la boîte de réclamation pour voir les réclamations et y répondre.

Si un problème d'authentification se produit, L'administrateur ne pourra pas accéder à l'application tant que le nom d'utilisateur et le mot de passe ne seront pas corrects.

### III.3 Modélisation statique

#### III.3.1 Diagramme de classes

Le diagramme de classe est une représentation statique des éléments constitutifs d'un système ainsi que de leurs relations. Il permet de visualiser les différentes classes impliquées dans le système.

#### III.3.2 Diagramme de classe global

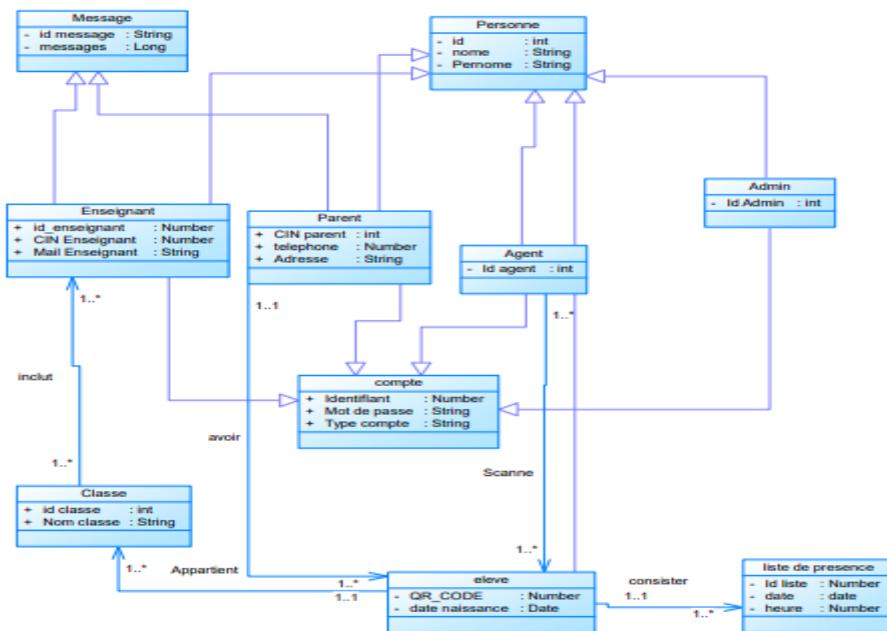


FIGURE 6 : diagramme de classes

**III.3.3 Modèle logique de données liée au diagramme de classe**

**compte** (Identifiant, Mot\_de\_passe, Type\_compte );

**admin** (Id\_admin, #Identifiant);

**parent** (CIN\_parent , nom\_parent , telephone , Adresse , #Identifiant);

**professeur** (Id\_professeur, Nom\_professeur , prenom\_prof ,CIN\_prenom, Mail\_prof, #Identifiant);

**classe** (Id\_classe, Nom\_classe , #ID\_ecole);

**Message** (Id\_Message, messages);

**Agent** (Id\_agent , Nom\_agent , Prenom\_agent , #Identifiant);

**liste\_presence** (Id\_liste , , #Id\_classe);

**Eleve** (CNE\_eleve, Nom\_eleve , prenom\_eleve , QR\_code , Date\_naissance, #CIN\_parent, #Id\_classe, #Id\_liste);

scanne = (#Id\_agent, #CNE\_eleve);

inclus = (#Id\_professeur, #Id\_classe);

recevoir = (#Id\_professeur, #Id\_Message);

envoyer = (#CIN\_parent, #Id\_Message);

## III.4 Diagramme d'Activité

### III.4.1 Définition

Un diagramme d'activité est un type de diagramme UML qui permet de représenter le comportement dynamique d'un système en décrivant la séquence d'actions d'un processus et en montrant les flux entre les actions. Les nœuds d'activité représentent les actions effectuées dans le système, tandis que les bords d'activité modélisent le flux de commande et de données entre les actions. Les diagrammes d'activité sont similaires aux organigrammes de traitement de l'information, mais peuvent également montrer les flux parallèles simultanés et les flux de remplacement.

### III.4.2 Diagramme d'Activité

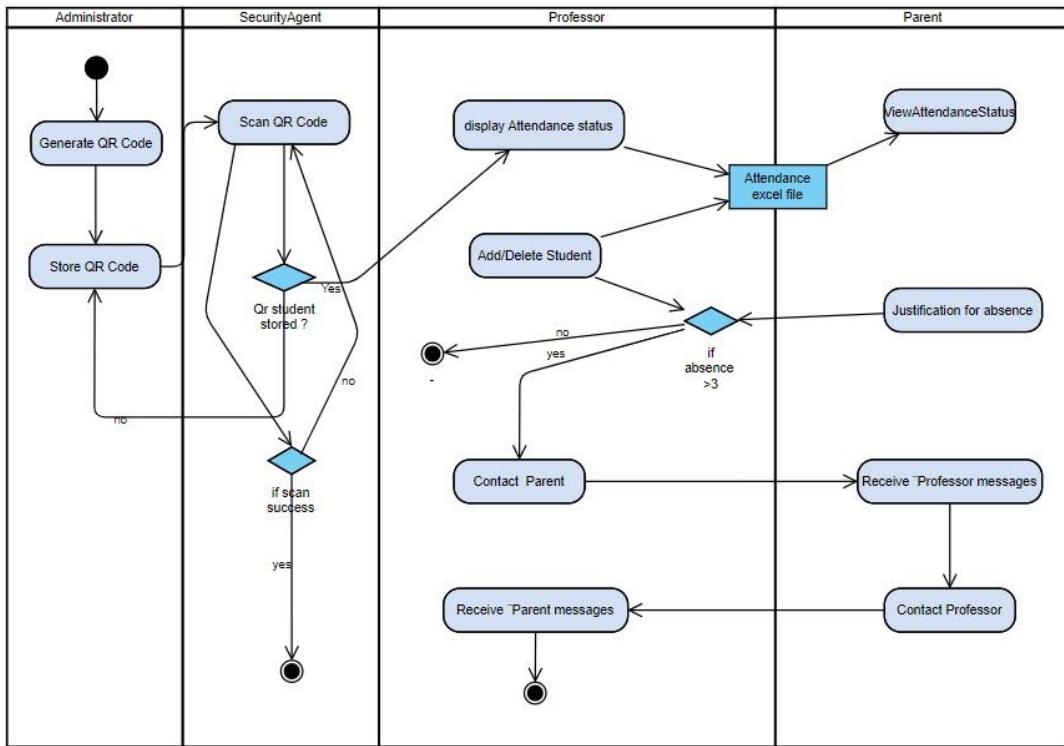
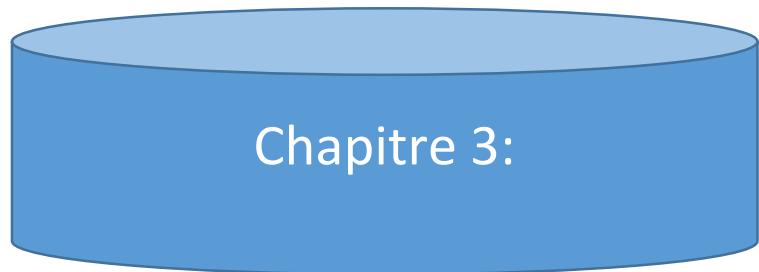


FIGURE 7 : diagramme d'activité

## IV. Conclusion

Ce chapitre a porté sur les différentes étapes de la conception détaillée. Nous avons commencé par affiner des diagrammes de séquence pour plusieurs cas d'utilisation afin de montrer les interactions entre les acteurs et le système, en suivant nous avons élaboré le diagramme de classe qui permet de présenter la structure de notre application, puis nous avons créé le diagramme de cas d'utilisation. Enfin, nous avons terminé le diagramme d'activité pour décrire les étapes détaillées des processus métiers impliqués dans le système.

Dans le prochain chapitre, nous fournirons des explications et des clarifications sur plusieurs concepts liés au contexte de notre travail.



# Phase de Réalisation



## I. Introduction

Dans ce chapitre, nous avons déjà parcouru les étapes nécessaires pour mener à bien la phase de réalisation. Le problème a été analysé en profondeur et nous avons établi une conception complète selon notre jugement. Cette conception décrit tous les besoins de l'application.

Dans un premier temps, nous décrirons l'environnement de développement (logiciel) ainsi que les différents outils utilisés. Ensuite, nous présenterons les différents aspects de fonctionnement de notre application.

## II. Environnement logiciel

- ❖ Visual Studio Code



*Figure8 : Visual Studio Code*

Visual Studio Code est un éditeur de code extensible développé par Microsoft pour Windows, Linux et macOS. 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

❖ Power Designer



Figure 9 : Power Designer

Power Designer est un logiciel de conception créée par la société Sysbase, qui permet de modéliser les traitements informatique en utilisent les diagrammes UML

❖ HTML



Figure 10 : HTML

HTML signifie "*Hypertext Markup Language*" en anglais, ce qui se traduit en français par "Langage de balisage hypertexte". C'est un langage informatique utilisé pour la création et la structuration de pages web.

❖ CSS



Figure 11 : CSS

CSS est un langage de style qui permet aux développeurs web de séparer la présentation d'un document de son contenu. En utilisant des règles de style pour différents éléments d'une page web, les développeurs peuvent créer des sites web visuellement attrayants et fonctionnels. L'utilisation de CSS facilite également la maintenance et la mise en forme de plusieurs pages d'un site web. En somme, CSS est un outil essentiel pour les développeurs et les designers web pour créer des sites web de qualité.

## ❖ JS

*Figure 12 : JavaScript*

JavaScript est un langage de programmation essentiel pour le développement web moderne. Il permet aux développeurs de créer des pages web interactives, des applications web, des jeux, des animations et des effets visuels. Avec sa capacité à manipuler le contenu HTML et CSS d'une page, ainsi qu'à interagir avec des serveurs web, JavaScript est un langage polyvalent et puissant qui continue d'évoluer et de s'adapter aux besoins changeants du web.

## ❖ PostgreSQL

*Figure 13: PostgreSQL*

PostgreSQL est un système de gestion de base de données relationnelle et objet. C'est un outil libre disponible selon les termes d'une licence de type BSD. Ce système est comparable à d'autres systèmes de gestion de base de données, qu'ils soient libres, ou propriétaires.

## ❖ Bootstrap

*Figure 14 : Bootstrap*

Bootstrap est un ensemble d'outils pratiques pour la conception de sites web et d'applications. Il comprend des codes HTML et CSS, des formulaires, des boutons, des outils de navigation et autres éléments interactifs, ainsi que des extensions JavaScript en option.

## Chapitre 3

### ❖ Django

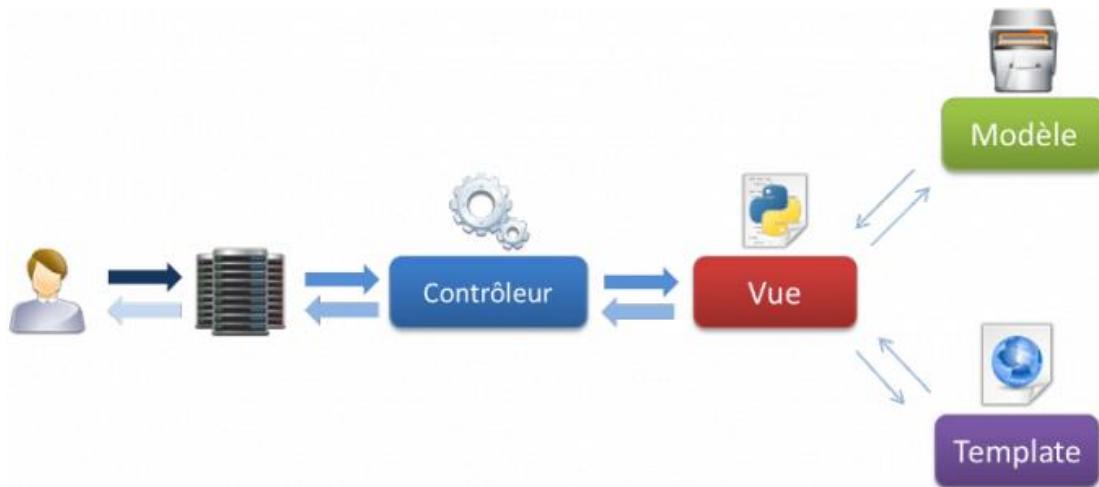


*Figure 15 : Django*

Django est un framework web populaire utilisé pour construire de nombreux sites web et applications web de haute qualité. Il bénéficie d'une grande communauté de développeurs actifs, ce qui signifie qu'il existe de nombreuses ressources disponibles pour l'apprentissage et la résolution de problèmes. Django est également hautement évolutif, ce qui lui permet de gérer un grand nombre d'utilisateurs et un trafic important. De plus, Django est conçu pour être sécurisé, avec une protection intégrée contre les vulnérabilités courantes des applications web. Globalement, Django est un framework web polyvalent et puissant qui peut être utilisé pour construire des applications web complexes et robustes.

#### II.1 Structure de Django

Django suit une structure de projet MVC (Modèle-Vue-Contrôleur) qui permet de séparer les différentes parties de l'application et de les rendre plus facilement gérables



*Figure 16 : Architecture de Django*

Dans Django, le modèle MVC est implémenté en utilisant un modèle d'architecture légèrement différent appelé **Modèle-Vue-Template (MVT)**. Dans MVT, le Modèle représente les données et le schéma de la base de données, la Vue gère la logique métier

## Chapitre 3

et la manipulation de données, et le Template affiche les données à l'utilisateur. Voici une brève explication de la façon dont chaque composant fonctionne dans l'implémentation de MVT de Django :

- **Modèle** : Un modèle est une classe Python qui définit les champs et les comportements des données que vous souhaitez stocker dans la base de données de votre application. L'ORM (Mapping objet-relationnel) de Django facilite la définition des modèles et l'interaction avec la base de données.
  - **Vue** : Une vue est une fonction ou une classe Python qui gère une demande HTTP spécifique et renvoie une réponse HTTP. Dans Django, les vues sont responsables de l'interrogation de la base de données et de la réalisation de la logique métier sur les données.
  - **Template** : Un template est un fichier texte qui définit la structure et la mise en page de l'HTML qui sera renvoyé à l'utilisateur. Le langage de template de Django vous permet d'afficher dynamiquement des données à partir des modèles de votre application.
- ❖ "Structure de répertoires standard d'un projet Django : Explication des répertoires et fichiers".
- **Le dossier racine** : C'est le dossier principal de votre projet Django. Il contient tous les fichiers et dossiers nécessaires à la création de l'application. Vous pouvez nommer ce dossier comme vous le souhaitez.
  - **Le dossier "applications"** : Ce dossier contient toutes les applications de votre projet. Chaque application est un ensemble de fonctionnalités indépendantes qui peuvent être réutilisées dans d'autres projets Django. Chaque application doit avoir son propre dossier.
  - **Le dossier "static"** : Ce dossier contient tous les fichiers statiques, tels que les fichiers CSS, JavaScript et les images, utilisés dans votre projet.
  - **Le dossier "templates"** : Ce dossier contient tous les modèles HTML utilisés dans votre projet.

- **Le fichier "manage.py"** : Ce fichier est utilisé pour gérer votre projet Django. Vous pouvez utiliser ce fichier pour effectuer des tâches telles que la création d'une base de données ou le lancement d'un serveur de développement.
- **Le fichier "settings.py"** : Ce fichier contient toutes les configurations de votre projet Django, telles que les paramètres de la base de données, les paramètres de sécurité et les paramètres de l'application.
- **Le fichier "urls.py"** : Ce fichier contient toutes les URL de votre projet, ainsi que les vues associées.

### III. Installation

→ Installation de Django :

**pip install Django**

→ créer et activer un environnement virtuel par les commandes :

**python -m venv name**

**py -m venv venv**

**cd .\venv\Scripts\**

**.\activate**

→ Pour Lancer le serveur sur votre machine locale :

**python manage.py runserver**

→ Et on a commencé d'installer les modèles qu'on avait besoin par les commandes suivantes :

**pip install opencv-python**

**pip install pyzbar**

## IV. QR\_CODE

Le code QR (Quick Response Code) est un code-barres bidimensionnel qui peut être lu par un scanner de code QR ou un appareil photo de smartphone. Il a été développé pour la première fois par une entreprise japonaise appelée Denso Wave en 1994 pour l'industrie automobile, mais est depuis devenu populaire pour une variété d'applications.

Voici quelques éléments clés sur les codes QR :



Figure 17 : QR CODE

- Les codes QR peuvent stocker une variété d'informations telles que des URL, du texte, des numéros de téléphone, et plus encore.
  - Ils peuvent être scannés à l'aide d'un scanner de code QR ou d'un appareil photo de smartphone avec une application de lecteur de code QR
- 
- Les codes QR peuvent être créés et personnalisés à l'aide d'outils en ligne gratuits ou de logiciels.
  - Les codes QR sont utilisés dans une variété d'applications, notamment la publicité, la billetterie, le paiement, et plus encore.
  - Les codes QR peuvent être utilisés pour offrir une expérience sans contact et sans contact dans diverses situations.
  - Les codes QR peuvent également être utilisés à des fins de suivi et d'analyse.

Lors de l'utilisation de codes QR, il est important d'assurer que le code est scannable, contient des informations précises, et mène à un site web ou une application sécurisée. Il est également essentiel d'assurer que les utilisateurs comprennent comment scanner et utiliser le code QR.

## V. Présentation de l'application

Le volet technique de ce chapitre étant achevé, nous allons maintenant consacrer cette partie à la présentation des principales interfaces de notre application de gestion d'absences. La première interface, qui est affichée aux administrateurs et aux simples utilisateurs, est l'écran d'accueil modélisé dans la figure suivante

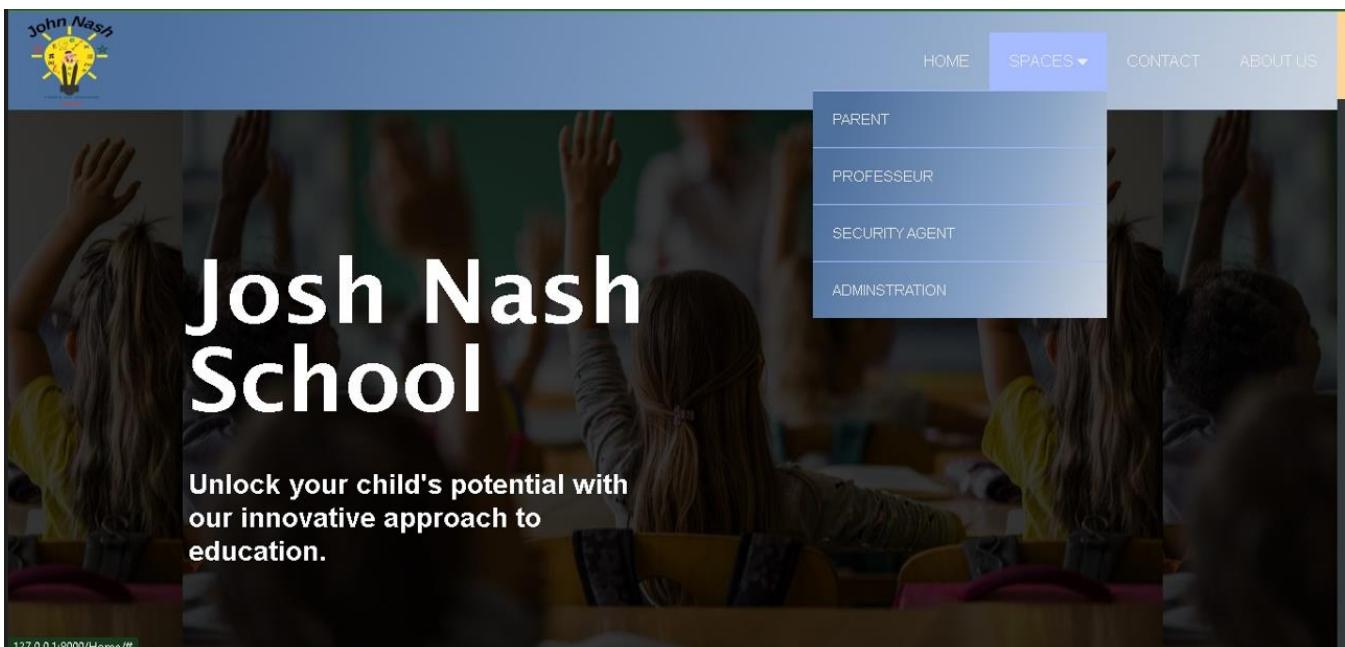
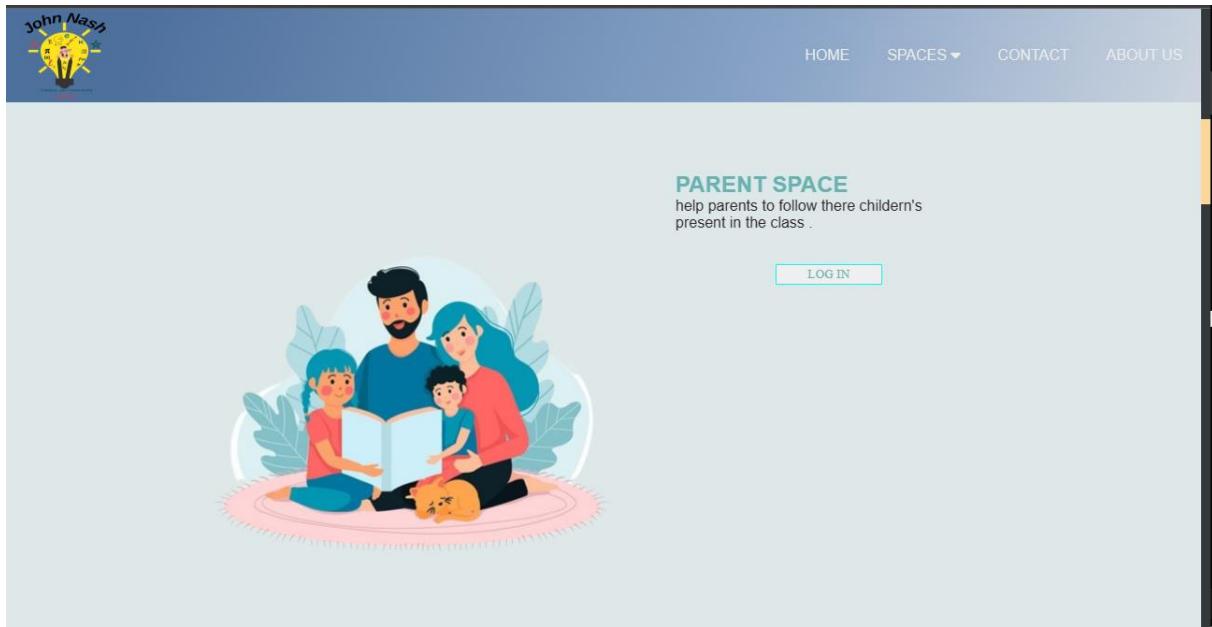
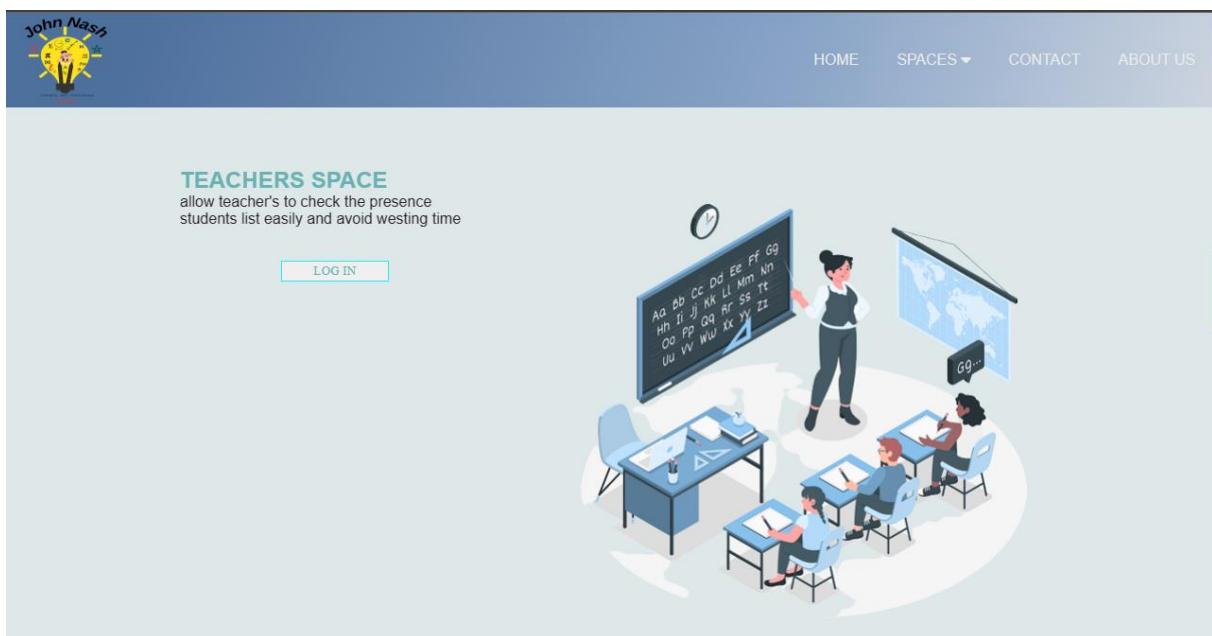


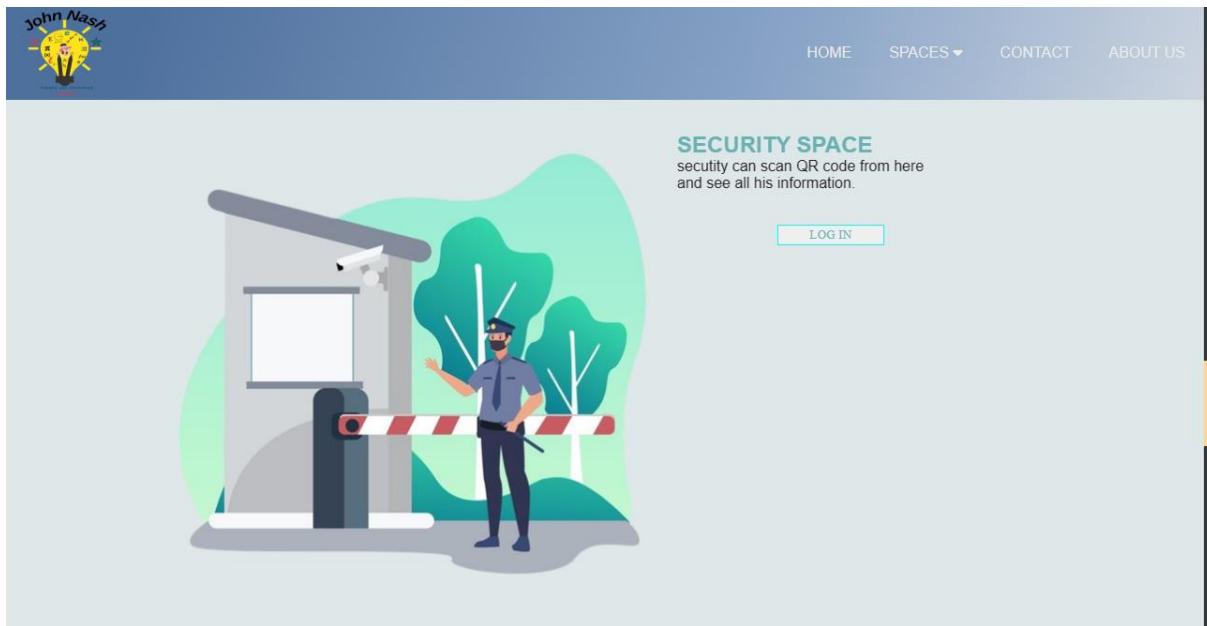
Figure 18 : Interface d'accueil



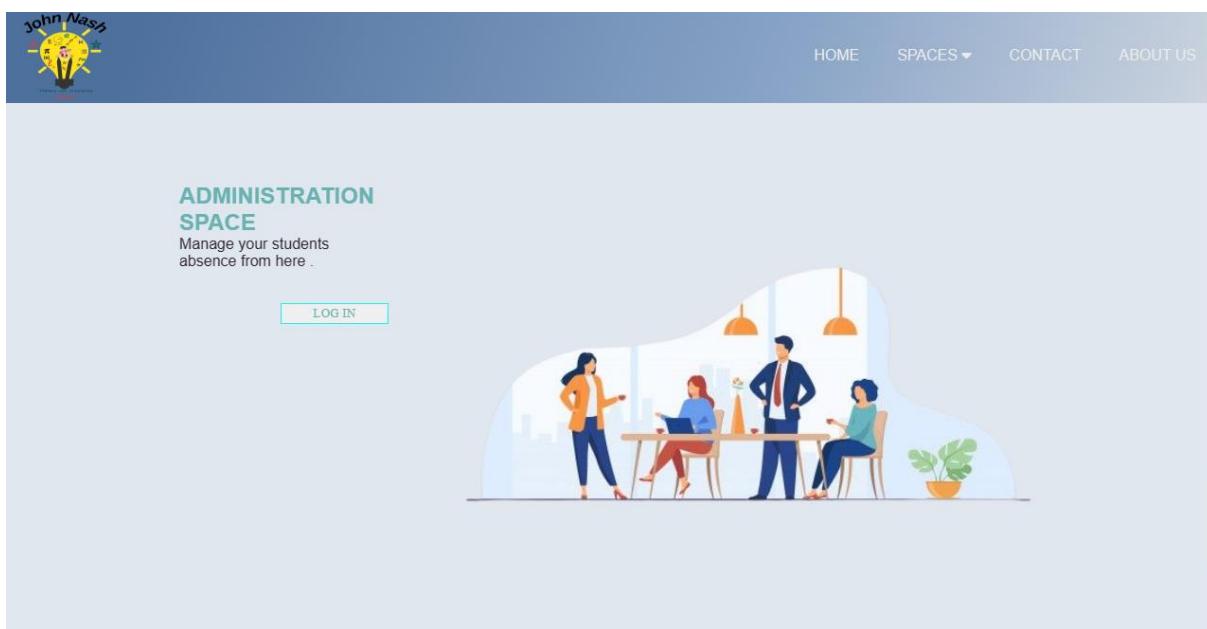
*Figure 19 : Interface d'accueil (Parent section)*



*Figure 20 : Interface d'accueil (Teachers section)*



*Figure 21 : Interface d'accueil (Security Agent section)*

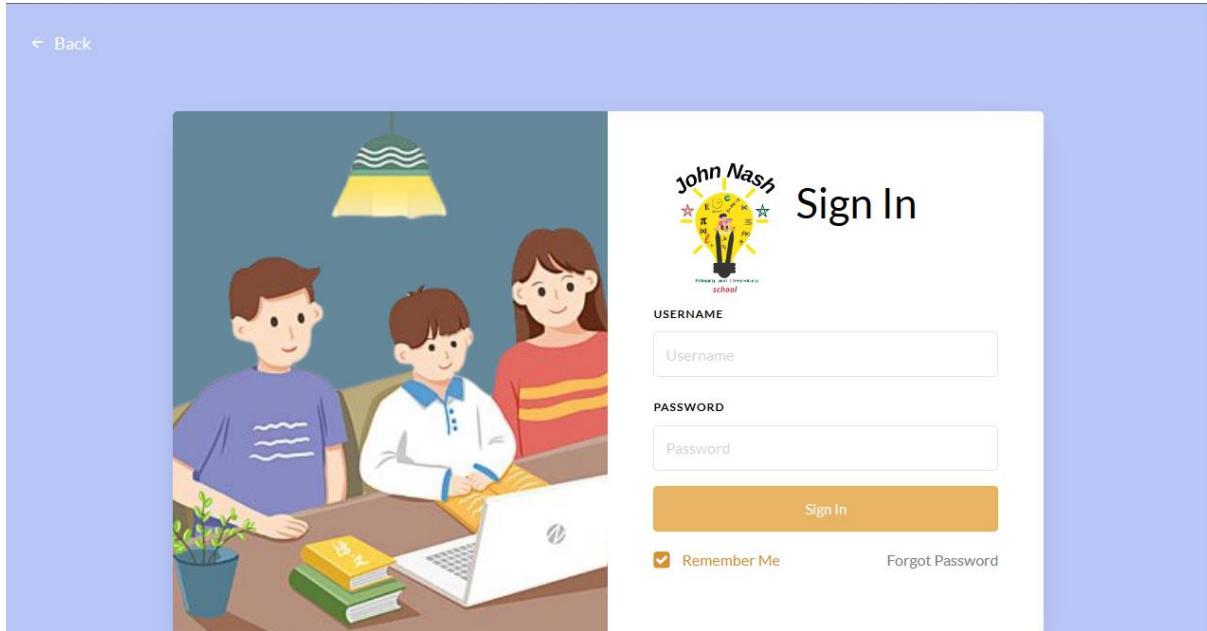


*Figure 22 : Interface d'accueil (Administration section)*

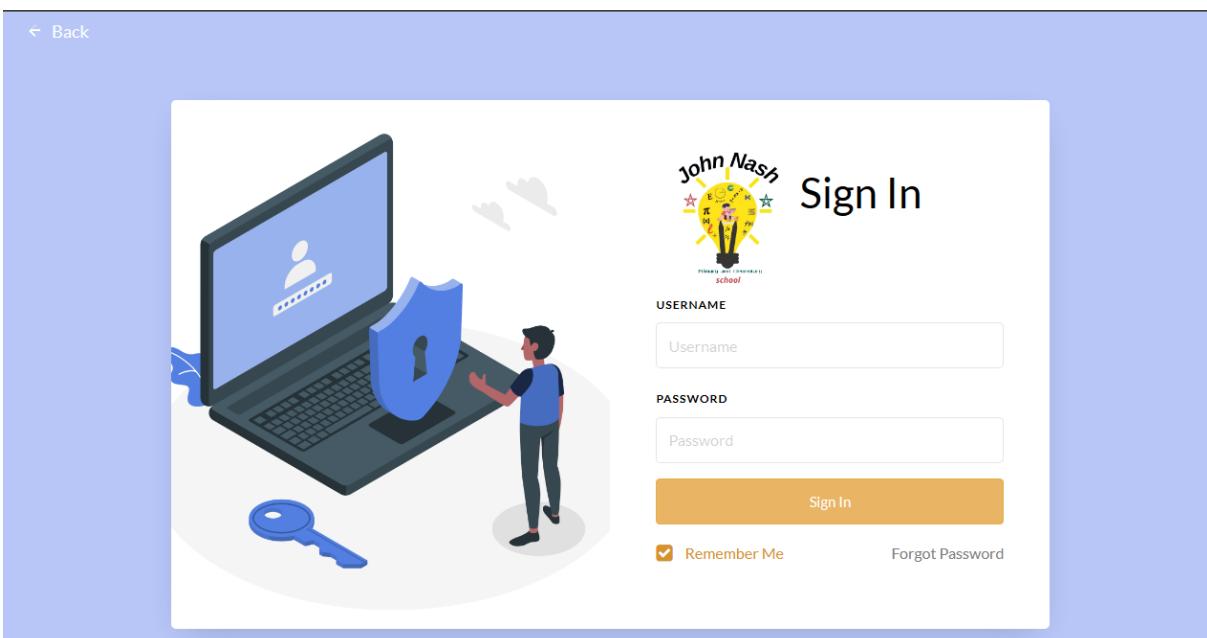
## V.1 les Espaces

### V.1.1 les interfaces de l'authentification

- Après avoir cliqué sur le bouton "se connecter", chaque interface mène à une page de connexion associée.



*Figure 23 : Interface Log In(Parent)*



*Figure 24: Interface Log In(Teacher)*

## Chapitre 3

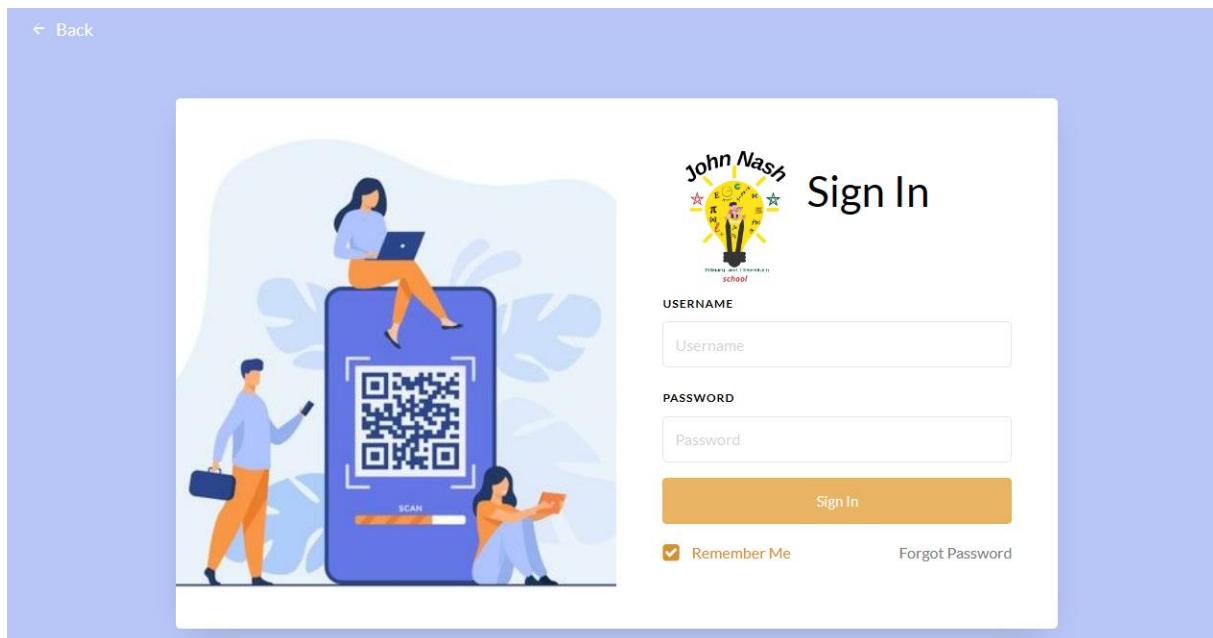


Figure 25 : Interface Log In(Agent)

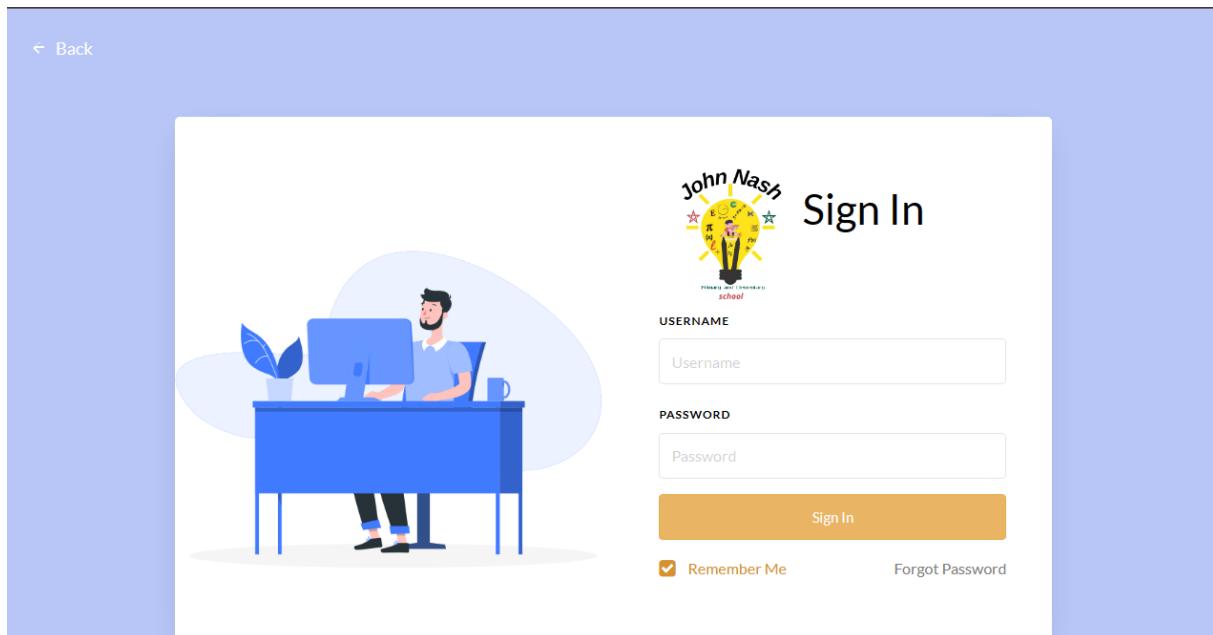
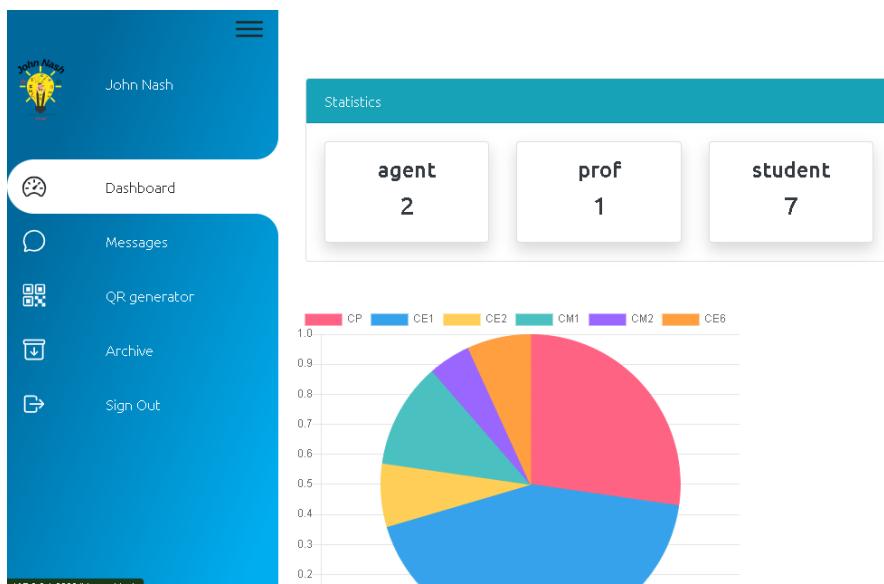


Figure 26 : Interface Log In(Administration)

### V.1.2 Interface Administrateur

- Après la connexion d'un administrateur sur son site, il y a un tableau de bord qui s'affiche et une boîte de messages pour les réclamations de problèmes dans sidebar,

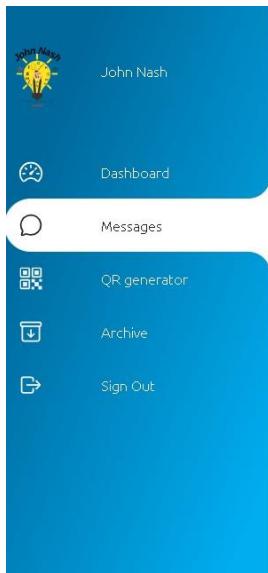


ainsi un générateur de codes QR et les archives d'absence d'élèves.

*Figure 27 : Interface Administration (Dashboard)*

- L'interface de tableau de bord affiche le nombre d'élèves, d'enseignants et d'agents dans l'école, ainsi que les statistiques d'absence de chaque classe.

## Chapitre 3



The screenshot shows the UPEI EST administration dashboard. On the left, there's a sidebar with icons for Dashboard, Messages, QR generator, Archive, and Sign Out. The main area is titled 'RECLAMATION BOX' and displays a table of user complaints:

id message	type_reclamation	date	E-Mail	Message
1	Probleme de connexion	April 9, 2023, 8:47 a.m.	Epsilon34@gmail.com	pppp
2	Probleme de connexion	April 9, 2023, 9:14 a.m.	Epsilon34@gmail.com	pppp
3	Probleme de connexion	April 9, 2023, 9:18 a.m.	Epsilon34@gmail.com	pppp
4	Probleme de connexion	April 9, 2023, 11:14 p.m.	nouaila.moussa04@gmail.com	probleme de connexion

*Figure 28 : Interface Administration (boîte de messages)*

- L'interface de la boîte de messages est un outil précieux pour les administrateurs, car elle leur permet de voir en un coup d'œil toutes les réclamations et les problèmes signalés par les utilisateurs. Cette fonctionnalité est particulièrement utile pour notre sites web , où les administrateurs peuvent être submergés de messages et de requêtes. En ayant une vue d'ensemble des problèmes, les administrateurs peuvent travailler plus efficacement pour résoudre les problèmes signalés et améliorer l'expérience utilisateur. De plus, la boîte de messages peut également servir de canal de communication direct entre les utilisateurs et les administrateurs, permettant ainsi une résolution rapide et efficace des problèmes.



**Inscription**

Prenom	Nom
eleve	Nom_eleve1
<b>CNE</b>	
R139389530	
<b>Cin Parent</b>	
266281	
<b>Date de naissance</b>	
08 / 03 / 2003	
<b>Classe</b>	
1	
<input type="button" value="Generate"/>	<input type="button" value="valider"/>

*Figure 29 : Interface Administration (Générateur de QR code)*

- Dans cette interface, l'administrateur doit insérer les informations de l'élève (nom, prénom, CNE, date de naissance, classe) ainsi que le CIN de son parent. Ensuite, pour créer le QR code, le bouton "*"générer"*" va crypter les informations et générer une chaîne de caractères qui sera transformée en QR code.

Le bouton "*"valider"*" permettra de stocker le QR code dans la base de données.

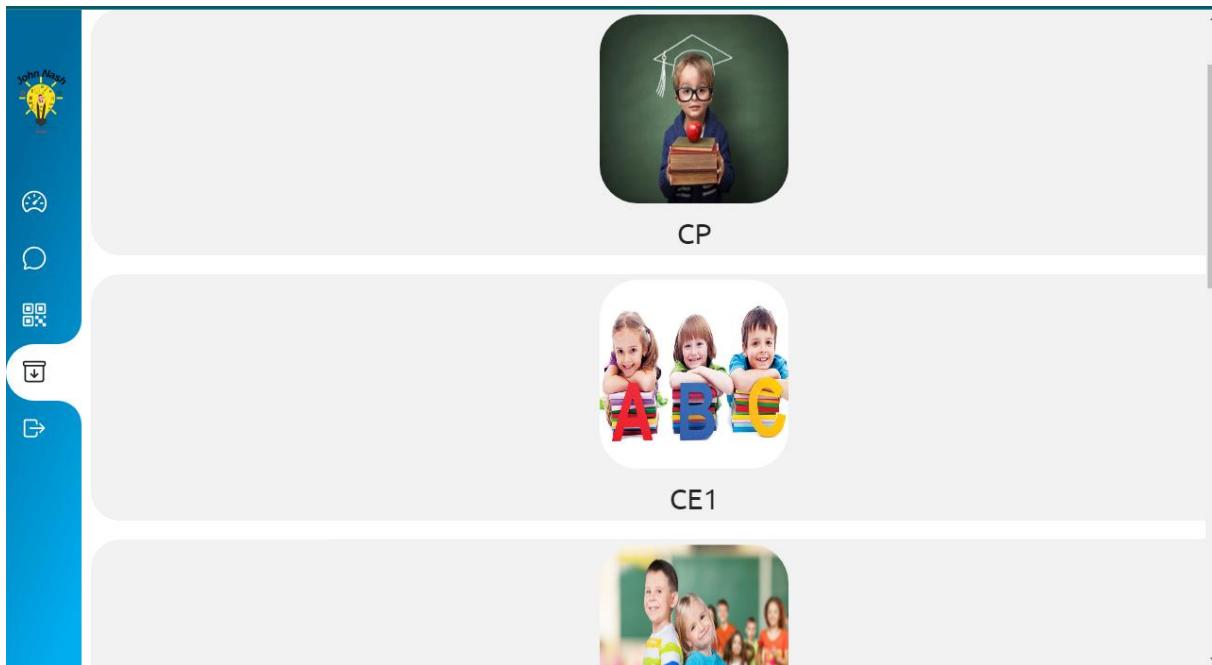
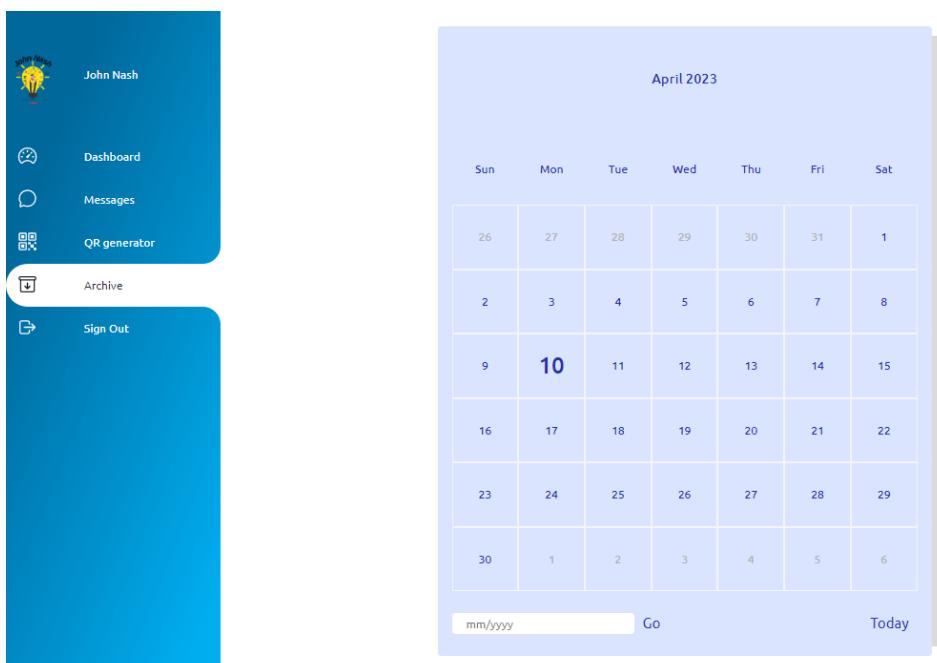


Figure 30 : Interface Administration (Archive : classes)

- Lorsque l'administrateur clique sur l'archive dans sidebar, les classes de différents niveaux s'affichent. Ensuite, l'administrateur clique sur la classe qu'il souhaite afficher dans l'archive, et le calendrier s'affiche comme indiqué dans la figure suivante :



➤ Figure 31 : Interface Administration (Archive : calendrier)

- Sur cette page, l'administrateur doit choisir le mois et l'année d'archive qu'il souhaite visualiser, puis cliquer sur le bouton "GO". Si l'input est vide ou si la date est invalide, un message d'alerte s'affiche. Ensuite, l'administrateur peut sélectionner le jour souhaité, et la liste de présence s'affiche comme illustré dans la figure suivante :

CNE	FIRST NAME	LAST NAME	Class	Presence
R129538	Nadia	Mountasike	CP	P
G9546674	Ahmed	Zaki	CP	A

Figure 32 : Interface Administration (Archive : liste d'archive)

- Restons dans l'archivage. Cette page affiche la liste complète des élèves avec leur statut de présence, les présences étant marquées par 'P' et les absences par 'A' dans la table.
- ❖ Dans la dernière section de "déconnexion", l'utilisateur peut quitter l'interface et sera redirigé vers la page de connexion comme illustré dans la "figure 26".

### V.1.3 interface Agent de Security

- L'interface de l'agent de sécurité représente l'ensemble des opérations que l'agent peut effectuer. Ces opérations sont présentées dans le sidebar, qui présente la page d'accueil le scan de code QR, l'archive des scans de sécurité et enfin la boîte de réclamation s'il y a des problèmes.



Figure 33: Interface Agent De Security (Accueil)

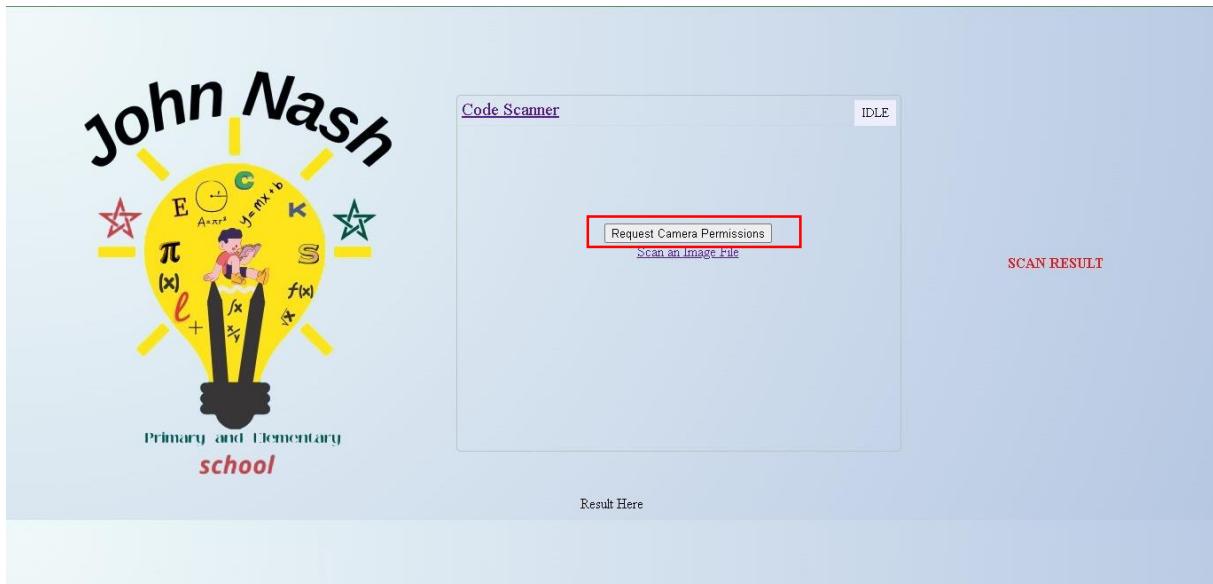


Figure 34 : Interface Agent De Security (Scan 1)

- Pour scanner le code QR dans cette interface, il suffit de cliquer sur le bouton "Demande d'autorisation de la caméra" pour activer la caméra.

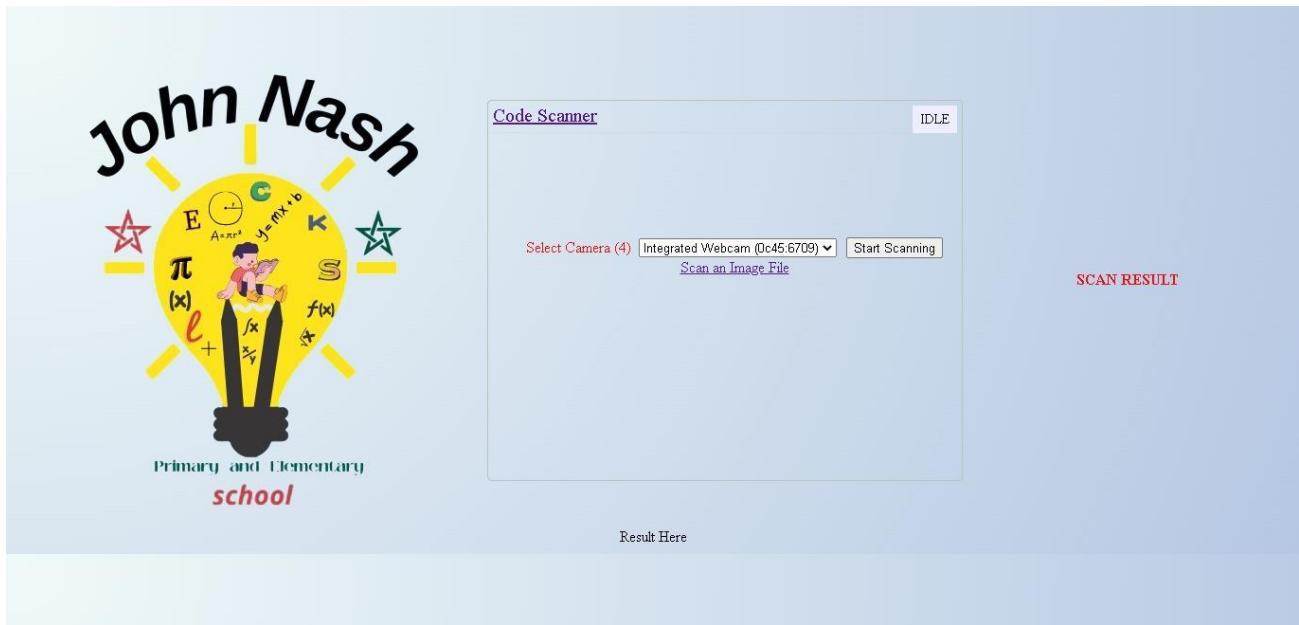
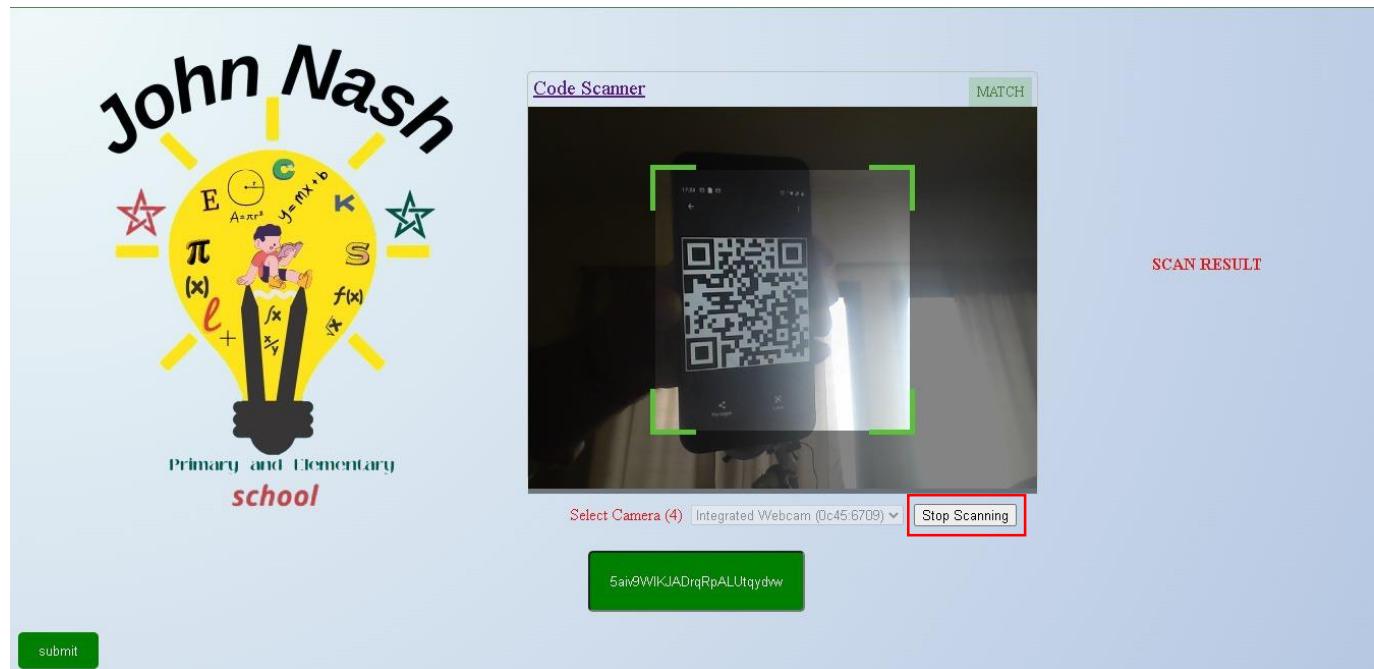


Figure 35 : Interface Agent De Security (Scan 2)

- Puis,dans cette interface, l'agent doit sélectionner la caméra et cliquer sur 'Start Scanning' pour démarrer le scan.
- Après avoir cliqué sur 'Start Scanning', l'interface affiche comme le figure suivante :



**Figure 36 : Interface Agent De Security (Scan 3)**

- Cette interface permet de scanner le code QR de l'élève et d'enregistrer sa présence dans la liste de présence. Pour cela, il suffit de cliquer sur le bouton "Submit".
- Le bouton 'Stop Scanning' arrête le scan de QR code .

History Table		
elevé	agent	classe
Moustarzika oussama	yassine	CP
Moustarzik Oussama	yassine	CP

**Figure 37 : Interface Agent De Security (Archive Scan)**

- Sur cette page, l'agent de sécurité peut également consulter tous les élèves qui ont été scannés avec leur classe.

#### V.1.4 Interface enseignant

- Dans cette interface, l'enseignant a la possibilité de consulter la liste des élèves ainsi que la liste des absences. Il peut ensuite accéder aux justifications associées à chaque élève absent. De plus, l'enseignant peut contacter l'administration en cas de problème et les parents concernant leurs enfants.

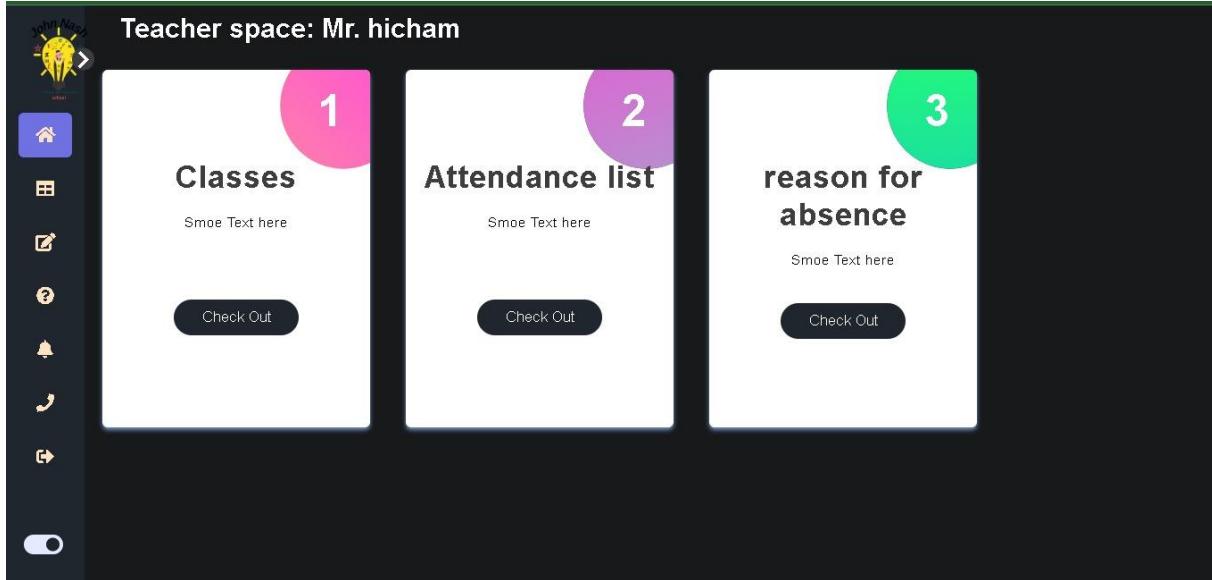


Figure 38 : Interface Teacher(Accueil)

- Cette figure représente l'accueil qui contient trois cartes :
  - "Classes" qui contient la liste des classes associées à chaque enseignant
  - "Raison d'absence" qui affiche les justifications d'absence de chaque élève absent
  - "Liste de présence" conduit à la liste de présence qui apparaît sous la forme de la figure suivante.

Élève	Classe	Date	Statut
hh hh	CP	April 8, 2023, 2:16 p.m.	Absent
Amrid Adam	CP	April 9, 2023, 2:47 a.m.	Présent
Moustarzik Oussama	CP	April 9, 2023, 8:40 p.m.	Présent

Figure 39 : Interface Teacher(Liste de présence)

- Cette interface affiche la liste de présence des élèves de chaque classe, ainsi que leur statut s'ils sont présents ou absents le jour en question.

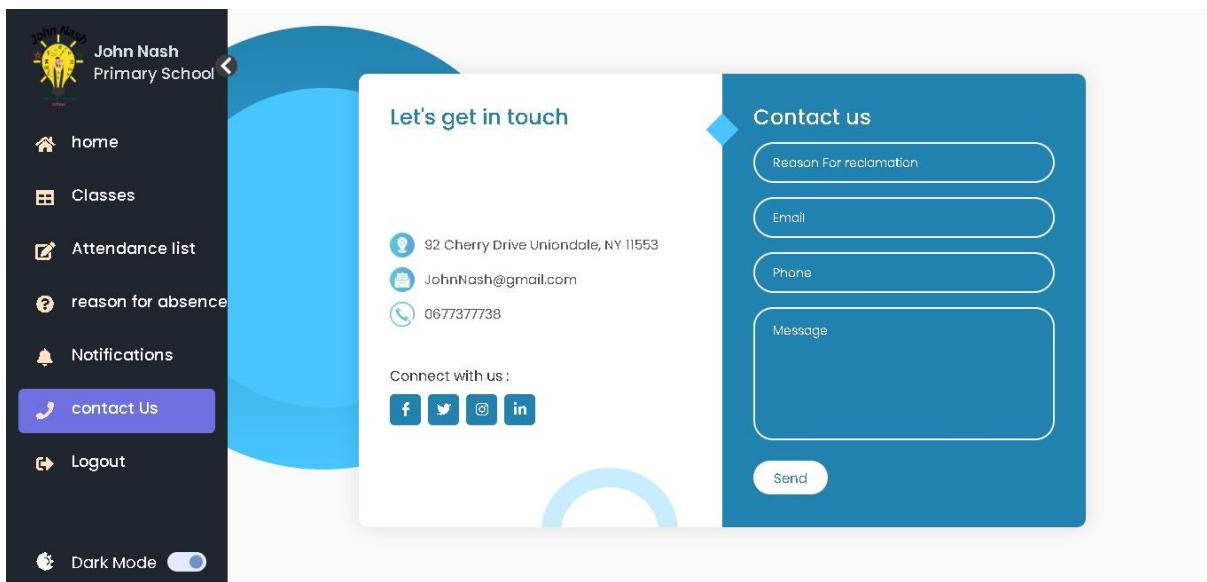


Figure 40 : Interface Teacher(Contact)

- Cette figure présente le formulaire de contact avec l'administration en cas de problème.

## V.1.5 Interface Parent

- Cette interface permet aux parents de suivre l'absence de leur enfant. Ils peuvent également générer le code QR de leur enfant, justifier son absence et contacter le professeur via le système de messagerie ou l'administration via le formulaire de contact.

Name	Date	Time	Status
Student 1	01/03/2023	1 PM	absent
Student 1	01/03/2023	8 AM	absent
Student 1	29/01/2023	1 PM	absent
Student 1	20/01/2023	8 AM	absent
Student 1	19/01/2023	1 PM	absent
Student 1	01/03/2023	8 am	absent
Student 1	01/03/2023	8 am	absent
Student 1	01/03/2023	8 am	absent

*Figure 41 : Interface Parent(Accueil)*

- L'interface d'accueil affiche aux parents l'absence de leur enfant.

Inscription	
Prenom	Nom
eleve	Nom_eleve1
CNE	R139389530
Cin Parent	266281
Date de naissance	08 / 03 / 2003
Classe	1
<input type="button" value="Generate"/> <input type="button" value="valider"/>	

*Figure 42: Interface Parent (generateur)*

- Le générateur de code QR est le même que celui de l'administrateur "figure29". Si un enfant perd son code QR, le parent utilise le générateur.



*Figure 43: Interface Parent (message)*

- Cette interface permet aux parents de contacter l'enseignant via un chat.  
*N.B : cette interface est la même dans l'interface de l'enseignant, mais les interfaces de chat nécessitent un serveur puissant pour fonctionner correctement.*

The figure shows a mobile application interface for a parent to submit an absence justification. The left sidebar menu includes: Home, QR generator, Messages, Justification service, Contact us, and Sign Out. The main screen has a form for entering personal information and a justification message.

First name	Last name
prénom	Cooper
Mail	
jhon@mail.com	
Justification	
please! justify the absence	

A large blue button at the bottom right says 'Send the form'.

*Figure 44: Interface Parent (justification formulaire)*

- Cette figure montre le formulaire rempli par le parent pour justifier l'absence de son enfant.

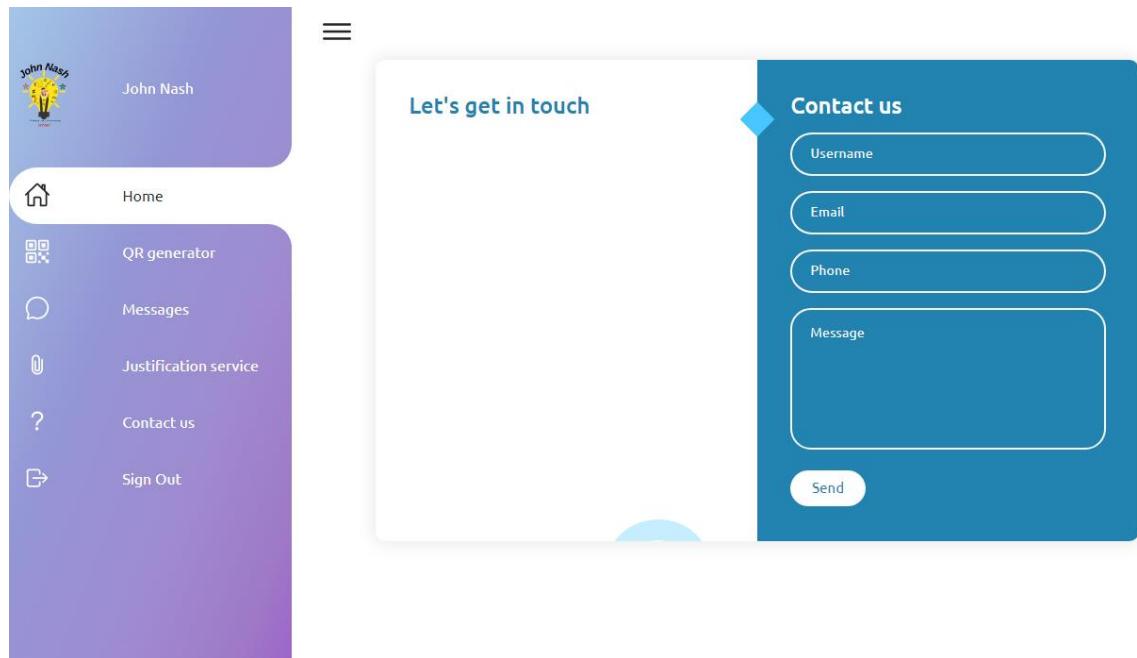


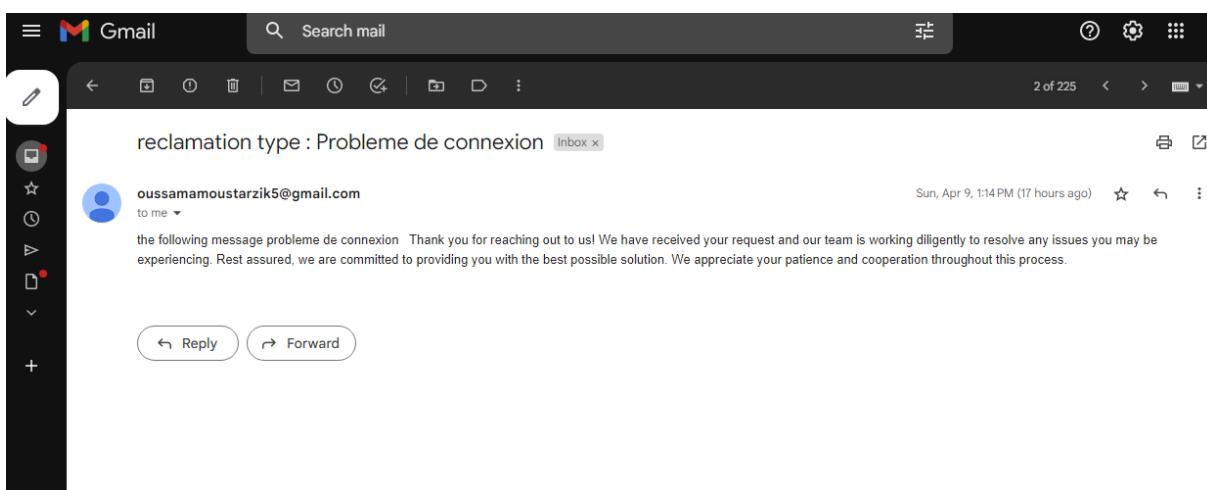
Figure 45: Interface Parent (contact formulaire)

- Cette page permet aux parent de contacter l'administration on cas de problèmes

## V.2 Contact interface

*Figure 46 : Formulaire de contact*

- Cette page permet aux utilisateurs de contacter l'administration pour des problèmes tels que des problèmes d'authentification. L'utilisateur doit remplir le formulaire et l'administrateur enverra un e-mail à l'adresse e-mail de l'utilisateur, comme indiqué dans la figure suivante :



*Figure 47 : E-mail réponse à réclamation*

## Chapitre 3

### V.3 About us interface

The screenshot shows the 'About Us' section of the John Nash school website. At the top left is the school's logo featuring a lightbulb and the name 'John Nash'. At the top right are navigation links: Home, About, Services, and Contact. Below the header, the title 'About Us' is centered. A descriptive paragraph states: 'We provide an absence management service that includes scanning the QR code of every student so they appear in the attendance list for every professor.' A 'Read More' button is located below this text. To the left of the main content is a photograph of two students, a boy and a girl, sitting at a desk and writing in notebooks. To the right is a section titled 'Our Students' with the tagline 'Our future is your future' and a 'Read More' button. Below this are three service sections: 'Online Courses' (represented by a book icon), 'Expert Teachers' (represented by a graduation cap icon), and 'Hands-on Learning' (represented by a person working on a computer icon). Each service section includes a brief description and a 'Read More' button.

*Figure 48 : About us Interface*

- Cette page affiche des informations sur la gestion des absences basée sur le QR code ainsi que d'autres fonctionnalités disponibles dans l'école John Nash.

## VI. conclusion

Ce chapitre offre un aperçu complet des éléments clés qui ont contribué à la création de notre application, y compris les plates-formes logicielles utilisées et les fonctionnalités de l'environnement de travail. De plus, nous avons introduit le concept de codes QR et présenté les différentes interfaces qui ont été développées tout au long du projet. Dans l'ensemble, nous pouvons affirmer avec confiance que nous avons atteint les objectifs du projet, mais il reste encore de la place pour la croissance et l'amélioration. Alors que nous continuons à affiner et à développer notre application, nous avons hâte d'explorer de nouvelles opportunités d'innovation et de progrès.

## Conclusion générale

La polyvalente formation dispensée par l'ESTC a pour mission de fournir à ses étudiants des compétences techniques et comportementales, tout en assurant une intégration adéquate dans le monde professionnel. Le travail présenté dans ce rapport a été réalisé dans le cadre d'un projet de fin d'études qui nous a été très bénéfique. En effet, cela nous a permis de nous familiariser avec les différents aspects de la réalisation d'un projet, ainsi que de travailler en équipe tout en respectant les différents partenaires du projet et en utilisant des outils précis dans le domaine du développement informatique. Pour concrétiser notre projet de fin d'études, nous avons commencé par nous former sur le cadre de travail. Nous avons appris le Framework Django ainsi que le langage de programmation Python, tout en déterminant ses limites pour assurer une expression adéquate des besoins. Après quatre mois de travail, en partant d'un cahier des charges établi à partir de la problématique, nous avons effectué les différentes étapes de réalisation de ce projet pour livrer une application web fonctionnelle. Ce projet a été une expérience à la fois intéressante et enrichissante, qui nous a permis d'améliorer nos connaissances et nos compétences dans le domaine du développement et de la conception de systèmes complexes. Toutefois, des perspectives d'amélioration de notre application restent indispensables. Nous envisageons donc d'ajouter de nouvelles fonctionnalités pour améliorer notre application.

## Webographie

<https://django-qr-code.readthedocs.io/en/latest/index.html>

[https://github.com/vssousa/qrcode\\_scanner](https://github.com/vssousa/qrcode_scanner)

[https://www.tutorialspoint.com/django/django\\_overview.htm](https://www.tutorialspoint.com/django/django_overview.htm)

<https://docs.djangoproject.com/en/4.1/howto/static-files/>

<https://pypi.org/project/pyzbar/>

<https://docplayer.net/56642317-Django-qr-code-documentation.html>

<https://www.w3schools.com/>

<https://goqr.me/api/doc/>

