

République Algérienne Démocratique et Populaire
Ministère de L'enseignement Supérieur et de La Recherche Scientifique
Université de Bordj Bou-Arreridj



Faculté des Mathématiques et d'Informatique
Département d'Informatique

Mémoire de fin d'études
En vue de L'Obtention du Diplôme de Master en Informatique
Réseau et multimédia

Thème :

*Conception et Réalisation d'Une Application
Mobile de Gestion des absences des étudiants*

Réalisé par :

Amrouche Younes

Taleb Abdelghani

Devant le jury composé de:

M. charikhi mourad

MCB à l'université de BBA

Président

Mme. Laifa Meriem

MCB à l'université de BBA

Examineur

M. Bendiaf Messaoud

MCB à l'université de BBA

Rapporteur

Promotion: 2019/2020

REMERCIEMENTS

Nous tenons d'abord à remercier Dieu le tout-puissant et miséricordieux de nous avoir donné la force et la patience d'accomplir ce modeste travail.

La première personne que nous tenons à remercier est notre encadrant Dr : BENDIAF MESSAOUD, pour son orientation, sa confiance et sa patience.

Nous tenons également à remercier les membres du jury d'avoir consacré une partie de leur temps à la lecture de ce mémoire et pour l'intérêt qu'ils ont porté à ce travail.

Nous tenons à exprimer nos sincères remerciements à tous les professeurs qui nous ont enseignés et qui par leurs compétences nous ont soutenus dans la poursuite de nos études.

Dédicace

Je remercie toujours Allah de m'avoir donné la capacité d'écrire et de réfléchir, la force d'y croire, la patience d'aller jusqu'au bout du rêve et le bonheur de lever mes mains vers le ciel et de dire " Ya Rab "

A celle qui m'a fait venir au monde, qui m'a donné l'amour, la tendresse et l'affection, à celle que son cœur est un univers au fond du quel se trouve le patron, qui m'a enseigné comment vivre et comment réussir, A toi maman «Tamazouzt ».

À mon père «Mohand Said» la source de tendresse, de noblesse et d'affection

À ma sœur Meriem et sa petite fille Malak et son mari Smail A Mon frère Zakaria

À Mes amis Abdelghani,Zakaria,Walid,Aimen,Brahim,Oussama

À toute ma famille Aux personnes qui partagent le bien, à tous ceux que j'aime et que je respecte je dédie ce modeste travail et je dis merci.

AMROUCHE YOUNES

Dédicace

*Je remercie toujours Allah de m'avoir donné la capacité d'écrire et de réfléchir,
la force de croire en moi, la patience d'aller jusqu'au bout de mes rêves et le
bonheur de lever mes mains vers le ciel et de dire " Ya Rab "*

*À celle qui m'a mis au monde, qui m'a donné l'amour, la tendresse et l'affection,
à celle dont le cœur est un univers au fond duquel se trouve le patron, qui m'a
enseigné comment vivre et comment réussir, A toi maman «Naima ».*

À mon père « Abdellah» la source de tendresse, de noblesse et d'affection

À Mes frères Abdessalam, Mohamed

À Mes amis Younes, Zakaria, Walid, Aimen, Brahim, Oussama

*À toute ma famille Aux personnes qui partagent le bien, à tous ceux que j'aime et
que je respecte je dédie ce modeste travail et je dis merci.*

TALEB ABDELGHANI

Table des matières

Table des matières

Liste des figures

Liste des tableaux

Introduction générale et problématique

I.1	Contexte	1
I.2	Problématique	2
I.3	Objectif et Contribution	2
I.4	Plan de mémoire	3

Chapitre I : Etat de l'art

II.1	Introduction.....	5
II.2	Les modèles existants.....	5
II.2.1	Zoho People.....	5
II.2.2	Bitrix24	6
II.2.3	When I Work.....	7
II.3	Description sommaire du modèle proposé.....	8
II.4	Objectifs du modèle proposé	8
II.5	Conclusion	8

Chapitre II : Analyse et conception

III.1	Introduction.....	10
III.2	Le langage UML	10
III.2.1	Définition	10
III.3	Conception détaillée de notre application.....	10
III.3.1	Présentation des acteurs.....	11
III.3.1.1	Admin :	11
III.3.1.2	Enseignant :	11
III.3.1.3	Etudiant :	11

III.3.2	Les cas d'utilisation.....	11
III.3.2.1	Définition	11
III.3.2.2	Diagramme de cas d'utilisation associé à l'admin.....	12
III.3.2.3	Diagramme de cas d'utilisation associé à l'enseignant	12
III.3.2.4	Diagramme de cas d'utilisation associé à l'étudiant	13
III.3.2.5	Diagramme de cas d'utilisation global	13
III.3.3	Description et représentation des scénarios.....	15
III.3.3.1	Le cas d'utilisation Authentification	15
III.3.3.2	Le cas d'utilisation autoriser la signature des présences	15
III.3.3.3	Le cas d'utilisation Signer la présence.....	16
III.3.4	Les diagrammes de Séquence	16
III.3.4.1	Définition	16
III.3.4.2	Le diagramme de séquence « authentification ».....	17
III.3.4.3	Le diagramme de séquence « ajouter étudiant »	18
III.3.4.4	Le diagramme de séquence « la signature des présences »	19
III.3.4.5	Le diagramme de séquence « autoriser la signature des présences ».....	20
III.3.4.6	Le diagramme de séquence « justification des absences des étudiants ».....	21
III.3.4.7	Le diagramme de séquence « consulter la liste des absences »	22
III.3.4.8	Le diagramme de séquence « consulter la liste des présences ».....	23
III.3.4.9	Le diagramme de séquence « consulter la liste des exclus »	23
III.3.5	Le diagramme de classe	24
III.3.5.1	Définition	24
III.3.5.2	Diagramme de classe.....	25
III.4	Conclusion	26

Chapitre III: Implémentation et bilan

IV.1	Introduction.....	28
IV.2	Environnement de développement	28
IV.2.1	Environnement Matériel	28
IV.2.2	Environnement Logiciel.....	29
IV.2.2.1	Le Langage java.....	29
IV.2.2.2	Le langage XML.....	29
IV.2.2.3	Le SDK (Software Development Kit)	29
IV.2.2.4	Android studio	30

IV.3	Implémentation de la base de données	30
IV.3.1	Server Google Firebase.....	30
IV.4	Présentation de l'application.....	30
IV.4.1	Le schéma global de l'application.....	30
IV.4.2	Présentation de quelques interfaces de notre application	31
IV.4.2.1	L'interface de chargement	31
IV.4.2.2	L'interface d'authentification	32
IV.4.2.3	Les menus des utilisateurs.....	32
IV.4.2.4	La Page « Ajouter étudiant »	36
IV.4.2.5	La Page « Justification des absences »	37
IV.4.2.6	La Page « Mes absences »	38
IV.4.2.7	La Page « Les absences »	39
IV.4.2.8	La Page « Les exclus »	40
IV.4.2.9	La Page « Présence »	41
IV.4.2.10	La Page « Signer la présence »	42
IV.4.2.11	La Page « Mon profil ».....	43
IV.5	Conclusion	44
Conclusion generale		46
Perspectives.....		47
Références		48

Liste des figures

Figure 1 : Zoho people.....	5
Figure 2 : Bitrix 24.....	6
Figure 3 : When I Work.....	7
Figure 4 : Diagramme de cas d'utilisation associé à l'admin.....	12
Figure 5 : Diagramme de cas d'utilisation associé à l'enseignant.....	13
Figure 6 : Diagramme de cas d'utilisation associé à l'étudiant.....	13
Figure 7 : Diagramme de cas d'utilisation global.....	14
Figure 8 : Diagramme de séquence authentification.....	17
Figure 9 : Diagramme de séquence ajouter étudiant.....	18
Figure 10 : Diagramme de séquence la signature des présences.....	19
Figure 11 : Diagramme de séquence autoriser la signature des présences.....	20
Figure 12 : Diagramme de séquence justification.....	21
Figure 13 : Diagramme de séquence consulter la liste des absences.....	22
Figure 14 : Diagramme de séquence consulter la liste des présences.....	23
Figure 15 : Diagramme de séquence consulter la liste des exclus.....	24
Figure 16 : Diagramme de classe.....	25
Figure 17 : Le schéma global de l'application.....	30
Figure 18 : L'interface de chargement.....	31
Figure 19 : L'interface d'authentification auprès du système.....	32
Figure 20 : Menu admin.....	33
Figure 21 : Menu enseignant.....	34
Figure 22 : Menu étudiant.....	35
Figure 23 : La page « Ajouter étudiant ».....	36
Figure 24 : La page « Justification des absences ».....	37
Figure 25 : La page « Mes absences ».....	38
Figure 26 : La page « Les absences ».....	39
Figure 27 : La page « Les exclus ».....	40
Figure 28 : La page « Présence ».....	41
Figure 29 : La page « Signer la présence ».....	42
Figure 30 : La page « Mon profil ».....	43

Liste des tableaux

Table 1 : fiche descriptive de cas d'utilisation authentication.....	15
Table 2 : fiche descriptive de cas d'utilisation autoriser la signature des présences.....	16
Table 3 : fiche descriptive de cas d'utilisation signer la présence.....	16

Introduction générale et problématique

I.1 Contexte

La gestion des absences est un calvaire. Ça ne devrait pas l'être !

La gestion des absences demande certes, de la concentration et de l'attention, mais il ne faut surtout pas perdre du temps avec ses procédures administratives. Il est important que nous puissions gagner du temps dans notre gestion des absences afin de nous concentrer sur les tâches stratégiques de notre métier. Afin de permettre aux managers et aux collaborateurs de se focaliser exclusivement sur les activités et la stratégie de développement de l'entreprise, nous devons nous munir des bons outils pour que nous puissions gérer facilement l'ensemble absences. Tel que le téléphone mobile, qui n'est pas seulement utilisé pour la communication et l'échange des messages courts, de nouveaux usages sont apparus tels que les jeux, la lecture audio, etc. Sur un plan plus pratique, le téléphone mobile dispose des variétés d'utilisations, cet appareil est devenu plus utile que jamais auparavant. L'essor du mobile a connu une évolution considérable en particulier dans les pays développés, les réseaux de télécommunications mobiles sont également en expansion : le nombre d'utilisateurs de mobile est en progression constante et la couverture territoriale est largement répandue. Grâce à la généralisation des téléphones portables tactiles à écrans larges ainsi qu'au développement des logiciels et des réseaux, les applications mobiles sont capables de satisfaire un large éventail de besoins.

Alors pourquoi ne pas développer une application mobile qui permet de faciliter le travail et améliorer la gestion d'absence.

I.2 Problématique

Le service de la gestion des absences accomplit ses tâches de gestion d'une manière presque manuelle en utilisant des logiciels de bureautique, ceci rend leurs tâches fastidieuses et difficiles, et le travail compliqué, lent et mal structuré.

Parmi ces problèmes :

- Problème de gestion, l'ajout et la suppression des absences se fait avec Excel, chaque filière est dans un fichier Excel à part ce qui mène à une perte de temps et une mauvaise organisation de travail.
- Difficulté d'accès à l'information, chaque liste d'étudiants est stockée par filière dans un fichier Excel indépendant, ce qui consiste le parcours de plusieurs fichiers pour consulter les absences des étudiants.
- Problème de gestion de l'information manque de base de données pour stocker les listes d'étudiants et leurs informations ce qui peut mener à une perte d'information.
- Problème de sécurité n'importe quelle personne peut accéder aux informations et les modifier.

I.3 Objectif et Contribution

L'objectif de notre travail est de réaliser une application mobile sous Android automatisant la gestion et le suivi des absences, afin de remédier aux problèmes présentés dans la section précédente,

L'application doit nous permettre de :

- Faciliter la consultation des absences des étudiants.
- Faciliter l'ajout des nouvelles absences.
- Diminuer les risques d'erreurs.
- Garantir la sécurité (Authentification).

Introduction générale et problématique

- Rendre le travail plus structuré et dynamique.

Pour cela, notre application « Gestion des absences des étudiants » devrait contenir trois parties :

La première destinée à l'étudiant :

Elle permet à l'étudiant de consulter la liste des absences, la liste des modules, la liste des enseignants et de signer sa présence à distance.

La deuxième partie est destinée à l'enseignant :

Elle permet à l'enseignant de consulter la liste des absences de chaque séance, la liste des exclus de chaque module et de donner l'autorisation aux étudiants pour signer leur présence.

La troisième partie est destinée à l'admin:

Elle permet à l'admin de justifier les absences des étudiants, de créer des comptes pour les étudiants et les enseignants et de gérer la base des données.

I.4 Plan de mémoire

En plus de ce chapitre d'introduction, notre travail sera divisé en trois chapitres :

Le deuxième chapitre comprend un état de l'art qui définit et explique certains des modèles de mesures de similarité et donne un petit aperçu du modèle que nous choisissons d'étudier.

Dans le troisième chapitre, nous essayerons de prévoir, d'étudier, de collecter les informations du système afin de gérer notre modèle. Pour cela on utilisera le langage UML qui permet de représenter des concepts abstraits (graphiques).

Dans le dernier chapitre, nous allons implémenter notre modèle en utilisant les langages de programmation Java, XML et Android Studio comme IDE puis nous verrons les interfaces et les fenêtres de notre application et ses résultats.

Chapitre I : Etat de l'art

II.1 Introduction

Il existe plusieurs modèles de gestion des absences, dans ce chapitre nous présenterons quelques modèles et collecterons les points importants pour créer notre modèle.

II.2 Les modèles existants

II.2.1 Zoho People

Zoho People est la solution RH complète qui comprend également une fonction de suivi des présences, il fournit un aperçu détaillé des heures et des absences des employés, vous aidant à planifier le lendemain en conséquence [1].

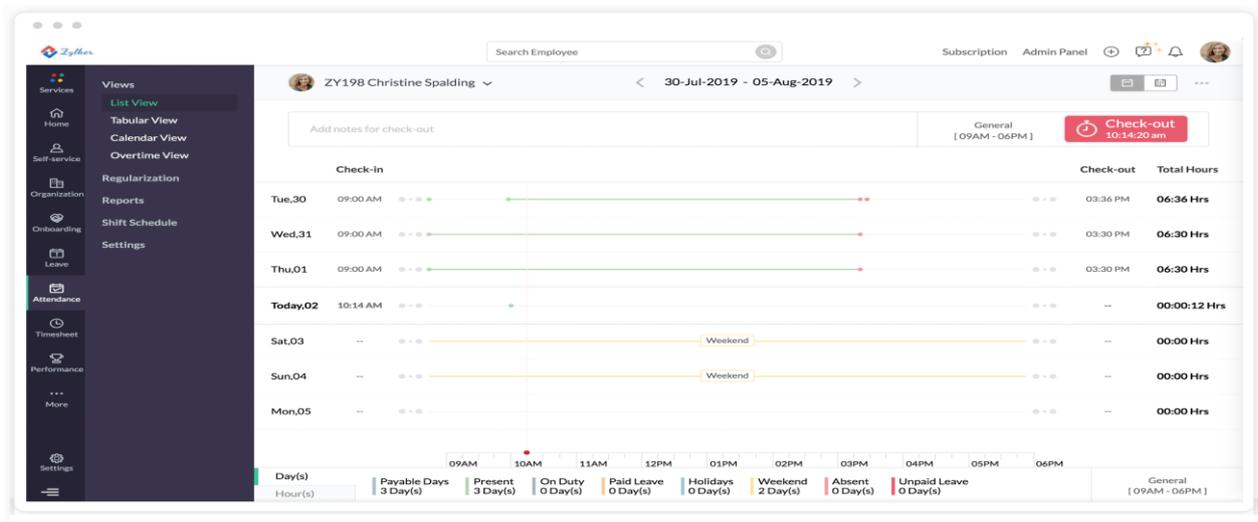


Figure 1 : Zoho people

CARACTÉRISTIQUES :

- Permettre aux employés de marquer les participants de n'importe où, avec la fonction de suivi géographique intégrée.
- Permet de gérer les absences à l'aide du planificateur d'absence.
- Permet d'atteindre les employés pour se connecter et se déconnecter à partir des appareils Web et mobiles [1].

Chapitre I : Etat de l'art

II.2.2 Bitrix24

Bitrix24 est un système RH gratuit et open source qui fournit des solutions de self-service de gestion du temps d'intranet, pour les employés [1].

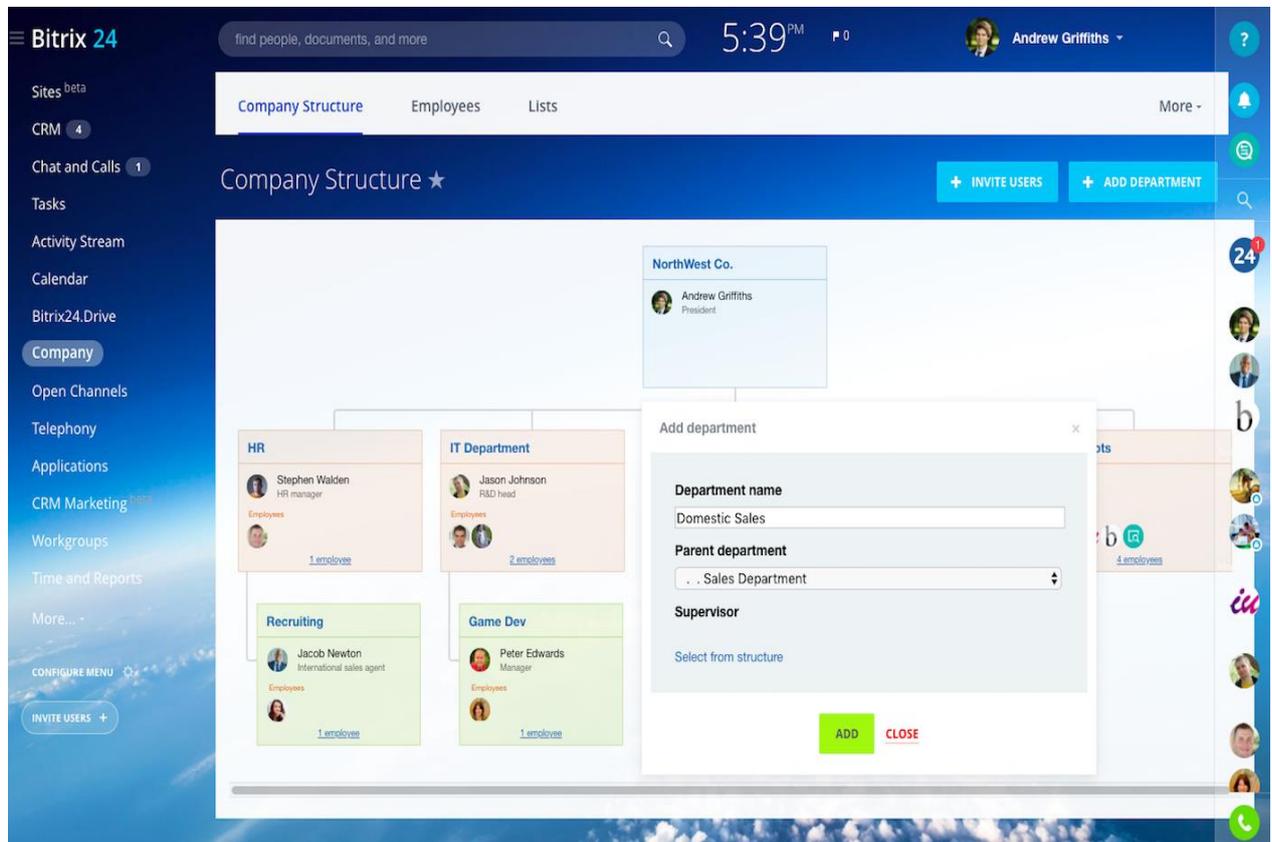


Figure 2 : Bitrix 24

CARACTÉRISTIQUES :

- Il offre la possibilité de se connecter et se déconnecter, de générer des rapports de journées de travail et des graphiques d'absence, et permet de demander des vacances.
- Il fournit des paramètres de journée de travail adaptatifs affiche la période de journée de travail minimale, permet de suivre les arrivées tardives, les départs anticipés, les rapports de productivité, etc.
- Ce programme flexible est idéal pour les petites et moyennes entreprises, car il fonctionne avec Android et iOS [1].

Chapitre I : Etat de l'art

II.2.3 When I Work

When I Work (Lorsque je travaille), un autre programme de gestion des présences, qui permet de planifier de partager rapidement avec l'équipe, de normaliser le calendrier 24h/24, de réduire les couts de main-d'œuvre, d'afficher des emplois, de suivre les candidats et plus encore [1].

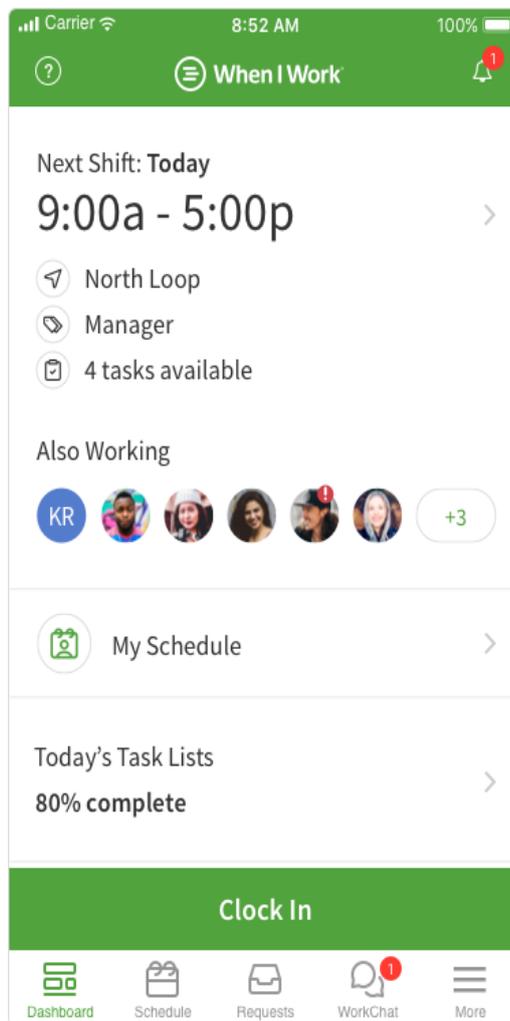


Figure 3 : When I Work

CARACTÉRISTIQUES :

- Il permet de planifier et suivre le temps de travail.
- Fournit un accès via des applications mobiles et de bureau.
- Disponible sous forme d'application mobile réactive et facile à configurer [1].

Chapitre I : Etat de l'art

II.3 Description sommaire du modèle proposé

Après avoir étudié les modèles existants, nous avons conclu que tous les modèles sont destinés uniquement aux employés et la majorité d'entre eux sont conçus pour les desktops. Nous avons donc décidé de créer une application mobile sous Android destinée aux étudiants. Notre application a une fonctionnalité de signer la présence à distance après l'autorisation de l'enseignant, cette dernière dépend de la fonction de GPS.

II.4 Objectifs du modèle proposé

Les objectifs de notre modèle sont :

- Faciliter la consultation des absences des étudiants.
- Faciliter l'ajout des nouvelles absences.
- Signer la présence à distance
- Garantir la sécurité (Authentification).
- Faciliter la génération des fiches d'absence.
- Créer les listes des exclus de façon automatique.

II.5 Conclusion

Dans ce chapitre, nous avons présenté les modèles existants de gestion des absences et le sommaire de notre modèle qui sera une application mobile sous Android pour la gestion des absences des étudiants et nous avons identifié les objectifs de notre modèle.

Chapitre II : Analyse et conception

III.1 Introduction

Après avoir vu en quoi consistait le cadre de notre application, nous introduisons dans le présent chapitre, la notion même de l'analyse et de la conception orientée objet de notre système, le but est de prévoir, d'étudier, de collecter les informations du système afin de le gérer. Pour cela nous utilisons le langage UML qui permet de représenter des concepts abstraits (graphiques), de limiter les ambiguïtés (vocabulaire précis, indépendant des langages orientés objets) et de modéliser les applications (selon une vision objets).

III.2 Le langage UML

III.2.1 Définition

Le langage UML (Unified Modeling Language, ou langage de modélisation unifié) a été pensé pour être un langage de modélisation visuelle commun, et riche sémantiquement et syntaxiquement. Il est destiné à l'architecture, la conception et la mise en œuvre de systèmes logiciels complexes par leur structure aussi bien que leur comportement. L'UML a des applications qui vont au-delà du développement logiciel, notamment pour les flux de processus dans l'industrie. Il ressemble aux plans utilisés dans d'autres domaines et se compose de différents types de diagrammes. Dans l'ensemble, les diagrammes UML décrivent la limite, la structure et le comportement du système et des objets qui s'y trouvent. L'UML n'est pas un langage de programmation, mais il existe des outils qui peuvent être utilisés pour générer du code en plusieurs langages à partir de diagrammes UML. L'UML a une relation directe avec l'analyse et la conception orientées objet [2].

III.3 Conception détaillée de notre application

Dans cette partie nous allons enfin pouvoir répondre à la question suivante :

- **Quelles sont les fonctionnalités de notre système ?**

Chapitre II : Analyse et conception

III.3.1 Présentation des acteurs

Dans notre application il y a trois acteurs, « Admin » « Enseignant » et « Etudiant » Nous pouvons afficher le travail de chaque personne.

III.3.1.1 Admin :

- Justifier les absences.
- Gérer la base des données.
- Gérer leurs profils.
- Créer des comptes des utilisateurs.

III.3.1.2 Enseignant :

- Autoriser les signatures des présences.
- Consulter la liste des présences.
- Consulter la liste des exclus.
- Gérer leurs profils.

III.3.1.3 Etudiant :

- Consulter les absences.
- Signer la présence.
- Gérer son profil.
- Consulter la liste des modules.
- Consulter la liste des enseignants.

III.3.2 Les cas d'utilisation

III.3.2.1 Définition

Un diagramme de cas d'utilisation capture le comportement d'un système, d'un sous-système, d'une classe ou d'un composant tel qu'un utilisateur extérieur le voit, il scinde la fonctionnalité du système en unités cohérentes, les cas d'utilisation, ayant un sens pour les acteurs, les cas

Chapitre II : Analyse et conception

d'utilisation permettent d'exprimer le besoin des utilisateurs d'un système, ils sont donc une vision orientée utilisateur de ce besoin au contraire d'une vision informatique [3].

III.3.2.2 Diagramme de cas d'utilisation associé à l'admin

Le diagramme suivant explique les différentes fonctions que l'admin peut faire dans notre application.

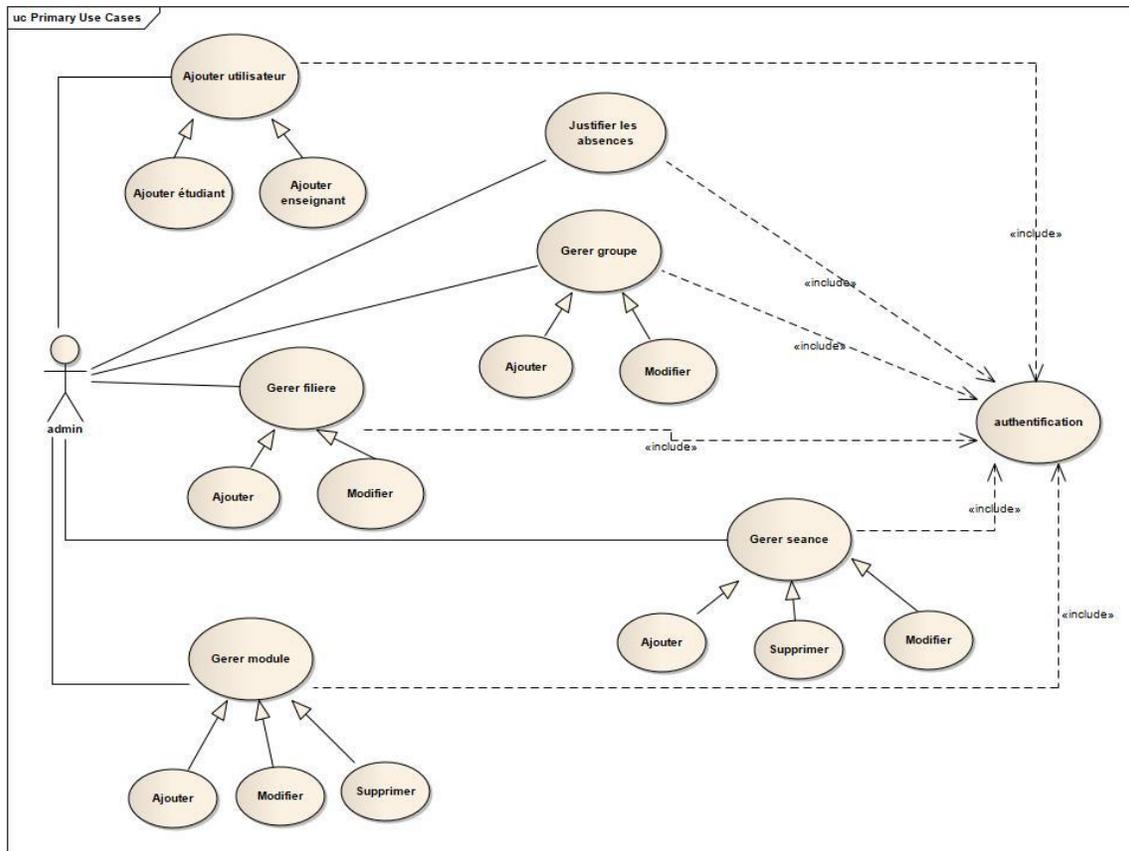


Figure 4 : Diagramme de cas d'utilisation associé à l'admin

III.3.2.3 Diagramme de cas d'utilisation associé à l'enseignant

Le diagramme suivant explique les différentes fonctions que l'enseignant peut faire dans notre application.

Chapitre II : Analyse et conception

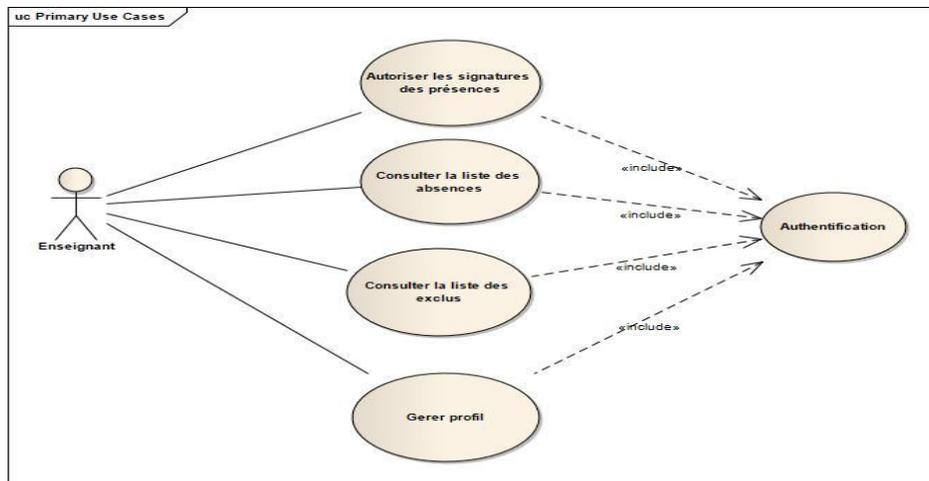


Figure 5 : Diagramme de cas d'utilisation associé à l'enseignant

III.3.2.4 Diagramme de cas d'utilisation associé à l'étudiant

Le diagramme suivant explique les différentes fonctions que l'étudiant peut faire dans notre application.

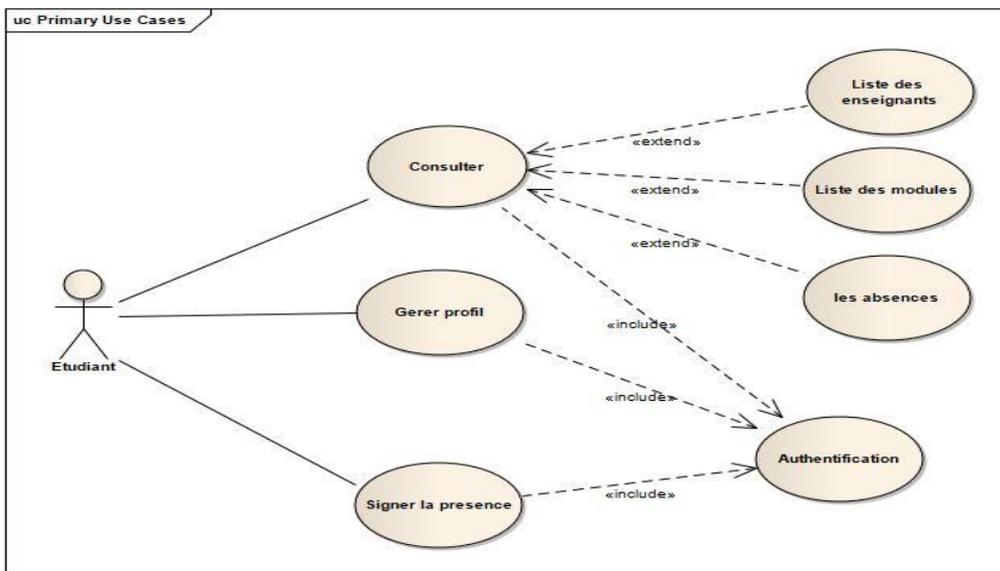


Figure 6 : Diagramme de cas d'utilisation associé à l'étudiant

III.3.2.5 Diagramme de cas d'utilisation global

Le diagramme suivant explique les différentes fonctions que tous les utilisateurs peuvent faire dans notre application.

Chapitre II : Analyse et conception

III.3.3 Description et représentation des scénarios

Dans les tableaux suivants, nous présenterons les cas d'utilisation relatifs à la signature des présences à distance. La signature des présences à distance contient trois cas d'utilisation : « l'authentification », « l'autorisation d'enseignant » et « la signature des étudiants ».

III.3.3.1 Le cas d'utilisation Authentification

Nom : Authentification
Acteur : Admin, enseignant, étudiant
Objectif : L'authentification permet d'accéder à des fonctionnalités réservées à un type d'utilisateur donné.
Scénario : <ul style="list-style-type: none">➤ L'utilisateur accède à la page d'authentification➤ Le système affiche le formulaire d'authentification.➤ L'utilisateur saisit son e-mail et son mot de passe.➤ Le système vérifie l'existence du compte et le type d'utilisateur (admin, enseignant, étudiant) et le « Device ID »➤ Le système renvoie l'interface correspondante.
Scénario alternatif : <ul style="list-style-type: none">➤ Aucun compte correspondant au couple e-mail/mot de passe indiqué :➤ le système lève une exception ; le cas d'utilisation se termine en échec.

Table 1 : fiche descriptive de cas d'utilisation authentification

III.3.3.2 Le cas d'utilisation autoriser la signature des présences

Nom : autoriser la signature des présences
Acteur : enseignant
Objectif : permet à l'enseignant d'autoriser la signature des présences

Chapitre II : Analyse et conception

Scénario : <ul style="list-style-type: none">➤ L'enseignant accède à la page d'autorisation de la signature après l'authentification➤ Le système affiche la liste des séances dans ce jour➤ L'enseignant choisit la séance et autorise la signature des présences➤ Les étudiants signent leurs présences (expliqué dans le tableau suivant)➤ L'enseignant ferme la signature des présences
Scénario alternatif : <ul style="list-style-type: none">➤ le système lève une exception ; le cas d'utilisation se termine en échec.

Table 2 : fiche descriptive de cas d'utilisation autoriser la signature des présences

III.3.3.3 Le cas d'utilisation Signer la présence

Nom : Signer la présence
Acteur : étudiant
Objectif : permet à l'étudiant de signer la présence dans une séance
Scénario : <ul style="list-style-type: none">➤ L'étudiant accède à la page de la signature après l'authentification➤ Le système affiche la liste des séances dans ce jour➤ L'étudiant choisit la séance et signe sa présence➤ Le système enregistre la présence
Scénario alternatif : <ul style="list-style-type: none">➤ Le système lève une exception ; le cas d'utilisation se termine en échec.➤ La signature de présence est fermée ou pas encore ouverte par l'enseignant.➤ L'étudiant n'est pas dans la salle vérifier par GPS

Table 3 : fiche descriptive de cas d'utilisation signer la présence

III.3.4 Les diagrammes de Séquence

III.3.4.1 Définition

Le diagramme de séquence fait partie des diagrammes comportementaux (dynamique) et plus précisément des diagrammes d'interactions. Il permet de représenter des échanges entre les

Chapitre II : Analyse et conception

différents objets et acteurs du système en Fonction du temps. A moins que le système à modéliser soit extrêmement simple, nous ne pouvons pas modéliser la dynamique globale du système dans un seul diagramme. Nous ferons donc appel à un ensemble de diagrammes de séquences chacun correspondant à une sous fonction du système, généralement d'ailleurs pour illustrer un cas d'utilisation [4].

III.3.4.2 Le diagramme de séquence « authentification »

Le premier scénario pour l'utilisateur consiste à s'authentifier auprès du système. Le diagramme suivant permet de représenter des échanges entre les différents objets et acteurs du système pour le cas d'utilisation « authentification ».

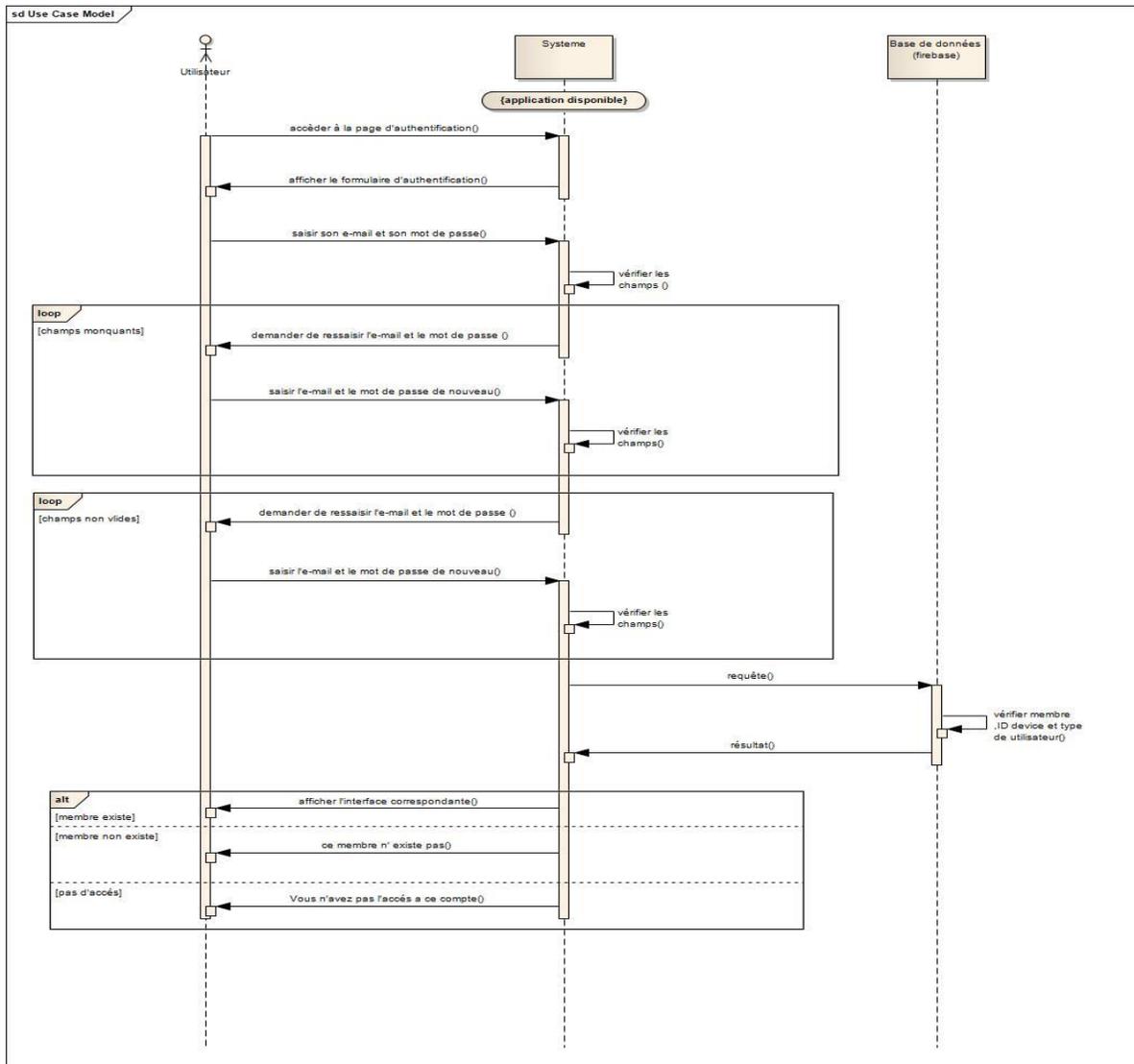


Figure 8 : Diagramme de séquence authentification

Chapitre II : Analyse et conception

III.3.4.3 Le diagramme de séquence « ajouter étudiant »

Le diagramme suivant permet de représenter des échanges entre les différents objets et acteurs du système pour le cas d'utilisation « ajouter étudiant » par l'admin après l'authentification auprès du système.

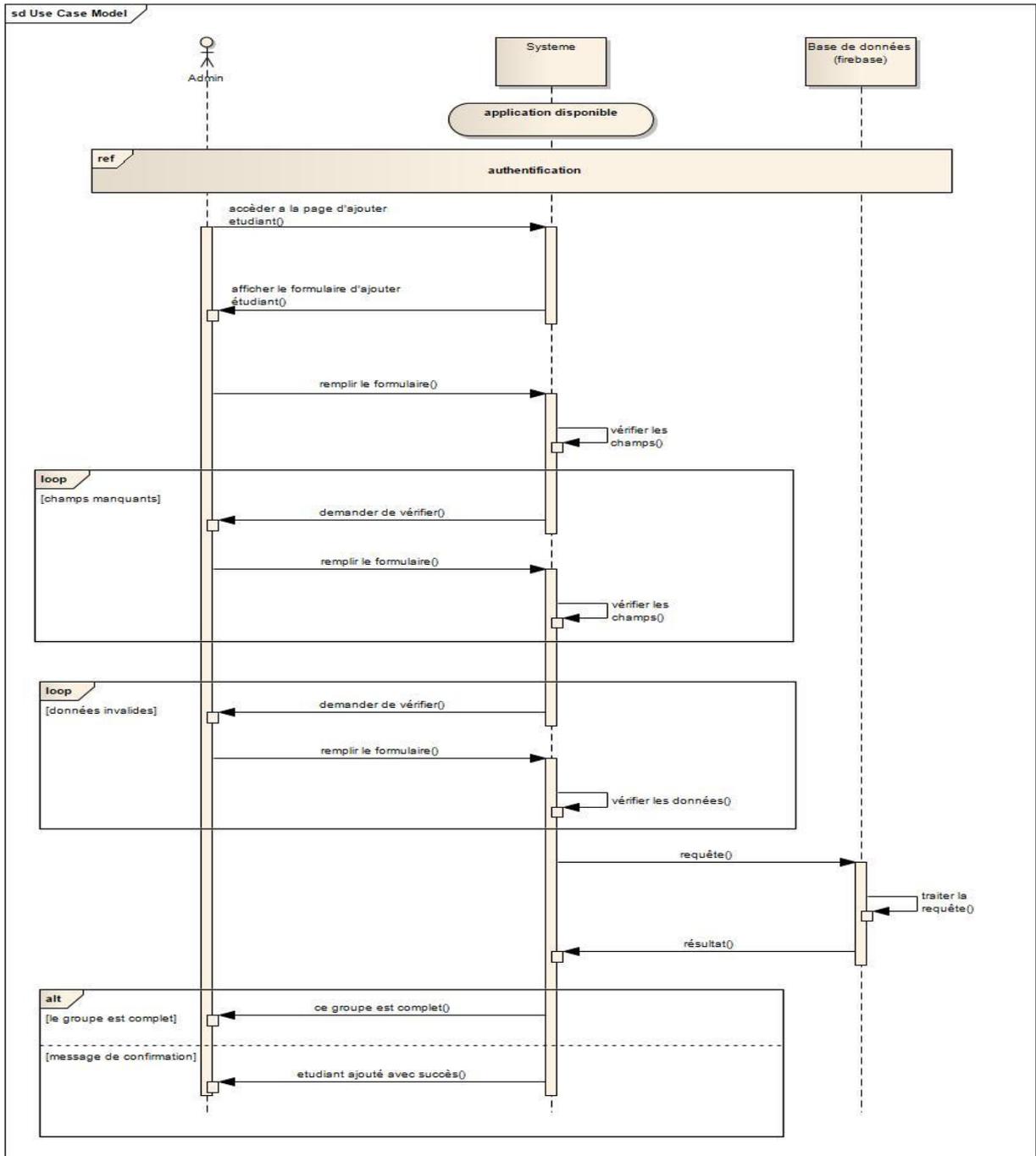


Figure 9 : Diagramme de séquence ajouter étudiant

Chapitre II : Analyse et conception

III.3.4.4 Le diagramme de séquence « la signature des présences »

Le diagramme suivant permet de représenter des échanges entre les différents objets et acteurs du système pour le cas d'utilisation « la signature des présences » par l'étudiant après l'authentification auprès du système et l'autorisation de l'enseignant.

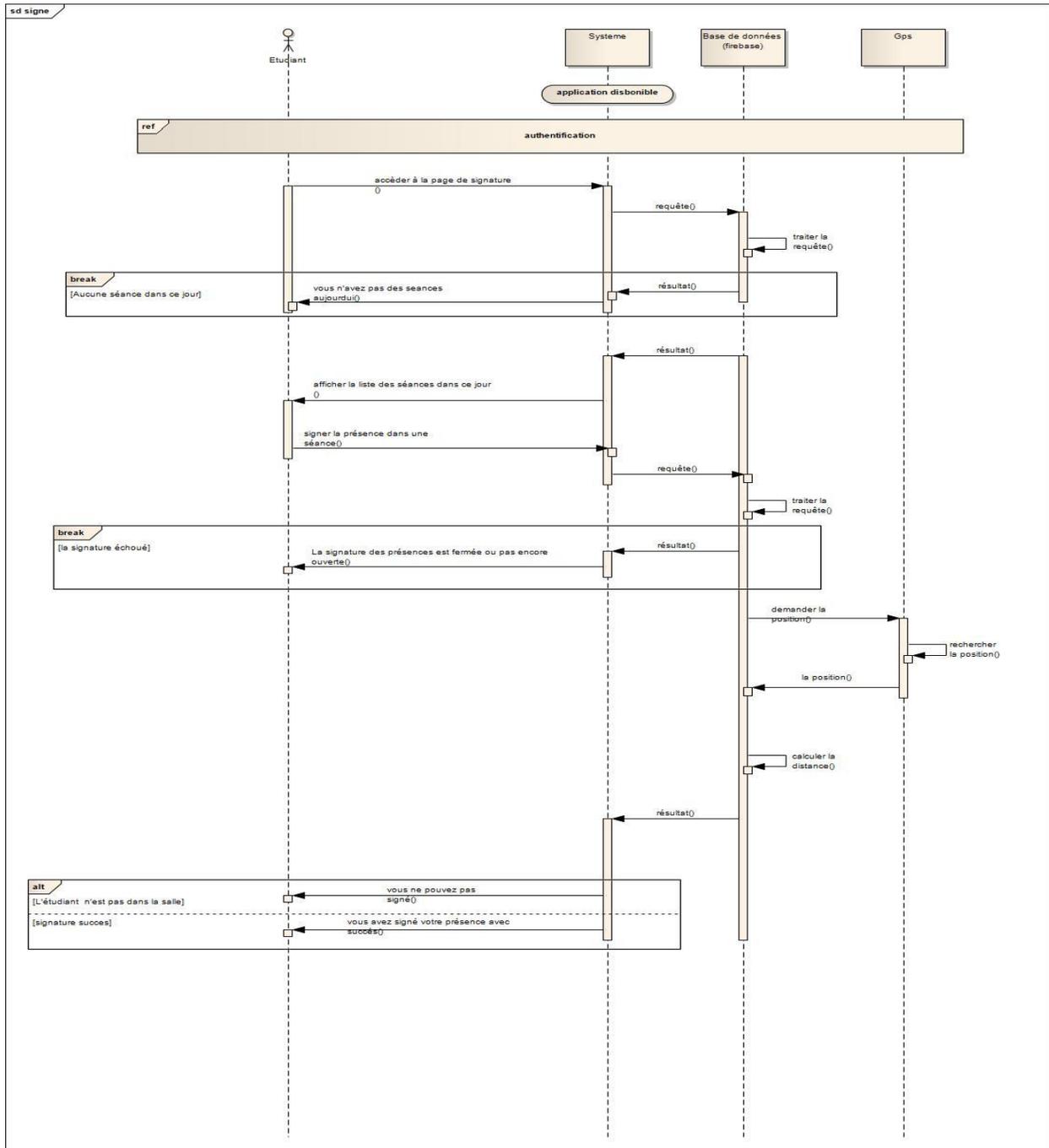


Figure 10 : Diagramme de séquence la signature des présences

Chapitre II : Analyse et conception

III.3.4.5 Le diagramme de séquence « autoriser la signature des présences »

Le diagramme suivant permet de représenter des échanges entre les différents objets et acteurs du système pour le cas d'utilisation « autoriser la signature des présences » par l'enseignant après l'authentification auprès du système.

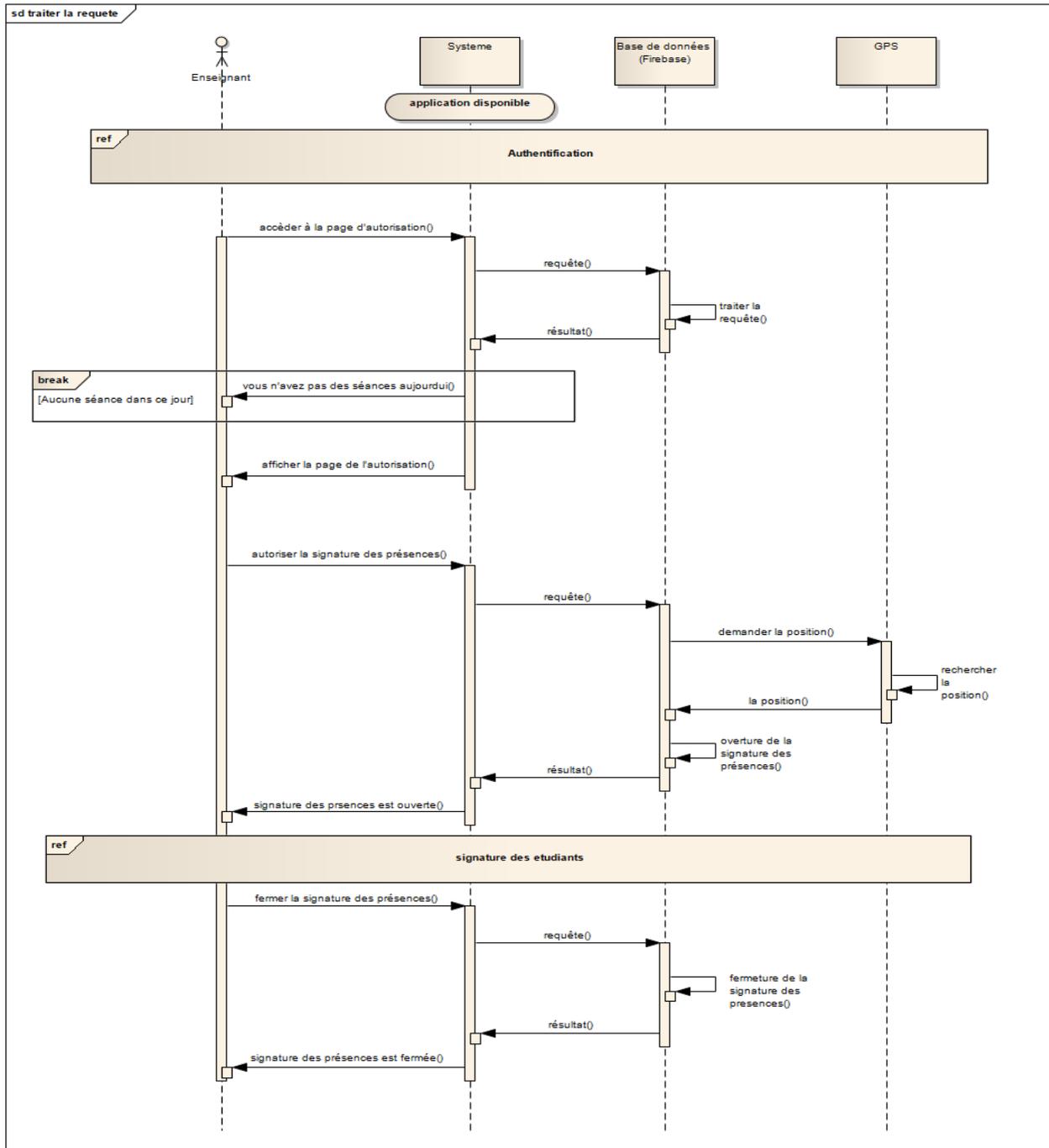


Figure 11 : Diagramme de séquence autoriser la signature des présences

Chapitre II : Analyse et conception

III.3.4.6 Le diagramme de séquence « justification des absences des étudiants »

Le diagramme suivant permet de représenter des échanges entre les différents objets et acteurs du système pour le cas d'utilisation « justification » qui permet à l'admin de justifier l'absence de l'étudiant.

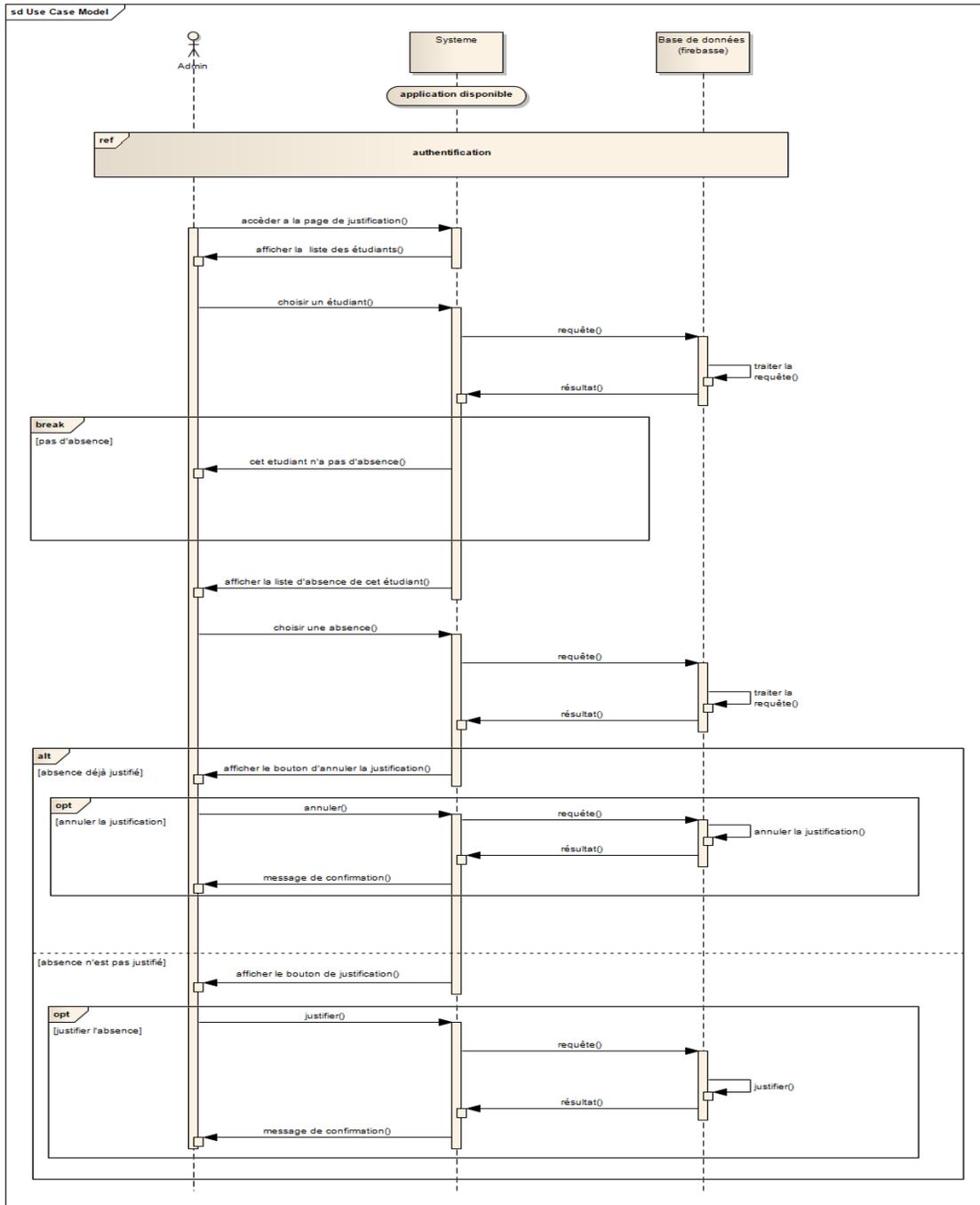


Figure 12 : Diagramme de séquence justification

Chapitre II : Analyse et conception

III.3.4.7 Le diagramme de séquence « consulter la liste des absences »

Le diagramme suivant permet de représenter des échanges entre les différents objets et acteurs du système pour le cas d'utilisation « consulter la liste des absences » par l'étudiant après l'authentification auprès du système.

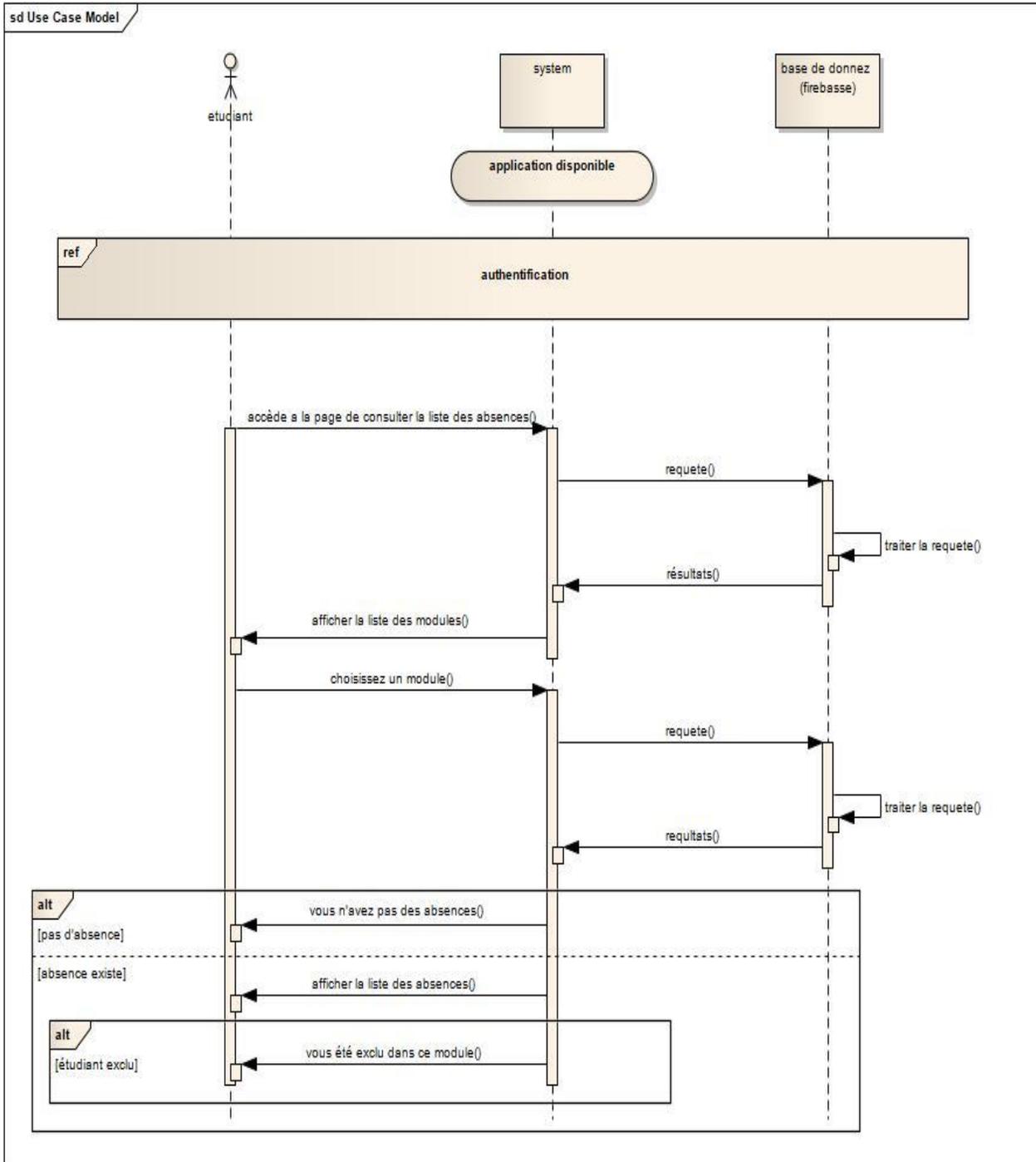


Figure 13 : Diagramme de séquence consulter la liste des absences

Chapitre II : Analyse et conception

III.3.4.8 Le diagramme de séquence « consulter la liste des présences »

Le diagramme suivant permet de représenter des échanges entre les différents objets et acteurs du système pour le cas d'utilisation « consulter la liste des présences » par l'enseignant après l'authentification auprès du système.

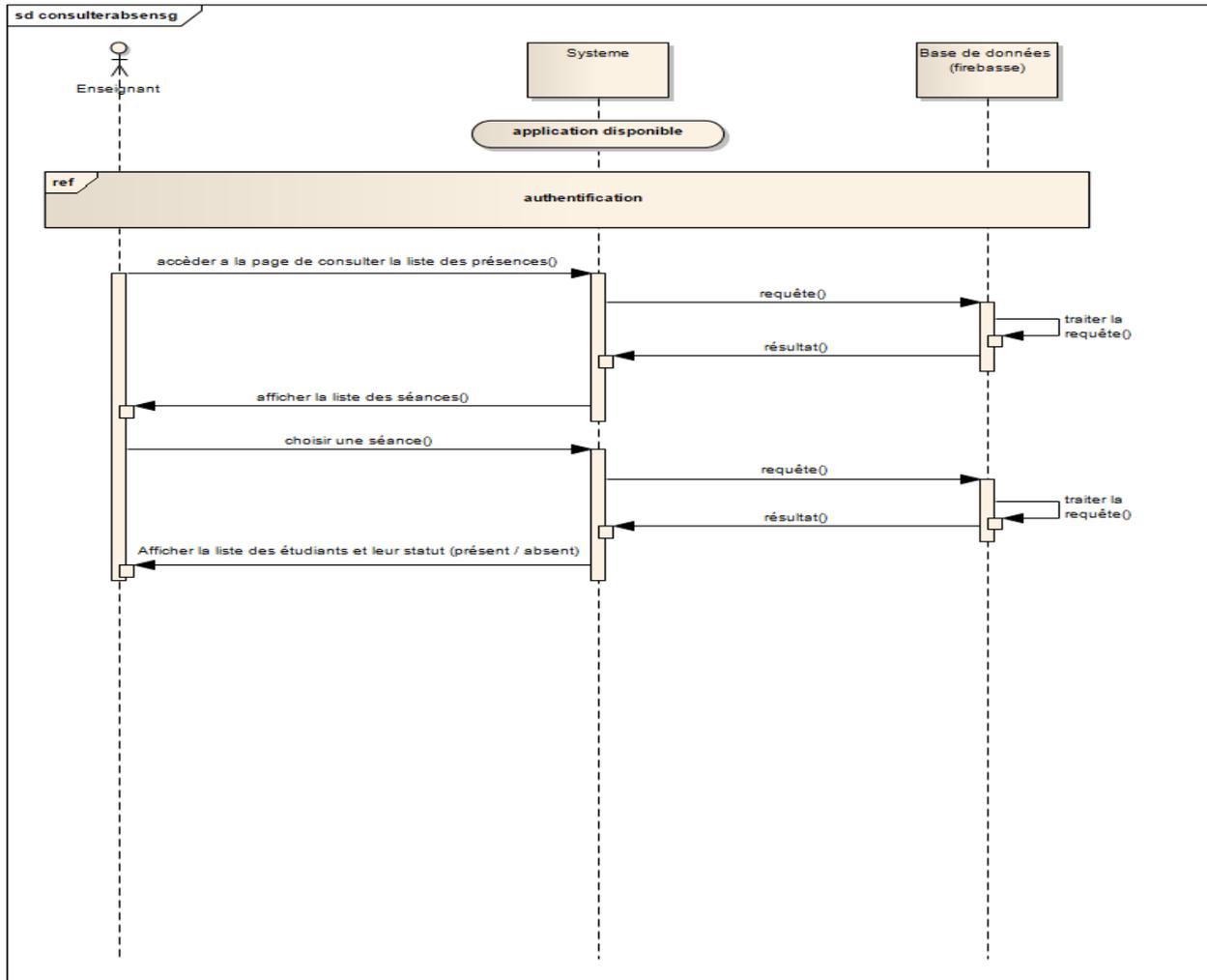


Figure 14 : Diagramme de séquence consulter la liste des présences

III.3.4.9 Le diagramme de séquence « consulter la liste des exclus »

Le diagramme suivant permet de représenter des échanges entre les différents objets et acteurs du système pour le cas d'utilisation « consulter la liste des exclus » par l'enseignant après l'authentification auprès du système.

Chapitre II : Analyse et conception

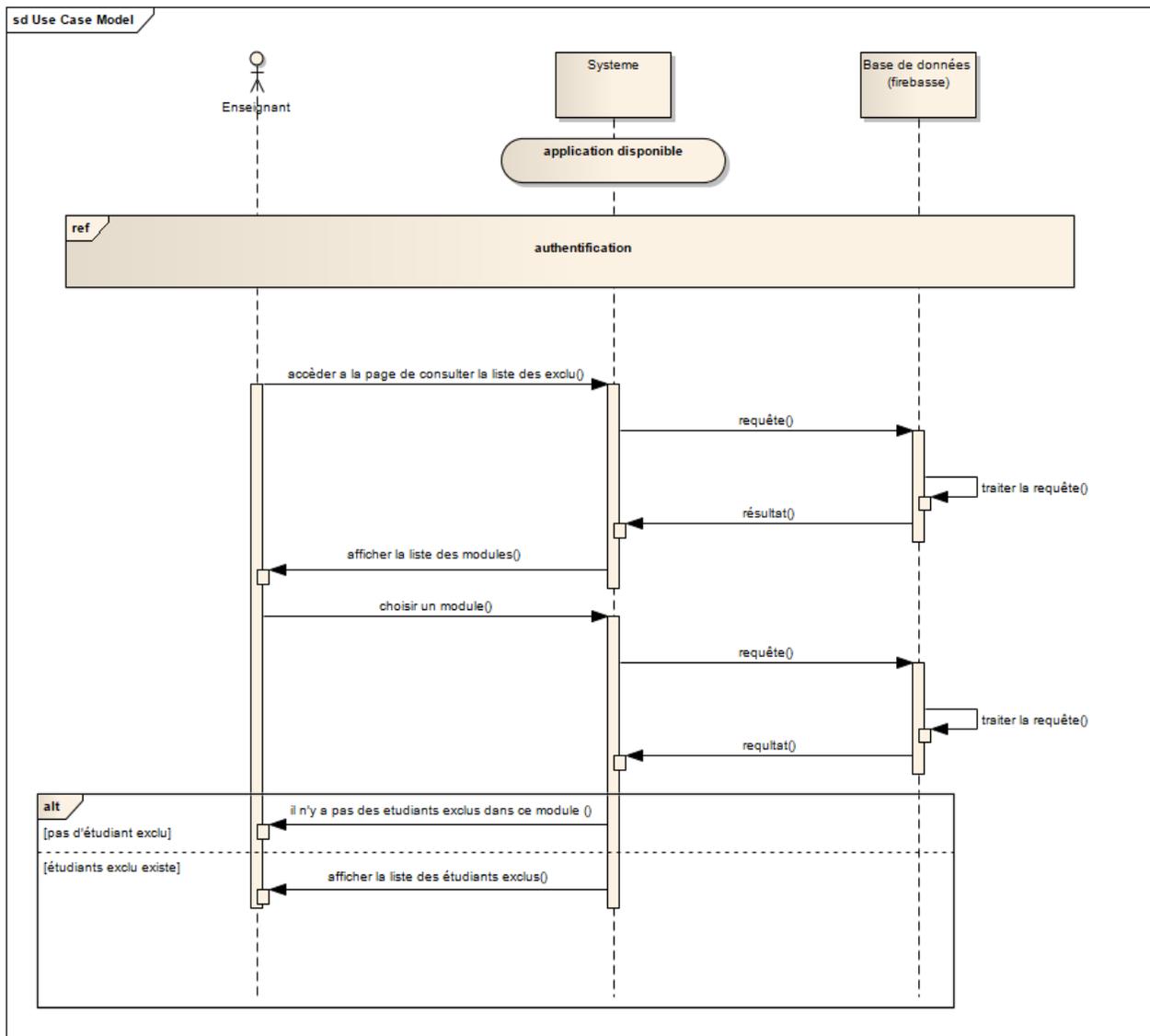


Figure 15 : Diagramme de séquence consulter la liste des exclus

III.3.5 Le diagramme de classe

III.3.5.1 Définition

Le diagramme de classe montre les blocs de construction de tout système orienté-objet. Les diagrammes de classes représentent une vue statique du modèle. Ou une partie du modèle, décrivant les attributs et les comportements qu'il a, plutôt que de détailler les méthodes pour atteindre les opérations. Les diagrammes de classes sont les plus utiles pour illustrer les relations entre les classes et les interfaces [5].

Chapitre II : Analyse et conception

III.3.5.2 Diagramme de classe

Le diagramme de classe de notre application est représenté par la figure suivante :

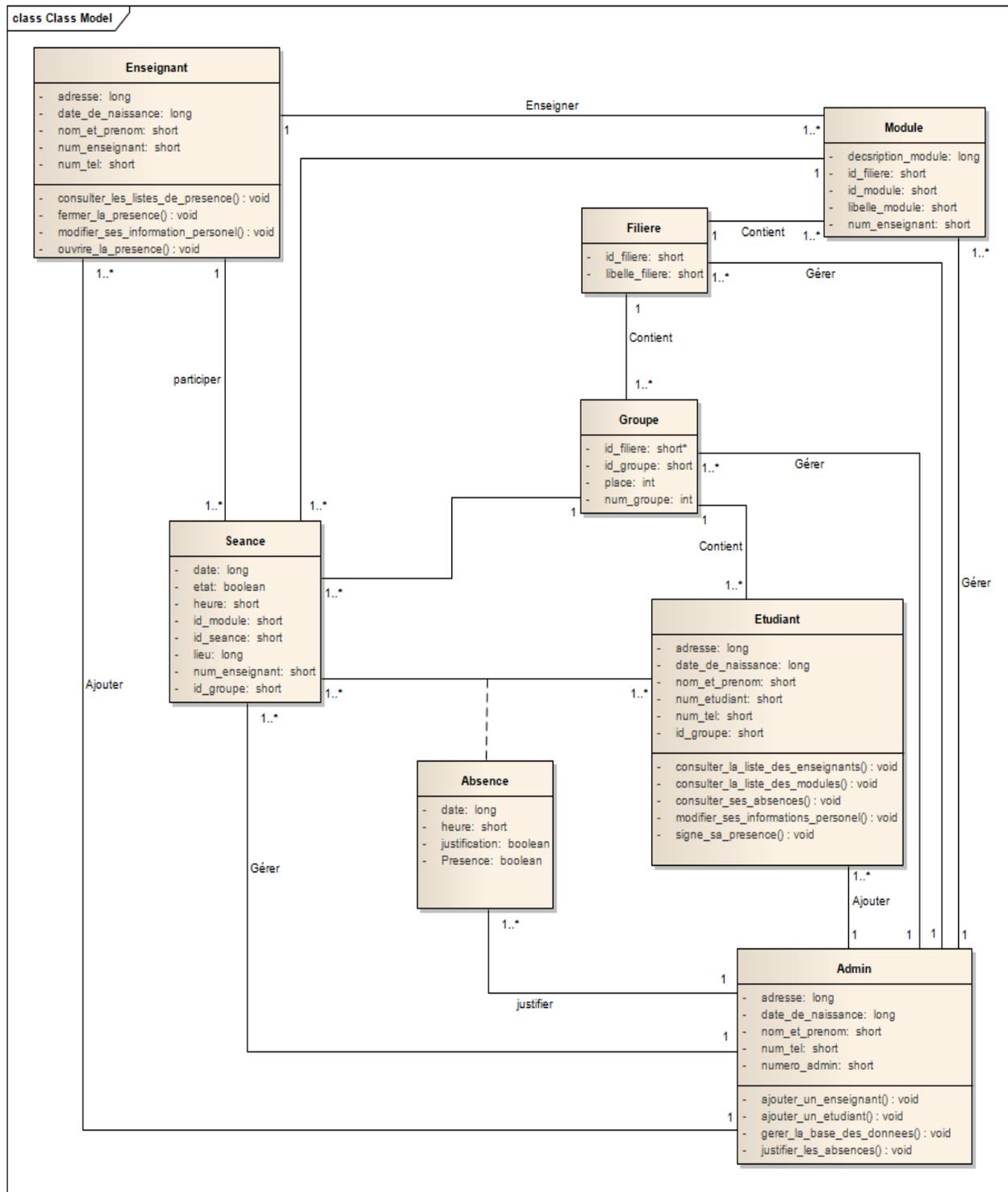


Figure 16 : Diagramme de classe

III.4 Conclusion

UML est un support de communication performant qui facilite la représentation et la compréhension de solution objet, il permet de représenter un système selon différentes vues complémentaires : les diagrammes. Dans ce chapitre, nous avons présenté les différents diagrammes utilisés pour la conception de notre projet qui contient une application mobile pour une gestion des absences des étudiants.

Chapitre III :

Implémentation et bilan

IV.1 Introduction

Nous avons vu dans les chapitres précédents en quoi consistait notre application, et ce chapitre est consacré à la partie pratique de la réalisation de notre application mobile, il comporte une description des outils de développement utilisés ainsi qu'une présentation de notre application mobile en décrivant chaque interface.

IV.2 Environnement de développement

Pour achever notre application, il faut savoir tout d'abord que les étapes de réalisation de notre travail reposent sur deux éléments : l'environnement matériel et l'environnement logiciel

IV.2.1 Environnement Matériel

Tout au long du développement de notre application mobile « application mobile pour une gestion des absences des étudiants » nous avons utilisé un téléphone de la marque (Xiaomi) et un ordinateur de la marque (HP).

Ordinateur : HP

- Processeur : Intel® Core i5 CPU @ 2.50GHz
- Ram : 12Go
- Disque Dure : 1 TB
- System exploitation: Windows 8.

Smartphone : Redmi note 8 by xiaomi

- System exploitation: Android
- Processeur snapdragon 665AIE
- RAM : 4 Go
- Stockage SSD 64 Go

IV.2.2 Environnement Logiciel

IV.2.2.1 Le Langage java

C'est un langage de programmation orienté objet, développé par Sun Microsystems. Il permet de créer des logiciels compatibles avec de nombreux systèmes d'exploitation (Windows, Linux, Macintosh, Solaris). Java donne aussi la possibilité de développer des programmes pour téléphones portables et assistants personnels. Enfin, ce langage peut être utilisé sur internet pour des petites applications intégrées à la page web (applet) ou encore comme langage serveur (jsp) [6].

IV.2.2.2 Le langage XML

Le langage XML (extended Markup Language) est un format général de documents orienté texte. Il s'est imposé comme un standard incontournable de l'informatique. Il est aussi bien utilisé pour le stockage de documents que pour la transmission de données entre applications. Sa simplicité, sa flexibilité et ses possibilités d'extension ont permis de l'adapter à de multiples domaines allant des données géographiques au dessin vectoriel en passant par les échanges commerciaux. De nombreuses technologies se sont développées autour de XML et enrichissent ainsi son environnement. Le langage XML dérive de SGML (Standard Generalized Markup Language) et de HTML (Hypertext Markup Language). Comme ces derniers, il s'agit d'un langage orienté texte et formé de balises qui permettent d'organiser les données de manière structurée [7].

IV.2.2.3 Le SDK (Software Development Kit)

C'est un kit de développement basé sur le langage Java. Il s'agit des outils que Google a fournis pour interagir avec Android pour la réalisation des applications. Le SDK propose, en plus, un émulateur Android. Ce dernier permet de lancer sur la machine du développeur un terminal virtuel représentant à l'écran un téléphone embarquant Android. C'est bien évidemment un outil indispensable pour le développement mobile. A chaque version d'Android est associée à une version de l'émulateur, permettant au développeur de tester et voir exactement à quoi ressemblera son application sur un matériel réel [8].

Chapitre III: Implémentation et bilan

IV.2.2.4 Android studio

Est un environnement de développement intégré, couramment abrégé en IDE, qui permet de développer et tester des applications, des outils et services dédiés au système d'exploitation Android. Principalement destiné aux développeurs, cette plateforme intègre une pléthore d'outils et fonctionnalités pour concevoir des applications de qualité [9].

IV.3 Implémentation de la base de données

Pour implémenter notre base de données, nous avons utilisé l'environnement de création de base de données Serveur Google Firebase.

IV.3.1 Server Google Firebase

La base de données Firebase est une base de données hébergée dans le cloud. Les données sont stockées au format JSON et synchronisées en temps réel avec chaque client connecté. Lorsque vous créez des applications multiplateformes avec nos SDK iOS, Android et JavaScript, tous vos clients partagent une instance de base de données en temps réel et reçoivent automatiquement des mises à jour avec les données les plus récentes [10].

IV.4 Présentation de l'application

Le volet technique de ce chapitre étant terminé, nous allons désormais consacrer cette partie du chapitre à la présentation des principales interfaces de notre application mobile.

IV.4.1 Le schéma global de l'application

Pour illustrer le menu qu'offre notre application, la figure suivante présente graphiquement le schéma global de l'application.

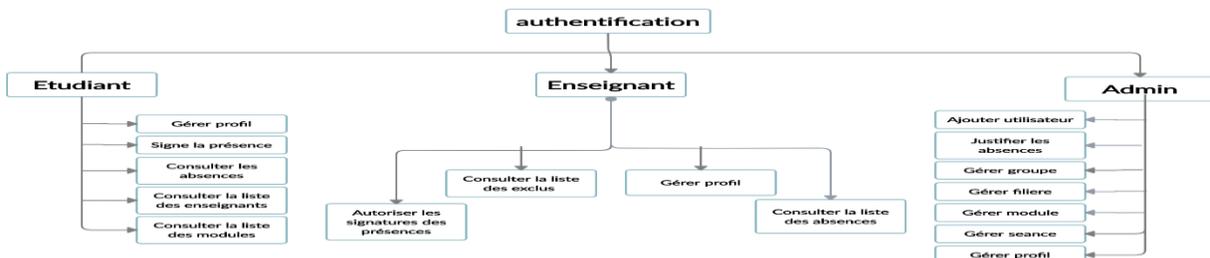


Figure 17 : Le schéma global de l'application

IV.4.2 Présentation de quelques interfaces de notre application

IV.4.2.1 L'interface de chargement

La figure suivante représente l'interface de chargement, la première interface rencontrée par l'utilisateur, après 2 secondes elle passe vers une autre activité « interface d'authentification auprès du système »



Figure 18 : L'interface de chargement

IV.4.2.2 L'interface d'authentification

Après l'interface du chargement, le système affiche l'interface d'authentification qui se présente dans la figure suivante :

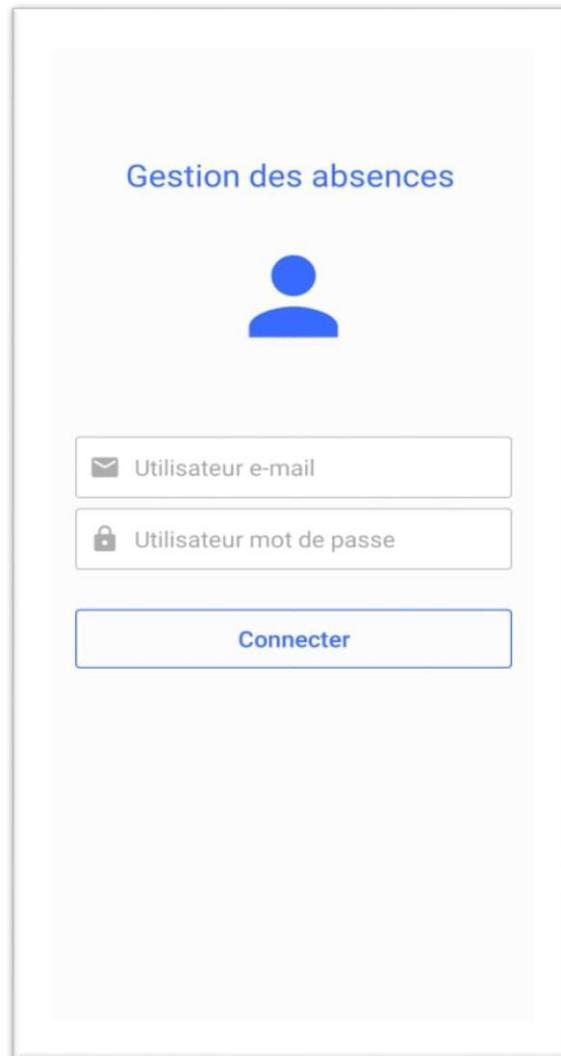


Figure 19 : L'interface d'authentification auprès du système

IV.4.2.3 Les menus des utilisateurs

Après l'authentification, le menu apparaît proposant plusieurs possibilités et services à chaque type des utilisateurs (étudiant, enseignant, admin).

Chapitre III: Implémentation et bilan

➤ Menu admin

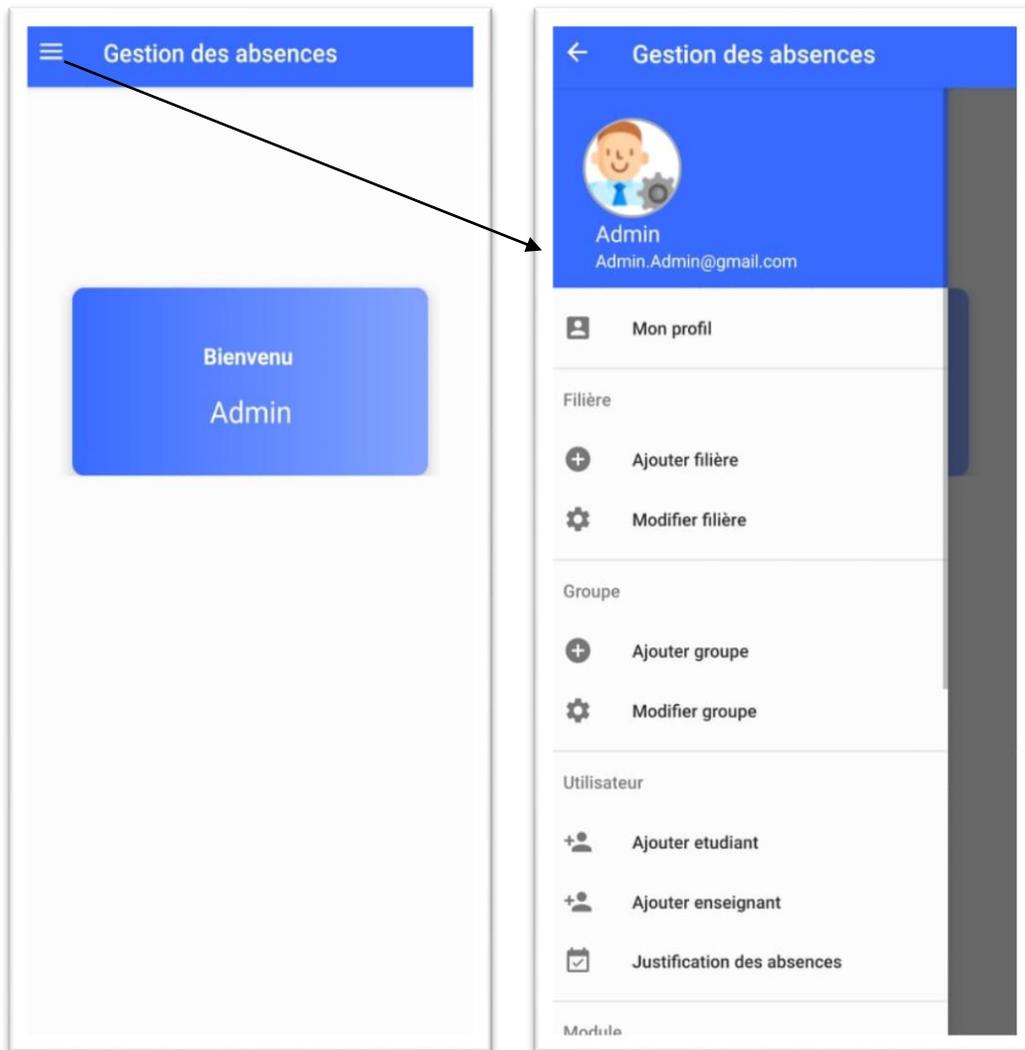


Figure 20 : Menu admin

Chapitre III: Implémentation et bilan

➤ Menu enseignant

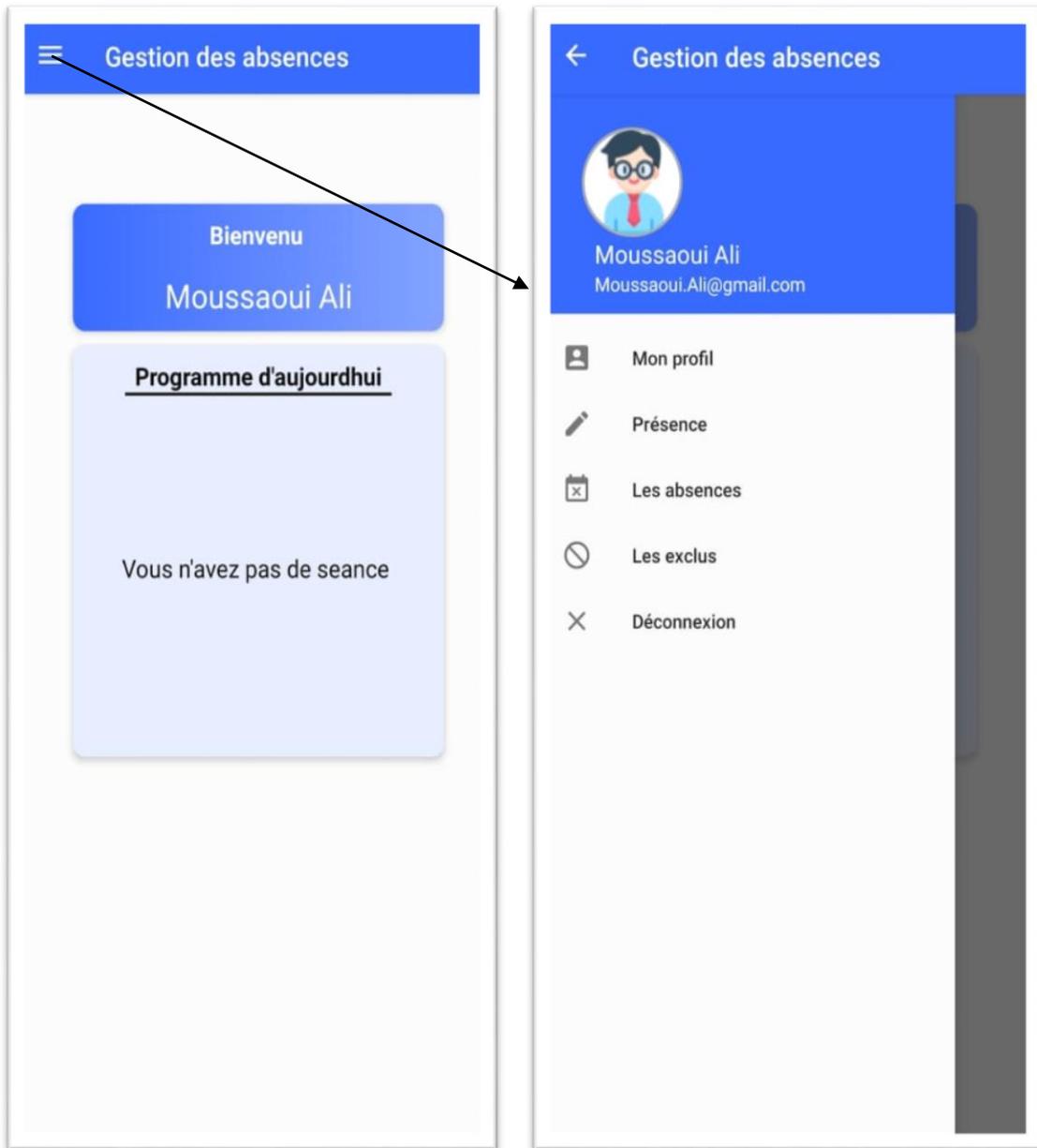


Figure 21 : Menu enseignant

Chapitre III: Implémentation et bilan

➤ Menu étudiant

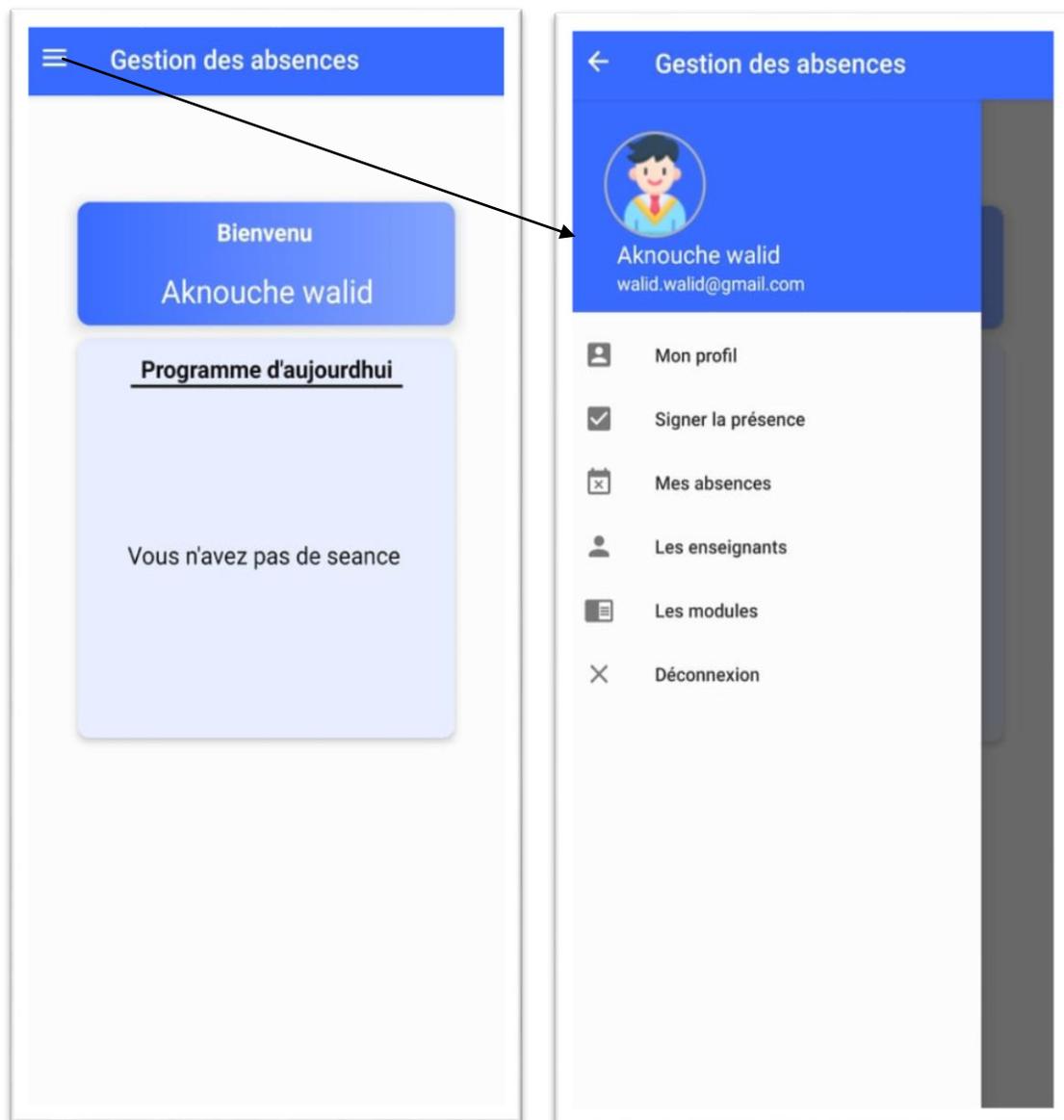


Figure 22 : Menu étudiant

À partir des menus précédents, les utilisateurs peuvent accéder aux différentes pages de notre application, chaque utilisateur a ses propres pages, et nous présenterons certaines d'entre-elles dans les figures suivantes :

IV.4.2.4 La Page « Ajouter étudiant »

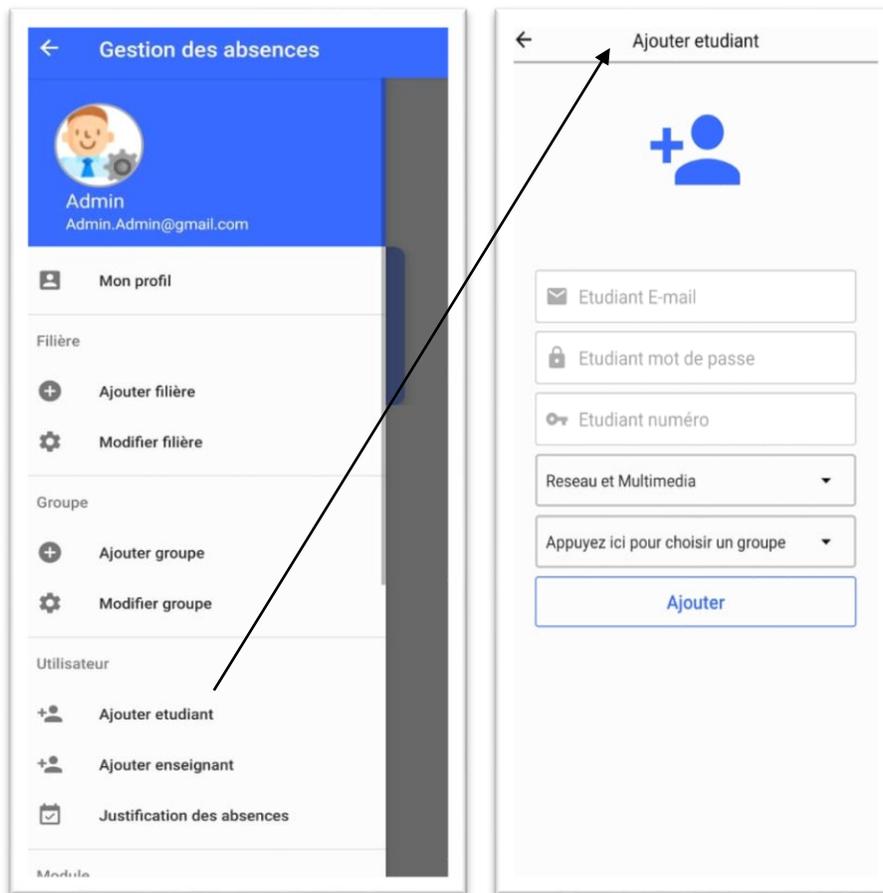


Figure 23 : La page « Ajouter étudiant »

La figure précédente montre le formulaire qui permet à l'admin d'ajouter un compte étudiant et de le placer dans une filière et un groupe.

Chapitre III: Implémentation et bilan

IV.4.2.5 La Page « Justification des absences »

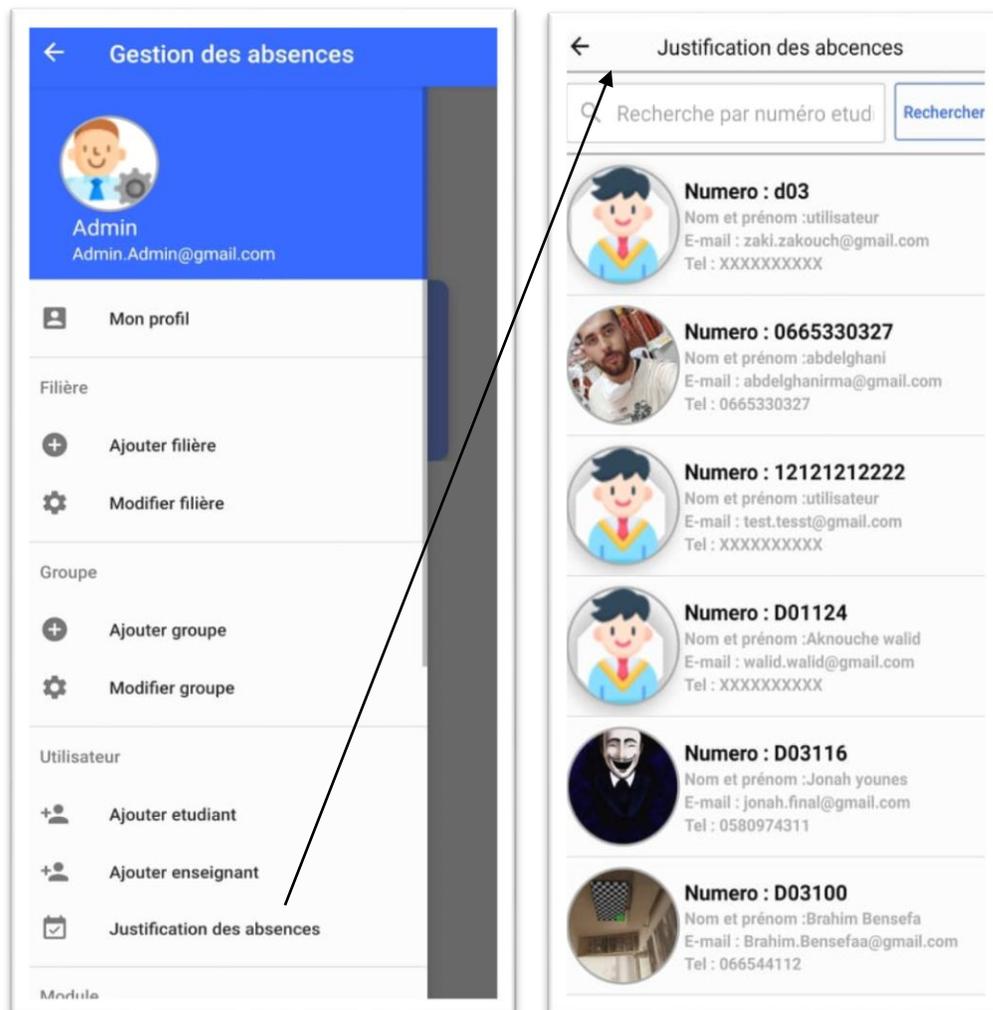


Figure 24 : La page « Justification des absences »

La figure précédente montre la page de « Justification des absences », l'admin ici peut chercher un étudiant par son numéro et voir les absences de cet étudiant et peut les justifier.

Chapitre III: Implémentation et bilan

IV.4.2.6 La Page « Mes absences »

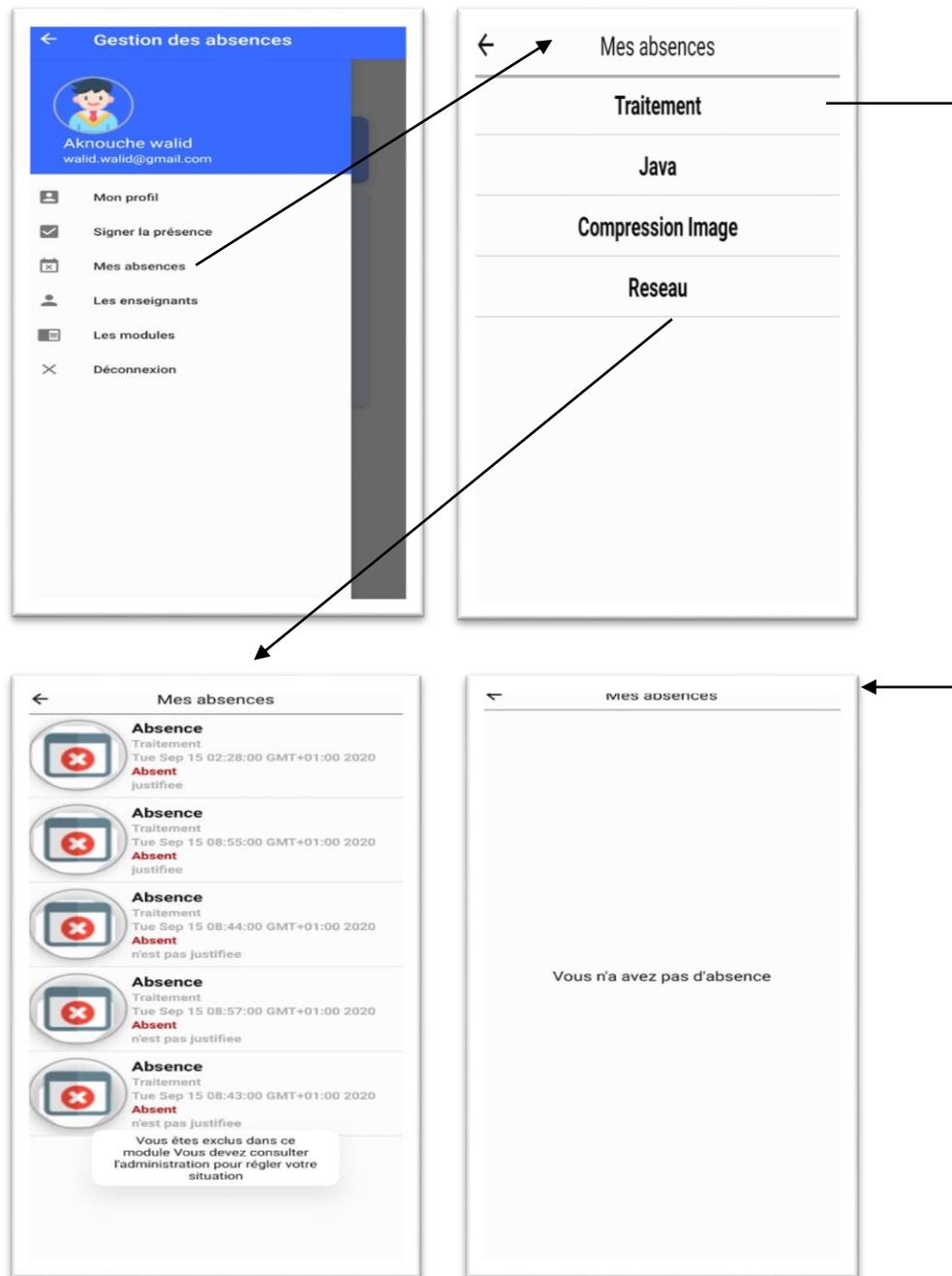


Figure 25 : La page « Mes absences »

La page de « Mes absences » permet à l'étudiant de consulter ses absences dans chaque module. En cas de 3 absences non justifiées ou 5 absences ou plus dans un module, la page affiche un message que cet étudiant est exclu.

Chapitre III: Implémentation et bilan

IV.4.2.7 La Page « Les absences »

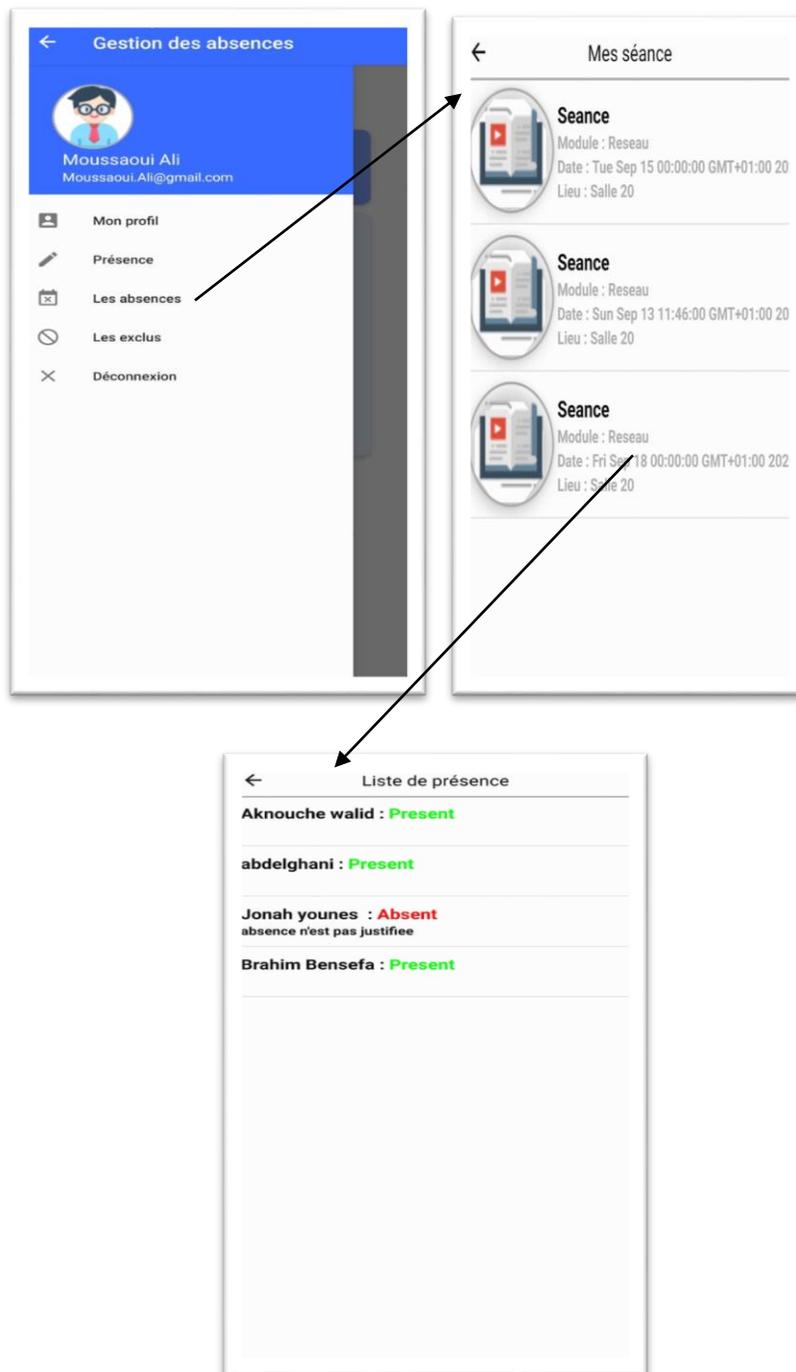


Figure 26 : La page « Les absences »

La page « Les absences » permet à l'enseignant de consulter les listes des présences des étudiants de chaque séance.

IV.4.2.8 La Page « Les exclus »

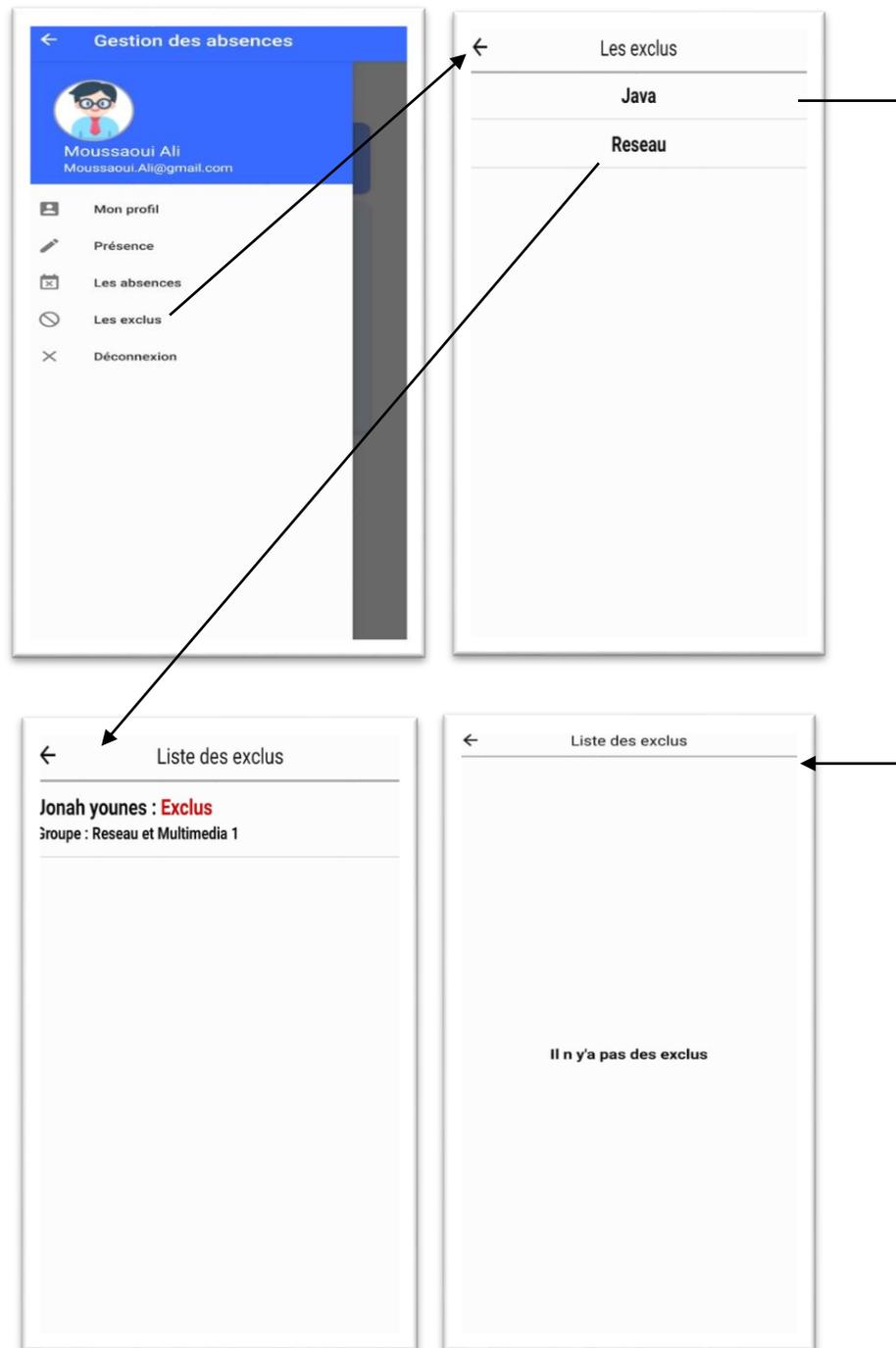


Figure 27 : La page « Les exclus »

La page « Les exclus » permet à l'enseignant de consulter les listes des étudiants exclus dans ses modules.

Chapitre III: Implémentation et bilan

IV.4.2.9 La Page « Présence »



Figure 28 : La page « Présence »

La page « Présence » permet à l'enseignant de donner l'autorisation à l'étudiant pour être capable de signer sa présence.

Chapitre III: Implémentation et bilan

IV.4.2.10 La Page « Signer la présence »

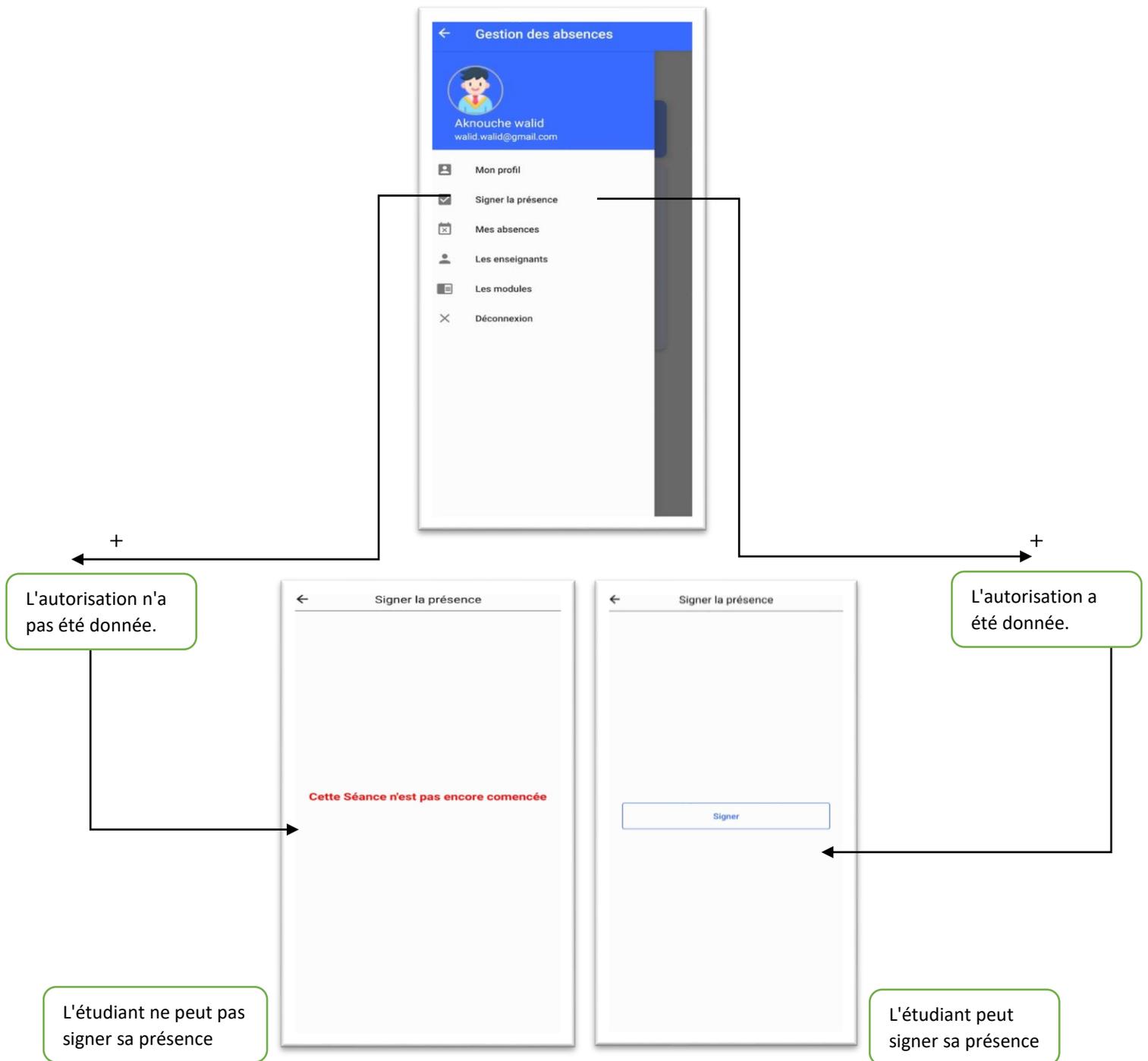


Figure 29 : La page « Signer la présence »

La page « Signer la présence » permet à l'étudiant de signer sa présence si l'autorisation d'enseignant a été donnée.

Chapitre III: Implémentation et bilan

IV.4.2.11 La Page « Mon profil »

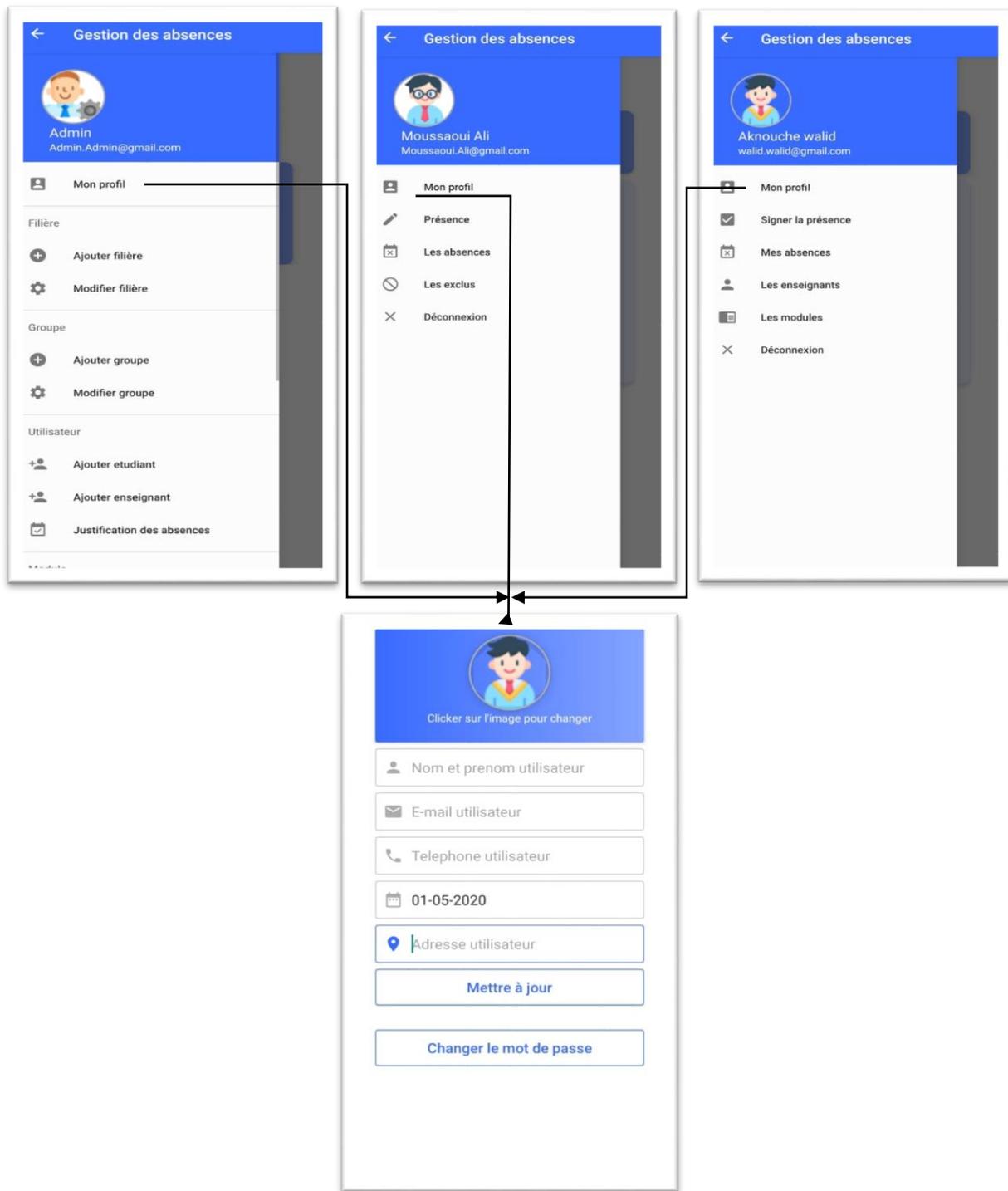


Figure 30 : La page « Mon profil »

La page « Mon profil » permet aux utilisateurs de mettre à jour leur profil et de changer leur mot de passe.

IV.5 Conclusion

La phase de réalisation est l'étape la plus importante dans le cycle de vie d'une application, dans ce chapitre, nous avons décrit brièvement le processus de réalisation de notre application en spécifiant l'environnement du développement associé à notre système, En effet, nous avons achevé l'implémentation tout en respectant la conception élaborée, en d'autres termes, nous détenons la version finale de l'application installée dans notre environnement de développement.

Conclusion générale et perspectives

Conclusion générale

Ce document de projet de fin d'études, consiste à concevoir une application mobile pour la gestion des absences, ce document a débuté par une présentation d'une problématique, ensuite nous avons introduit les objectifs et les contributions de notre modèle, puis nous avons fait un plan du mémoire. Dans la seconde partie de ce mémoire, nous avons présenté et expliqué certains des modèles de gestion des absences et collecté les points importants pour créer notre modèle. Dans la troisième partie, nous avons présenté la conception de notre application en utilisant le langage UML : nous avons identifié les acteurs qui joueront un rôle dans l'outil de gestion à mettre en œuvre, nous avons défini par la suite les fonctionnalités de ces acteurs à travers des diagrammes de cas d'utilisation et de séquence, à la fin, nous avons étendu la représentation de ces diagrammes au niveau conceptuel en construisant un diagramme de classes. Enfin, nous avons présenté l'implémentation de notre application, nous avons décrit les choix technologiques effectués concernant l'environnement de développement de notre application et l'architecture matérielle du système, suivi d'une présentation des différentes interfaces de notre application mobile pour la gestion des absences des étudiants.

Ce projet a fait l'objet d'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 des systèmes complexes.

Perspectives

Des perspectives d'amélioration de notre application restent toutefois indispensables. Nous envisageons ainsi d'ajouter des nouvelles fonctionnalités dans le futur.

L'empreinte digitale la fonction qu'on doit ajouter, comme ça l'étudiant signe sa présence avec son empreinte digitale

Références

- [1] <https://www.a7la-home.com/fr/13-best-attendance-management-software/>
(Accédé: 28/09/2020)
- [2] <https://www.lucidchart.com/pages/fr/langage-uml>
(Accédé: 22/08/2020)
- [3] <https://laurent-audibert.developpez.com/Cours-UML/?page=diagramme-cas-utilisation>
(Accédé: 23/08/2020)
- [4] <http://remy-manu.no-ip.biz/UML/Cours/coursUML5.pdf>
(Accédé: 16/09/2020)
- [5] https://www.sparxsystems.fr/resources/uml2_tutorial/uml2_classdiagram.html
(Accédé : 16/09/2020)
- [6] <https://www.futura-sciences.com/tech/definitions/internet-java-485/>
(Accédé : 18/09/2020)
- [7] <https://www.irif.fr/~carton/Enseignement/XML/Cours/Annexes/introduction.html>
(Accédé : 18/09/2020)
- [8] <https://www.ionos.fr/digitalguide/sites-internet/developpement-web/software-development-kit/> (Accédé : 18/09/2020)
- [9] <https://www.clubic.com/telecharger-fiche433537-android-studio-2.html>
(Accédé : 18/09/2020)
- [10] <https://firebase.google.com/docs/database>
(Accédé : 18/09/2020)

Résumé

Durant ce modeste travail, nous nous sommes intéressés aux développements d'application mobile sous la plateforme « Android »

Notre travail a été réalisé en deux parties : Une partie théorique, dans laquelle nous avons étudié quelques modèles et collecté les points importants pour créer notre modèle. Et une partie pratique, dans laquelle nous avons réalisé une application Android, qui a facilité et amélioré la gestion des absences des étudiants.

Mots-clés : application mobile, gestion des absences

Abstract

During this modest work, we were interested in the development of mobile applications under the "Android" platform.

Our work was carried out in two parts: A theoretical part in which we study some models and collect the important points to create our model And a practical part in which we have created an Android application, which facilitates and improves the management of student absences

Key word: mobile application, absence management

ملخص

خلال هذا العمل المتواضع، ارتأينا أن نهتم بدراسة تطبيقات الهاتف الذكي تحت منصة "أندرويد" وقد أجرينا العمل في جزئين جزء نظري ندرس فيه بعض النماذج ونجمع النقاط المهمة لإنشاء نموذجنا وجزء عملي قمنا من خلاله بعمل تطبيق أندرويد، يسهل ويحسن إدارة غيابات الطلاب

كلمات مفتاحية: تطبيقات الهاتف، تسيير الغيابات