



**Ministère de l'Enseignement Supérieur et de la
Recherche Scientifique
Direction Générale des Études Technologiques**



**Institut Supérieur des Études Technologiques de Djerba
Département Technologies de l'informatique**

Code projet
------------------------	-------

**Rapport de
STAGE DE FIN D'ETUDES**

Présenté en vue de l'obtention du diplôme

**Licence Appliquée en Technologies de
L'Informatique**

Parcours: DSI

**Développement d'une application web et mobile pour
la gestion d'un foyer universitaire**

Elaboré par :
LAFI Oumayma & CHEBLI Kais

Encadré par :
Mme.BLAGHJI Manel

Effectué à :
Entreprise : Foyer universitaire Ulysse Djerba
Encadreur : Mr ESSID Samir

Année universitaire : 2021/2022

Remerciements

La réalisation de ce mémoire a été possible grâce au concours de plusieurs personnes à qui nous voudrons témoigner toute mon gratitude.

Nous tenons à remercier M Essid Samir, notre encadrant au sein du Foyer Ulysse Djerba pour nous avoir accueilli au sein de l'entreprise et pour ce qui nous avons été dédié au cours de notre stage de fin d'études.

C'est avec grand respect et gratitude que nous transmettons nos vifs remerciements à notre encadrant Mme Blaghji Manel pour son encadrement attentif, ses remarques, ses idées directives, sa pédagogie, son soutien tout au long du notre projet, son enthousiasme et ses qualités professionnelles et ses suggestions constructives et pertinentes qui ont été d'un apport inestimable dans l'élaboration de ce travail. Qu'elle trouve ici l'expression de notre profond respect.

Nous souhaitons également exprimer nos remerciements les plus sincères à tous les enseignants qui nous avons transmis leur savoir et tous les membres du département informatique de l'ISET.

Dédicaces

Avec l'expression de ma reconnaissance, je dédie ce modeste travail à ceux qui, quels que soient les termes embrassés, je n'arriverais jamais à leur exprimer mon amour sincère.

A l'homme, mon précieux offre du dieu, qui doit ma vie, ma réussite et tout mon respecte :
mon cher père **Ali Lafi** que Dieu pardonne son âme.

A la femme, qui a souffert sans me laisser souffrir, qui n'a jamais dit non à mes exigences et qui n'a épargné aucun effort pour me rendre heureuse : mon adorable mère **Yamna Lafi**.

A mes chères sœurs, mon frère et à toute le reste de la famille qui n'ont pas cessée de me conseiller, encourager et soutenir tout au long de mes études. Que Dieu les protège et leurs offre la chance et le bonheur.

À M. Blaghji Manel

Merci pour votre bonne volonté d'accepter de m'encadrer, pour tout le temps qu'elle m'a octroyé et pour tous les conseils qu'elle m'a prodigués.

Merci d'avoir pris le temps de m'aider au cours de cette année et de m'avoir accompagné dans la maîtrise de mes connaissances.

Vous êtes le professeur qui a réussi à m'inspirer, à me donner confiance en moi et en l'avenir mais aussi qui a réussi à me donner l'envie d'apprendre.

À mes amies

A ceux qui étaient toujours à mes côtés, et qui m'ont accompagné durant mon chemin d'études supérieures, mes aimables amis, collègues d'étude, je vous remercie pour tous les bons moments que nous avons vécus ensemble.

Sans oublier mon binôme **Kais** pour son soutien moral, sa patience et sa compréhension tout au long de ce projet.

Et à tous ceux qui nous ont assistés, dans la réalisation et le bon déroulement de ce travail.

Lafi Oumaima

Dédicaces

Du profond de mon cœur, je dédie ce travail à tous ceux qui me sont chers :

À ma famille

Tous les mots du monde ne sauraient exprimer l'immense amour que je vous porte, ni la profonde gratitude que je vous témoigne pour tous les efforts et les sacrifices que vous n'avez jamais cessé de consentir pour mon instruction et mon bien-être. C'est à travers vos encouragements que j'ai opté pour cette profession, et c'est à travers vos critiques que je me suis réalisé. J'espère avoir répondu aux espoirs que vous avez fondés en moi. Je vous rends hommage par ce modeste travail en guise de ma reconnaissance éternelle et de mon infini amour. Que Dieu tout puissant vous garde et vous procure santé, bonheur et longue vie pour que vous demeuriez le flambeau illuminant le chemin de vos enfants.

À M. Blaghji Manel

Merci pour votre bonne volonté d'accepter de m'encadrer, pour tout le temps qu'elle m'a octroyé et pour tous les conseils qu'elle m'a prodigués.

Merci d'avoir pris le temps de m'aider au cours de cette année et de m'avoir accompagné dans la maîtrise de mes connaissances.

Vous êtes le professeur qui a réussi à m'inspirer, à me donner confiance en moi et en l'avenir mais aussi qui a réussi à me donner l'envie d'apprendre.

À mes amis

A ceux qui étaient toujours à mes côtés, et qui m'ont accompagné durant mon chemin d'études supérieures, mes aimables amis, collègues d'étude, je vous remercie pour tous les bons moments que nous avons vécus ensemble.

Sans oublier mon binôme **Oumayma** pour son soutien moral, sa patience et sa compréhension tout au long de ce projet.

Et à tous ceux qui nous ont assistés, dans la réalisation et le bon déroulement de ce travail.

Chebli Kais

Table des matières

Introduction Générale.....	1
Chapitre 1 : Présentation générale du projet	2
Introduction	2
I. Présentation du foyer universitaire Ulysse Djerba	2
1) Description des taches.....	3
II. Présentation du projet	4
1) Etude de l'existant.....	4
2) Solution proposée	4
III. Méthodologie et modélisation adoptée	5
1) Méthodologie de conception.....	5
2) Méthodologie de travail.....	6
IV. Framework Scrum	7
1) Équipe et Rôle	7
2) Fixation du Sprint time Box.....	7
3) Les cérémonies	8
a) Sprint planning.....	8
b) Daily meeting	8
c) Sprint Review	8
d) Sprint Rétrospective	8
Conclusion.....	8
Chapitre 2 : Spécification des besoins	9
Introduction	9
I. Spécification des besoins	9
1) Identification des acteurs.....	9
2) Exigences fonctionnelles	10
3) Exigences non fonctionnelles.....	12
II. Modélisation des besoins fonctionnels	12
1) Diagrammes de cas d'utilisation	12
a) Diagramme de cas d'utilisation de l'administrateur	13

b)	Diagramme de cas d'utilisation du responsable des clubs	14
c)	Diagramme de cas d'utilisation du technicien.....	14
d)	Diagramme de cas d'utilisation du psychologue	15
e)	Diagramme de cas d'utilisation de l'étudiant.....	15
1)	Description textuelle	16
a)	Description textuelle du diagramme de cas d'utilisation « S'authentifier »... 	16
b)	Description textuelle du diagramme de cas d'utilisation « Créer compte » ...	17
c)	Description textuelle du diagramme de cas d'utilisation « Gérer profil ».....	17
d)	Description textuelle du diagramme de cas d'utilisation « Passer réservation »	18
e)	Description textuelle du diagramme de cas d'utilisation « Gérer Réservations »	18
III.	Le product backlog.....	19
Conclusion.....		21
Chapitre 3 : Etude Conceptuelle.....		22
Introduction		22
I. Architecture du système		22
1)	L'architecture 3 tiers	22
2)	Le design pattern d'architecture MVC.....	23
II. Conception du sprint 1.....		24
1)	Sprint Goal	24
2)	Sprint Backlog.....	24
3)	Diagramme de séquence « S'authentifier »	25
III. Conception du sprint 2.....		26
1)	Sprint Goal	26
2)	Sprint backlog	26
3)	Diagramme de séquence « Valider compte étudiant »	28
4)	Diagramme de séquence « créer compte personnel »	29
IV. Conception du sprint 3.....		29
1)	Sprint Goal	29
3)	Diagramme de séquence « Gestion de profil »	31
4)	Diagramme de séquence « Traiter les réclamations »	32

5) Diagramme de séquence « Ajouter des évènements »	32
V. Conception du sprint 4.....	33
1) Sprint Goal	33
a) Diagramme de séquence « Créer compte étudiant »	34
VI. Conception du sprint 5.....	35
a) Sprint Goal	35
a) Diagramme de séquence « Passer réservation ».....	36
b) Diagramme de séquence « Passer réclamation ».....	37
VII. Diagramme de classes	38
1) Diagramme de classes	38
2) Correspondance entre modèles objets et relationnels	39
Conclusion.....	39
Chapitre 4 : Réalisation de l'application web et mobile	40
Introduction	40
I. Choix Techniques	40
1) Les outils de développement	40
2) Les Frameworks	41
b) Application Mobile.....	42
c) API RESTful.....	43
II. Environnement logiciel	44
III. Design graphique.....	45
IV. Les interfaces	46
1) Application web (partie front)	46
a) Interface d'accueil.....	46
b) Interface d'authentification	46
a) Interface d'inscription	47
b) Interface de téléchargement de l'application	48
2) Application web (partie back).....	48
a) Interface administrateur	48
b) Interface liste des personnels	50
c) Interface liste des secteurs	50
d) Interface liste des évènements	51

e)	Interface liste des réclamations.....	52
f)	Interface liste des actualités	52
g)	Interface liste des étudiants.....	54
3)	Application mobile	55
a)	Interface publique.....	55
b)	Interface d'inscription et d'authentification	55
c)	Interface réglage profil	56
d)	Interface menu	57
e)	Interface prise rendez-vous avec le psychologue.....	58
f)	Interface liste des clubs.....	59
	Conclusion.....	60
	Conclusion générale	61
	Néographie	62

Table des figures

Figure 1: Localisation du foyer Ulysse Djerba	2
Figure 2 : Organigramme du foyer	3
Figure 3:Logo Uml	5
Figure 4:Le processus Scrum	6
Figure 5:Equipe Scrum.....	7
Figure 6: Diagramme de cas d'utilisation de l'administrateur	13
Figure 7: Diagramme de cas d'utilisation de responsable de club	14
Figure 8: Diagramme de cas d'utilisation du technicien	14
Figure 9: Diagramme de cas d'utilisation du psychologue.....	15
Figure 10: Diagramme de cas d'utilisation de l'étudiant	15
Figure 11: Diagramme de cas d'utilisation de l'étudiant	16
Figure 12: Architecture 3 tiers	22
Figure 13: MVC.....	23
Figure 14: diagramme de séquence « s'authentifier »	25
Figure 15: diagramme de séquence « valider étudiant »	28
Figure 16: diagramme de séquence « créer compte personnel »	29
Figue 17: diagramme de séquence « réglage profil »	31
Figure 18: diagramme de séquence « traiter réclamation ».....	32
Figure 19: diagramme de séquence « ajouter évènement ».....	32
Figure 20: diagramme de séquence « créer compte étudiant »	34
Figure 21: diagramme de séquence « réserver chambre »	36
Figure 22: diagramme de séquence « passer réclamation »	37
Figure 23: diagramme de classes	38
Figure 24: API RESTful	43
Figure 25: Logo my Ulysse	45
Figure 26: Interface d'accueil	46
Figure 27: Page s'authentifier pour l'étudiant (web)	47
Figure 28: Page d'inscription pour l'étudiant (web).....	47
Figure 29: Page de téléchargement de l'application mobile.....	48
Figure 30: Page Welcome	49
Figure 31: Page d'authentification	49
Figure 32: Liste personnels.....	50

Figure 33: Liste des secteurs	51
Figure 34: Liste des évènements	51
Figure 35: Liste des réclamations	52
Figure 36: Interface liste des actualités	52
Figure 37:Interface d'ajout des actualités.....	53
Figure 38:Intercae d'alerte d'ajout des actualités	53
Figure 39:Interface liste des étudiants	54
Figure 40: Interface alerte de suppression des étudiants	54
Figure 41: Inetrface publique.....	55
Figure 42: Interface inscription des étudiants.....	56
Figure 43: Interface connexion des étudiants	56
Figure 44: Interface réglage compte	57
Figure 45: Interface menu	57
Figure 46: Interface liste secteurs et liste chambres	58
Figure 47: Interface prendre rendez-vous	59
Figure 48 : Interface liste clubs	59
Figure 49: Interface liste clubs	60

Table des tableaux

Tableau 1:Les acteurs	10
Tableau 2: Description textuelle du cas d'utilisation s'authentifier	16
Tableau 3: Description textuelle du cas d'utilisation Créer compte.....	17
Tableau 4:Description textuelle du cas d'utilisation Gérer profil	17
Tableau 5: Description textuelle du cas d'utilisation Passer réservation.....	18
Tableau 6:Cas d'utilisation Gérer réservation	18
Tableau 7: Product backlog	21
Tableau 8: Backlog sprint 1.....	25
Tableau 9: Backlog sprint 2.....	27
Tableau 10: Backlog sprint 3.....	30
Tableau 11: Backlog sprint 4.....	33
Tableau 12: Backlog sprint 5.....	36
Tableau 13: L'environnement logiciel	44

Introduction Générale

Le logement dans le monde entier a connait une croissance sans précédente mais en Tunisie la réservation de logement est dirigée principalement aux hôtels.

Les logements en ligne existent depuis des années dans les pays avancés technologiquement pour faciliter la réservation des chambres et gagner le temps.

C'est dans ce cadre s'introduit notre projet de fin d'étude effectué au sein du foyer universitaire Ulysse Djerba et qui consiste à développer une plateforme web et mobile permettant l'administration du foyer et offrant un support de communication pour les étudiants. En plus, cette plateforme permet à l'étudiant de réserver sa chambre sans avoir besoin de se déplacer afin de réduire l'encombrement dans l'administration du foyer et gagner le temps lors de la rentrée universitaire.

Notre travail se traduit dans ce rapport qui développe les différentes phases de développement de notre projet et qui sont organisées en quatre chapitres de la manière suivante :

- Dans le premier chapitre, nous présenterons d'une manière générale l'organisme d'accueil, le cadre général de travail, une étude de l'existant et enfin la méthodologie adoptée.
- Dans le deuxième chapitre, nous présenterons la spécification des besoins fonctionnels et non fonctionnels du projet ainsi que le backlog du projet.
- Le troisième chapitre est dédié à l'étude conceptuelle de notre système. Dans ce chapitre, nous présenterons la modélisation des différents diagrammes nécessaires facilitant la compréhension du système.
- Le quatrième chapitre présente l'environnement de travail, les outils logiciels que nous avons utilisés pour la réalisation de notre projet ainsi que quelques imprimés écrans.

Chapitre 1 : Présentation générale du projet

Introduction

Ce chapitre introductif est consacré à la présentation de l'organisme d'accueil ainsi que le cadre général du projet. Nous allons présenter, dans un premier temps, le foyer universitaire Ulysse Djerba. Nous présentons par la suite, les problèmes et les enjeux de l'organisme d'accueil dans le but de dévoiler les objectifs de notre travail.

I. Présentation du foyer universitaire Ulysse Djerba

Le foyer universitaire Ulysse Djerba est un foyer universitaire public a été fondé le 25 Aout 1999.

Il a commencé à accueillir les étudiants depuis Septembre 2000.

La capacité maximale d'accueil est de 592 étudiants répartis entre 304 étudiantes et 288 étudiants.

Le nombre des chambres est 296 dont 152 réservées aux étudiantes et 144 réservées aux étudiants.

- **Dénomination :** Foyer Ulysse Djerba
- **Adresse :** Djerba route Houmet Essouk 2km, à l'interface de Bourgo Mall.
- **Téléphone :** 75 732 300
- **Fax :** 75 732 515
- **Horaires d'ouvertures :** 24h/24h.
- **Surface :** 8000 mètres carrées couvert sur un espace de 2.5 Hectares.

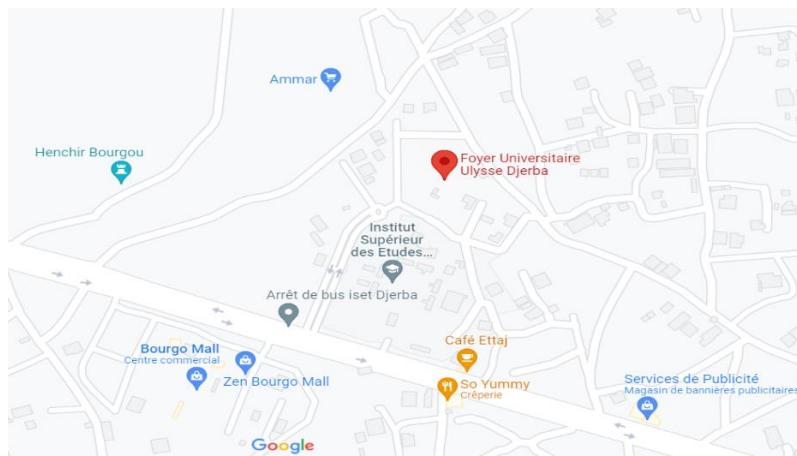


Figure 1: Localisation du foyer Ulysse Djerba

L'organigramme du foyer est le suivant :

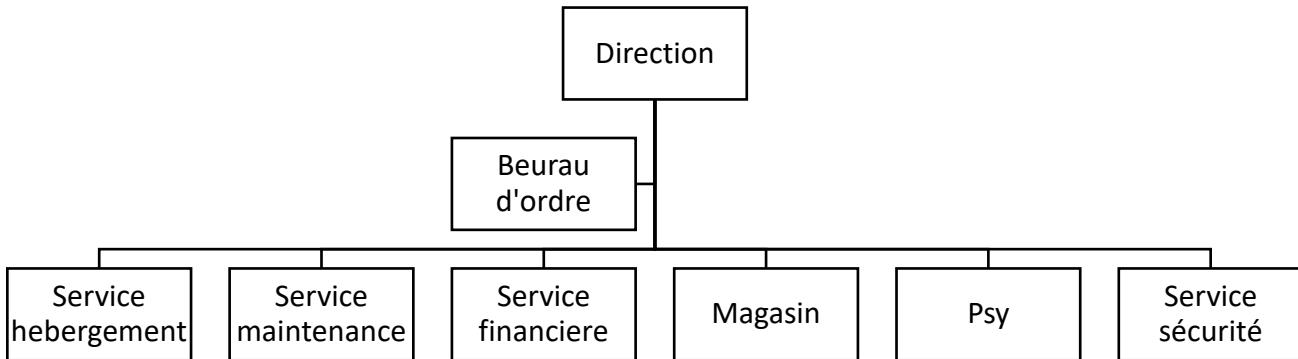


Figure 2 : Organigramme du foyer

1) Description des tâches

- **Direction** : le principal responsable de l'établissement est le directeur. Il veille à exploiter et à développer le patrimoine du foyer et à réaliser ses objectifs.
- **Bureau d'ordre** : il a pour rôle d'assurer la collaboration et la coordination entre les différents services ainsi que l'enregistrement des réceptions et des envois de toutes les correspondances de l'année en cours.
- **Service d'hébergement** : enregistrement des informations des étudiants et des chambres.
- **Service Maintenance** : ce service se préoccupe de tout ce qui est infrastructure.
- **Service financier** : contrôle le paiement des étudiants pour chaque tranche et le cautionnement à la fin de l'année universitaire ainsi que le paiement des mandats et factures.
- **Magasin** : le magasinier contrôle les biens immobiliers et s'occupe des produits consommables. Il assiste à leur achat et leur stockage lors de leur arrivée.
- **Service sécurité** : dispose des agents pour assurer la sécurité du foyer.

II. Présentation du projet

En se basant sur la propagation des réservations des logements en ligne, le foyer Ulysse Djerba nous propose de créer une plateforme qui réunit les étudiants de l'Iset de Djerba qui ont le droit de se loger au foyer.

1) Etude de l'existant

Le foyer universitaire Ulysse Djerba n'a pas un support d'information comme un site web ou bien une application mobile. En effet, pour passer l'information, parfois les responsables du foyer utilisent la page Facebook et là plus part du temps ils ont eu recours aux manières classiques comme le tableau d'affichage. A cause de ces méthodes traditionnelles, plusieurs étudiants ratent les évènements et les activités des clubs et d'ailleurs même les clubs sont inconnus par certains étudiants.

De plus, lorsque les étudiants ont des problèmes dans leurs chambres, ils sont obligés de noter l'anomalie rencontrée dans un cahier de réclamation dans le bureau d'accueil.

D'autre part, le premier jour de la rentrée les étudiants passent plusieurs heures pour terminer la réservation de chambre à cause de l'embouteillage.

2) Solution proposée

Après l'analyse de l'existant, nous sommes amenés à suggérer une solution qui permet de surmonter les problèmes constatés.

Ainsi, nous proposons une solution (un site web et une application mobile) représentant un support de communication et proposant un ensemble de fonctionnalités aux étudiants résidants dans le foyer. Cette solution permet aux étudiants de réserver leurs chambres avant la rentrée universitaire, réclamer les différentes anomalies rencontrées (panne, fuite, etc.), réserver un rendez-vous pour rencontrer le psychiatre du foyer et leur permet de rester branchés en leur présentant les différents événements et actualités du foyer universitaire.

Notre plateforme est composée de trois parties :

Une partie BackOffice :

C'est la partie administration qui permet la gestion du site web. Elle permet à l'administrateur de gérer les utilisateurs, de valider les demandes d'inscription des étudiants et de gérer les évènements, les clubs et les réclamations.

Les personnels du foyer à savoir le responsable des clubs, le technicien et le psychologue ont accès à cette partie mais chacun a un accès restreint à ses fonctionnalités.

Une partie Front End :

Cette partie est destinée aux publics permettant de représenter le foyer, ses services et une galerie contenant des images des différents coins du foyer. C'est aussi la partie qui permet aux étudiants de s'inscrire et de réserver les chambres par la suite.

Une partie Mobile :

C'est une application mobile qui permet aux étudiants de se connecter via un adresse e-mail et un mot de passe afin d'accéder aux différents services : réservation chambre, consultation des actualités et évènements, participation aux évènements, rendez-vous avec le psychologue, etc.

L'application mobile n'est pas directement connectée à la base de données, mais via L'API RESTful qui joue le rôle d'interfaçage. En fait, l'APRI REST reçoit des requêtes http qui vient de l'application mobile.

III. Méthodologie et modélisation adoptée

1) Méthodologie de conception

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. [1]



Figure 3:Logo Uml

UML permet grâce à sa représentation graphique, d'exprimer visuellement une solution objet, de faciliter la compréhension de l'évolution de la solution.

Cette étude se déroule en deux phases. Durant la première phase nous analyserons les besoins attendus de l'application par le biais des diagrammes de cas d'utilisations et les diagrammes de séquences pour approfondir la compréhension de l'exécution des fonctionnalités de l'application à développer. La deuxième phase s'intéresse au point de vue données à travers les diagrammes de classes.

2) Méthodologie de travail

Utiliser UML tout seul n'est pas suffisant pour modéliser correctement et convenablement un système logiciel, nous avons besoin d'une pratique, d'un guide qui nous donne des étapes et des manières pour exploiter les diagrammes UML, dans ce cadre viennent les méthodes agiles pour proposer les bonnes pratiques aidant le concepteur à avoir une méthode de conception.

Pour cela, nous avons choisi de travailler avec la méthode SCRUM.

Scrum est une méthode de développement agile orientée projet informatique dont les ressources sont régulièrement actualisées. La méthode Scrum tire son nom du monde du rugby, Scrum = mêlée. Le principe de base étant d'être toujours prêt à réorienter le projet au fil de son avancement. C'est une approche dynamique et participative de la conduite du projet. La mêlée est une phase de jeu essentielle au rugby. Elle permet au jeu de repartir sur d'autres bases. La réunion dans la méthode Scrum relaie la métaphore. [2]

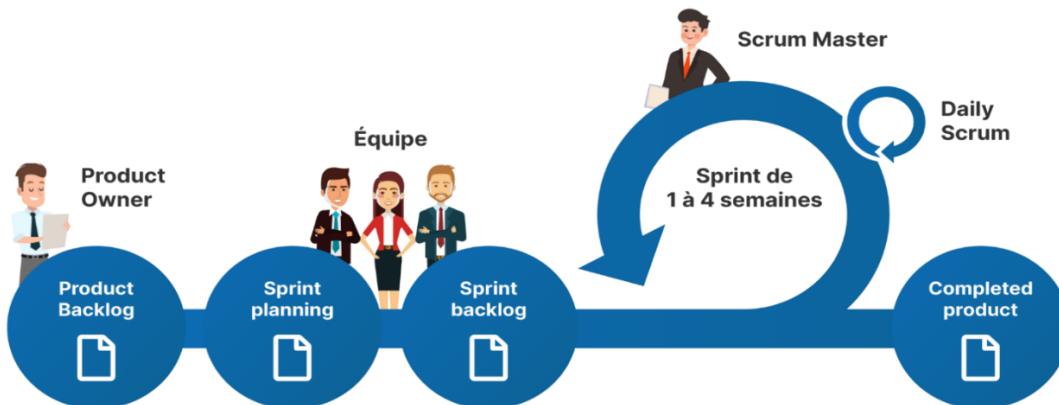


Figure 4:Le processus Scrum

IV. Framework Scrum

Nous ne prétendons pas que notre projet répond aux exigences de SCRUM à cent pour cent mais nous nous en inspirons.

1) Équipe et Rôle

L'Équipe Scrum comprend un propriétaire de produit (Product Owner), une Équipe de Développement (Développement Team) et un Scrum Master.

- **Développement Team :** Ce sont les personnes chargées de la réalisation des sprints.
- **Product Owner :** C'est l'acteur clé du projet. C'est une partie des responsabilités du propriétaire par l'entreprise qui a une vision de ses besoins et le transmettre à l'équipe Scrum.
- **Scrum Master :** Il assure globalement le bon déroulement des programmes et protège l'équipe de tout problème extérieur. Il assure globalement l'organisation des réunions et la bonne application de la méthode agile.

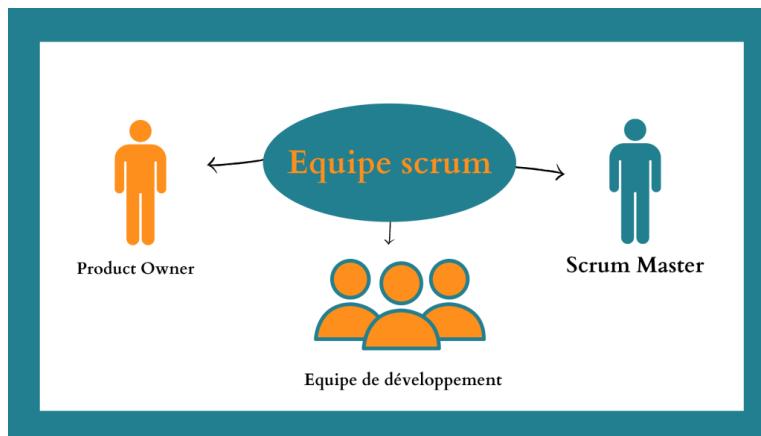


Figure 5:Equipe Scrum

Dans notre cas, les rôles sont répartis comme suit :

- **Product Owner :** Foyer Ulysse Djerba
- **Scrum Master :** Essid Samir
- **Développement Team :** Chebli Kais & Lafi Oumaima

2) Fixation du Sprint time Box

Avant d'entamer notre travail, nous fixons la durée des sprints à savoir les <sprints Time-Box>. Pour notre projet, nous avons choisi que les sprints dure entre 7 et 20 jours.

3) Les cérémonies

a) Sprint planning

Nous organisons, avant chaque sprint, une réunion de planification, le sprint planning meeting. Ce planning sélectionne dans le Product backlog les exigences les plus prioritaires pour le client. Elles seront développées, testées et livrées à la fin du sprint. Elles constituent le sprint backlog, un sous ensemble du Product backlog.

b) Daily meeting

L'objectif du Daily meeting est de s'assurer ensemble que nous avons bien atteint notre objectif chaque jours le Daily meeting se base sur trois questions :

Qu'est-ce que tu as fait hier ?

Quels sont les problèmes que tu as rencontrés ?

Qu'est-ce que tu as prévu aujourd'hui ?

c) Sprint Review

C'est la réunion qui marque à la fin de développement. L'équipe présente les résultats du Sprint et Le Product Owner valide ou rejette les items du Sprint Backlog en fonction de la Définition of Done.

d) Sprint Rétrospective

Toute l'équipe Scrum participe à la réunion. La rétrospective a lieu juste après la revue de sprint et les intervenant qui sont venus y assister peuvent rester pour la rétrospective, à titre d'observateurs. Cependant, la confiance est nécessaire pour le succès d'une rétrospective et la présence de certaines personnes peut être animée par le Scrum Master. Mais, notamment dans les environnements difficiles, il est préférable que ce soit une personne extérieure à l'équipe qui joue le rôle de facilitateur de cette réunion le but de la réunion est d'améliorer le processus pour le prochain sprint.

Conclusion

A travers ce chapitre, nous avons présenté le cadre de notre projet de fin d'études et nous avons identifié l'objectif à atteindre et le travail à faire. Compte tenu de cela, nous présenterons la spécification des besoins fonctionnels et non-fonctionnels du projet dans le chapitre suivant.

Chapitre 2 : Spécification des besoins

Introduction

Après avoir présenté le projet au niveau du chapitre précédent et avant de commencer la phase de développement, nous élaborons à ce niveau la spécification des besoins qui sert à bien organiser et clarifier les fonctionnalités du projet.

Ce chapitre consiste donc à déterminer l'ensemble des besoins de notre système et modéliser ses interactions, suivant la spécification de la méthodologie UML, tout en respectant certaines contraintes.

I. Spécification des besoins

Dans cette partie, nous détaillons les exigences spécifiques qui se manifestent en des besoins fonctionnels et des besoins non fonctionnels dont le but est d'identifier les services que doit fournir notre application.

1) Identification des acteurs

« Un acteur » représente l'abstraction d'un rôle joué par des entités externes (utilisateur, dispositif matériel ou autre système) qui interagissent directement avec le système étudié. Tout simplement un acteur est une entité physique (personne) ou abstraite (logiciel) capable d'utiliser le système afin de répondre à un besoin bien défini.

Les acteurs de notre application sont :

Acteur	Rôles
Administrateur	C'est l'administrateur de l'application web/mobile, il valide les réservations des chambres et les utilisateurs qui font l'inscription ainsi que la gestion du site.
Etudiant	L'étudiant n'a pas besoin de compte pour consulter le site du foyer, mais il est obligé de créer un compte pour passer une réservation et pour pouvoir accéder aux différentes fonctionnalités de l'application mobile.
Responsable des clubs	Le responsable des clubs est un personnel du foyer qui doit avoir un compte. A travers notre plateforme, il peut gérer les clubs et

	les événements.
Technicien	Chaque technicien doit avoir un compte. A travers notre plateforme, il peut gérer les réclamations.
Psychologue	Chaque psychologue doit avoir un compte. A travers notre plateforme, il peut gérer les rendez-vous.

Tableau 1:Les acteurs

2) Exigences fonctionnelles

Les principaux besoins fonctionnels de notre application se résument autour des points suivants :

BF1 : Le système doit permettre à l'administrateur de s'authentifier et de se déconnecter

BF2 : Le système doit permettre à l'étudiant de s'authentifier et de se déconnecter

BF3 : Le système doit permettre au responsable club de s'authentifier et de se déconnecter

BF4 : Le système doit permettre au technicien de s'authentifier et de se déconnecter

BF5 : Le système doit permettre au psychologue de s'authentifier et de se déconnecter

BF6 : L'application Web doit permettre à l'administrateur de :

- BF6.1 : Gérer les personnels.
- BF6.2 : Gérer les étudiants.
- BF6.3 : Gérer les réservations.
- BF6.4 : Gérer les réclamations.
- BF6.5 : Gérer les rendez-vous avec le psychologue.
- BF6.6 : Gérer les clubs.
- BF6.7 : Gérer les événements.
- BF6.7 : Gérer les actualités.
- BF6.8 : Gérer les chambres.
- BF6.9 : Gérer les secteurs.

BF7 : L'application Web doit permettre aux étudiants de :

- BF7.1 : Créer un compte.
- BF7.2 : Passer une réservation.
- BF7.3 : Gérer son compte.
- BF7.4 : Télécharger l'application mobile.

BF8: L'application Web doit permettre au Responsable clubs de :

- BF8.1 : Gérer profil.
- BF8.2 : Gérer les clubs.
- BF8.3 : Gérer les événements.
- BF8.4 : Gérer les actualités.

BF9 : L'application Web doit permettre au psychologue de :

- BF9.1 : Gérer profil.
- BF9.2 : Gérer les rendez-vous.
- BF9.3 : Gérer des disponibilités.

BF10 : L'application mobile doit permettre aux étudiants de :

- BF10.1 : S'authentifier.
- BF10.2 : Réserver une chambre.
- BF10.3 : Consulter les clubs.
- BF10.4 : Joindre un club.
- BF10.5 : Consulter les évènements.
- BF10.6 : Participer à un évènement.
- BF10.7 : Passer une réclamation.
- BF10.8 : Prendre un rendez-vous avec le psychologue.
- BF10.9 : Consulter les actualités.

3) Exigences non fonctionnelles

Les besoins non fonctionnels sont importants car ils agissent de façon indirecte sur le résultat et sur la performance de la plateforme, ce qui fait qu'ils ne doivent pas être négligés. Pour cela, il faut répondre aux exigences suivantes :

BNF1 : Sécurité : Les applications doivent être sécurisées et garantissent la protection et la confidentialité des données.

BNF2 : Performance : les applications doivent être performantes à travers leurs fonctionnalités et répondent d'une manière optimale et non-ambiguë à toutes les exigences d'utilisation.

BNF3 : Compatibilité

- Compatibilité sur différents systèmes d'exploitation.
- Compatibilité sur différentes plateformes

II. Modélisation des besoins fonctionnels

Une étude approfondie des besoins fonctionnels est indispensable avant d'entamer le développement pour obtenir, d'une manière plus formelle, une vue globale sur les exigences de notre application.

Cette partie présente alors une modélisation des besoins en faisant recours aux concepts fondamentaux d'UML, à savoir le diagramme de cas d'utilisation.

1) Diagrammes de cas d'utilisation

Un cas d'utilisation « use case » représente un ensemble de séquences d'actions qui sont réalisées par le système et qui produisent un résultat observable intéressant pour un acteur particulier. Il permet de décrire ce que le futur système devra faire, sans spécifier comment il le fera. [3]

a) Diagramme de cas d'utilisation de l'administrateur

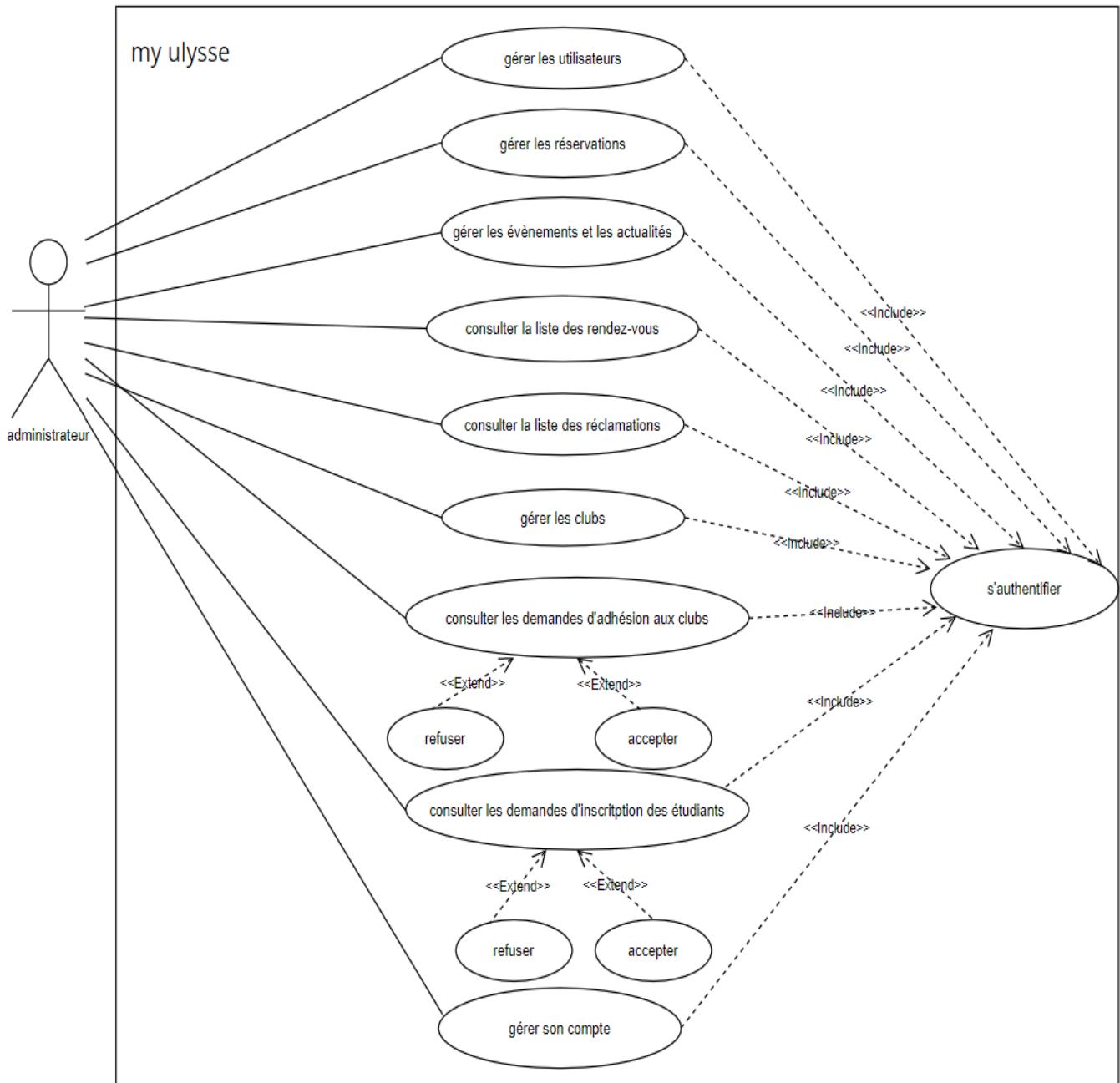


Figure 6: Diagramme de cas d'utilisation de l'administrateur

b) Diagramme de cas d'utilisation du responsable des clubs

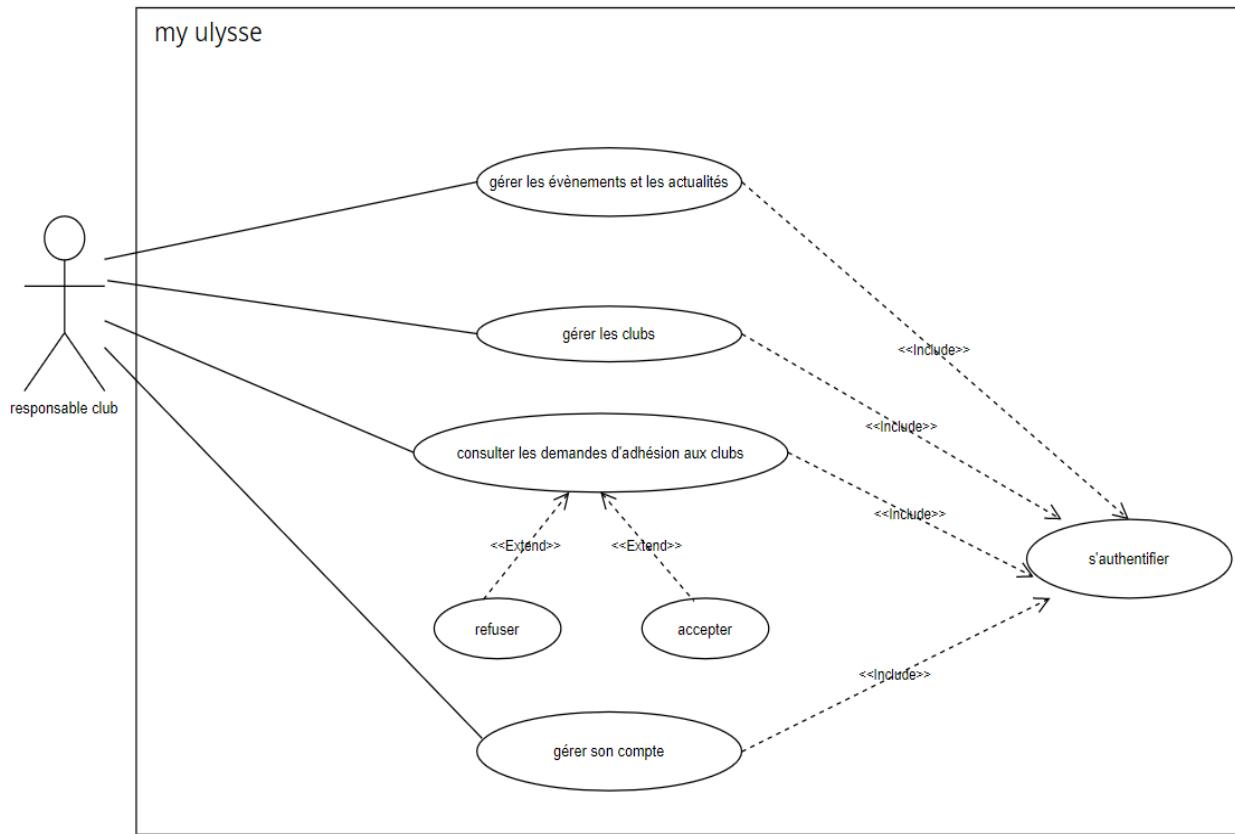


Figure 7: Diagramme de cas d'utilisation de responsable de club

c) Diagramme de cas d'utilisation du technicien

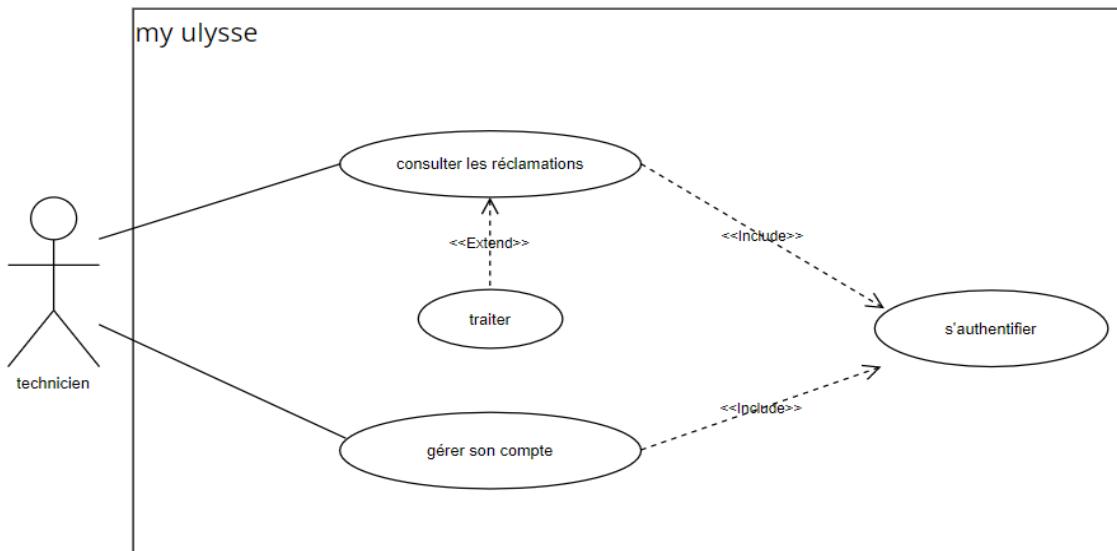


Figure 8: Diagramme de cas d'utilisation du technicien

d) Diagramme de cas d'utilisation du psychologue

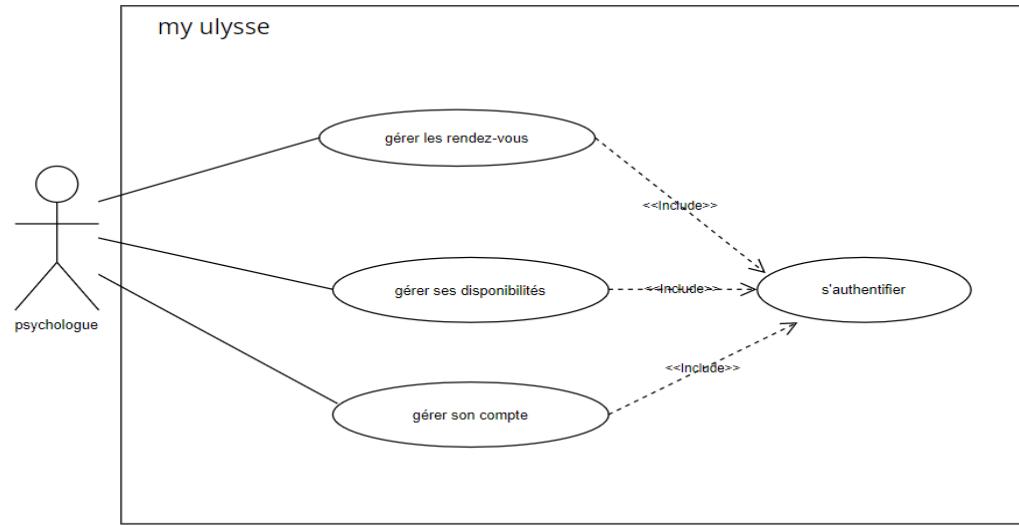


Figure 9: Diagramme de cas d'utilisation du psychologue

e) Diagramme de cas d'utilisation de l'étudiant

➤ Application mobile

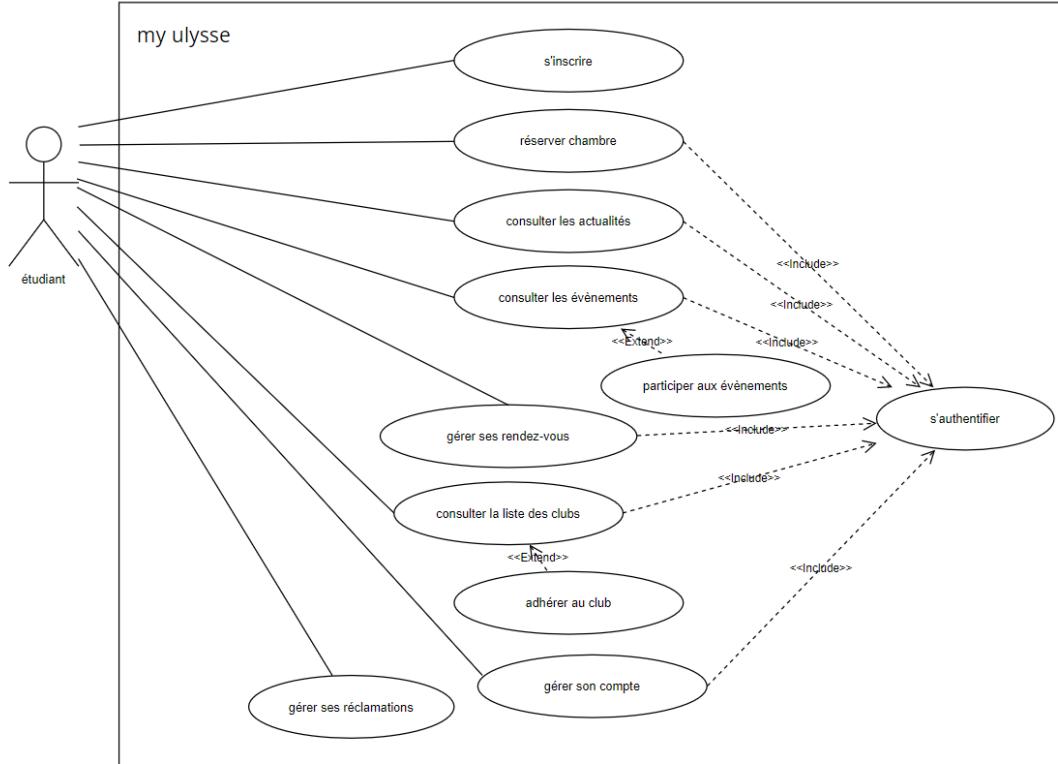


Figure 10: Diagramme de cas d'utilisation de l'étudiant

➤ Application web

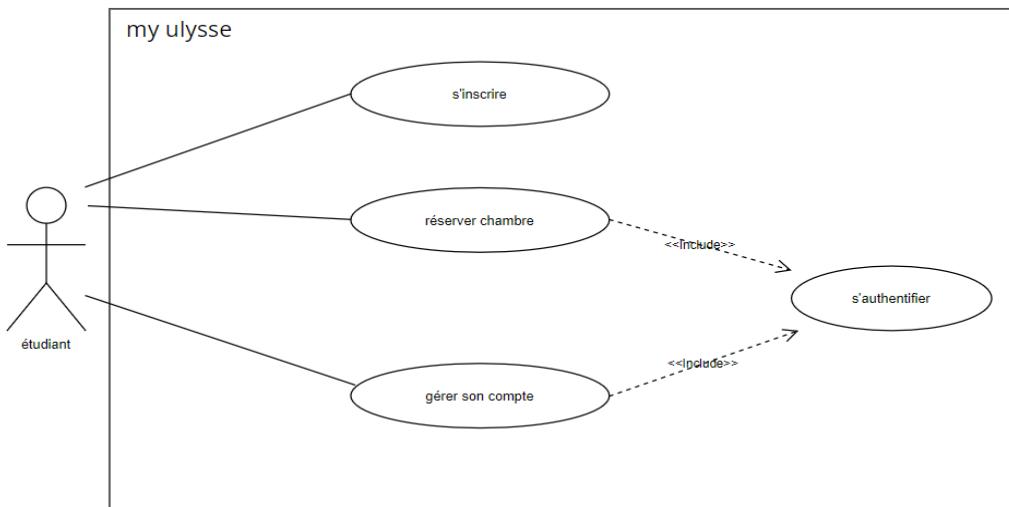


Figure 11: Diagramme de cas d'utilisation de l'étudiant

1) Description textuelle

a) Description textuelle du diagramme de cas d'utilisation « S'authentifier »

Cas d'utilisation: S'authentifier	
Acteur	Administrateur, étudiant, responsable club, technicien, psychologue
Précondition	Connexion internet L'utilisateur demande de s'authentifier
Post-condition	L'utilisateur est authentifié
Scénario nominal	1. L'utilisateur ouvre le formulaire d'authentification 2. L'utilisateur entre son e-mail et le mot de passe 3. Le système vérifie la validité des informations entrées 4. Le système lui redirige à son interface selon le type de l'utilisateur : administrateur, étudiant, responsable club, technicien, psychologue.
Exception	2.a Si l'utilisateur a oublié son mot de passe, il demande la page de restauration du mot de passe. Il saisit son email et le système lui envoie un email pour restauration du mot de passe. 3.a Si les informations entrées par l'utilisateur ne sont pas valides ou vides, le système lui affiche un message d'erreur lui informant qu'il doit vérifier ses informations.

Tableau 2: Description textuelle du cas d'utilisation s'authentifier

b) Description textuelle du diagramme de cas d'utilisation « Créeer compte »

Cas d'utilisation : Créeer compte	
Acteur	Etudiant
Précondition	Connexion internet étudiant demande de s'inscrire
Post-condition	étudiant s'inscrit et attend la validation par l'administrateur
Scénario nominal	1. L'étudiant ouvre le formulaire d'inscription 2. L'étudiant entre son nom, e-mail et le mot de passe 3. Le système vérifie la validité des informations entrées 4. Le système ajoute l'étudiant à la base de données et lui confirme la création de son compte sous réserve de validation. 5. Un email est envoyé à l'administrateur pour valider le compte qui a fait l'inscription. 6. Un email est envoyé à l'étudiant pour lui informer de la validation de son compte.
Exception	3.a Si les informations entrées par l'utilisateur ne sont pas valides ou vides, le système lui affiche un message d'erreur lui informant qu'il doit vérifier ses informations et/ou l'email existe déjà.

Tableau 3: Description textuelle du cas d'utilisation Créeer compte**c) Description textuelle du diagramme de cas d'utilisation « Gérer profil »**

Titre	Gérer profil
Acteur	Admin, Etudiant, Responsable club, technicien, psychologue
Résumé	L'utilisateur qui est connecté capable de gérer son profil
Précondition	Le utilisateur doit s'authentifier
Scénario-nominal	1- S'authentifier 2- Consulter son profil 3- Gérer son profil 4-Affichage de son profil (adresse email , nom , tel ..) dans la même page et modifier le profil et valider

Tableau 4:Description textuelle du cas d'utilisation Gérer profil

d) Description textuelle du diagramme de cas d'utilisation « Passer réservation »

Titre	Passer réservation
Acteur	Etudiant
Résumé	A tout moment, l'étudiant peut accéder à notre site web pour pouvoir passer une réservation.
Précondition	L'étudiant doit s'authentifier Chambre disponible
Scénario-nominal	1- S'authentifier 2- Choisir un secteur 3- Choisir une chambre 4- Remplissage du formulaire de réservation. (Année du baccalauréat et le nom du binôme). 5- Valider la réservation

Tableau 5: Description textuelle du cas d'utilisation Passer réservation**e) Description textuelle du diagramme de cas d'utilisation « Gérer Réservations »**

Titre	Gérer réservation
Acteur	Administrateur
Résumé	L'administrateur peut consulter les réservations, et traiter aussi l'état d'une demande en cours pour la valider.
Précondition	L'administrateur du foyer doit s'authentifier Réservation existe.
Scénario-nominal	1- S'authentifier 2- Consulter liste des réservations. 3- Traiter l'état par l'administrateur. 4-l'envoi d'email aux étudiants.

Tableau 6:Cas d'utilisation Gérer réservation

III. Le product backlog

ID	User Story	En tant que	Description	Priorité	Durée	Sprint
1	Interface Administrateur	Administrateur	L'admin consulte cette interface pour gérer plusieurs actions	moyenne	2 jours	Sprint 1
2	Interface psychologue	Psychologue	L'interface du psychologue	moyenne	1 jour	
3	Interface responsable club	Responsable club	L'interface du responsable des clubs	moyenne	1 jour	
4	Interface technicien	Technicien	L'interface du technicien	moyenne	1 jour	
6	Authentification	Technicien, étudiant, responsable club, admin, psychologue	Authentification et accès à l'espace utilisateur	moyenne	1 jour	
7	Réinitialisation de mot de passe avec les emails	Technicien, admin, Responsable club, psychologue	Réinitialisation du mot de passe en cas d'oublié	moyenne	2 jours	
8	Gestion de profil	Administrateur , psychologue, étudiant, responsable club, technicien	L'administrateur peuvent modifier son compte et les autres comptes dès les utilisateurs	haut	3 jours	
9	Validation compte étudiant	Administrateur	L'administrateur peut valider un compte étudiant pour lui donner l'accès	haute	2 jours	
10	Gestion des utilisateurs	Administrateur	L'administrateur peut gérer les utilisateurs	moyenne	4 jours	2 Sprint
11	Gestion des étudiants	Administrateur	L'administrateur peut gérer les étudiants	moyenne	1 jour	
12	Gestion des réservations	Administrateur	L'administrateur peut gérer les réservation	moyenne	5 jours	
13	Consulter la liste des rendez-vous	Administrateur	L'administrateur peut consulter la liste des rendez-vous	faible	1 jour	

14	Consulter la liste des réclamations	Administrateur	L'administrateur peut consulter la liste des réclamations	faible	1 jour	
15	Gestion des événements et des actualités	Administrateur	L'administrateur peut gérer les événements et les actualités	moyenne	2 jours	
16	Gestion des clubs	Administrateur	L'administrateur peut gérer les clubs	moyenne	2 jours	
17	Gérer les demandes d'adhésion au clubs	Administrateur	L'administrateur peut gérer les demandes d'adhésion	faible	1 jour	
18	Gestion des réclamations	Technicien	Le technicien peut gérer les réclamations	moyenne	2 jours	Sprint 3
19	Gestion des rendez-vous	Psychologue	Le psy peut gérer les rendez-vous	moyenne	2 jours	
20	Gestion des événements et des actualités	Responsable des clubs	Le responsable clubs peut gérer les événements et les actualités	moyenne	2 jours	
21	Gestion des clubs	Responsable des clubs	Le responsable clubs peut gérer les clubs	moyenne	2 jours	
22	Gérer les demandes d'adhésion aux clubs	Responsable des clubs	Le responsable clubs peut gérer les demandes d'adhésion	haute	2 jours	
23	Interface publique	Etudiant	L'étudiant peut consulter cette interface	moyenne	3 jours	Sprint 4
24	Inscription étudiant	Etudiant	Inscription des étudiants au site	moyenne	2 jours	
25	Passer une réservation	Etudiant	L'étudiant peut réserver une chambre	moyenne	3 jours	
26	Api authentification, inscription	Etudiant	Authentification et inscription d'étudiant	haute	2 jours	
27	L'interface connexion inscription	Etudiant	Chaque étudiant peut consulter cette interface pour s'authentifier ou inscrire	moyenne	2 jours	
28	Interface publique	Etudiant	Chaque étudiant peut visiter cette interface	moyenne	2 jours	

29	Passer une réservation	Etudiant	Après la validation du compte , l'étudiant passe une réservation	moyenne	1 jour	Sprint 5
30	Api liste des clubs	Etudiant	Chaque client peut trouver la liste des clubs	haute	4 jours	
31	Consulter la liste de club	Etudiant	L'étudiant peut consulter la liste des clubs	faible	1 jour	
32	Adhérer aux clubs	Etudiant	L'étudiant peut adhérer un club	faible	1 jour	
33	Api liste des événements et des actualités	Etudiant	Retourner la liste des événements et des actualités	haute	4 jours	
34	Consulter les événements et les actualités	Etudiant	L'étudiant peut consulter la liste des événements et des actualités	faible	1 jour	
35	Participer aux événements	Etudiant	L'étudiant peut participer aux événements	faible	1 jour	
36	Gérer les rendez-vous	Etudiant	L'étudiant peut gérer ses rendez-vous	moyenne	1 jour	
37	Gérer les réclamations	Etudiant	L'étudiant peut gérer ses réclamations	moyenne	1 jour	
38	Gérer son compte	Etudiant	L'étudiant peut gérer son compte	moyenne	1 jour	

Tableau 7: Product backlog

Conclusion

Dans ce chapitre, nous avons présenté les exigences fonctionnelles et non fonctionnelles de notre système ainsi que le raffinement des diagrammes des cas d'utilisation pour montrer les interactions entre les différents acteurs et le système. Ensuite, nous avons présenté les différents sprints à travers le Product backlog.

Le chapitre prochain sera consacré à l'étude conceptuelle.

Chapitre 3 : Etude Conceptuelle

Introduction

La partie conception est la partie la plus importante dans la réalisation de n'importe quel projet. Elle permet de décrire d'une manière non ambiguë le fonctionnement d'un système afin de simplifier sa réalisation. Ce chapitre sera consacré à la conception de notre application.

Dans ce chapitre, nous identifions en premier lieu l'architecture de notre système. Par la suite, nous présentons quelques diagrammes de séquences et le diagramme de classes de notre projet.

I. Architecture du système

1) L'architecture 3 tiers

L'architecture à trois niveaux est une architecture d'application logicielle bien établie qui organise les applications en trois niveaux de calcul logiques et physiques : le niveau Présentation, ou interface utilisateur, le niveau Application, où les données sont traitées et le niveau Données, où les données associées à l'application sont stockées et gérées. [4]

L'architecture 3 tiers est divisée en trois niveaux ou couches :

- Couche de présentation.
- Couche de traitement.
- Couche d'accès aux données.

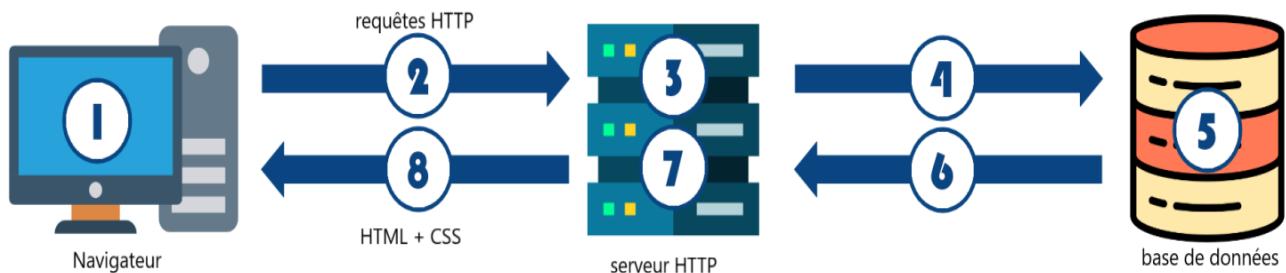


Figure 12: Architecture 3 tiers

2) Le design pattern d'architecture MVC

L'architecture MVC (Model-View-Controller), est l'une des architectures logicielles les plus utilisées pour les applications Web. Elle permet de créer une application web pour bien gérer la structuration d'un projet en trois parties. [5]

Le motif MVC se compose de trois modules ou objets principaux à utiliser dans le développement de logiciels :

- Un **modèle** représentant la structure logique sous-jacente des données dans une application logicielle, ainsi que la classe supérieure qui y est associée. Ce modèle d'objet ne contient aucune information sur l'interface utilisateur.
- Une **vue**, autrement dit un ensemble de classes représentant les éléments de l'interface utilisateur (tous ceux que l'utilisateur voit à l'écran et avec lesquels il peut interagir : boutons, boîtes de dialogue, etc.).
- Un **contrôleur** représentant les classes qui se connectent au modèle et à la vue, et servant à la communication entre les classes dans le modèle et la vue.

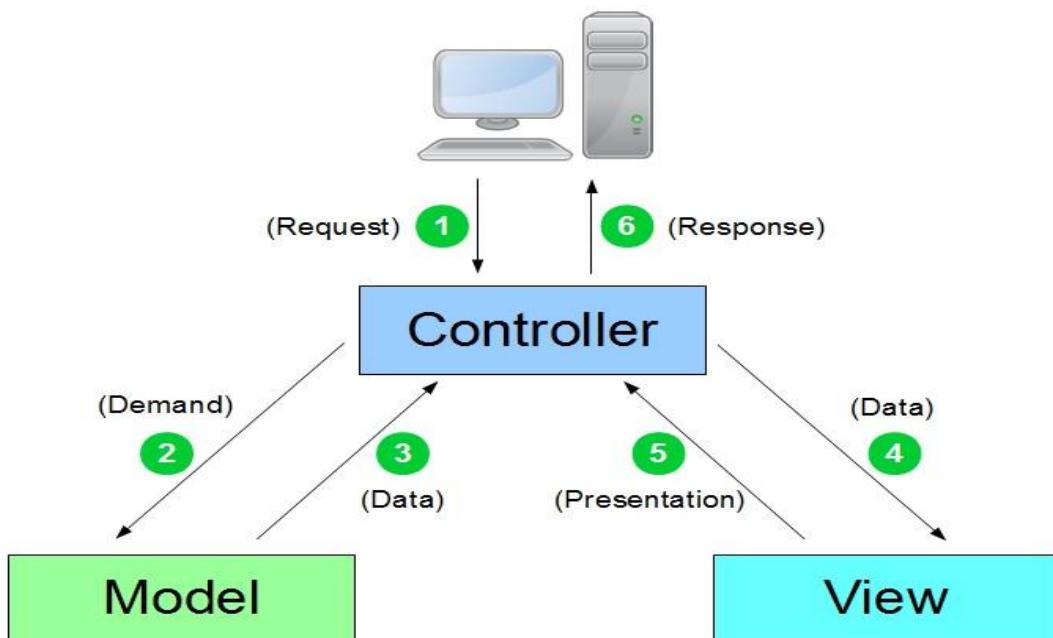


Figure 13: MVC

II. Conception du sprint 1

Dans le monde des méthodes Agiles, le sprint désigne le cycle de développement pendant lequel vont s'enchaîner un certain nombre de tâches pour, à terme, s'achever par la conception d'un produit final.

Cela se caractérise par un rassemblement de personnes impliquées dans un projet afin de se concentrer sur le développement de ce projet.

1) Sprint Goal

Est une brève explication de ce que l'équipe prévoit d'accomplir au cours d'un sprint Agile. Il s'agit d'un objectif tangible écrit ensemble par l'équipe et le Product Owner et limité dans le temps à la durée du sprint. En d'autres termes, les objectifs de sprint clarifient votre objectif lors d'un incrément de programme.

Notre but ici est de réaliser les différentes interfaces de l'application mobile et l'application web.

2) Sprint Backlog

Le sprint backlog, c'est la liste de l'ensemble des éléments à réaliser tout au long du sprint afin d'être en mesure de livrer l'incrément à la fin de celui-ci. Ces éléments en question proviennent du Product backlog qui a été construit en amont, à la base du projet Scrum. [6]

Une fois, nous avons défini le but du sprint, il est temps de décider quelles sont les fonctionnalités et leurs degrés d'importance incluent dans ce dernier. Plus précisément, quels noms de Backlog du produit seront inclus dans le Backlog du sprint.

User story	Tâches	Degré de difficulté	Priorité
l'interface privé	Intégration des pages de l'application web	Facile	Elevé
	Création d'un Template Bootstrap		
	Intégration de la maquette dans les fichiers blade de la Framework Laravel		
l'interface de l'administrateur	Intégration de la page pour l'administrateur	Facile	Elevé
	Création d'un Template Bootstrap		
	Intégrer la maquette dans les fichiers blade de la Framework Laravel		
l'interface de psychologue,	Intégration de la page pour le psychologue, technicien et la responsable clubs.		

technicien et responsable du club	Création d'un Template Bootstrap Intégrer la maquette dans les fichiers blade de la Framework Laravel	Facile	
l'authentification	Intégration et testes nécessaires : Un formulaire login. Chaque personne est redirigée vers son interface grâce au middleware	Moyenne	
réinitialisation du mot de passe avec les emails	Intégration des tests nécessaires Vérification par email et changement du mot de passe pour l'administrateur.	Moyenne	
Gestion de profil	Chaque utilisateur peut modifier son compte	Moyenne	

Tableau 8: Backlog sprint 1

3) Diagramme de séquence « S'authentifier »

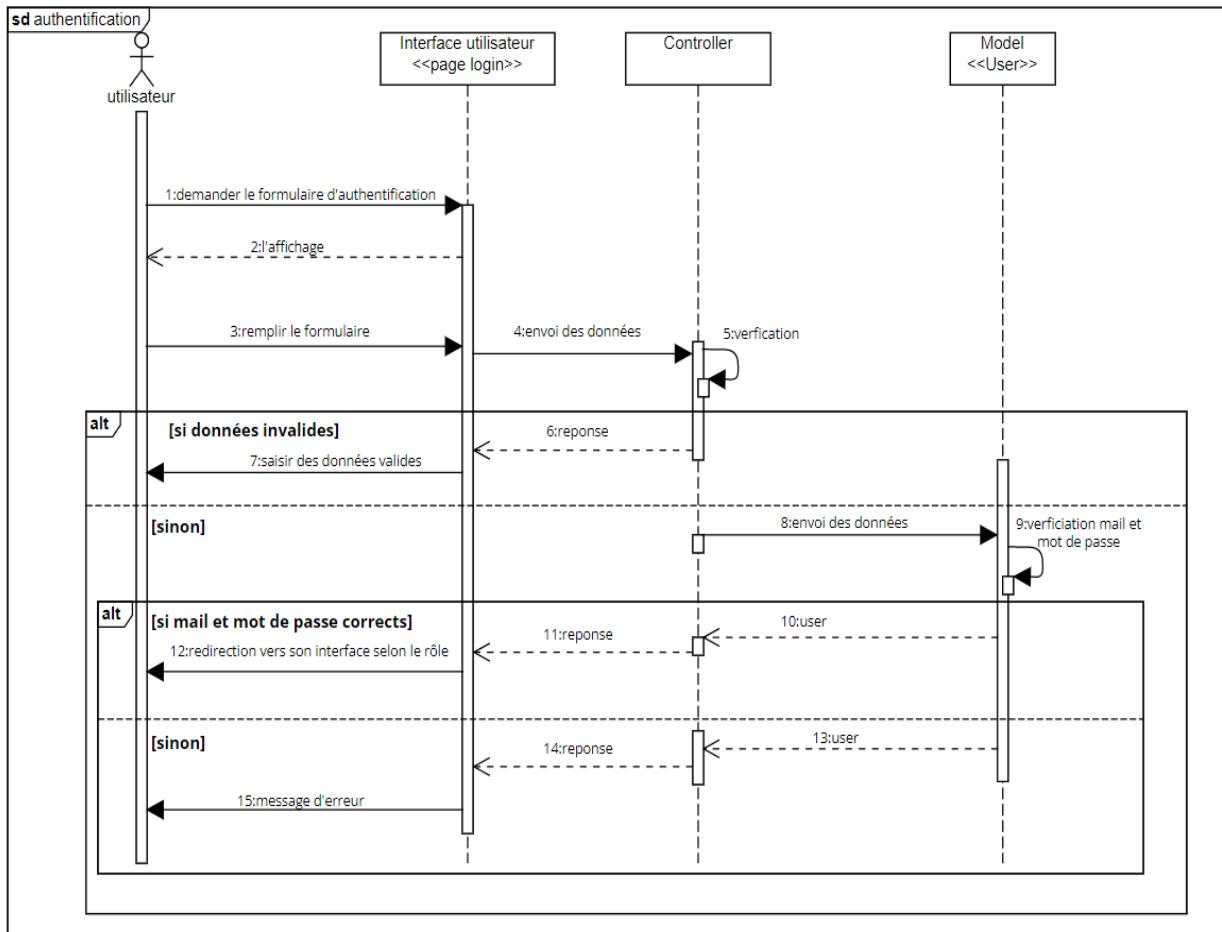


Figure 14: diagramme de séquence « s'authentifier »

III. Conception du sprint 2

Dans ce qui suit on va déterminer l'étude détaillée du deuxième sprint.

1) Sprint Goal

Le but de ce sprint est de développer les tâches de l'administrateur la validation des comptes étudiants, la gestion des utilisateurs, la gestion des évènements, les clubs, les rendez-vous avec le psychologue, les réclamations et les réservations des chambres.

2) Sprint backlog

Le tableau ci-dessous résume le Backlog de notre second sprint.

User story	Tâches	Degré de difficulté	Priorité
Validation de compte étudiant	Intégration des pages de l'application web. Création d'un Template Bootstrap. Intégrer la maquette dans les fichiers blade de la Framework Laravel.	Moyenne	
Gestion des étudiants	Validation ou bien l'invalidation des comptes des étudiants et la gestion des comptes des étudiants.	Moyenne	
Gestion des utilisateurs	L'ajout, la modification et la suppression des utilisateurs.	Moyenne	
Gérer les demandes d'adhésion au clubs	L'administrateur peut gérer les demandes d'adhésion.	Moyenne	Elevé
Gestion des clubs	Consultation de la liste des clubs. L'ajout la modification et la suppression des clubs.	Facile	
Gestion des évènements et des actualités	Consultation de la liste des évènements. L'ajout de la modification et la suppression des évènements et des actualités.	Moyenne	
Consultation la liste des	Consultation de la liste des réclamations.	Facile	

réclamations			
Gestion des réservations	Consultation de la liste des réservations. Le validation ou bien l'invalidation des réservation.	Moyenne	
Consultation la liste des rendez-vous	Consultation de la liste des rendez-vous.	Facile	

Tableau 9: Backlog sprint 2

3) Diagramme de séquence « Valider compte étudiant »

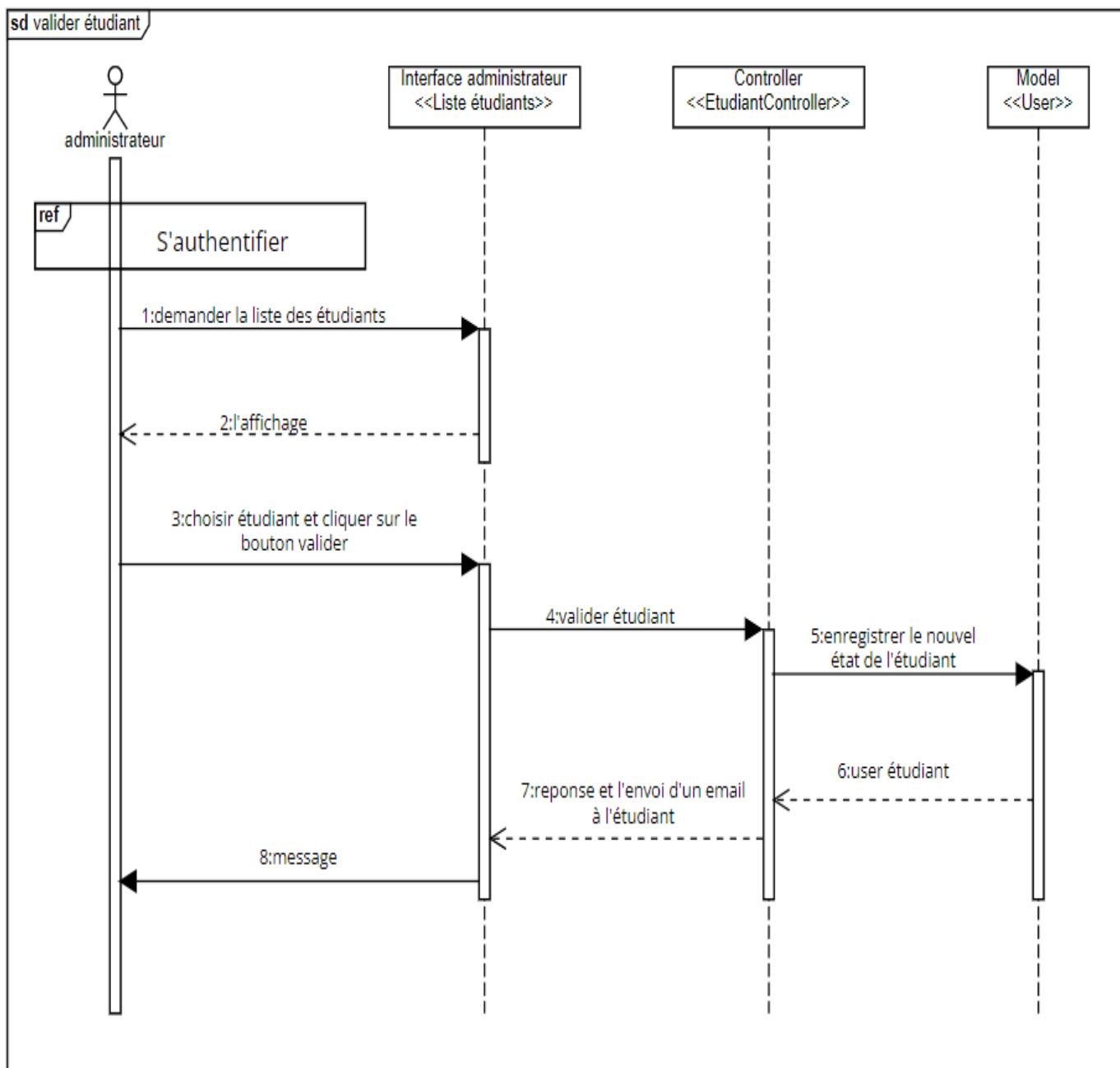


Figure 15: diagramme de séquence « valider étudiant »

4) Diagramme de séquence « créer compte personnel »

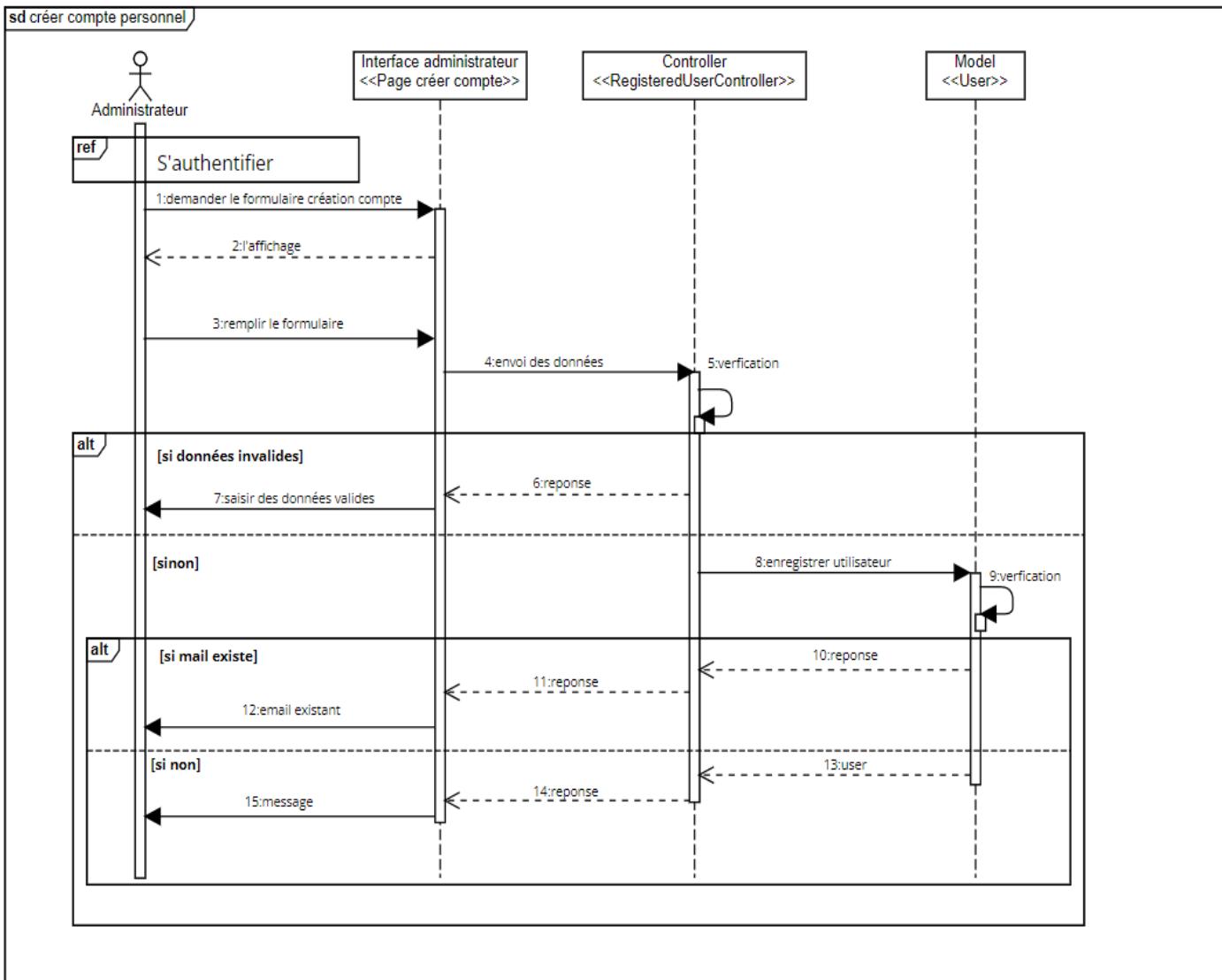


Figure 16: diagramme de séquence « créer compte personnel »

IV. Conception du sprint 3

En partant sur le même principe que les sprints précédents, nous commençons par définir le but du sprint.

1) Sprint Goal

Le but de ce sprint est de réaliser la partie concernant les techniciens, le psychologue et la responsable des clubs et leurs tâches.

2) Sprint Backlog

Le tableau ci-dessous résume le Backlog de notre troisième sprint.

User story	Tâches	Degré de difficulté	Priorité
La gestion de profil pour le technicien	Intégration d'un formulaire pour la modification des coordonnées (nom, email, cin ..) de technicien connecté .	Moyenne	Élevé
	L'affichage des données dans la même page.		
Consultation la liste des réclamations	L'interface pour consulter la liste des réclamations	Facile	
Traiter une réclamation	Une button pour faire le traitement les réclamations	Facile	
La gestion de profil pour le psychologue	Intégration d'un formulaire pour la modification des coordonnées (nom, email, cin ..) de psychologue connecté .	Moyenne	
	L'affichage des données dans la même page.		
Consultation la liste des rendez-vous	L'interface pour consulter la liste des rendez-vous	Facile	
Accepter/annuler un rendez-vous	Deux buttons pour faire accepter ou bien annuler les rendez-vous	Facile	
La gestion de profil pour le responsable des clubs	Intégration d'un formulaire pour la modification des coordonnées (nom, email, cin....) de responsable des clubs connecté. L'affichage des données dans la même page.	Moyenne	
Gestion des évènements	L'interface pour consulter les événements et suppression Intégration d'un formulaire pour ajouter ou bien modifier des événements.	Facile	
Gestion des clubs	L'interface pour consulter les club et suppression L'interface pour consulter la liste de demander d'adhésion aux clubs Intégration d'un formulaire pour ajouter ou bien modifier des clubs.	Moyenne	

Tableau 10: Backlog sprint 3

3) Diagramme de séquence « Gestion de profil »

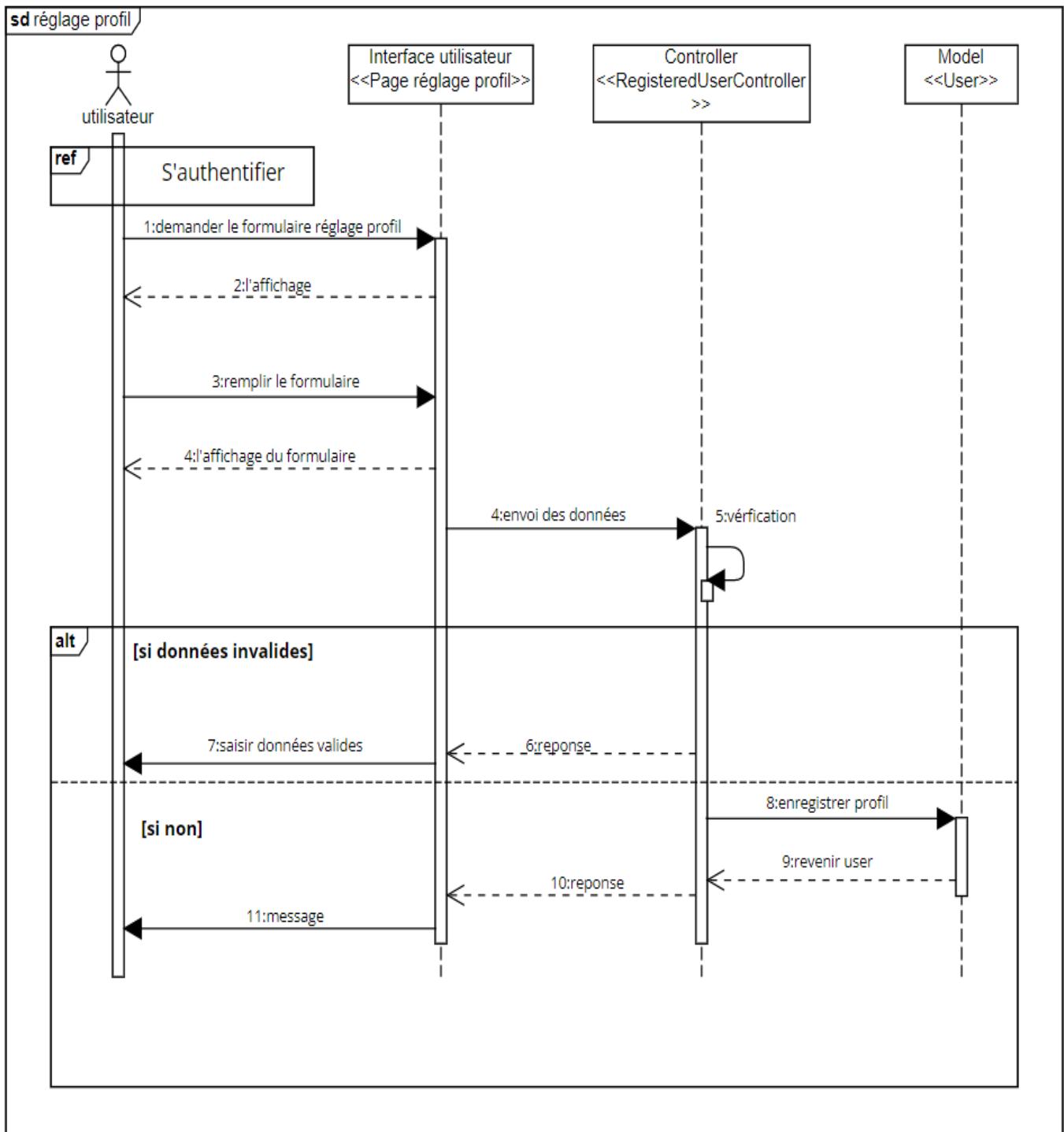


Figure 17: diagramme de séquence « réglage profil »

4) Diagramme de séquence « Traiter les réclamations »

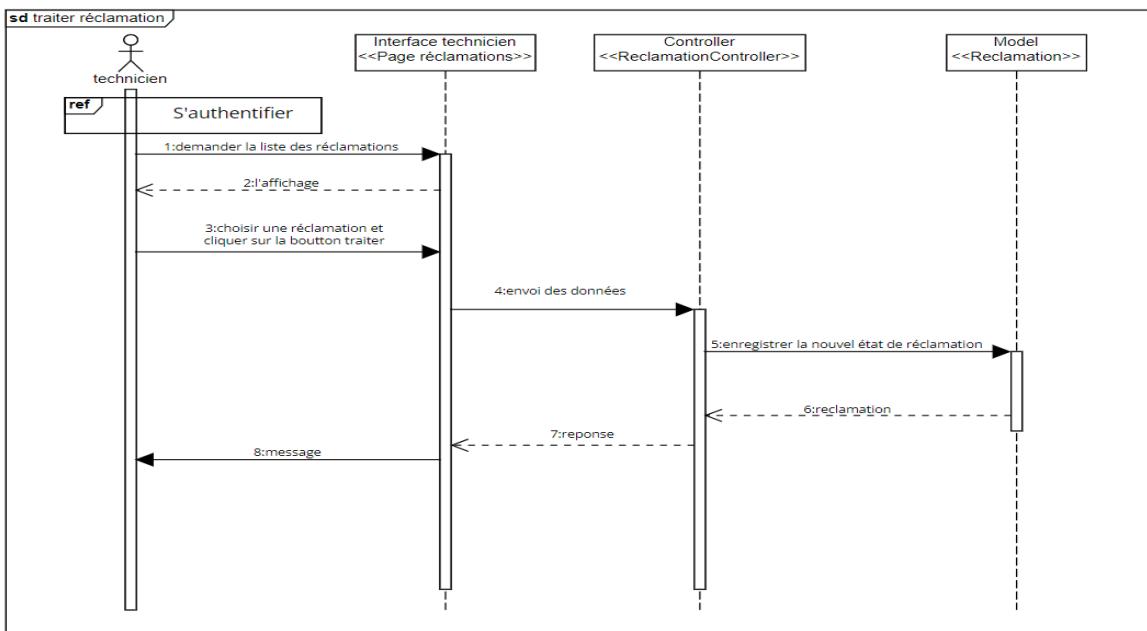


Figure 18: diagramme de séquence « traiter réclamation »

5) Diagramme de séquence « Ajouter des évènements »

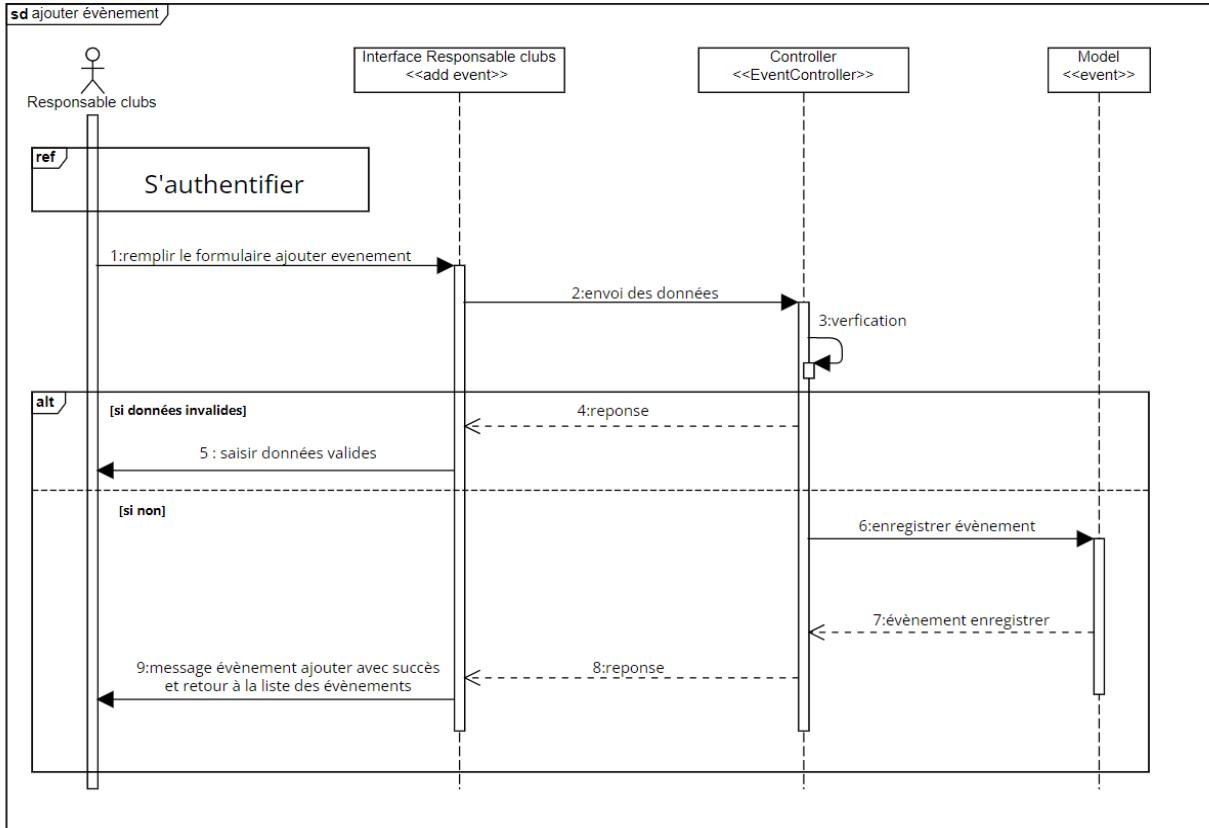


Figure 19: diagramme de séquence « ajouter évènement »

V. Conception du sprint 4

En partant sur le même principe que les sprints précédents, nous commençons par définir le but du sprint.

1) Sprint Goal

Le but de ce sprint est de commencer la partie concernant les étudiants et leurs tâches comme la gestion de profil, réservation de chambre.

2) Sprint Backlog

Le tableau ci-dessous résume le Backlog de notre quatrième sprint.

User story	Tâches	Degré de difficulté	Priorité	
Interface publique	Intégration des pages de l'application web Création d'un template Bootstrap	Facile	Élevé	
La gestion de profil pour l'étudiant	Intégration d'un formulaire pour la modification des coordonnées (nom, email, cin ..) de l'étudiant connecté .	Moyenne		
	L'affichage des données dans la même page.			
Consultation et la réservation de chambre	L'interface pour consulter les secteurs et les chambres disponibles Intégration d'un formulaire pour le passage d'une réservation avec les données de l'étudiant	Moyenne		

Tableau 11: Backlog sprint 4

a) Diagramme de séquence « Créer compte étudiant »

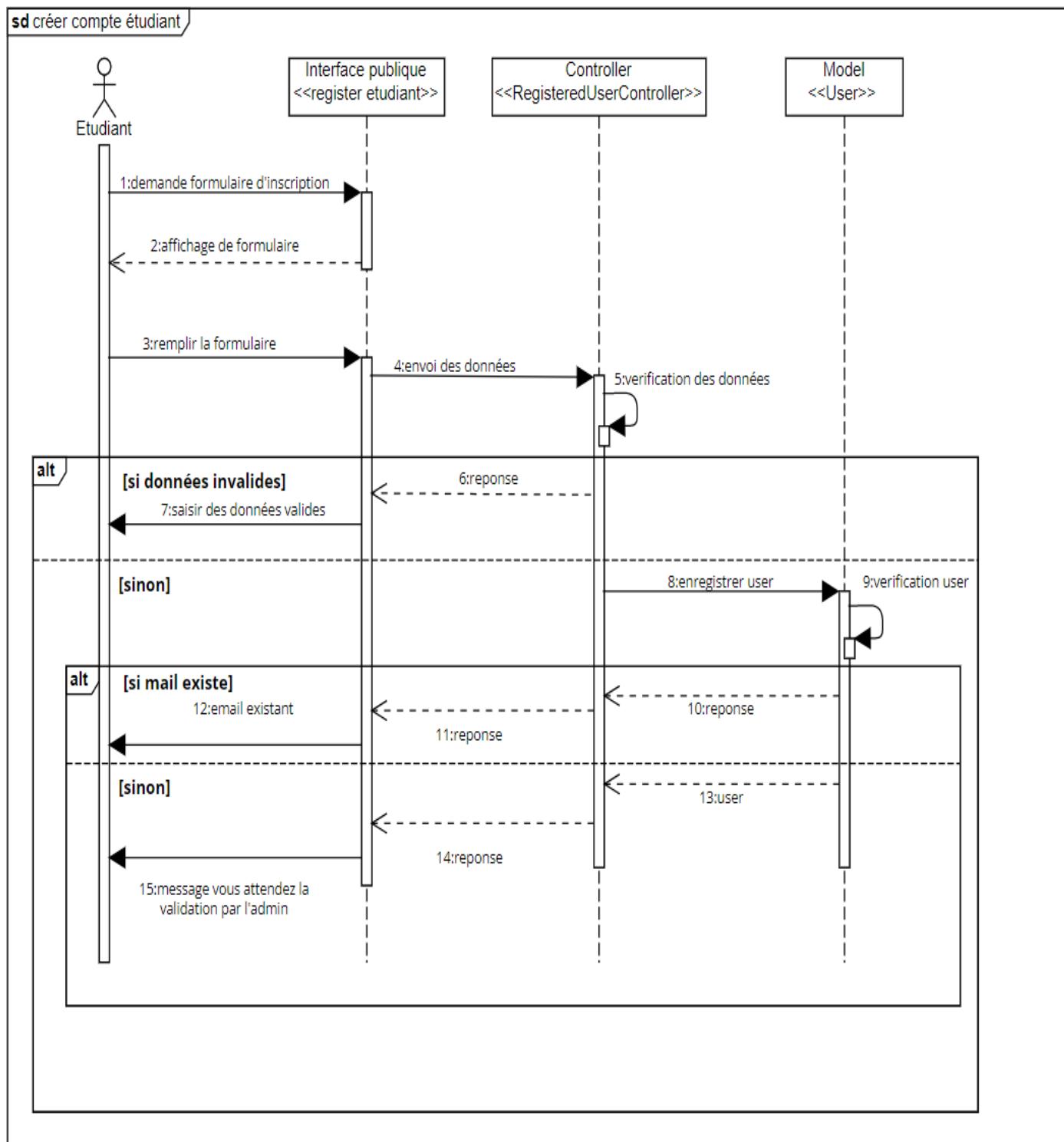


Figure 20: diagramme de séquence « créer compte étudiant »

VI. Conception du sprint 5

En partant sur le même principe que les sprints précédents, nous commençons par définir le but du sprint.

a) Sprint Goal

Le but de ce sprint est de commencer la partie concernant les étudiants et leurs tâches comme la gestion de profil, réservation de chambre, joindre le club, participation à les évènements, passage de réclamation, prend d'un rendez-vous avec le psychologue et en même temps la partie de consultation des historiques.

2) Sprint Backlog

Le tableau ci-dessous résume le Backlog de notre cinquième sprint.

User story	Tâches	Degré de difficulté	Priorité
Interface authentification et inscription	L'interface pour que l'étudiant faire une inscription et se connecter.	Moyenne	
La gestion de profil pour l'étudiant	Intégration d'un formulaire pour la modification des coordonnées (nom, email, cin ..) de l'étudiant connecté .	Moyenne	
	L'affichage des données dans la même page.		
Consultation et la réservation de chambre	L'interface pour consulter les secteurs et les chambres disponibles Intégration d'un formulaire pour le passage d'une réservation avec les données de l'étudiant	Moyenne	
Consultation et l'adhéissions du club	L'interface pour la consultation des clubs. Intégration d'un formulaire pour joindre les clubs	Moyenne	
Consultation des évènements et des actualités et participer aux évènement	L'interface pour la consultation des évènements et des actualités. Intégration d'un formulaire pour la participation aux évènements	Moyenne	Elevé
Gérer les rendez-vous	L'interface pour consulter les rendez-vous. Intégration d'un formulaire pour prendre les rendez-vous.	Moyenne	

Gérer les réclamations	L'interface pour consulter les réclamations. Intégration d'un formulaire pour passer les réclamations.	Moyenne	
------------------------	---	---------	--

Tableau 12: Backlog sprint 5

a) Diagramme de séquence « Passer réservation »

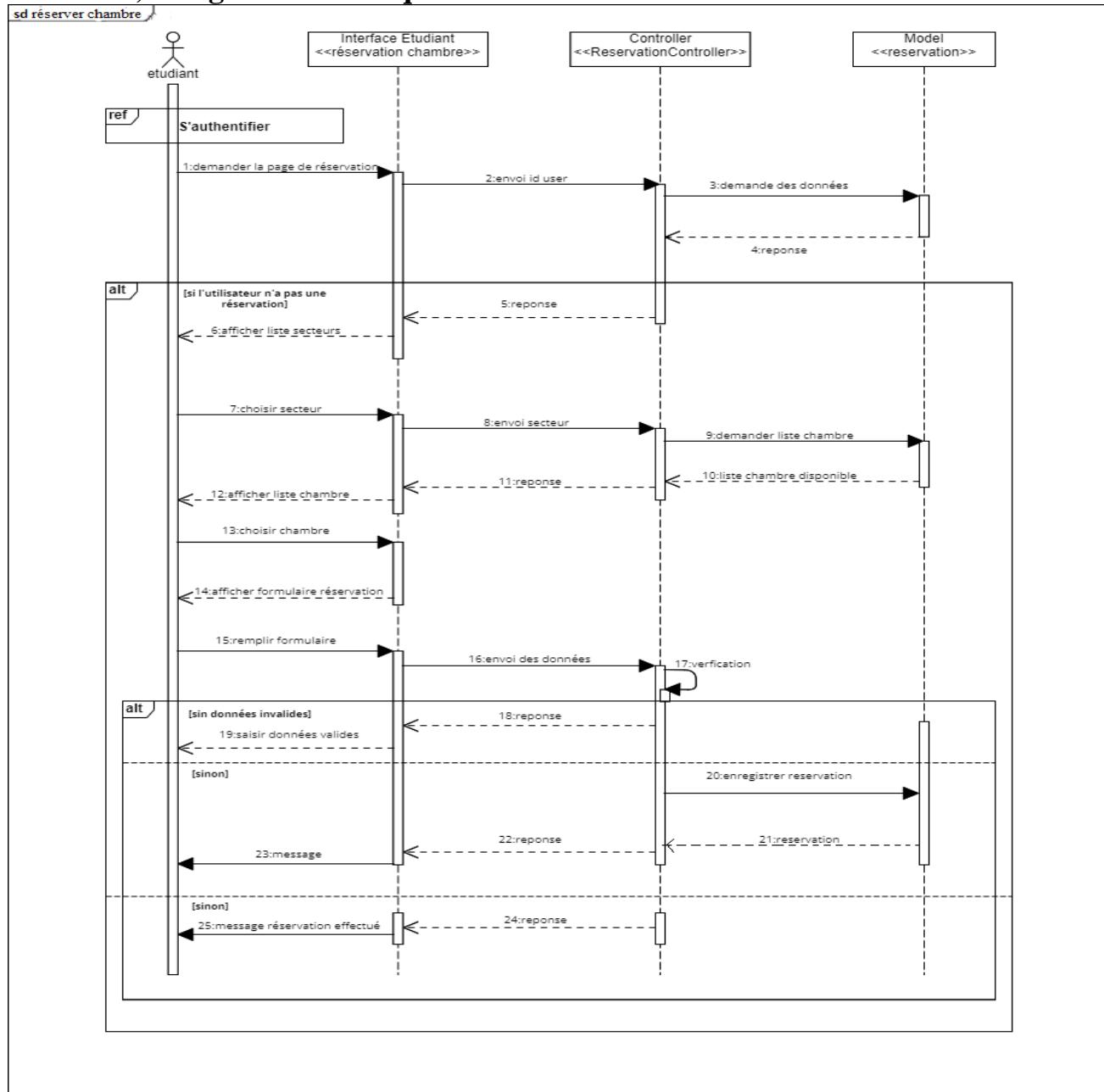


Figure 21: diagramme de séquence « réserver chambre »

b) Diagramme de séquence « Passer réclamation »

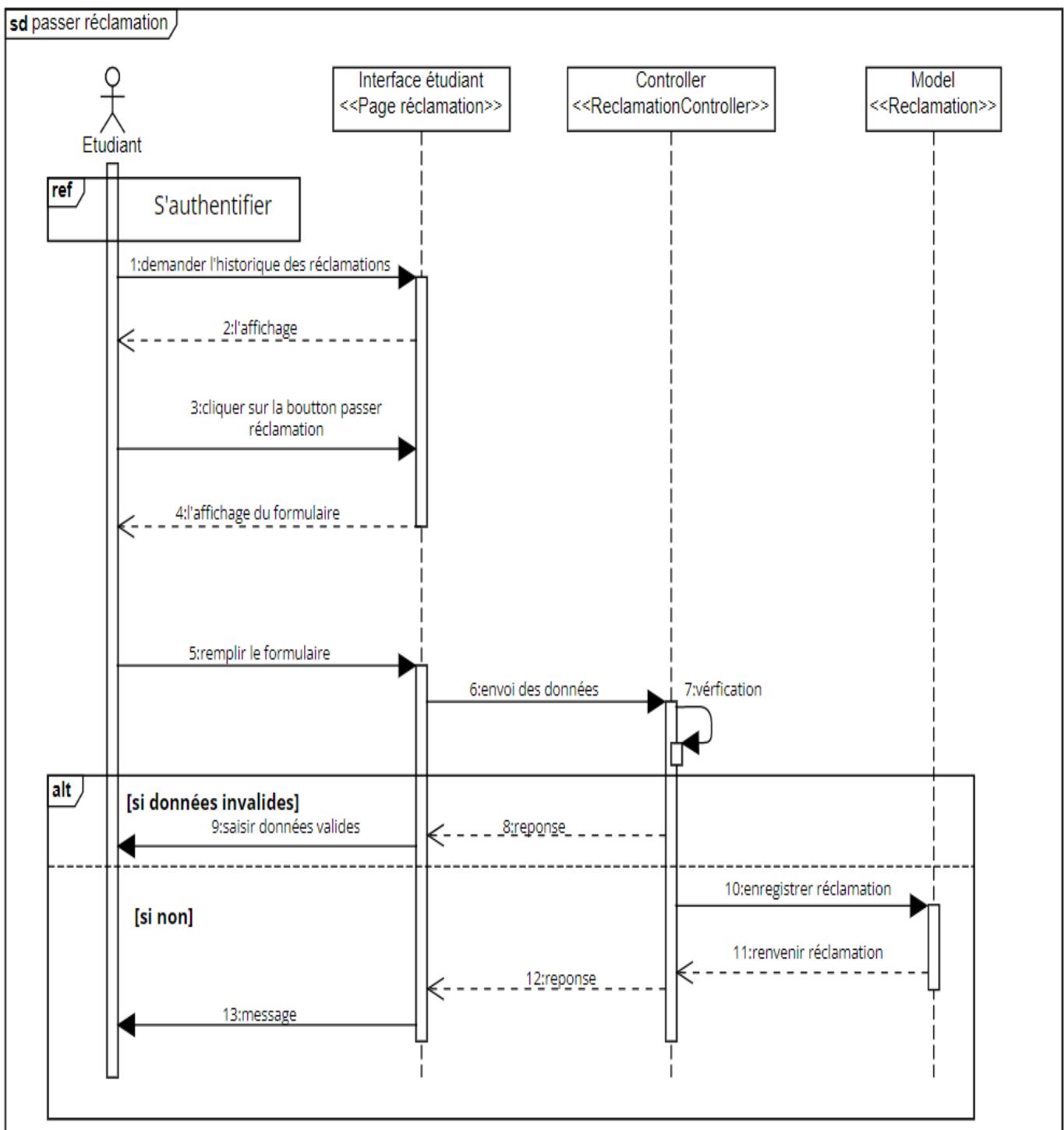


Figure 22: diagramme de séquence « passer réclamation »

VII. Diagramme de classes

1) Diagramme de classes

Le diagramme de classes est un schéma utilisé en génie logiciel pour présenter les classes et les interfaces des systèmes ainsi que leurs relations. Ce diagramme fait partie de la partie statique d'UML, ne s'intéressant pas aux aspects temporels et dynamiques.

La figure suivante représente le diagramme de classes de notre application :

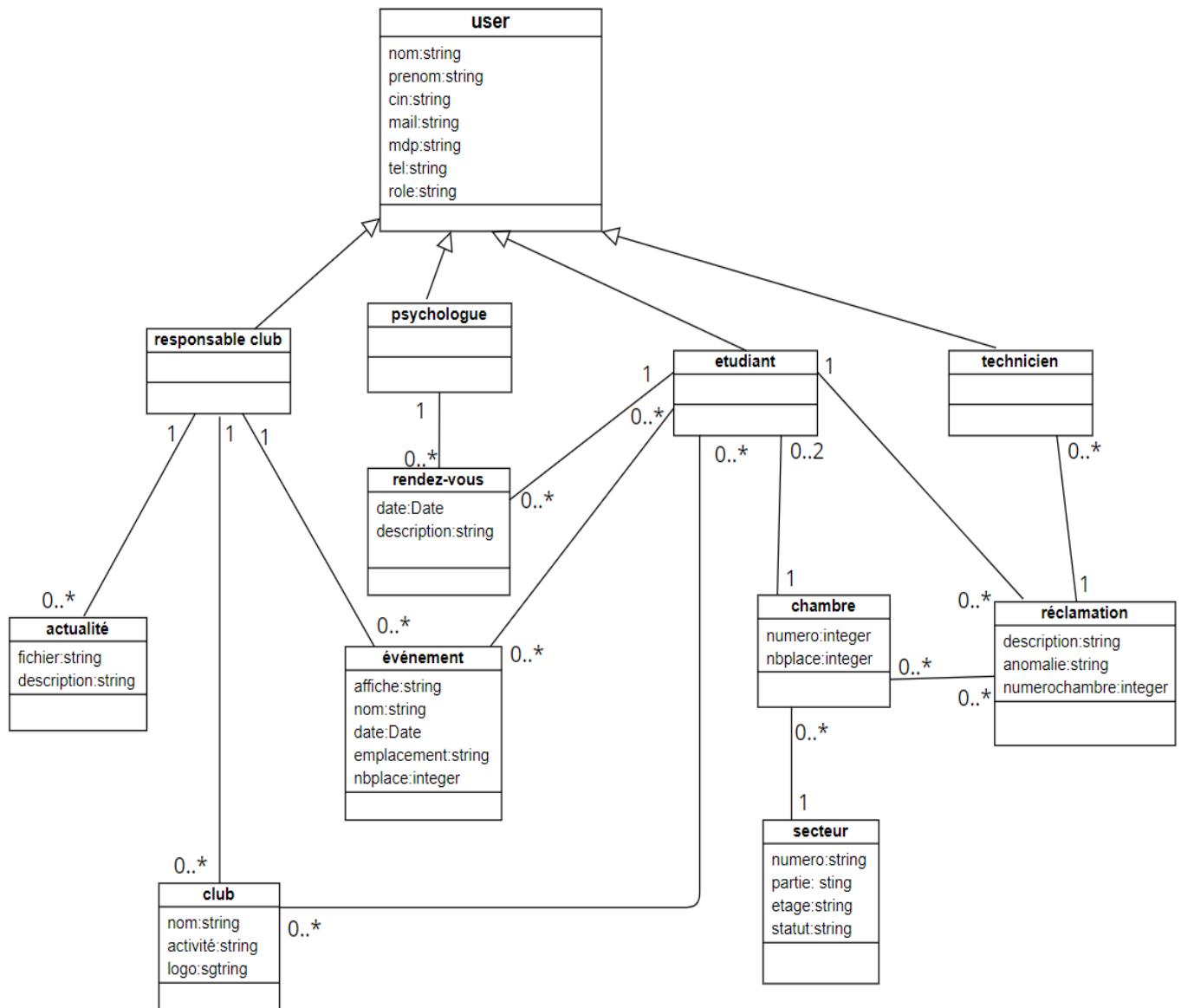


Figure 23: diagramme de classes

2) Correspondance entre modèles objets et relationnels

Users (id, name, email, password, telephone, adresse, cin, role, validation, created_at, updated_at);

Actualites (id, fichier, descrirption, created_at, updated_at);

Events (id, nom, affiche, emplacement, date, nbplace, created_at, updated_at);

Clubs (id, nom, logo, activite, image, created_at, updated_at);

Reservations (id, numero, partie, name, bac, binome, etat, #chamber_id, #user_id, created_at, updated_at);

Reclamations (id, name, numerochambre, anomalie, description, created_at, updated_at);

Rendez-vouses(id, date, description, created_at, updated_at,);

Secteurs (id, numerosecteur, partie, etage, statut created_at, updated_at);

Chambres (id, numero, nbplace created_at, updated_at, #secteur_id);

Joindreclub (id, confirmation, #user_id, #club_id);

Participationevents (id, niveau, #event_id, #user_id);

Conclusion

Ce chapitre a été consacré à différentes étapes de la conception détaillée. En effet nous avons eu recours à différents diagrammes de conception diagramme de séquence et de classe afin de montrer les interactions entre les acteurs et le système et présenté la structure de notre application.

Cette étape nous prépare la phase de développement et de réalisation qui sera l'objectif de prochain chapitre.

Chapitre 4 : Réalisation de l'application web et mobile

Introduction

Nous abordons dans ce chapitre le dernier volet de ce rapport, qui a pour objectif d'exposer la phase de réalisation. La phase de réalisation est considérée comme étant la concrétisation finale de toute la méthode de conception.

Ce chapitre, nous présentons les techniques utilisées, l'environnement logiciel de développement ainsi que les étapes de réalisation de notre solution en nous basant sur quelques captures d'écrans.

I. Choix Techniques

1) Les outils de développement



PHP (officiellement, ce sigle est un acronyme récursif pour PHP Hypertext Preprocessor) est un langage de scripts généraliste et Open Source, spécialement conçu pour le développement d'applications web. Il peut être intégré facilement au HTML. [7]

JavaScript



JavaScript est un langage de programmation qui permet d'implémenter des mécanismes complexes sur une page web. [8]



Les feuilles de styles (en anglais "Cascading Style Sheets", abrégé CSS) sont un langage qui permet de gérer la présentation d'une page Web. Le langage CSS est une recommandation du World Wide Web Consortium (W3C), au même titre que HTML ou XML.[9]

HTML



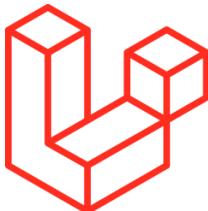
Signifie « HyperText Markup Language » qu'on peut traduire par « langage de balises pour l'hypertexte ». Il est utilisé afin de créer et de représenter le contenu d'une page web et sa structure. [10]

2) Les Frameworks

a) Application Web

Dans la partie web nous avons utilisées LARAVEL et BOOTSTRAP comme des Framework pour la développement l'application.

❖ Laravel



Laravel est un puissant Framework PHP basée sur le modèle de conception MVC, conçu pour les développeurs qui ont besoin d'une boîte à outils simple et élégante pour créer des applications web complètes. [11]

⊕ Nous avons choisi la Framework Laravel pour plusieurs avantages :

➤ Bibliothèques orientées objet

Laravel est l'un des meilleurs Framework PHP car il possède des bibliothèques orientées objet et d'autres pré installées, qui ne se trouvent dans aucun autre framework PHP.

➤ Documentation et communauté

Laravel possède une puissante communauté de développeurs qui fournit en permanence une assistance pour la rendre plus flexible et évolutive. Ainsi, si vous souhaitez apporter des fonctionnalités complexes, de nombreuses documentations sont à votre disposition. Tests unitaires avec les tests unitaires de Laravel, chaque module de votre application Web est testé avant la mise en ligne du site.

➤ **Artisan**

Laravel fournit un outil de ligne de commande intégré appelé Artisan. Cet outil aide à créer une architecture de code « squelette » et de base de données, ainsi que leurs migrations. La gestion de la base de données en devient plus facile. L'outil Artisan permet d'effectuer presque toutes les tâches de programmation répétitives et fastidieuses.

❖ **Bootstrap**



Bootstrap est une collection d'outils permettant de créer des designs sur des sites et des applications Web. [12]

b) Application Mobile

Dans la partie mobile nous avons utilisées l'IONIC comme un Framework pour la développement l'application.

❖ **Ionic**



Ionic Framework est une boîte à outils d'interface utilisateur open source permettant de créer des applications mobiles, des applications de bureau et des applications Web progressives performantes et de haute qualité à l'aide de technologies Web telles que HTML, CSS et JavaScript. Il permet aux développeurs de créer une seule fois et de s'exécuter partout. [13]

✚ Nous avons choisi la Framework Ionic pour plusieurs avantages :

➤ **Facile à apprendre**

Si le développeur a une connaissance de base des frameworks CSS, HTML ou JavaScript, il est très facile d'apprendre et de développer des applications à l'aide du Framework Ionic. Cela permet également aux sociétés de développement de passer à Ionic si leurs exigences et leurs besoins nécessitent le développement d'applications hybrides.

➤ **Documentation facile**

Ionic fournit une documentation très bonne et bien structurée. La documentation officielle couvrait la plupart des choses requises pour les développeurs.

➤ **Multiplateforme**

L'application ionique peut être déployée sur plusieurs plates-formes telles que iOS natif, Android, Desktop et le Web, le tout avec une seule base de code. Ces applications peuvent écrire une fois et s'exécutent partout.

❖ **VueJs**



Vue.JS, ou simplement Vue, est un Framework progressif pour les interfaces utilisateur pour les applications et sites JavaScript. Il s'agit d'un des frameworks front-end JS les plus populaires. [14]

c) **API RESTful**

Une API compatible REST, ou « RESTful », est une interface de programmation d'application qui fait appel à des requêtes HTTP pour obtenir (GET), placer (PUT), publier (POST) et supprimer (DELETE) des données. [15]

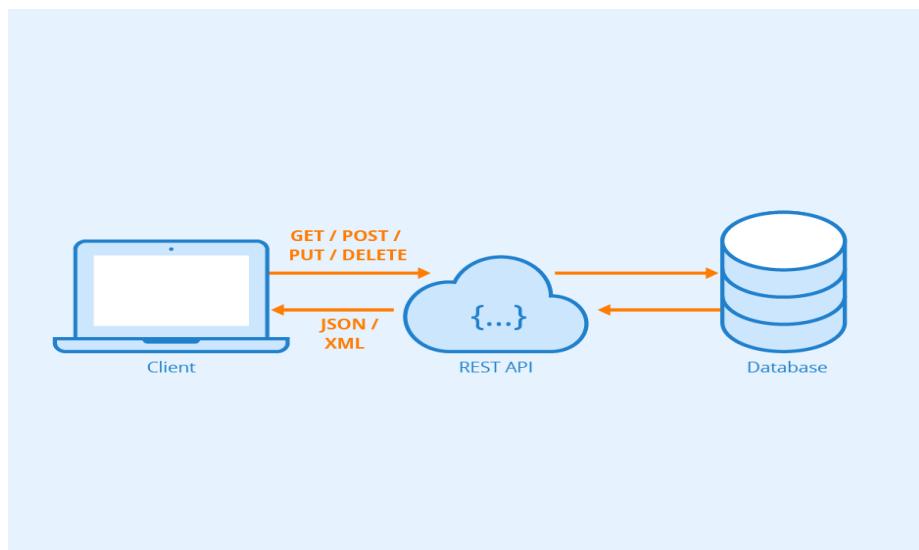


Figure 24: API RESTful

II. Environnement logiciel

Nom de Logiciel	Logo	Utilisation
Visual Studio Code		Visual Studio Code est un éditeur de code open-source développé par Microsoft supportant un très grand nombre de langages grâce à des extensions. Il supporte l'auto complétion, la coloration syntaxique, le débogage, et les commandes git. [16]
Laragon		Laragon est un environnement de développement universel portable, isolé, rapide et puissant pour PHP, Node.js, Python, Java, Go, Ruby. Il est rapide, léger, facile à utiliser et facile à étendre. [17]
Visual Paradigm		Visual Paradigm Online ("VP Online") est un outil de création de diagrammes basé sur le Web qui prend en charge un grand nombre de diagrammes commerciaux et techniques. [18]
Heidi SQL		HeidiSQL est un outil d'administration de base de données possédant un éditeur SQL et un constructeur de requête. Il a été développé et optimisé pour être utilisé avec le SGBD relationnel MySQL disponible commercialement ou gratuitement. [19]
Postman		Postman est un logiciel qui permet d'appeler ou bien tester une API. [20]
Canva		Canva est un outil de design et de publication en ligne dont la mission est de permettre à chacun dans le monde de concevoir ce qu'il souhaite et de publier où il le souhaite. [21]
Word 2016		Le logiciel Word permet de faire un traitement de texte et aussi de réaliser un tableau et de la mettre en forme. [22]

Tableau 13: L'environnement logiciel

III. Design graphique

Le graphisme est une discipline qui consiste à créer, choisir et utiliser des éléments graphiques (dessins, caractères typographiques, photos, couleurs, etc.) pour élaborer un objet de communication et/ou de culture.

Un **logo** ou logotype est une composition figurée servant à identifier visuellement, de façon immédiate une entreprise, une marque, une association, une institution, un produit, un service, un événement ou toute autre sorte d'organisations dans le but de se faire connaître et reconnaître des publics et marchés auquel il s'adresse et de se différencier des autres entités d'un même secteur. Donc dans cette étape nous avons créé un logo pour l'application web et mobile qui symbolise MY ULYSSE (foyer universitaire Ulysse Djerba).



Figure 25: Logo my Ulysse

Dans notre cas, nous avons choisi ce logo qui comporte un petit prototype qui symbolise des chambres et leurs clés.

En tenant compte de l'authenticité de l'île de Djerba, nous avons choisi la couleur symbolique de l'île qui est la couleur bleue.

IV. Les interfaces

1) Application web (partie front)

Dans cette partie, nous illustrons quelques interfaces de notre applications Web partie front-end.

a) Interface d'accueil

L'interface d'accueil est une interface publique. A travers cette interface, un étudiant a la possibilité de visualiser les informations du foyer universitaire Ulysse Djerba, consulter les photos de la galerie et télécharger l'application mobile « My Ulysse ». Il peut s'inscrire pour pouvoir réserver une chambre et modifier son profil.

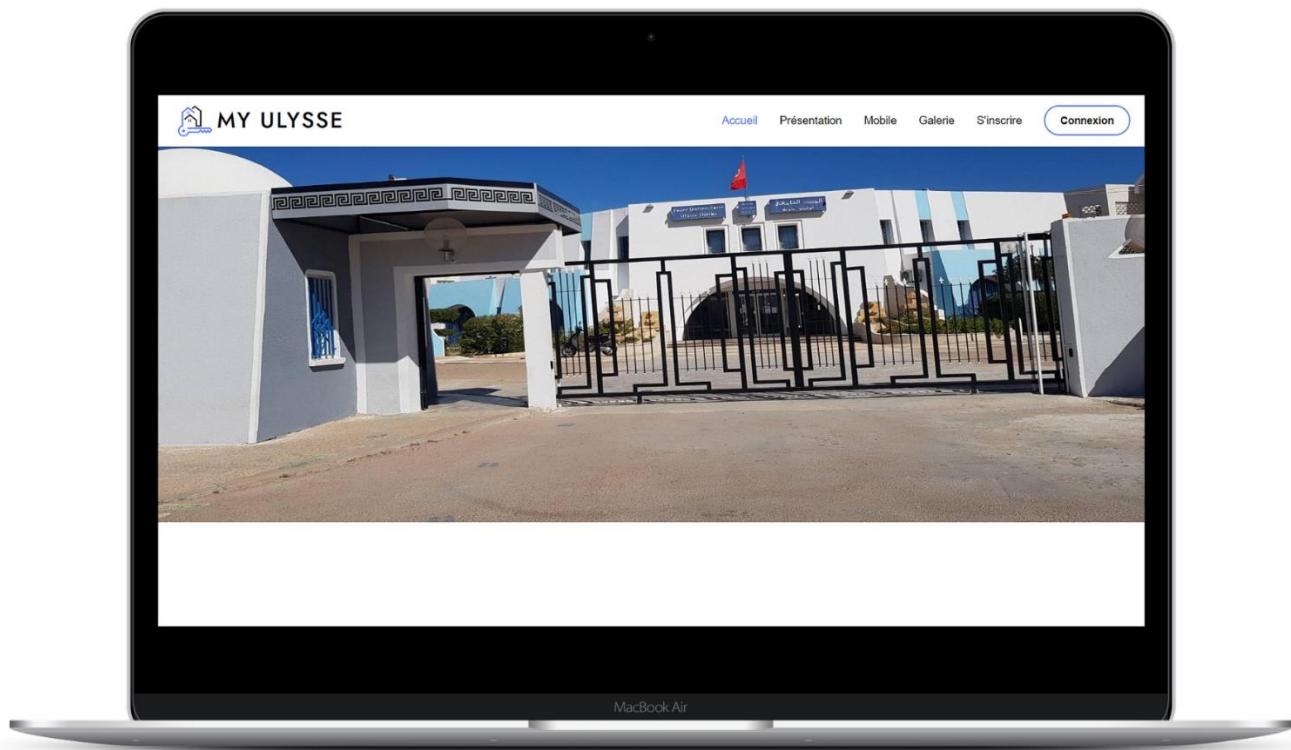


Figure 26: Interface d'accueil

b) Interface d'authentification

La figure suivante présente l'interface d'authentification permettant à chaque étudiant de se connecter.

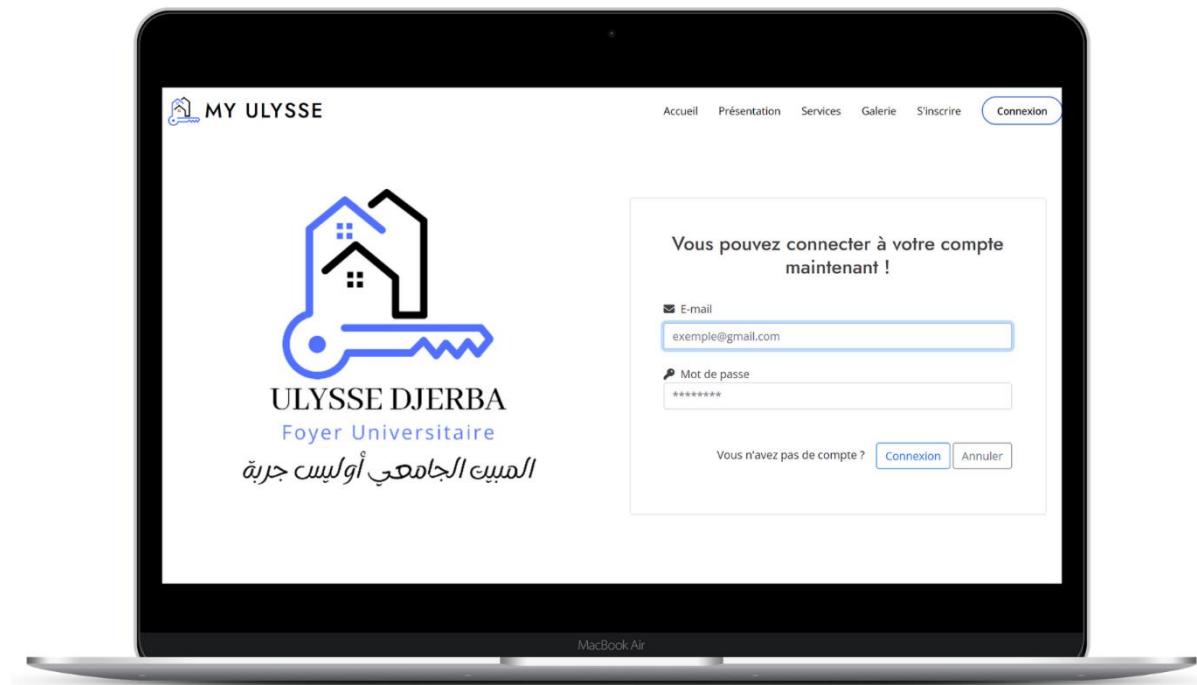


Figure 27: Page s'authentifier pour l'étudiant (web)

a) Interface d'inscription

La figure suivante présente l'interface d'inscription permettant à chaque étudiant d'inscription.

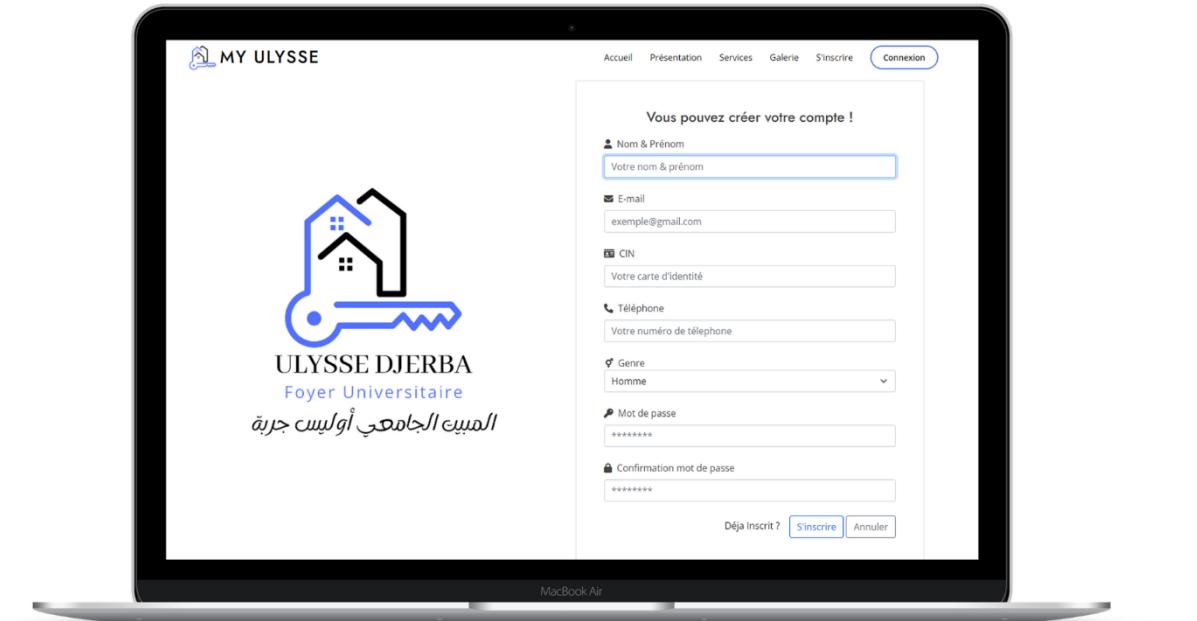


Figure 28: Page d'inscription pour l'étudiant (web)

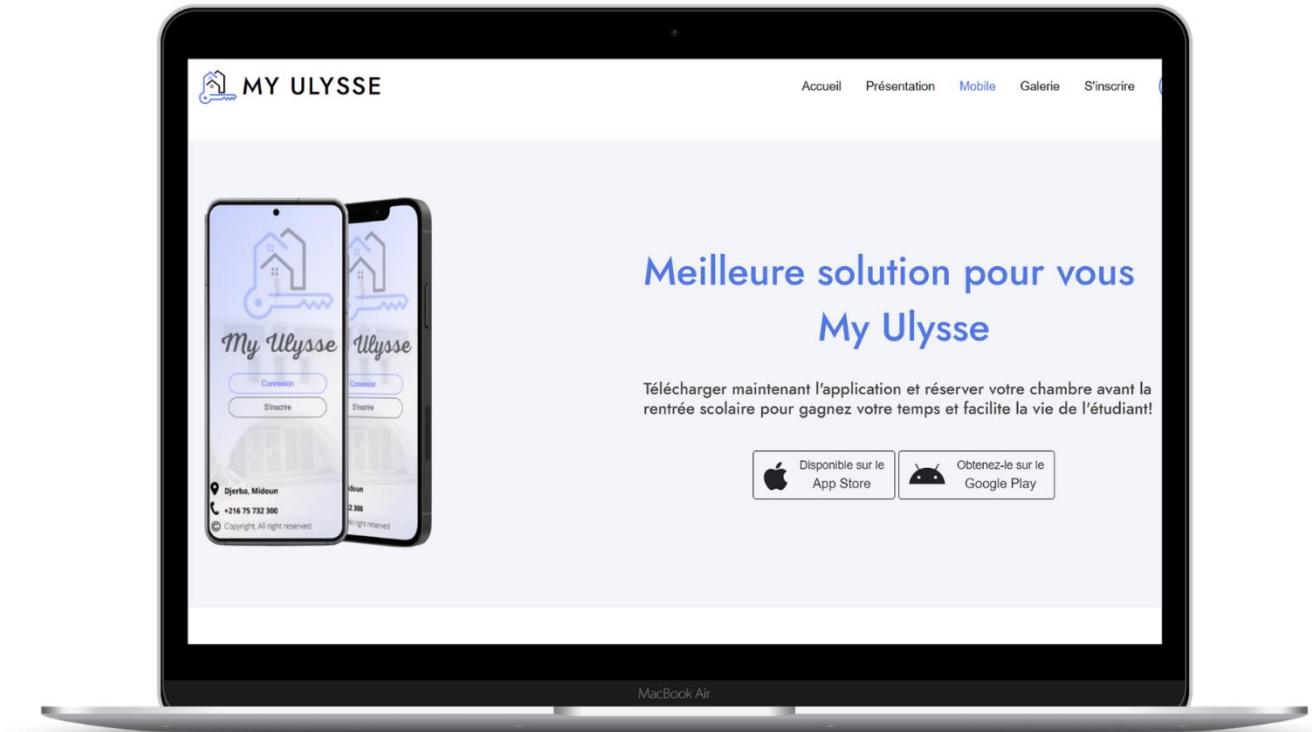
b) Interface de téléchargement de l'application

Figure 29: Page de téléchargement de l'application mobile

2) Application web (partie back)

Dans cette partie, nous illustrons quelques interfaces de notre application Web partie back-end.

a) Interface administrateur

La partie backOffice permet aux personnels du foyer universitaire (l'administrateur, le psychologue, responsable des clubs et le technicien) de s'authentifier et accéder aux différentes fonctionnalités de gestion de l'application « My Ulysse ». La première interface illustrée par la figure ci-dessous permet de d'accéder à la page d'authentification en cliquant sur le lien « connexion »



Figure 30: Page Welcome

La figure suivante présente l'interface d'authentification permettant à l'administrateur, le psychologue, responsable des clubs ou bien le technicien de se connecter.

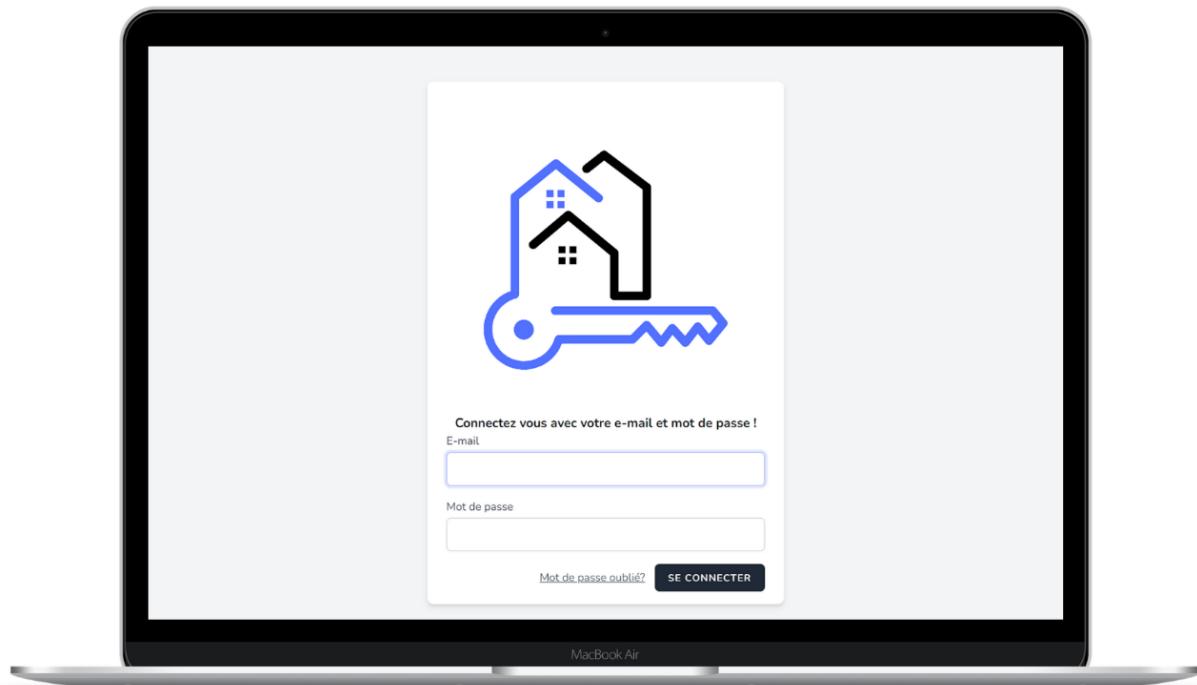
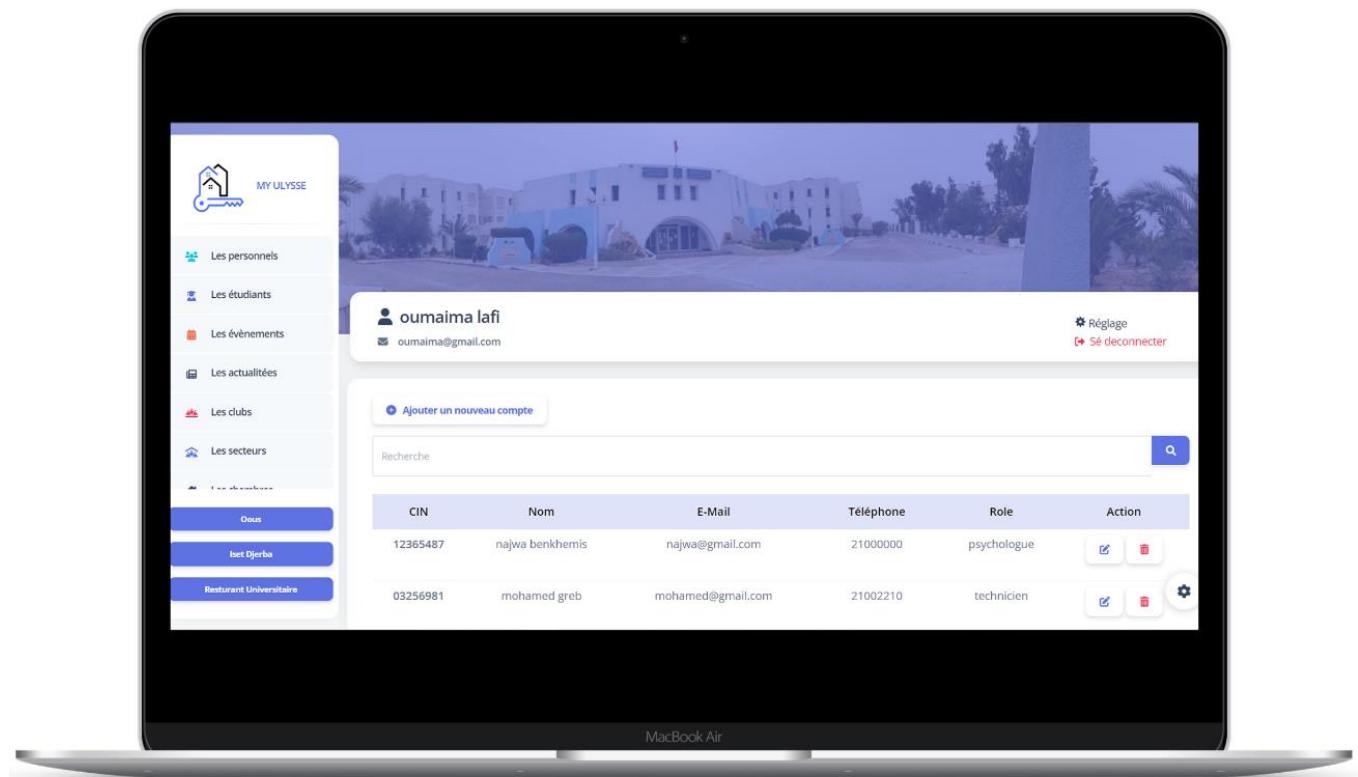


Figure 31: Page d'authentification

b) Interface liste des personnels

La figure suivante présente l'interface de liste des personnels permettant à l'administrateur d'ajouter un nouveau personnel, modifier ou bien supprimer.

**Figure 32: Liste personnels****c) Interface liste des secteurs**

La figure suivante présente l'interface liste des secteurs permettant à l'administrateur d'ajouter un nouveau secteur, modifier ou bien supprimer.

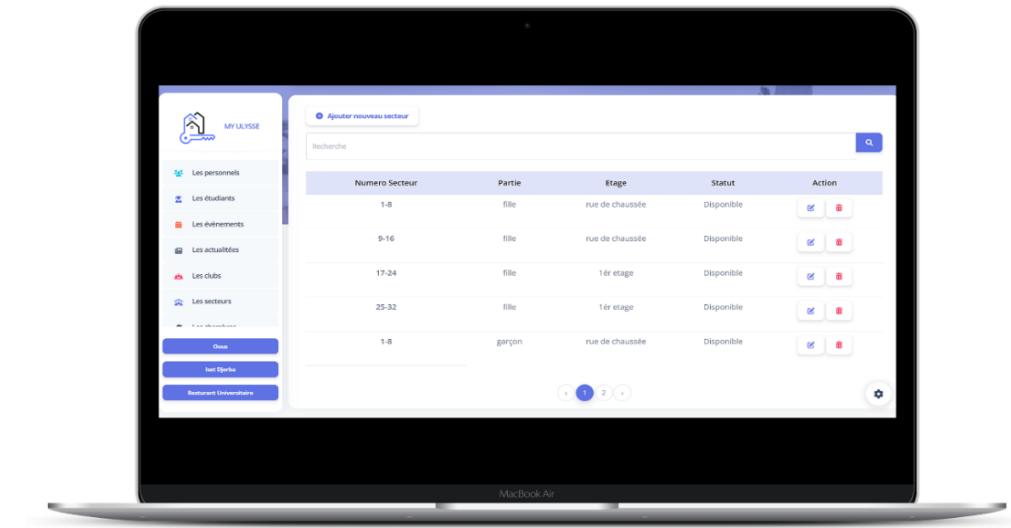


Figure 33: Liste des secteurs

d) Interface liste des évènements

La figure suivante présente l'interface liste des secteurs permettant à l'administrateur et le responsable des clubs d'ajouter un nouvel évènement, modifier ou bien supprimer.

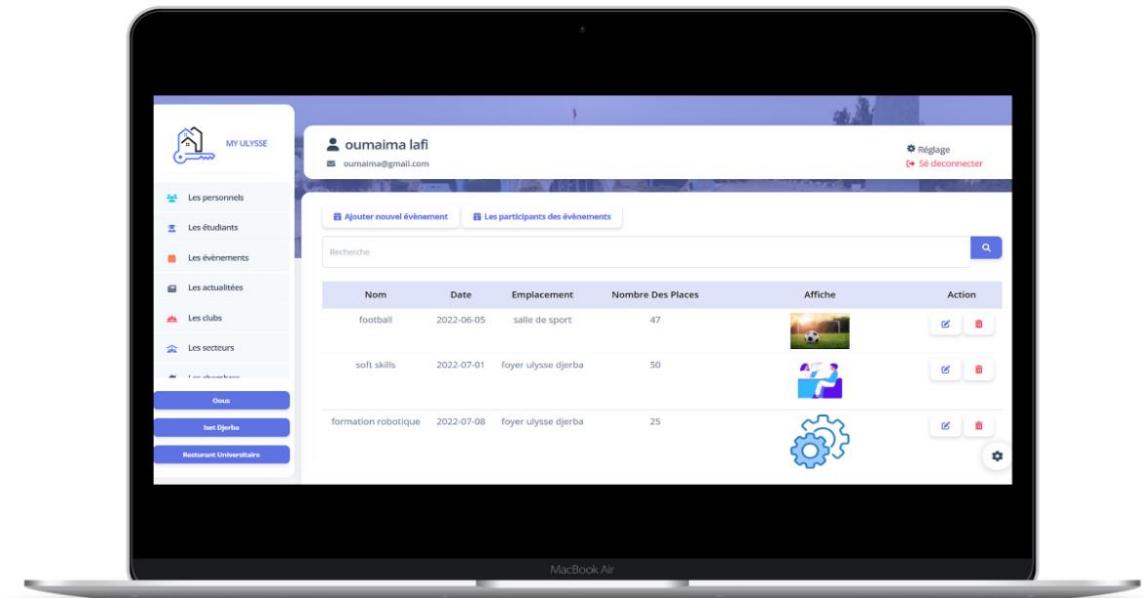


Figure 34: Liste des évènements

e) Interface liste des réclamations

La figure suivante présente l'interface liste des réclamations permettant au technicien de traiter une réclamation.

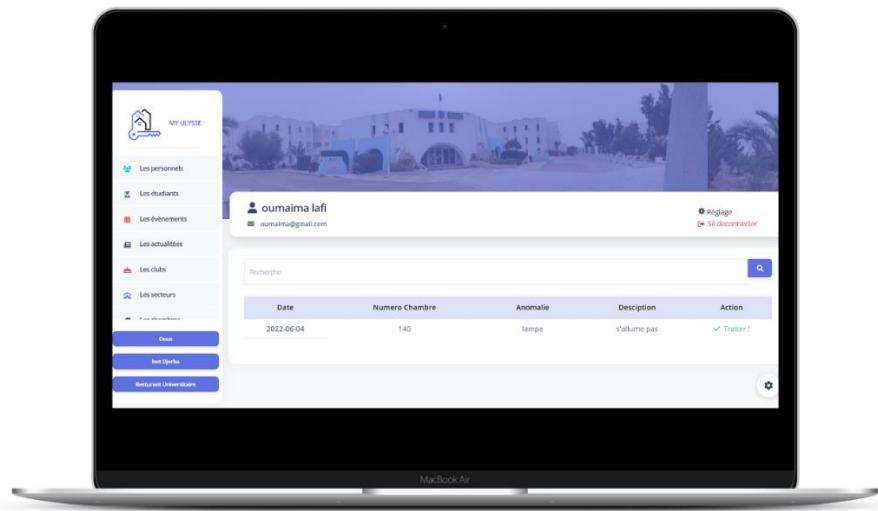


Figure 35: Liste des réclamations

f) Interface liste des actualités

La figure suivante présente l'interface liste des actualités permet à l'administrateur ou bien le responsable des clubs d'ajouter, modifier ou bien supprimer des actualités.

Ici nous avons montré le bouton d'ajout des nouvelles actualités avec le cercle rouge et la flèche.

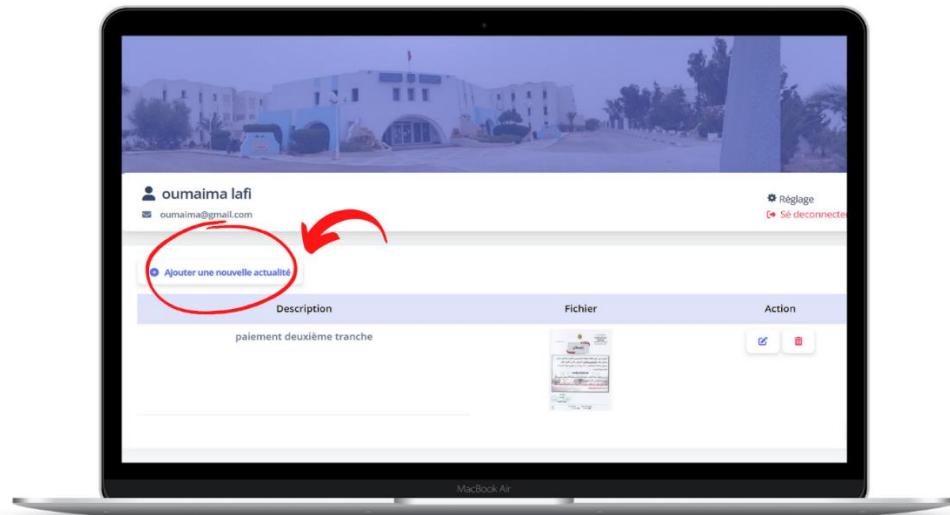


Figure 36: Interface liste des actualités

L'interface suivante montre la formulaire d'ajout des actualités et l'enregistrement de l'actualité ajoutée avec le bouton enregistrer comme le montre la figure.

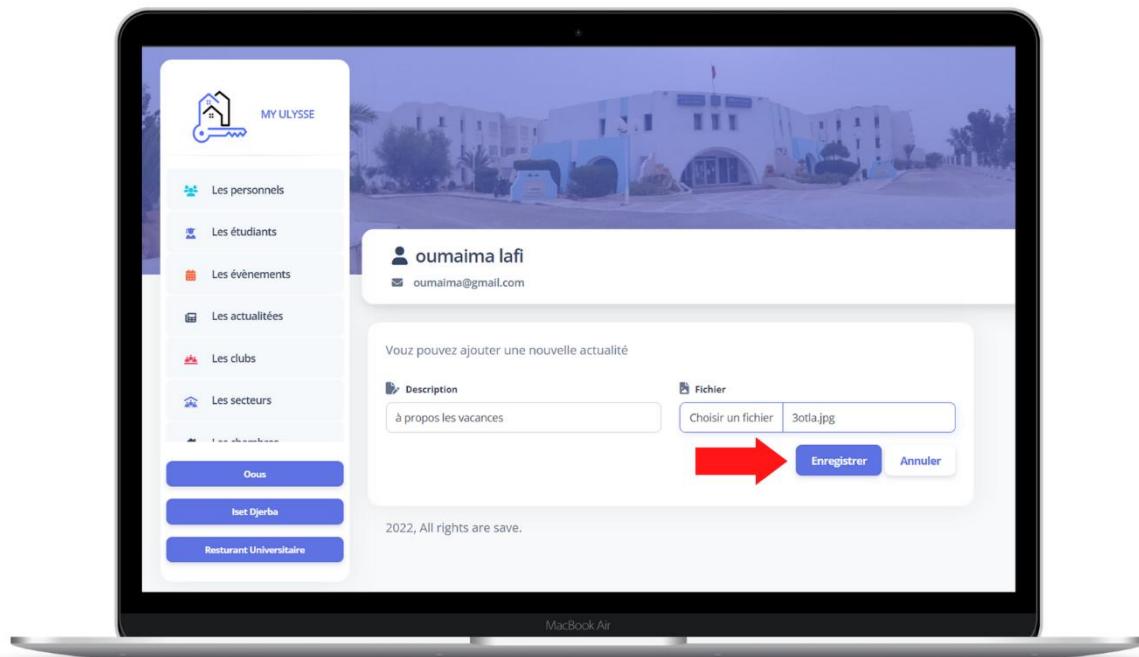


Figure 37:Interface d'ajout des actualités

Après l'ajout d'une actualités l'administrateur ou bien le responsable des clubs doit cliquer sur le bouton ok d'alerte pour confirmer l'ajout et revenir à la liste des actualités.

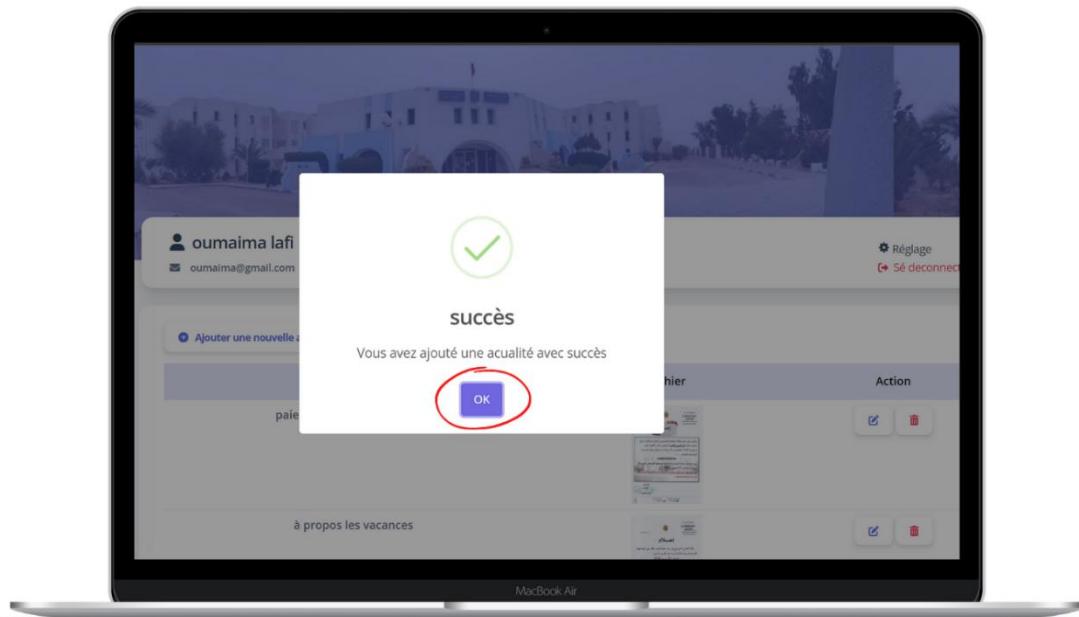


Figure 38:Intercae d'alerte d'ajout des actualités

g) Interface liste des étudiants

La figure suivante présente l'interface liste des étudiants permet à l'administrateur d'accepter ou bien refuser les demandes d'inscription des étudiants aussi de modifier ou bien supprimer les étudiants qui sont déjà inscrit.

Ici nous avons montré le bouton de suppression des étudiants avec le cercle rouge et la flèche.

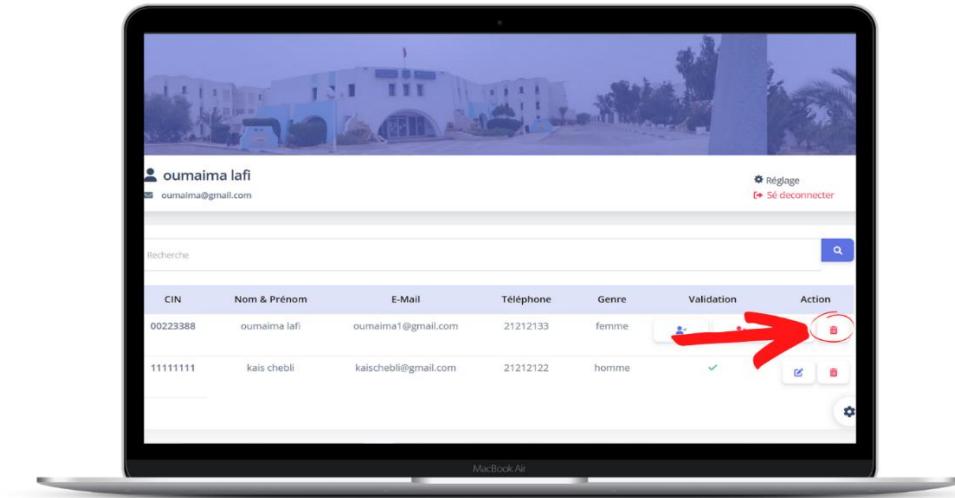


Figure 39:Interface liste des étudiants

Après la choisir de l'étudiant d'après l'administrateur pour lui supprimer, lorsqu'il clique sur le bouton de suppression une alerte est afficher pour confirmer la suppression ou bien l'annuler, dans le cas de la suppression une alerte de confirmation est affiché et dans l'autre cas l'administrateur revenir à la liste des étudiants.

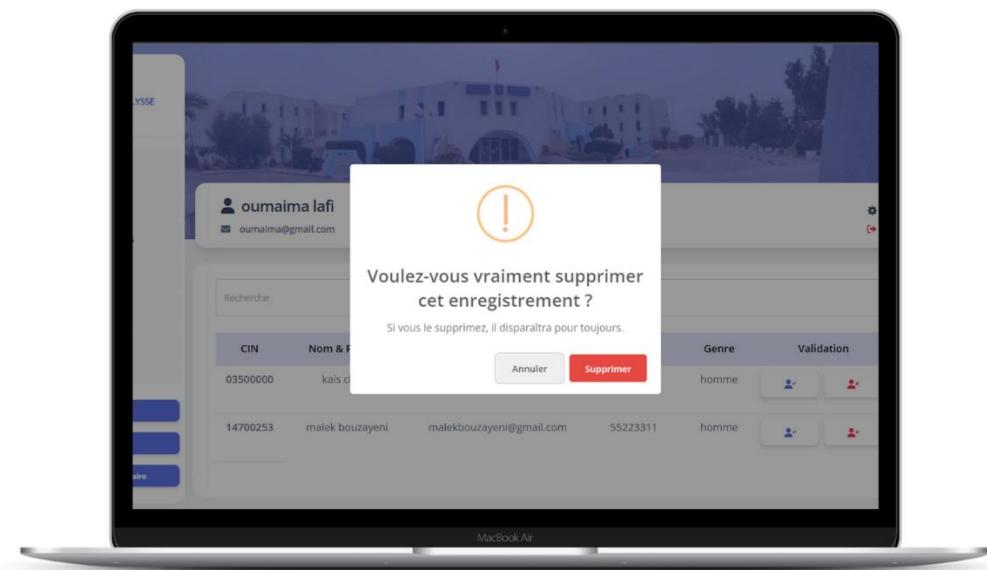


Figure 40: Interface alerte de suppression des étudiants

3) Application mobile

a) Interface publique

L'interface suivante montre la page publique de l'application mobile. Chaque étudiant peut faire une inscription et attendre la validation de l'administrateur pour pouvoir se connecter ou bien il peut se connecter s'il a déjà un compte.



Figure 41: Inetrface publique

b) Interface d'inscription et d'authentification

L'interface suivante montre la page d'inscription des étudiants. Chaque étudiant doit faire une inscription et attendre la validation de l'administrateur pour pouvoir se connecter.



Figure 42: Interface inscription des étudiants

Pour accéder à l'application mobile, l'étudiant doit se connecter avec son email et son mot de passe.



Figure 43: Interface connexion des étudiants

c) Interface réglage profil

Après l'authentification chaque étudiant peut consulter et modifier son compte.

Cette figure montre l'interface réglage compte.



Figure 44: Interface réglage compte

d) Interface menu

Une fois authentifié, l'étudiant peut accéder à l'application et peut visualiser l'ensemble des fonctionnalités de l'application à travers le menu suivant :

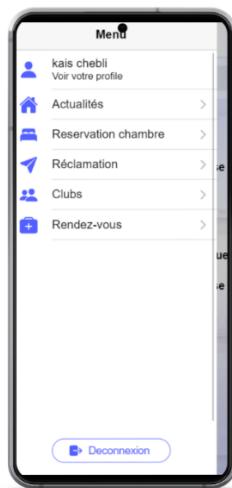


Figure 45: Interface menu

e) Les interfaces de réservation

Pour passer une réservation d'une chambre, l'étudiant choisit un secteur parmi les secteurs affichés. Après avoir choisi le secteur, l'application affiche la liste des chambres disponibles dans ce secteur. Le scénario de réservation est représenté par les figures suivantes :

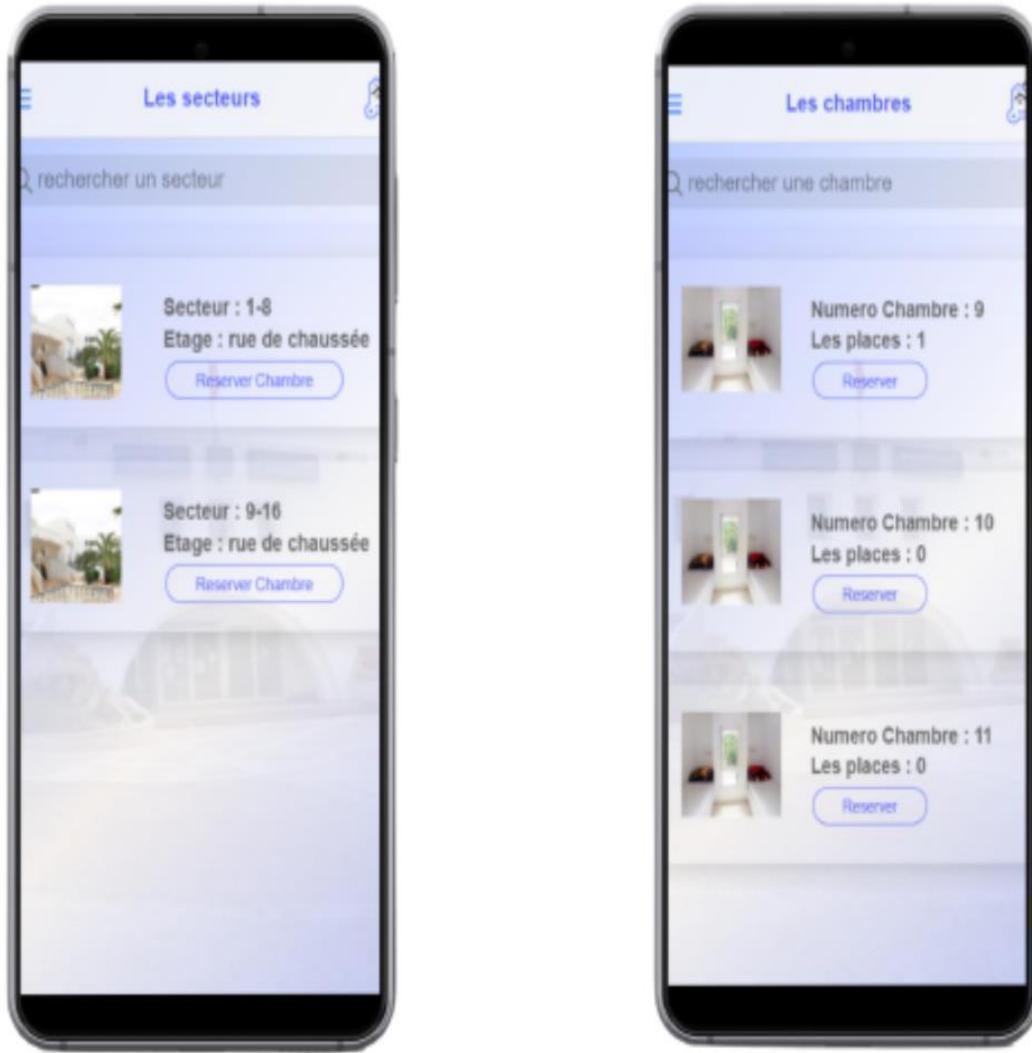


Figure 46: Interface liste secteurs et liste chambres

e) Interface prise rendez-vous avec le psychologue

Cette interface montre le formulaire pour prendre un rendez-vous avec le psychologue dans le foyer universitaire.



Figure 47: Interface prendre rendez-vous

f) Interface liste des clubs

Cette interface montre la liste des clubs existe dans le foyer universitaire et on a montré le bouton de joindre au club avec le cercle rouge et la flèche.

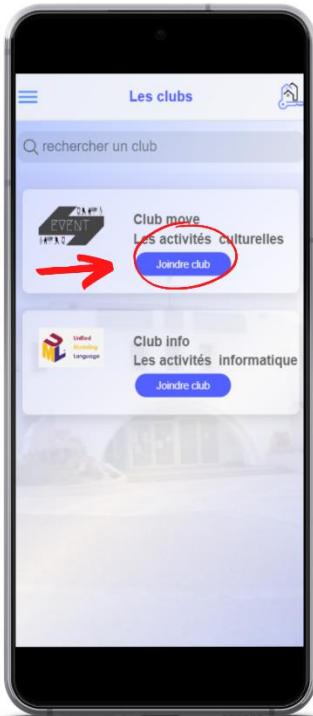


Figure 48 : Interface liste clubs

Cette interface montre la liste des clubs qui déjà l'étudiants joindre l'un des clubs existe après l'envoi d'un formulaire.

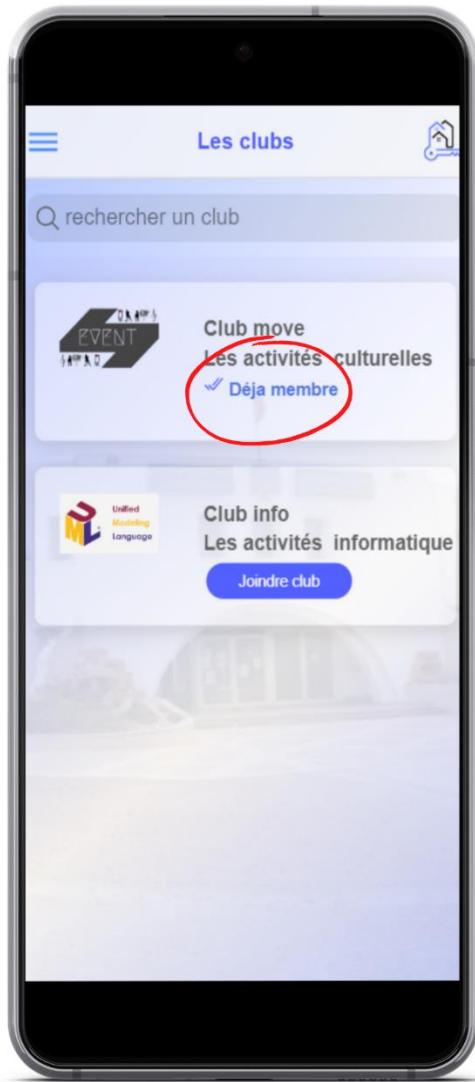


Figure 49: Interface liste clubs

Conclusion

Tout au long de ce chapitre, nous avons présenté l'environnement de travail logiciel avec lequel nous avons développé nos applications web et mobile. Ensuite, nous avons présenté les différentes interfaces des applications développées.

Conclusion générale

Les travaux présentés dans ce rapport s'inscrivent dans le cadre de notre projet de fin d'études intitulé «Développement d'une application web et mobile pour la gestion d'un foyer universitaire».

Le but de notre travail était de réaliser une application web et mobile qui facilite la vie des étudiants dans le foyer et assurer la gestion du foyer universitaire.

La réalisation de ce travail a passé par quatre phases de base. En premier lieu, nous avons présenté l'organisme d'accueil, le cadre de travail et l'étude de l'existant.

En deuxième lieu, nous avons effectué une étude qui nous a permis d'identifier les besoins de notre application. Ensuite nous avons entamé la partie conception. Enfin, nous avons détaillé la phase de réalisation en exposant le choix des technologies utilisés et en présentant quelques interfaces de notre projet.

Nous estimons que ce projet nous a été bénéfique sur tous les plans à savoir le plan technique et le plan humain. Sur le plan technique, nous avons appris à manipuler une partie front end et une partie back end avec la Framework Laravel et HeidiSQL pour la génération des tables et pleins d'autres outils, aussi on a manipulé l'application mobile avec la Framework Ionic et Vuejs.

En ce qui concerne le plan humain, nous avons eu la chance d'améliorer nos capacités de communication et de travail au sein d'une équipe.

En conclusion, nous souhaitons par la suite héberger les applications et ajouter un module de notifications dans l'application mobile. De plus, nous envisagerons d'ajouter un plan du foyer qui schématise les différents secteurs et chambres afin de faciliter la réservation pour les nouveaux étudiants.

Nétographie

- [1] <https://www.lucidchart.com/pages/fr/langage-uml>
- [2] <https://www.piloter.org/projet/methode/scrum.htm>
- [3] <http://www.mysti2d.net/henaffbagolet/ETC/C061/utilisation%20des%20repr%C3%A9sentations%20des%20syst%C3%A8mes/files/Utilisation%20des%20repr%C3%A9sentations%20de%20syst%C3%A8mes.html?Diagrammedecasutilisation.html>
- [4] <https://www.ibm.com/fr-fr/cloud/learn/three-tier-architecture>
- [5] <https://rosedienglab.defarsci.org/a-quoi-sert-une-architecture-mvc-son-fonctionnement/>
- [6] <https://klaxoon.com/communaute/sprint-backlog-lister-les-taches-a-realiser-pendant-le-sprint#:~:text=Le%20sprint%20backlog%20c'est,la%20base%20du%20projet%20Scrum>
- [7] <https://www.php.net/manual/fr/intro-whatis.php>
- [8] https://developer.mozilla.org/fr/docs/Learn/JavaScript/First_steps/What_is_JavaScript
- [9] <https://www.futura-sciences.com/tech/definitions/internet-css-4050/>
- [10] <https://developer.mozilla.org/fr/docs/Web/HTML>
- [11] <https://didaquest.org/wiki/Laravel>
- [12] <https://www.wizishop.fr/lexique-e-commerce/bootstrap>
- [13] <https://www.javatpoint.com/what-is-ionic-framework>
- [14] <https://www.zdnet.fr/lexique-it/vuejs-une-definition-39928683.htm>
- [15] <https://www.lemagit.fr/definition/API-RESTful>
- [16] <https://framalibre.org/content/visual-studio-code>
- [17] <https://laragon.org/docs/index.html>
- [18] <https://online.visual-paradigm.com/fr/about-us/press-kit/>
- [19] <https://www.encyclopedie.fr/definition/HeidiSQL>
- [20] <https://www.eewee.fr/postman-cest-quoi/>
- [21] https://www.canva.com/fr_fr/about/
- [22] https://elearn.univ-cen.dz/pluginfile.php/112805/mod_resource/content/1/Cours%20Dinformatique.pdf