



Ministère de l'Enseignement Supérieur, 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 d'obtention du diplôme de

**Mastère Professionnel en Technologies
de l'Information et de Communication
et Innovation Touristique**

Parcours :(TICIT)

Plateforme de gestion de l'Omra

Elaboré par :

Souabni Wael

Encadré par :

BLAGHGI Mejdi

Effectué à :

**Entreprise : ISRAA TRAVEL AGENCY
Encadreur : SMAALI SADDEM**

Année universitaire : 2021/2022

Dédicaces

Je dédie ce travail à :

Mes chers parents

Qui m'ont toujours soutenu et motivé dans mes études.

*Sans eux, certainement, je n'aurais pas pu arriver à ce
niveau.*

Mes chers frères et mon adorable sœur

Pour leurs grands amours et leurs soutiens.

Mes chers amis

Qui m'ont encouragé durant l'élaboration de ce travail.

À toute ma famille et à tous ceux que j'aime.

Remerciements

Au terme de mon projet de fin d'études, je tiens à remercier toutes les personnes qui, par leurs conseils, leurs suggestions, ou par leurs simples présences, j'ai pu rendre ce travail aussi fructueux et efficace que plaisant.

Je voudrais aussi remercier mon encadreur de l'ISET M. Mejdi BLAGHGI, pour son soutien et ses recommandations judicieuses.

Je tiens aussi à remercier toute l'équipe de l'agence et plus particulièrement M. Saddam Smaali pour leur encadrement.

Je dois chaque chose de ma connaissance à mes enseignants de l'ISET qui ont si bien mené leurs nobles tâches à nous enseigner les bases de l'informatique. Je les remercie non seulement pour le savoir qu'ils m'ont transmis, mais aussi pour la fierté et l'ambition que leurs personnes nous inspirent.

Enfin, mes sincères remerciements s'adressent à ma famille pour leur encouragement tout au long de mes études.

Sommaire

Introduction générale.....	1
Chapitre 1 Contexte général du projet	2
Introduction.....	2
1. Présentation du cadre du stage	2
2. L'organisation de L'agence	2
3. Etude de l'existant	4
3.1. Les objectifs du stage	4
3.2. Description de l'existant	4
3.3. Critique de l'existant	4
3.4. Solution proposée.....	5
4. Cahier de charges	5
4.1. Sujet.....	5
4.2. Objectifs du projet.....	5
4.3. Public cible.....	5
4.4. Spécifications de besoins	6
4.4.1. Identification des acteurs	6
4.4.2. Besoins fonctionnels.....	6
4.5. Outils de développement et de communication	7
Conclusion	7
Chapitre 2 Étude Conceptuelle	8
Introduction.....	8
1. Méthodologie de conception	8
2. Diagrammes de cas d'utilisation.....	8
2.1. Diagramme de cas d'utilisation «global»	9
2.2. Diagramme de cas d'utilisation «administrateur»	11
2.3. Diagramme de cas d'utilisation «pèlerin».....	13

2.4. Diagramme de cas d'utilisation «accompagnateur»	14
3. Diagrammes des séquences.....	15
3.1. Diagramme de séquence « Ajouter utilisateur»	16
3.2. Diagramme de séquence « valider pèlerin»	17
3.3. Diagramme de séquence « Ajouter pèlerin dans un package».....	18
4. Diagramme des classes	19
5. Diagramme de déploiement	20
Conclusion	21
Chapitre 3 Réalisation	22
Introduction.....	22
1. Choix technologiques	22
1.1. Vue JS.....	22
1.2. Laravel	23
1.3. Laravue	23
1.4. Flutter	23
2. Environnement de développement	24
2.1. Laragon.....	24
2.2. Visuel Code.....	24
2.3. PhpMyAdmin.....	24
2.4. Advanced REST Client (ARC).....	25
2.5. Git hub	25
3. Réalisation.....	25
3.1. Application Web	25
3.1.1. Authentification	25
3.1.2. Tableau de bord.....	26
3.1.3. Interface de Gestion Utilisateurs.....	27
3.1.4. Interface de gestion de package	28
3.1.5. Interface de gestion de pèlerins.....	29

3.1.6. Interface de gestion d'accompagnateur	30
3.1.7. Interface de gestion de message.....	31
3.1.8. Interface de gestion de paiement.....	32
3.2. Application Mobile Pèlerin.....	33
3.2.1. Authentification	34
3.2.2. Accueil.....	34
3.2.3. Interface de liste packages et ses détails.....	35
3.2.4. Interface de formulaire de participation à un package.	35
3.2.5. Interface de Historique de participation	36
3.2.6. Interface de Liste des accompagnateurs	36
3.2.7. Interface d'envoyer un message.....	37
3.3. Application Mobile Accompagnateur	37
3.3.1. Interface de détail package et ajouter un pèlerin	38
3.4. Api rest	39
Conclusion	39
Conclusion	40
Bibliographie & Néographie.....	41

Liste Des Figures

Figure 1 : Logo de Israa-Travel	2
Figure 2 : Organisation d'Israa Travel Agency Djerba	3
Figure 3 : Exemple d'acteur.....	9
Figure 4 : Exemple d'un cas d'utilisation	9
Figure 5 : Diagramme de cas d'utilisation global	10
Figure 6 : Diagramme de cas d'utilisation de l'administrateur	11
Figure 7 : Diagramme cas d'utilisation de pèlerin.....	13
Figure 8 : Diagrammes de cas d'utilisation accompagnateur.....	14
Figure 9 : Diagramme de séquence « Se authentifier»	15
Figure 10 : Diagramme de séquence « Ajouter utilisateur»	16
Figure 11 : Diagramme de séquence « Valider pèlerin»	17
Figure 12 : Diagramme de séquence « S'inscrire de pèlerin dans un package»	18
Figure 13 : Diagramme de classe	19
Figure 14 : Diagramme de déploiement	21
Figure 15 : Logo Vue Js.....	22
Figure 16 : Logo laravel	23
Figure 17 : Logo laravue.....	23
Figure 18: Logo flutter	23
Figure 19 : Logo laragon	24
Figure 20 : Logo Visual Studio code.....	24
Figure 21 : Logo phpMyAdmin	24
Figure 22 : Logo Advanced Rest Client	25
Figure 23 : Logo GitHub	25
Figure 24 : Interface d'authentification.....	26
Figure 25 : Interface de dashboard	26
Figure 26 : Interface de gestion utilisateur	27
Figure 27 : Interface d'Ajout d'utilisateur	27
Figure 28 : Interface de gestion de package.....	28
Figure 29 : Interface de détail de package	28
Figure 30 : Interface de gestion pèlerins.....	29
Figure 31 : Interface de modification pèlerin.....	29

Figure 32 : Interface gestion des accompagnateurs.....	30
Figure 33 : Interface de gestion relation accompagnateurs - packages	30
Figure 34 : Interface de Gestion de message	31
Figure 35 : Interface de réponse d'un message	31
Figure 36 : Interface de gestion de paiement	32
Figure 37 : Interface de validation champ de formulaire	32
Figure 38 : Affichage d'un toast de succès	33
Figure 39 : Splash screen.....	33
Figure 40 : Interface d'authentification	34
Figure 41 : Interface d'inscription.....	34
Figure 42 : Interface d'accueil.....	34
Figure 43 : Interface de Liste package de vols et ses details	35
Figure 44 : Formulaire d'inscription au package Omra	35
Figure 45: liste d'historique de participations	36
Figure 46 : Interface liste d'accompagnateur	36
Figure 47 : Interface profil et envoyer un message	37
Figure 48 : Interface d'authentification	37
Figure 49 : Interface d'inscription.....	37
Figure 50 : Interface de détail du package	38
Figure 51 : Interface d'ajout de pèlerin	38
Figure 52 : Interface de calendrier	38
Figure 53 : interface de liste de pelerins par vol	38

Liste Des Tableaux

Tableau 1 : Description du cas d'utilisation Gestion des packages	12
Tableau 2 : Description du cas d'utilisation inscrire à un package	13
Tableau 3 : Description du cas d'utilisation envoyer un message	14
Tableau 4 : Description de la classe « Compte Utilisateur »	20
Tableau 5 : Avantages et inconvénients Vue JS	22

Introduction générale

Vu la crise de Covid et son influence sur le secteur touristique, plusieurs organismes ont choisi de développer des solutions personnalisées à leurs clients. Parmi ces derniers, on trouve Israa-Travel Agency qui a choisi de développer son propre produit pour la gestion de l'Omra en se basant sur une proposition que j'avais fourni.

Ayant une passion pour le développement web et mobile, j'ai eu l'opportunité d'exploiter mes compétences pour soumettre l'ébauche de ce projet qui vise à trouver une solution logicielle pour ce problème.

Ce rapport, qui présente le fruit de mon travail, est organisé en chapitres comme suite :

- Le premier sera dédié au contexte général du stage. Je présente l'organisme d'accueil ainsi l'application et je décris le besoin fonctionnel de mon projet.
- Dans le deuxième chapitre j'aborde la phase de conception.
- Dans le troisième chapitre, je décris l'environnement de travail, les choix techniques et un aperçu sur les interfaces réalisées.

Ce rapport s'achève par une conclusion générale qui résume le travail effectué.

Chapitre 1 Contexte général du projet

Introduction

Dans ce chapitre, je présente l'organisme d'accueil, et par la suite, je décris les objectifs de mon stage de projet de fin d'études.

1. Présentation du cadre du stage

J'ai effectué mon projet de fin d'études dans l'agence de voyage «Israa Travel Agency». Elle a été créée en 2005 à l'Avenue Taieb Mhiri DOUZ. Elle met à la disposition de ses clients deux autres points de vente : Rue la liberté 1008 au centre-ville de Tunis, et à Djerba Midoun Av Farhat Hached.



Figure 1 : Logo de Israa-Travel

2. L'organisation de L'agence

Elle se présente comme suit :

- Le gérant qui a un but de diriger la société et prend la plupart des décisions qui concernent l'agence.
- Le Direction se divise en deux branches :
 - Directeur Administratif et Financier (DAF) : son rôle est de définir et superviser la gestion administrative (juridique, procédures, etc...) et financière (comptabilité, contrôle de gestion, trésorerie, etc...).
 - Directeur Général : se charge de définir la stratégie commerciale de l'entreprise et anticipe les mouvements de son marché.
- Les différents départements tels que le département Comptabilité, département Commercial et département Informatique :

- Département Comptabilité : son rôle est d'assurer la comptabilisation des flux financiers et de produire les documents comptables obligatoires : bilan. De même, il est chargé des relations avec les banques et les services fiscaux.
- Département Commercial : son rôle est de transmission des informations concernant le service aux clients et de le fidéliser.
- Département Informatique : assure la création des affiches publicitaires de l'agence sur les réseaux sociaux. ainsi, le paramétrage de quelques parties du site web.

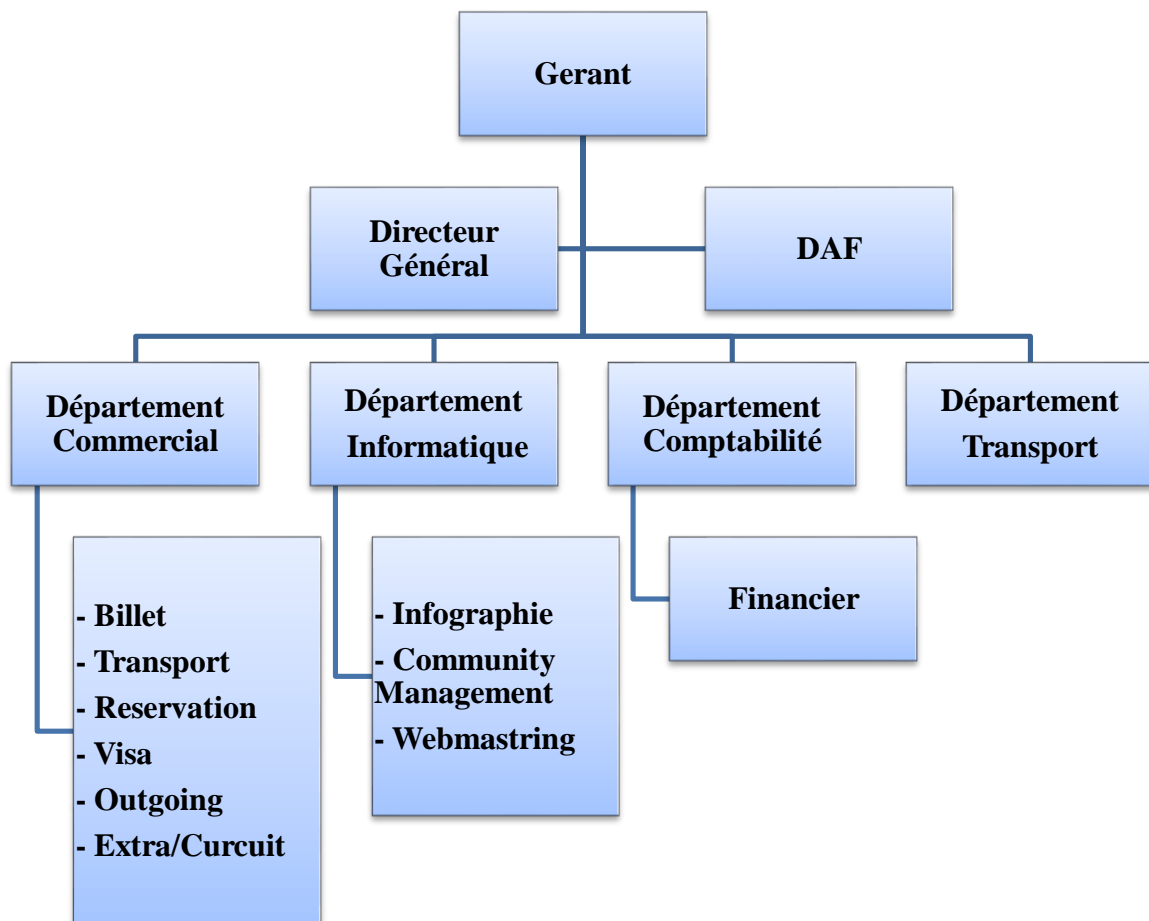


Figure 2 : Organisation d'Israa Travel Agency Djerba

3. Etude de l'existant

3.1. Les objectifs du stage

Ce stage, a en premier lieu comme objectif, de découvrir le monde du travail et de l'entreprise, ainsi que de mettre en pratique mes connaissances acquises à fin de s'en sortir dans des situations complexes. Il permettra également d'évaluer mes compétences en résolution des problèmes qui peuvent être liés au monde socio-professionnel et en particulier au monde informatique.

Durant ce stage, je me suis demandé de faire la rédaction d'un rapport, un premier apprentissage d'envergure à la recherche documentaire et à la rédaction scientifique, sollicitant ainsi nos capacités d'analyse, d'esprit critique et de synthèse.

Enfin, l'objectif technique de mon stage du projet de fin d'études est de concevoir développer une application web et mobile dynamique permettant de résoudre un besoin de la société.

3.2. Description de l'existant

En se basant sur la demande croissante et le relance de la Omra, les pèlerins et les accompagnateurs sont obligés de se déplacer vers l'un des locaux d'agence d'Israa Travel Agence pour obtenir l'information de l'un de ces commerciaux (comme les dates, le nom des hôtels, les prix ...etc.).

Puis, le pèlerin est demandé de présenter son passeport à l'agent, qui va ajouter son nom, son prénom, le numéro de passeport sur un fichier Word partagé entre les autres agents commerciaux de l'agence.

Actuellement, la société ne dispose d'aucun outil informatique spécifique permettant la gestion d'Omra, ce qui nécessite une gestion manuelle avec des fichiers Word.

3.3. Critique de l'existant

Après analyse de l'existant, j'ai pu relever un certain nombre des difficultés que rencontrent les fonctionnaires de la société dont la gestion de l'Omra :

- La transmission d'informations est difficile lorsque les agents sont chargés par d'autres travaux.
- L'information d'agent peut-être approximatives concernant le détail d'un package.

- Le malentendu entre les agents lors de la création du fichier Word.

3.4. Solution proposée

Afin de pallier le problème observé, et de faciliter la tâche de la gestion de l'Omra, j'ai proposé une solution informatisée avec plusieurs fonctionnalités. Cette solution est de développer une application web dynamique qui doit être exploitée d'une manière facile et efficace, ainsi que deux applications mobiles.

4. Cahier de charges

Il est important de rappeler que, dans tout projet, le cahier des charges est un contrat entre le concepteur/développeur du projet et l'utilisateur final de celui-ci. Il doit être respecté lors de la conception et de la réalisation du projet.

Dans le cadre de mon projet, le cahier des charges renferme :

- les objectifs du projet.
- le public cible.
- les spécifications des besoins.

4.1. Sujet

Création d'une plateforme de gestion de la Omra

4.2. Objectifs du projet

Il s'agit de concevoir et d'implémenter une plateforme de gestion de la Omra ainsi qu'une application mobile reliée à ce site permettant de :

- Assurer la gestion des utilisateurs, des pèlerins, d'accompagnateurs, du package de Omra.
- Assurer la réservation en ligne.

4.3. Public cible

- Pèlerins.
- Accompagnateurs.
- Agent de l'agence de voyage.

4.4. Spécifications de besoins

4.4.1. Identification des acteurs

- Le Pèlerin : cet acteur est le visiteur ayant déjà créé un compte sur la plateforme, il peut donc suivre le processus d'achat d'un des packs de la Omra.
- L'accompagnateur : cet acteur a déjà la possibilité d'ajouter d'autres pèlerins à la Omra.
- L'administrateur : pour les sites web on l'appelle généralement « le webmaster ». C'est celui qui assure le dynamisme du site et veille sur les mises à jour des produits, de leurs prix, de leurs disponibilités, et de la gestion d'utilisateurs.

4.4.2. Besoins fonctionnels

Un besoin fonctionnel est un besoin direct exprimant une réponse à une action, demande ou information système.

Après l'étude de mon projet, je constate qu'il doit permettre à :

L'administrateur de :

- Gérer les packages.
- Gérer les utilisateurs.
- Gérer les pèlerins et les accompagnateurs.

Pèlerin :

- S'inscrire.
- Explorer l'application Mobile.
- Explorer la liste des accompagnateurs.
- Réserver son pack.
- Consulter les listes de réservation.

Accompagnateur :

- S'inscrire
- Explorer les packages.
- Ajouter des pèlerins.

4.5. Outils de développement et de communication

- Vue Js pour la création d'interface d'application web.
- Laravel pour la création de l'api.
- Flutter pour le développement mobile.
- StarUML pour la schématisation des diagrammes de la conception.

Conclusion

Dans ce chapitre, j'ai présenté dans un premier temps l'organisme ainsi que les critiques de l'existant. Dans un second temps, j'ai proposé une solution envisagée en vue de minimiser les problèmes.

Chapitre 2 Étude Conceptuelle

Introduction

La conception est une étape primordiale dans le cycle de vie d'une application, elle a pour objectif de faire l'étude des données et des traitements à effectuer. C'est en général dans cette phase que s'applique les techniques de modélisation.

1. Méthodologie de conception

Cette étude sera réalisée selon le formalisme UML. UML⁽¹⁾ pour Unified Modeling Language, que l'on peut traduire par "langage de modélisation unifié " est une notation permettant de modéliser un problème de façon standard.

UML est un langage formel et normalisé, il est un support de communication performant. Il 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 on analysera les besoins attendus de l'application par le biais des diagrammes de cas d'utilisation 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. On dressera ainsi le diagramme de classes.

2. Diagrammes de cas d'utilisation

Les diagrammes de cas d'utilisation⁽²⁾ (DCU) sont des diagrammes UML utilisés pour donner une vision globale du comportement fonctionnel d'un système logiciel. Il est composé de 3 composants :

Acteurs : Un acteur est l'idéalisation d'un rôle joué par une personne externe, un processus ou une chose qui interagit avec un système.

Il se représente par un petit bonhomme avec son nom inscrit dessous.



Figure 3 : Exemple d'acteur

Le cas d'utilisation : Un cas d'utilisation est une unité cohérente représentant une fonctionnalité visible de l'extérieur.

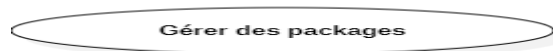


Figure 4 : Exemple d'un cas d'utilisation

2.1. Diagramme de cas d'utilisation «global»

La figure suivante illustre le diagramme de cas d'utilisation qui permet de représenter les grandes fonctionnalités nécessaires aux utilisateurs du système.

C'est le premier diagramme du modèle UML, celui qui assure la relation entre l'utilisateur et les objets que le système met en œuvre.

Ce diagramme n'est qu'une vue globale de notre projet, les diagrammes suivant vont bien détailler l'enchaînement des services du projet. Mon projet comporte trois acteurs qui interagissent chacun selon son rôle avec le système.

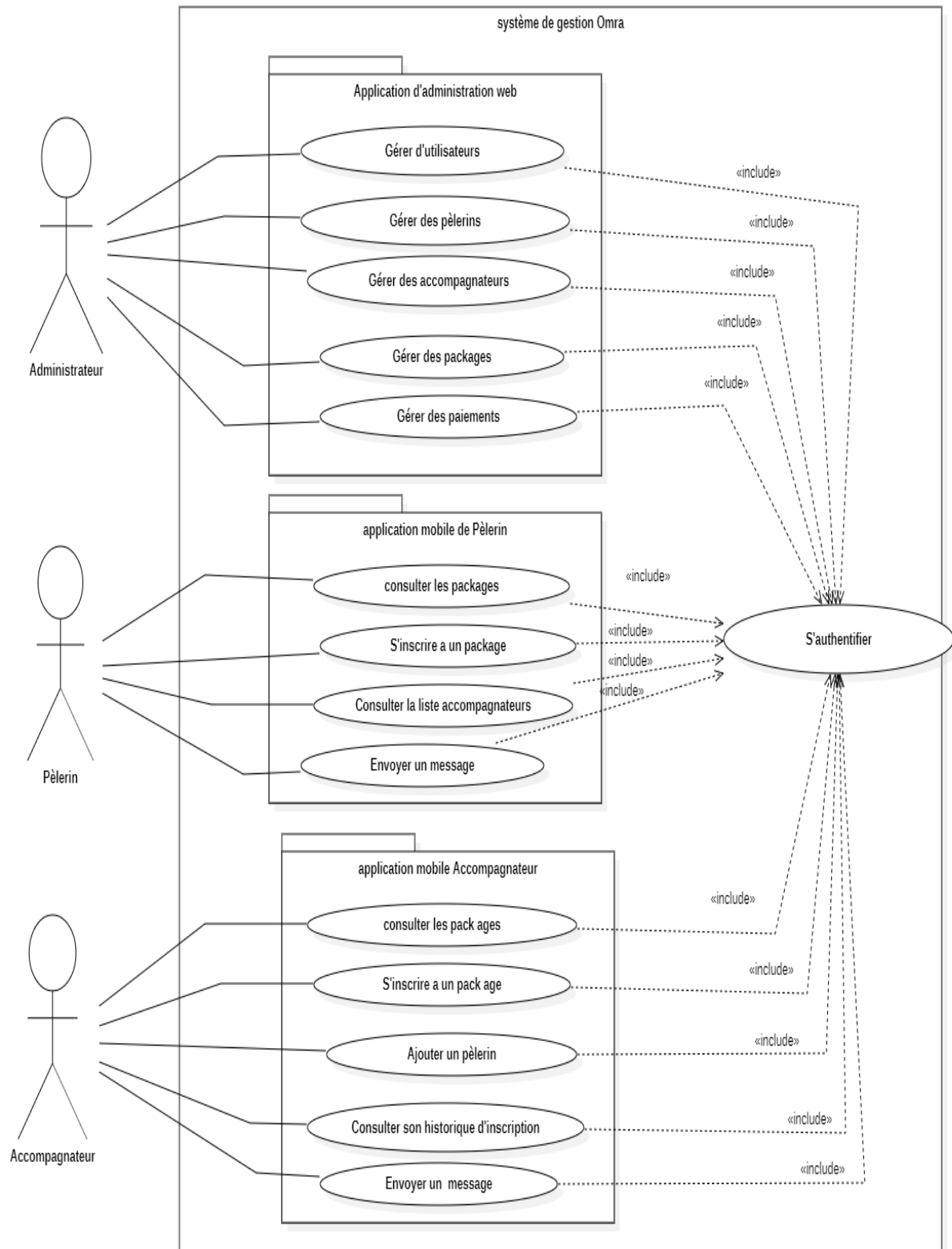


Figure 5 : Diagramme de cas d'utilisation global

2.2. Diagramme de cas d'utilisation «administrateur»

L'administrateur du système peut gérer (ajout, suppression, modification) tous les utilisateurs Ou les package. Ainsi la validation des inscriptions pour les pèlerins, de publier ou masquer les données d'un accompagnateur et de répondre aux messages clients. L'administrateur est le seul qui a le privilège de suppression.

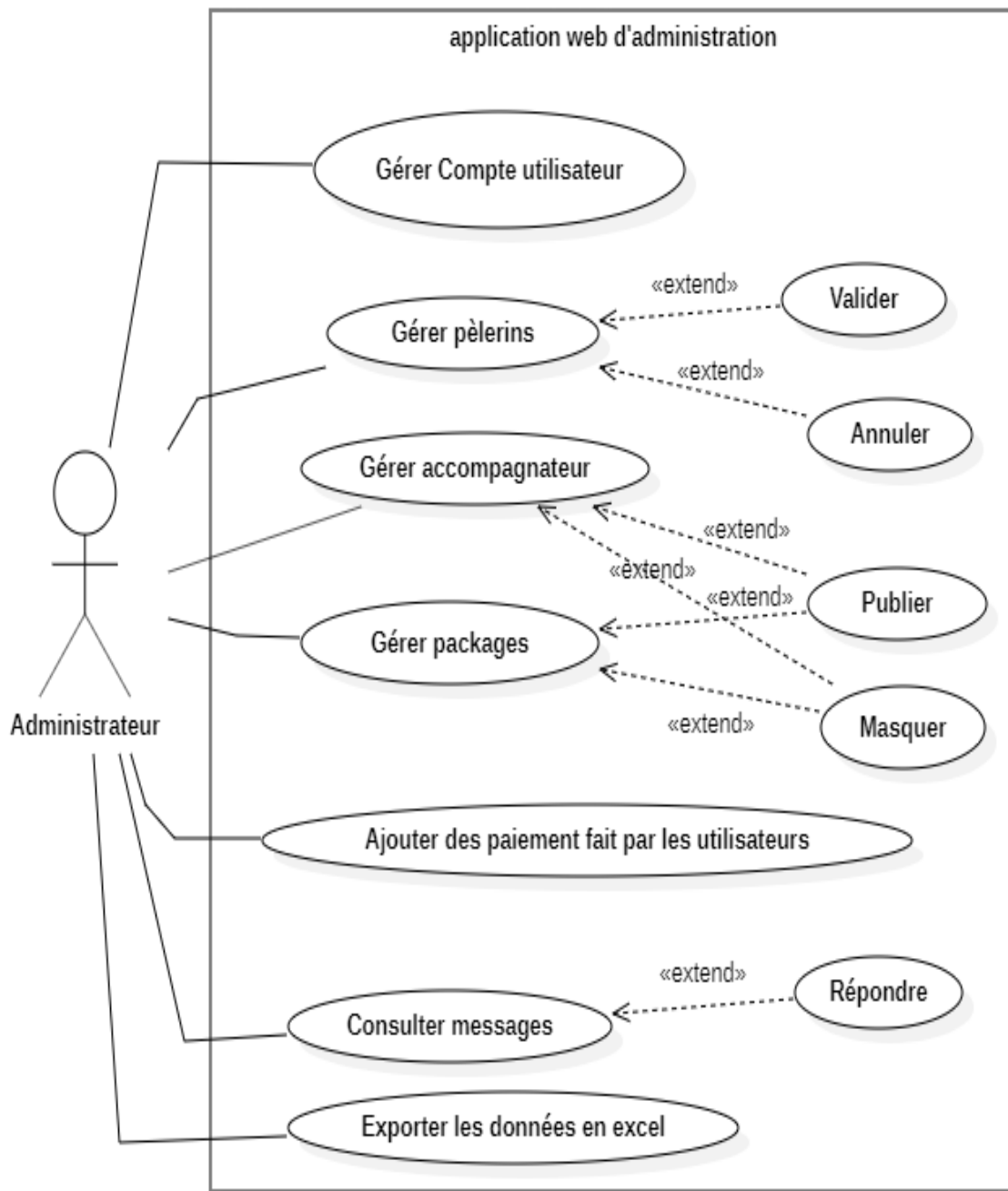


Figure 6 : Diagramme de cas d'utilisation de l'administrateur

✓ Description textuelle de la gestion des packages :

Cas d'utilisation	Gestion des packages
Résumé	L'administrateur peut gérer les packages existant sur la base ; il peut ajouter, modifier, supprimer, publier ou masquer
Acteurs	Administrateur
Scénario nominale	<ul style="list-style-type: none"> -L'administrateur saisit les données de package -Le système enregistre le package -L'administrateur reçoit un message de succès -L'administrateur choisit un package -L'administrateur change les informations de package -Le système enregistre les nouvelles informations. -L'administrateur reçoit un message de succès -L'administrateur supprime le package -Un message de vérification s'affiche. -L'administrateur reçoit un message de succès -L'administrateur publie le package -L'administrateur reçoit un message de succès -L'administrateur masque le package -L'administrateur reçoit un message de succès
Scénario d'erreur	<ul style="list-style-type: none"> -L'administrateur ne saisit rien -L'administrateur ne choisit pas un package
Précondition	-L'administrateur doit être authentifié.
Post-condition	<ul style="list-style-type: none"> - Package ajouté - Package modifié - Package supprimé - Package publié - Package masqué

Tableau 1 : Description du cas d'utilisation Gestion des packages

2.3. Diagramme de cas d'utilisation «pèlerin»

Le pèlerin peut s'inscrire à l'application ou à un package, consulter la liste des accompagnateurs, envoyer un message de réclamation en cas des besoins et de consulter son historique des inscriptions.

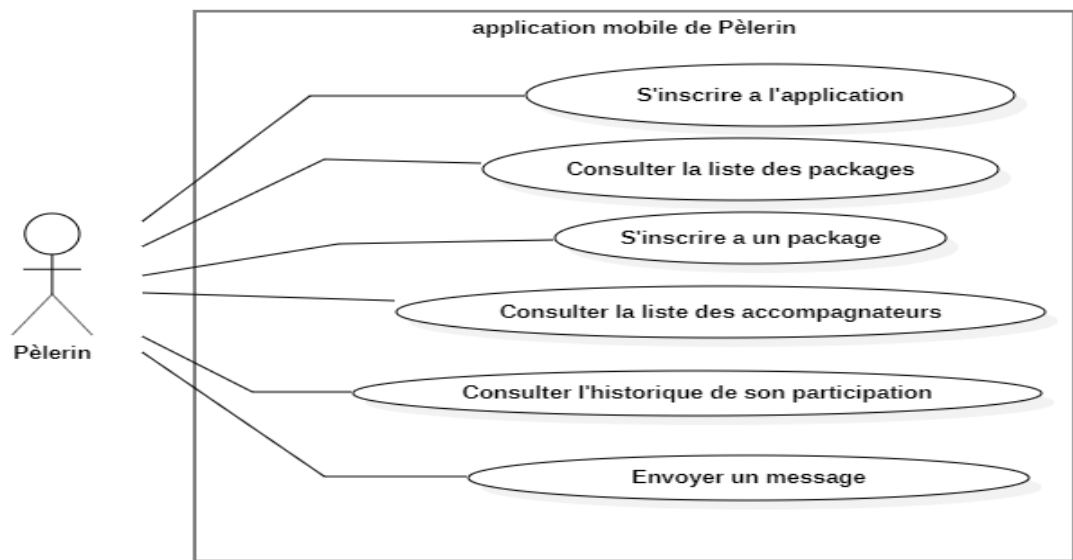


Figure 7 : Diagramme cas d'utilisation de pelerin

✓ Description textuelle du s'inscrire à un package :

Cas d'utilisation	S'inscrire à un package
Résumé	Le pèlerin peut s'inscrire un package
Acteurs	Pèlerin
Scénario nominale	<ul style="list-style-type: none"> -Le pèlerin consulte la liste des packages. -Le pèlerin choisit un package. -Le pèlerin consulte le détail de package -le pèlerin clique sur le Botton inscrire -Le pèlerin remplit le formulaire d'inscriptions. - le pèlerin envoie le formulaire.
Scénario d'erreur	<ul style="list-style-type: none"> -Le pèlerin ne choisit pas un package -Le pèlerin ne remplit pas le formulaire
Précondition	- Le pèlerin doit être authentifié.
Post-condition	- Inscription ajouté

Tableau 2 : Description du cas d'utilisation inscrire à un package

2.4. Diagramme de cas d'utilisation «accompagnateur»

L'accompagnateur peut s'inscrire à l'application ou ajouter des pèlerins, envoyer un message de réclamation en cas de besoins et de consulter l'historique de ses inscriptions.

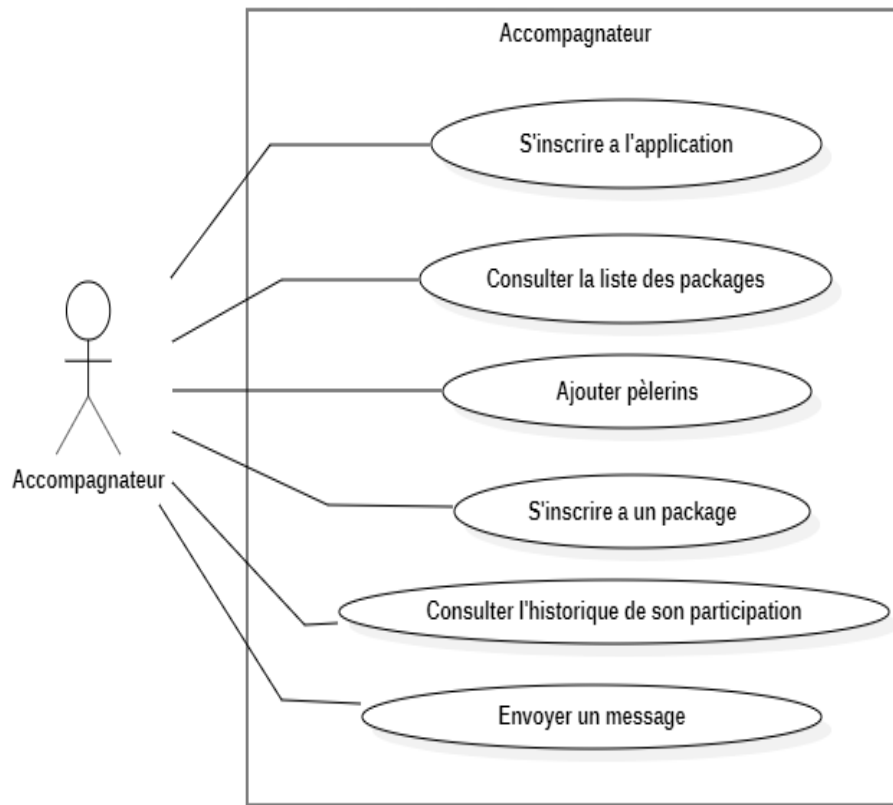


Figure 8 : Diagrammes de cas d'utilisation accompagnateur

✓ Description textuelle du cas d'utilisation envoyer un message :

Cas d'utilisation	Envoyer un message
Résumé	L'accompagnateur peut envoyer un message
Acteurs	Accompagnateur
Scénario nominale	-L'accompagnateur ouvre le formulaire de message -L'accompagnateur remplis ce dernier. - L'accompagnateur envoie le formulaire.
Scénario d'erreur	- L'accompagnateur n'écrit rien dans le message
Précondition	- L'accompagnateur doit être authentifié.
Post-condition	- Message envoyé

Tableau 3 : Description du cas d'utilisation envoyer un message

3. Diagrammes des séquences

Le **diagramme de séquence**⁽³⁾ est un diagramme d'interaction qui représente les objets participant à une interaction et les messages qu'ils échangent, organisés en séquences horaires. C'est une description de ce que fait un système indépendamment de la manière dont il le fait.

3.1. Diagramme de séquence « Se authentifier »

La figure ci-dessous décrit le scénario d'authentification, tels que ce diagramme :

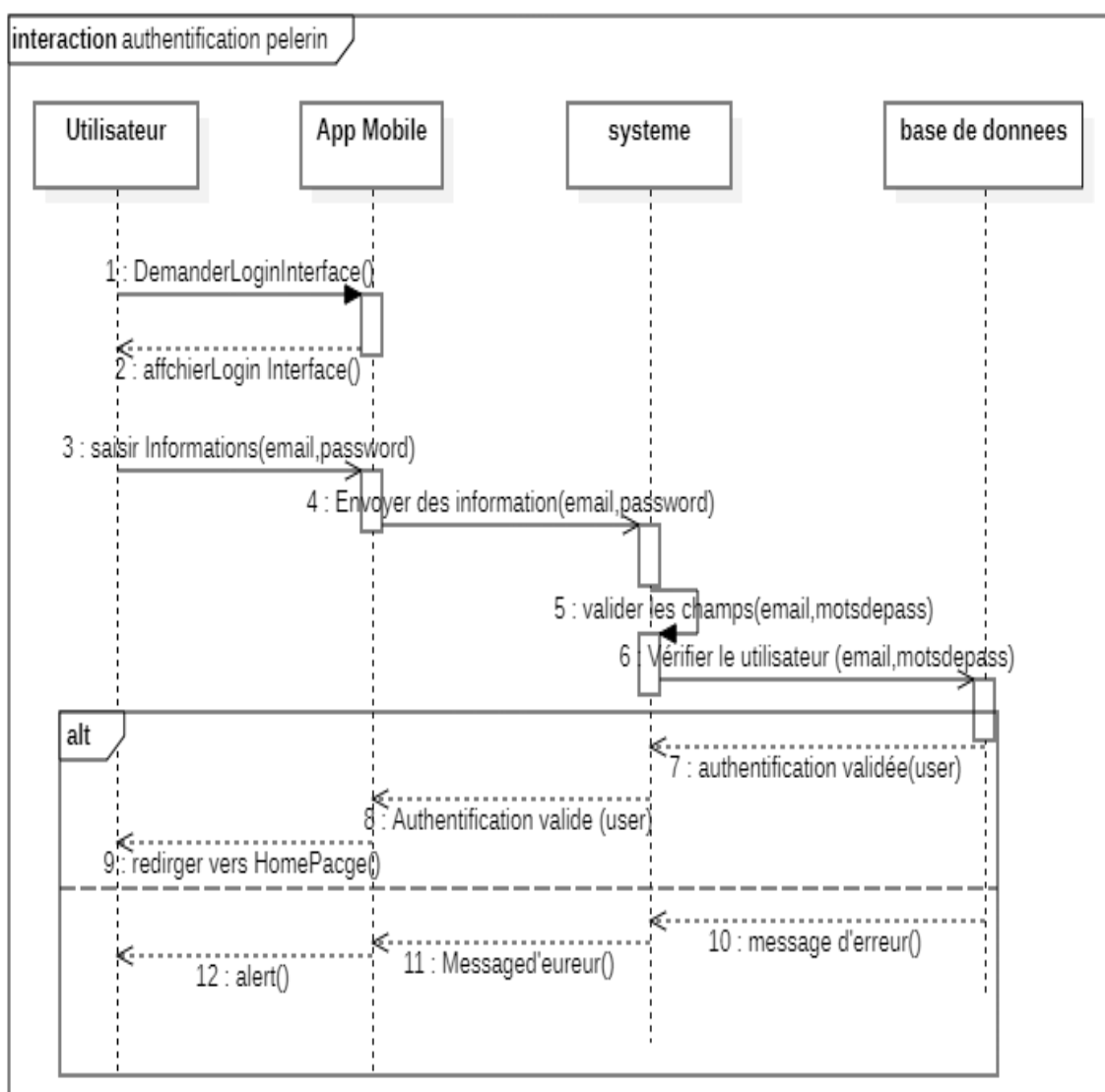


Figure 9 : Diagramme de séquence « Se authentifier »

Pour qu'un pèlerin soit authentifié .Il doit lancer l'application, remplir son email et son mots de passe puis qu'il l'envoie. Le système va vérifier les données envoyées par ce client et va vérifier

l'existence d'un utilisateur .Puis si il existe il va retourner tous les coordonnet de ce dernier et le rediriger vers la page d'accueil .sinon il va lui fournir une alerte d'erreur.

3.1. Diagramme de séquence « Ajouter utilisateur»

La figure ci-dessous décrit le scénario du « Ajouter utilisateur»

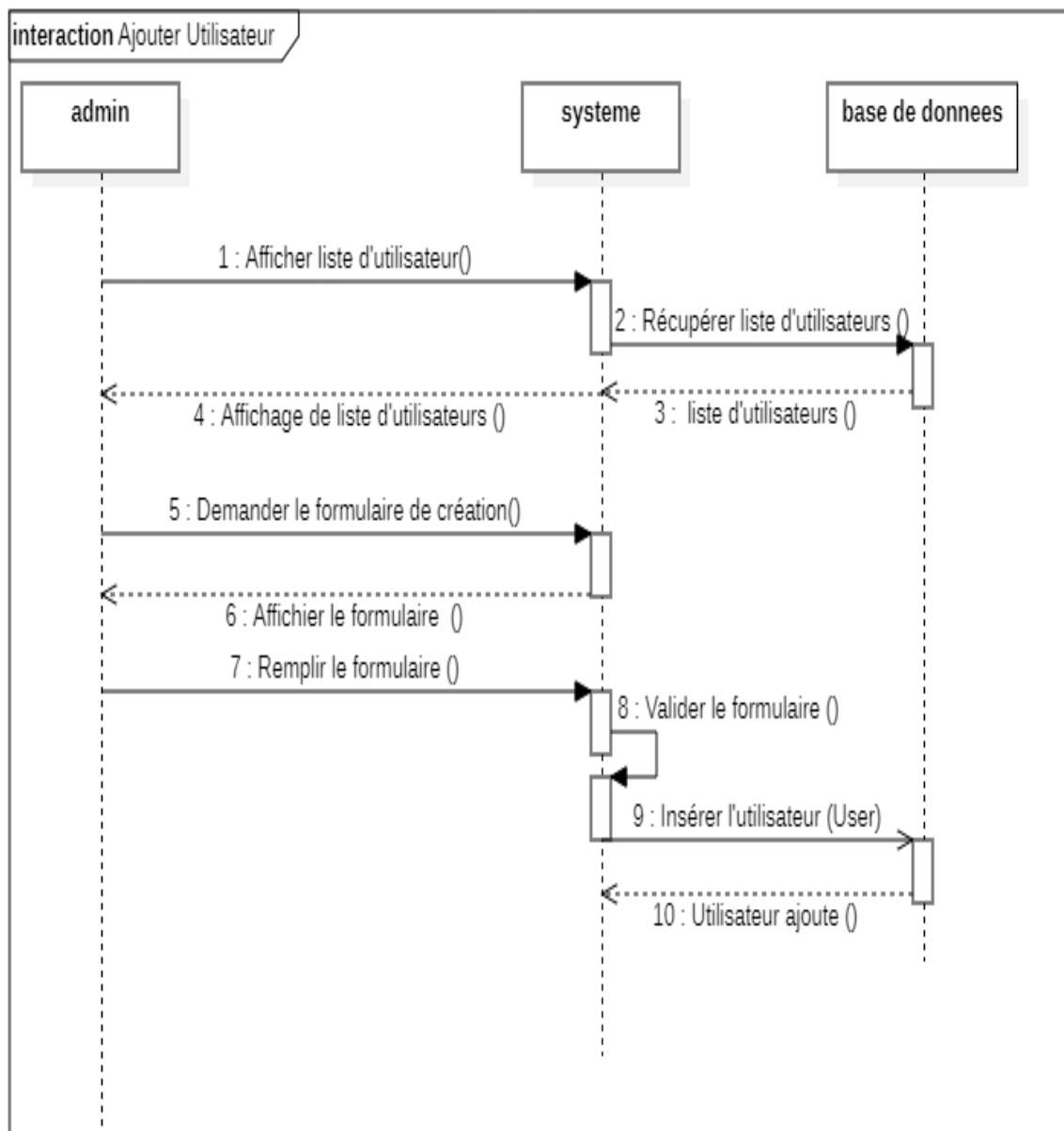


Figure 10 : Diagramme de séquence « Ajouter utilisateur»

Pour ajouter un utilisateur, l'administrateur est demandé d'ouvrir tous d'abord la page de listing de tous les utilisateurs. Puis clique sur le bouton '+' pour ouvrir un formulaire de création d'utilisateur. Il saisit les données nécessaires .Une vérification fait par le système ; en cas que

les donnes sont valides alors Il y a message de succès qui va s’afficher au administrateur .sinon des erreurs vont être affichées pour les donnes invalides.

3.2. Diagramme de séquence « valider pèlerin»

La figure ci-dessous décrit le scénario « valider pèlerin»

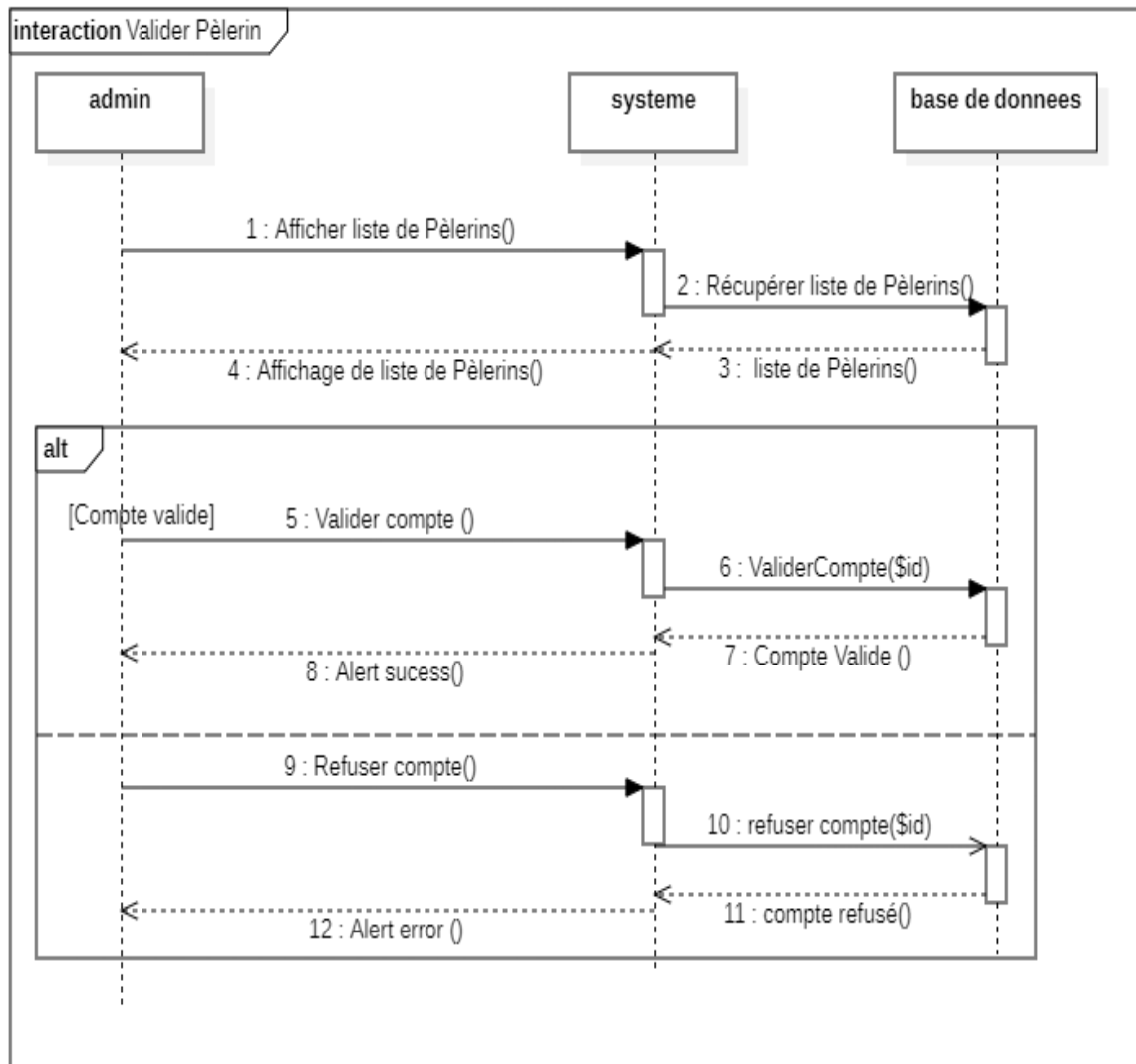


Figure 11 : Diagramme de séquence « Valider pèlerin»

Pour valider un compte pèlerin, l’administrateur est demandé d’ouvrir la liste de pèlerins et de valider le compte qui veut ou le refuser en clique sur le bouton qui se trouve devant le nom de chaque compte pèlerin

3.3. Diagramme de séquence « Ajouter pèlerin dans un package »

La figure ci-dessous décrit le scénario «S’inscrire de pèlerin dans un packages»

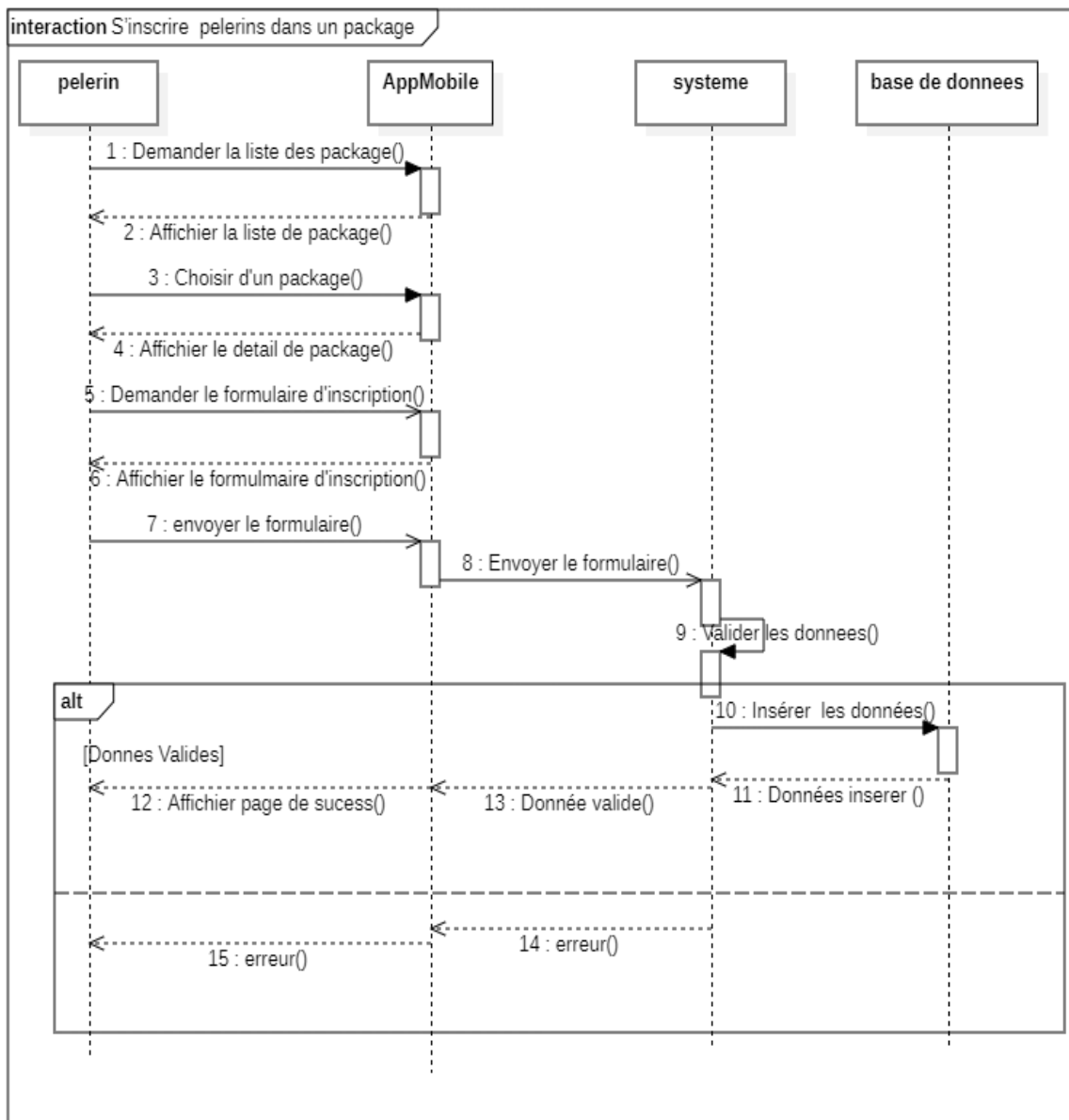


Figure 12 : Diagramme de séquence « S’inscrire de pèlerin dans un package »

Pour qu’un utilisateur soit inscrit dans un package d’Omra. Le pèlerin doit tout d’abord consulter la liste des packages, puis cliqué sur le bouton d’inscription de ce qu’il lui convient. Le Système va lui affiche un formulaire pour qu’il le remplir avec des données valables de son passeport .Puis le système va vérifier les donnes saisies et va l’ajouter dans la base de données en cas où ils sont valides sinon il va retourner un message d’erreur.

4. Diagramme des classes

Le **diagramme des classes**⁽⁴⁾ représente les classes constituant le système et les associations entre elles. Il exprime d'une manière générale la structure statique d'un système en termes de classes et de relations entre ces classes.

La figure ci-dessous représente le diagramme des classes de mon système :

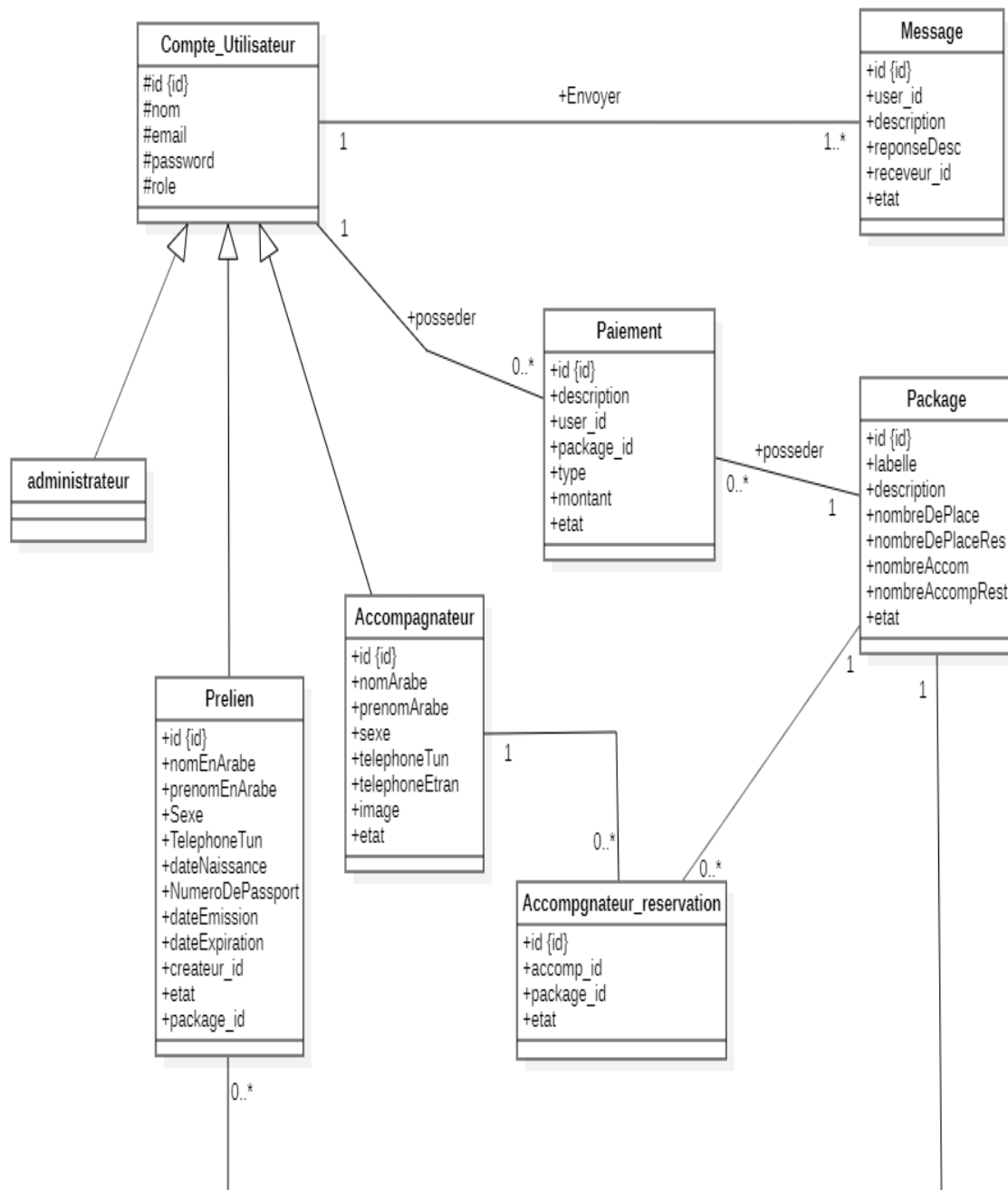


Figure 13 : Diagramme de classe

✓ Description textuelle

▪ Les classes admin, accompagnateurs et pèlerins héritent de la classe utilisateur, pour utiliser les mêmes attributs et méthodes.

- Chaque package contient plusieurs pèlerins.
- Chaque pèlerin peut avoir plusieurs participations.
- Chaque accompagnateur peut avoir plusieurs participations.

Attributs	
ID : Identifiant Nom : nom Email : l'adresse email d'administrateur Mot de passe : mot de passe d'administrateur	
Méthode	<ul style="list-style-type: none"> ✓ Connecter () : pour se connecter à la plateforme et ouvrir une session. ✓ Inscription () : pour s'inscrire à la plateforme ✓ Déconnexion () : pour se déconnecter et fermer une session.

Tableau 4 : Description de la classe « Compte Utilisateur »

5. Diagramme de déploiement

Un **diagramme de déploiement**⁽⁵⁾ est une vue statique qui sert à représenter l'utilisation de l'infrastructure physique par le système et la manière dont les composants du système sont répartis ainsi que la relation entre eux.

Mon système contient cinq nœuds :

- Le serveur web qui contient Laravel.
- Un serveur de base de données contient le système de gestion des bases de données MySQL.
- Une application web.
- Une application mobile client.
- Une application mobile accompagnateur

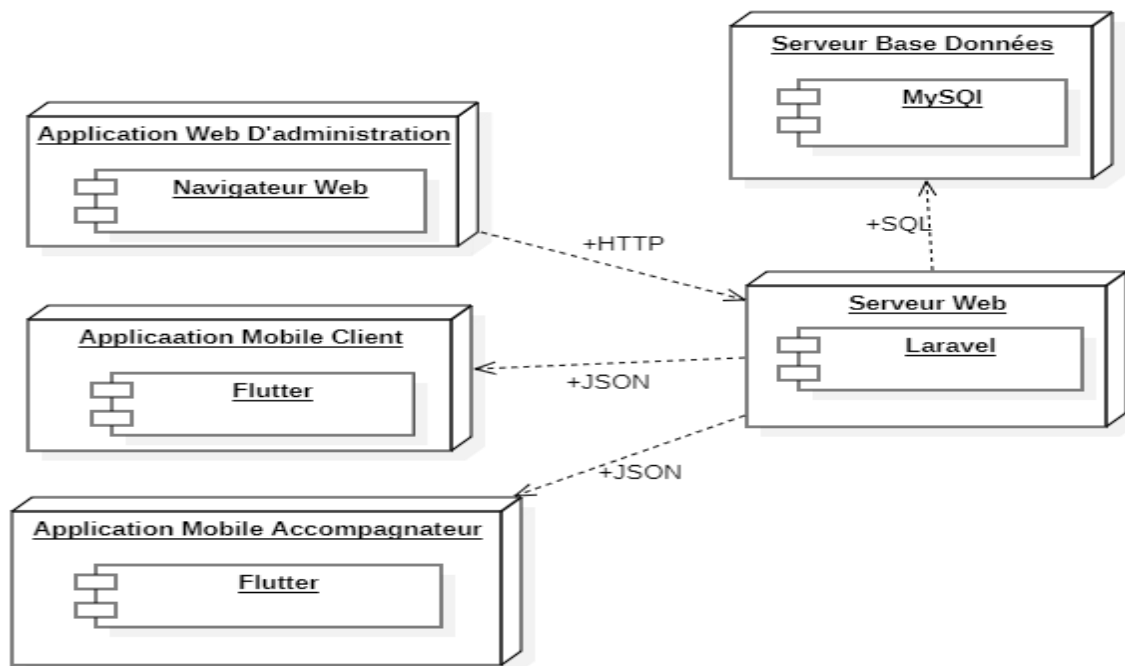


Figure 14 : Diagramme de déploiement

Conclusion

Dans ce chapitre, j'ai élaboré l'étude conceptuelle de projet. J'ai suivi la méthode de conception UML. L'élaboration des séquences sert à faciliter la tâche de réalisation et de développement qui sera présentée dans le chapitre suivant.

Chapitre 3 Réalisation

Introduction

Ce chapitre représente le dernier volet de ce rapport, il sera consacré à l'implémentation de mon système. Je vais commencer par la présentation des ressources logicielles utilisées, puis je vais passer à la présentation des captures d'écran dans le but de mettre en évidence l'aspect ergonomique et fonctionnel des interfaces développées.

1. Choix technologiques

Pour choisir la bonne technologie de développement, il faut tout d'abord réaliser une petite étude comparative entre les technologies qui se trouve sur le marché.

1.1. Vue JS

Vue.js⁽⁸⁾ est un Framework JavaScript, lancé en 2013, qui convient parfaitement à la création d'interfaces utilisateur hautement adaptables et d'applications Single-page complexes.



Figure 15 : Logo Vue Js

Avantages	<ul style="list-style-type: none">✓ Documentation détaillée✓ Ne nécessite aucune connaissance supplémentaire : HTML et JavaScript feront le travail.
Inconvénients	<ul style="list-style-type: none">✓ Antécédents chinois✓ Les mises à jour fréquentes obligent les développeurs à réapprendre

Tableau 5 : Avantages et inconvénients Vue JS

1.2. Laravel

Laravel⁽⁶⁾ est un Framework web open-source écrit en PHP respectant le principe modèle-vue-contrôleur et entièrement développé en programmation orientée objet. Laravel est distribué sous licence MIT, avec ses sources hébergées sur Git Hub. Ce Framework a été créé par Taylor Otwell en juin 2011.

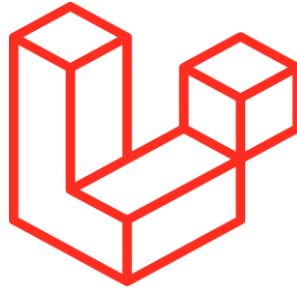


Figure 16 : Logo laravel

1.3. Laravue

Laravue⁽⁷⁾ est un projet qui utilise le Framework Laravel en backend pour sa puissance et Vue js comme haute performance sur le frontend. Il était inspiré de vue-element-admin et semble être une solution complète pour un niveau d'application d'entreprise.

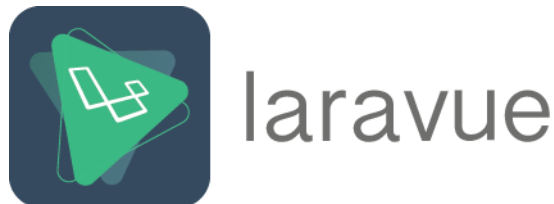


Figure 17 : Logo laravue

1.4. Flutter

Flutter⁽⁸⁾, le Framework de Google permettant de concevoir des applications Mobile multiplateforme pour Android et Ios. Il est généralement Utiliser pour Tous ce qui est interface utilisateur.



Figure 18: Logo flutter

2. Environnement de développement

2.1. Laragon

Laragon⁽⁹⁾ est un environnement de développement Web dédié au système d'exploitation Windows. Il est accompagné de différentes composantes à savoir :

*NGINX / Apache (serveur web)

*PHP (langage interprété côté serveur)

*MySQL (base de données)



Figure 19 : Logo laragon

2.2. Visuel Code

Visual Studio Code⁽¹⁰⁾ est un éditeur de code extensible développé par Microsoft pour Windows, Linux et macOS2.



Figure 20 : Logo Visual Studio code

2.3. PhpMyAdmin

PhpMyAdmin⁽¹¹⁾ (PMA) est une application Web de gestion pour les systèmes de gestion de base de données MySQL réalisée principalement en PHP et distribuée sous licence GNU GPL.

Cette interface facilite au développeur de faire la gestion des tables de la base de données ; En assurant l'ajout, la suppression et la Modification des tables .Il était en version 0.9.0 le 9 septembre 1998. Il est évolué jusqu'à nos jour, pour attendre : 5.0.1.



Figure 21 : Logo phpMyAdmin

2.4. Advanced REST Client (ARC)

C'est⁽¹²⁾ un outil de test API. Ce qui permet aux développeurs Web de créer et de tester des requêtes HTTP personnalisées.



Figure 22 : Logo Advanced Rest Client

2.5. Git hub

Git Hub⁽¹³⁾ est un site web et un service de cloud qui aide les développeurs à stocker et à gérer leur code, ainsi qu'à suivre et contrôler les modifications qui lui sont apportées.



Figure 23 : Logo GitHub

3. Réalisation

3.1. Application Web

Dans cette partie, je vais vous présenter les différentes interfaces de l'application web

3.1.1. Authentification

Comme toute application, l'authentification et l'inscription sont les modules de base. Le premier portail de mon système est l'authentification, dont l'administrateur de mon application s'authentifie.



Figure 24 : Interface d'authentification

3.1.2. Tableau de bord

Cette interface de tableau de bord est l'accueil de l'application, où l'utilisateur peut naviguer dans toute l'application.

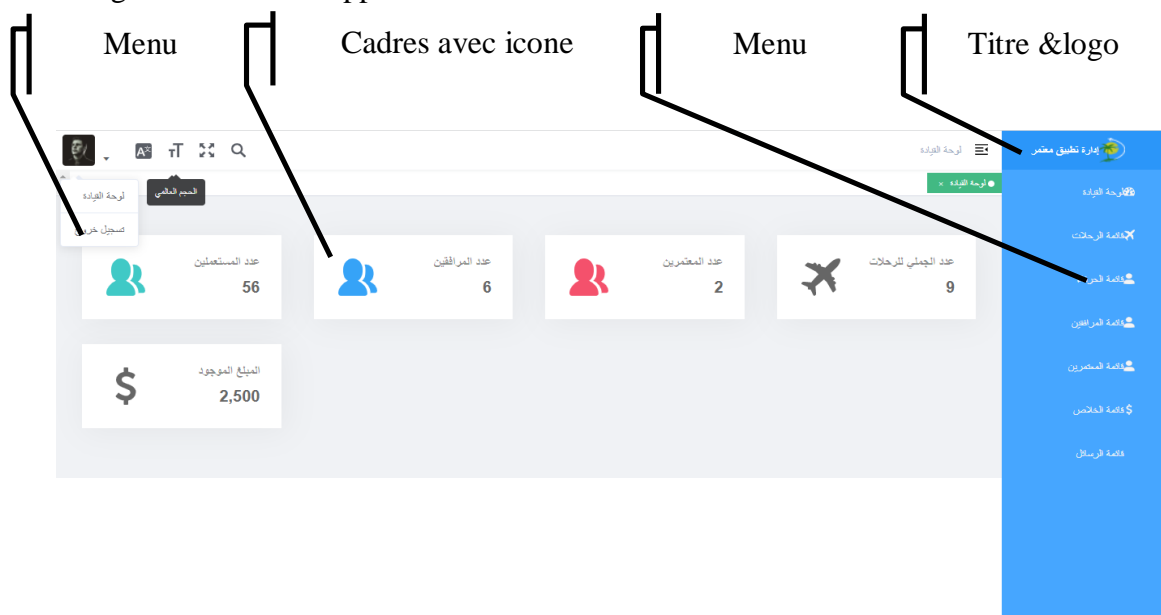


Figure 25 : Interface de dashboard

3.1.3. Interface de Gestion Utilisateurs

Dans cette partie, l'administrateur peut ajouter, modifier, supprimer des utilisateurs.

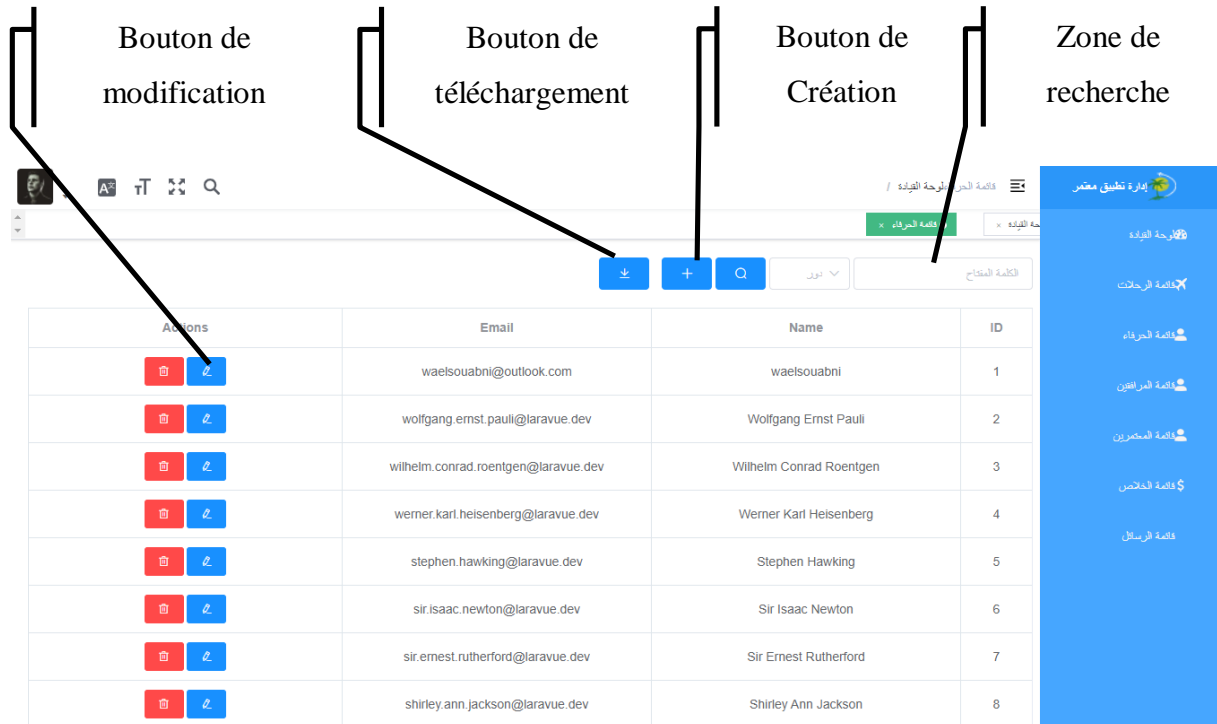


Figure 26 : Interface de gestion utilisateur

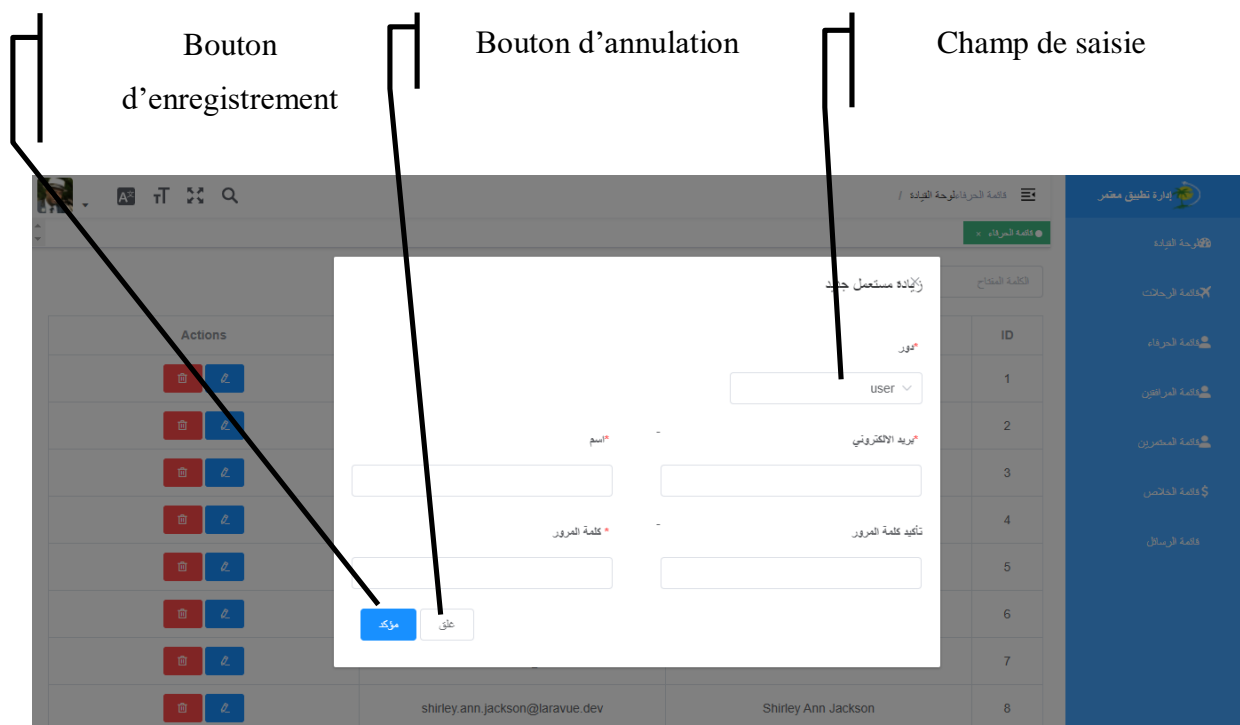


Figure 27 : Interface d'ajout d'utilisateur

3.1.4. Interface de gestion de package

Dans cette partie, l'administrateur peut ajouter, modifier, supprimer, de publier ou masquer un package.

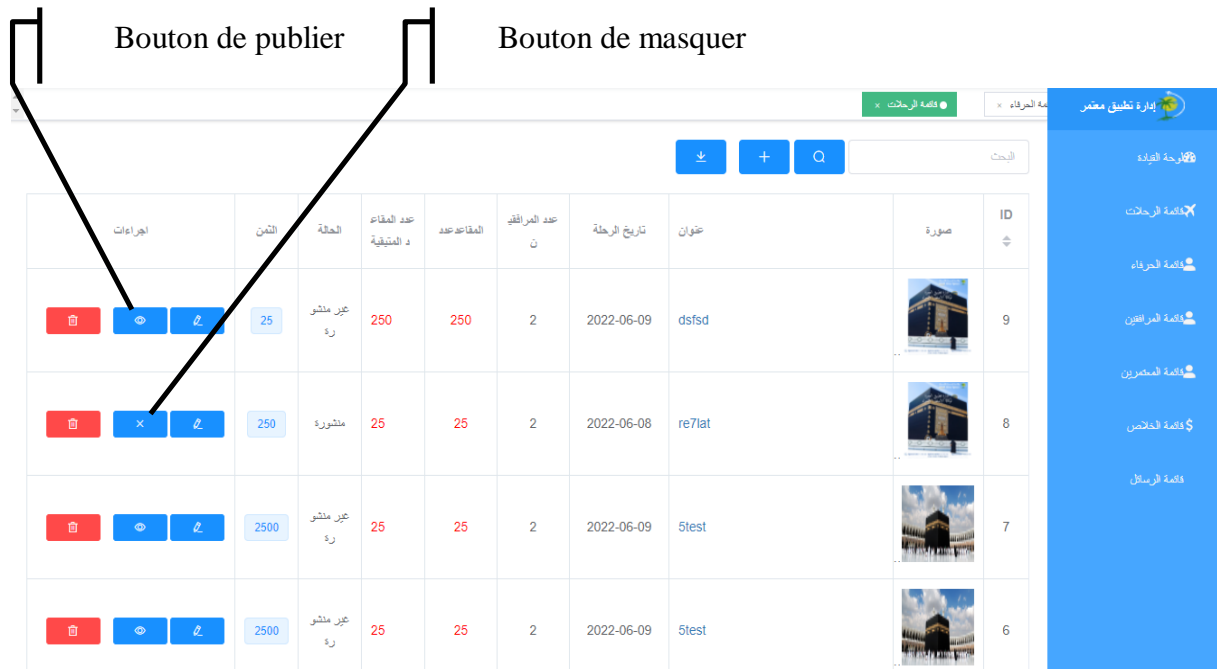


Figure 28 : Interface de gestion de package

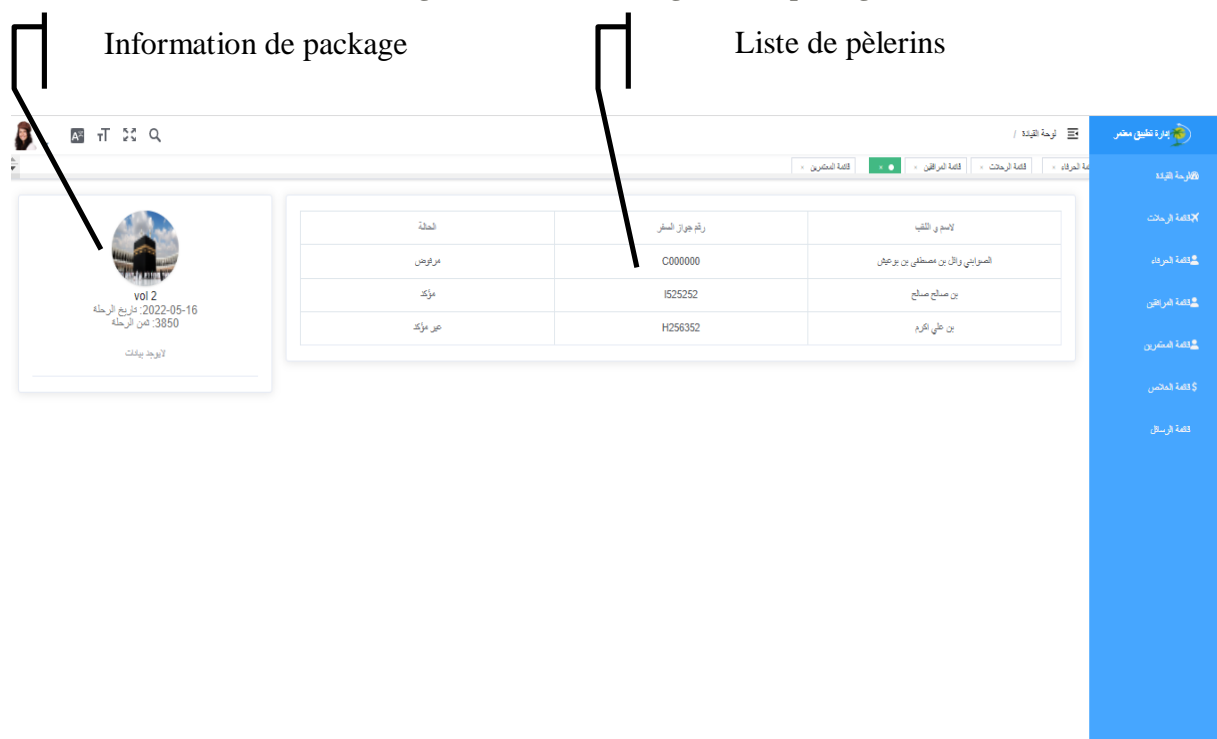


Figure 29 : Interface de détail de package

3.1.5. Interface de gestion de pèlerins

Dans cette partie, l'administrateur peut ajouter, modifier, supprimer, accepter ou refuser des pèlerins.

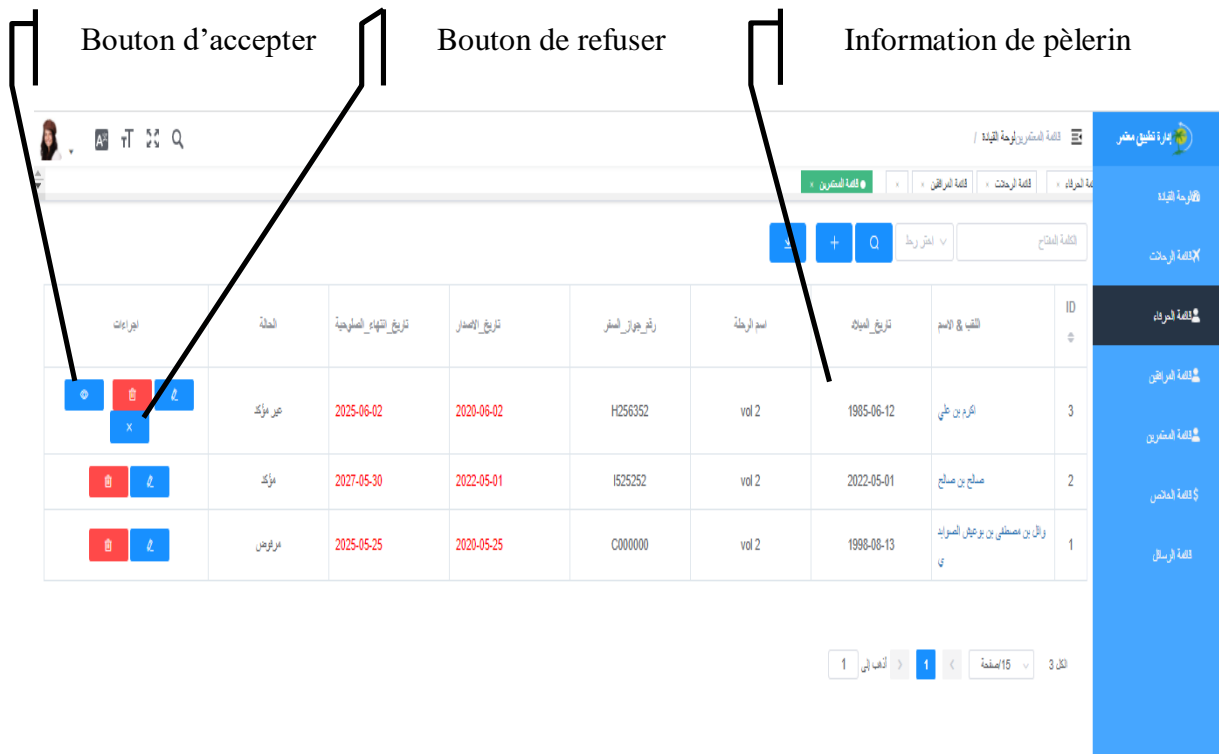


Figure 30 : Interface de gestion pèlerins

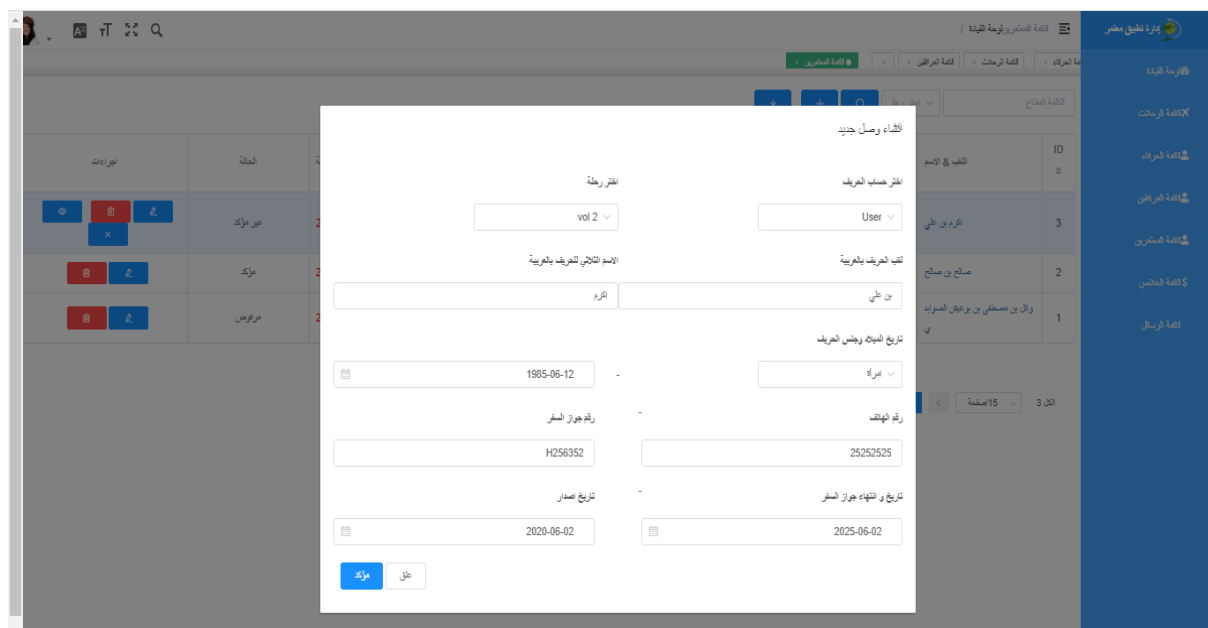


Figure 31 : Interface de modification pèlerin

3.1.6. Interface de gestion d'accompagnateur

Dans cette partie, l'administrateur peut ajouter, modifier, supprimer, publier ou masquer des accompagnateurs.

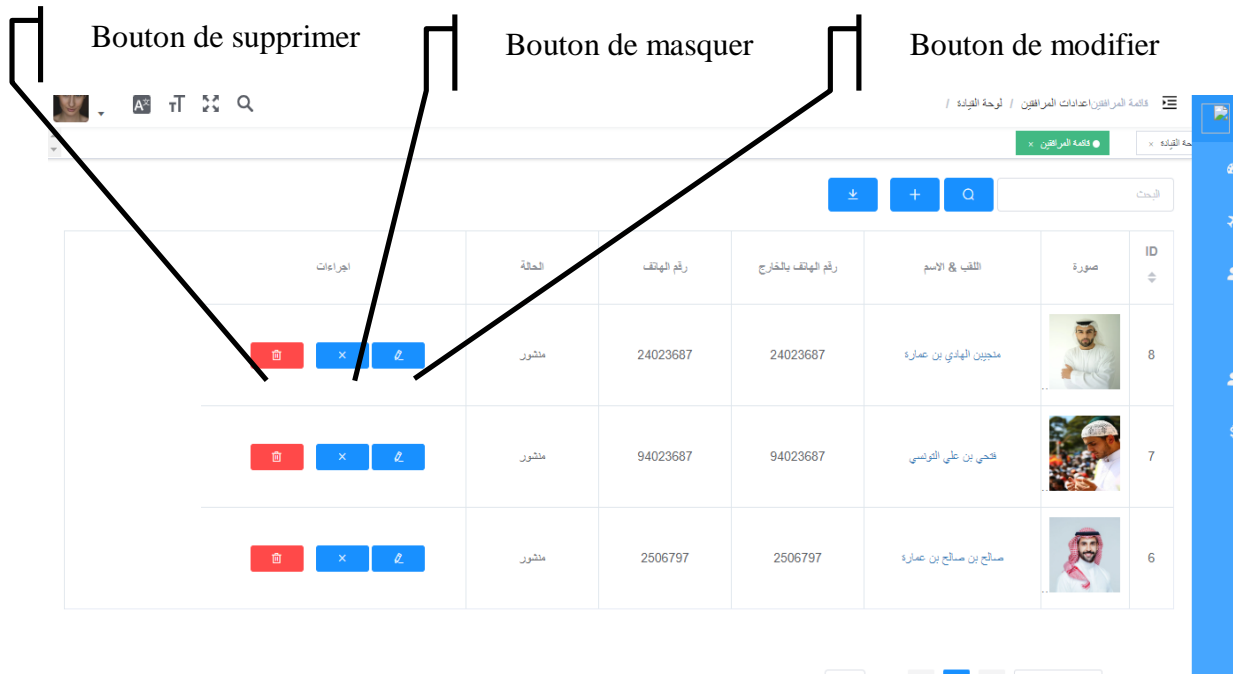


Figure 32 : Interface gestion des accompagnateurs

Et tant que on peut permet au accompagnateur de s'inscrire a plusieurs packages, On trouve aussi une interface de gestion de cette relation entre l'accompagnateur et les packages.

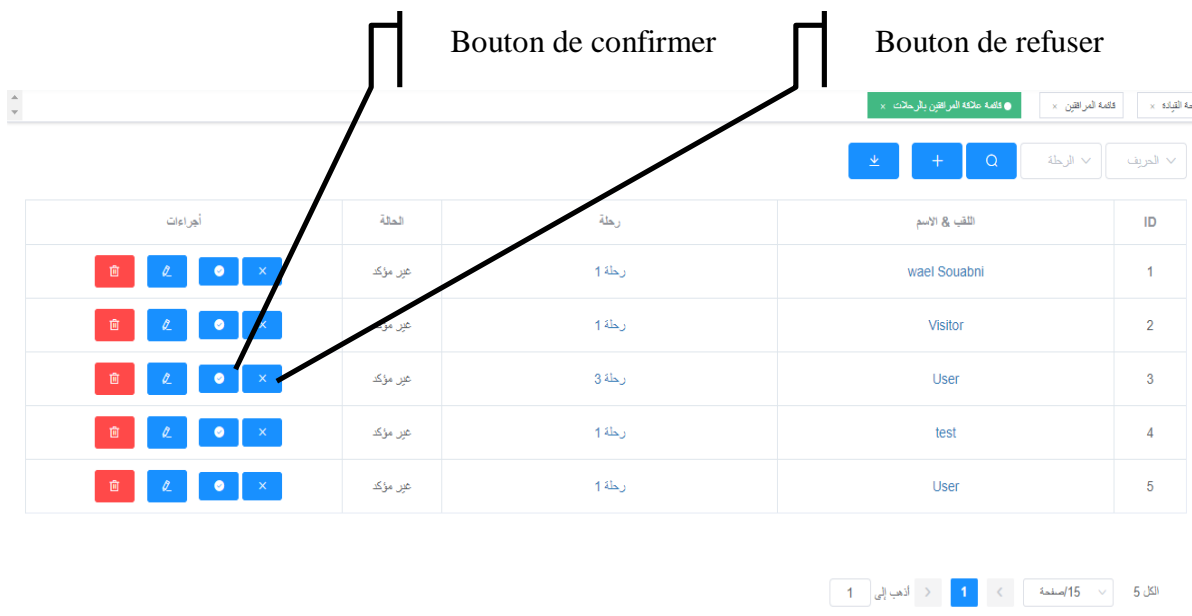


Figure 33 : Interface de gestion relation accompagnateurs - packages

3.1.7. Interface de gestion de message

Dans cette partie, l'administrateur peut consulter et répondre un message par l'envoi automatique d'un mail au client avec le texte de réponse .Ainsi de supprimer un mail existant.



Figure 34 : Interface de Gestion de message

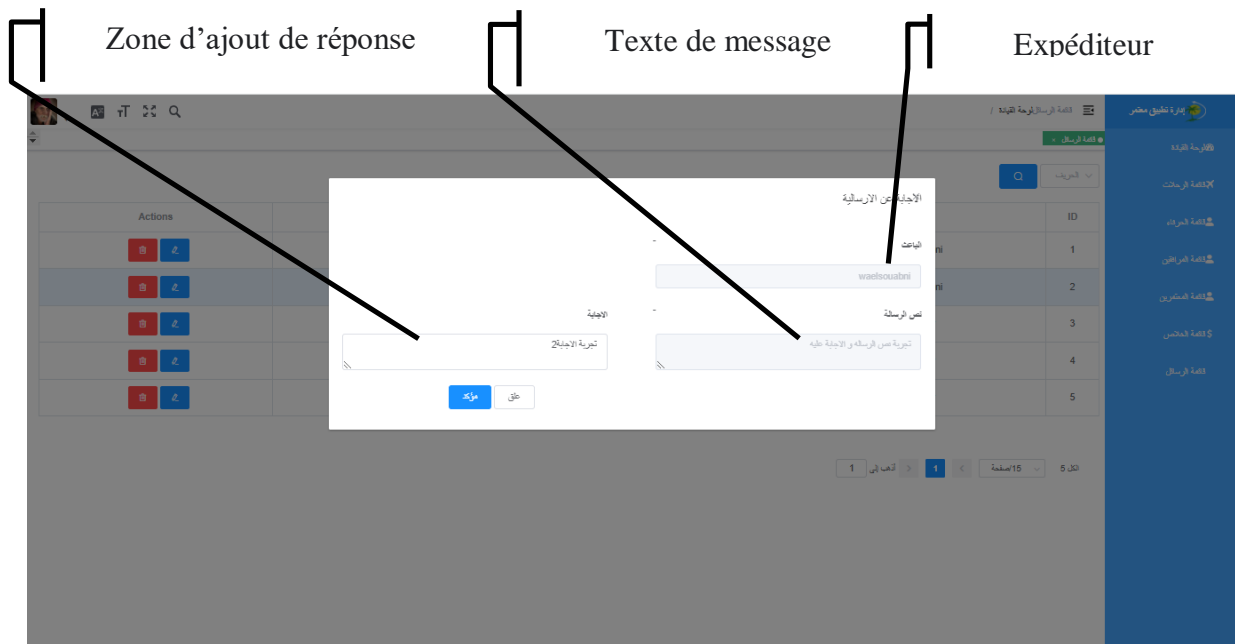


Figure 35 : Interface de réponse d'un message

3.1.8. Interface de gestion de paiement

Dans cette partie, l'administrateur peut ajouter ou supprimer un paiement qui a été faite par l'un de ses clients concernant le service Omra. Et il peut faire ça filtration selon le vol.

ID	الاسم	رحلة	مبلغ	أجراءات
1	waelsouabni	vol 2	2500	
2	Albert Einstein	test	2500	
3	User	vol 2	2500	

Figure 36 : Interface de gestion de paiement

Mais il ne faut pas oublier que tous les formulaires de création ou de modification se passent par une phase de validation .En fait ce dernier est une phase primordiale qui se fait tout d'abord avec le vue js chez le client et on le poursuivre sur le serveur pour assurer la validité des données saisies par le utilisateur.

Figure 37 : Interface de validation champ de formulaire

Il faut noter aussi que à la fin de chaque action faite par l'administrateur, il y aura un affichage d'un toast qui annonce le succès ou l'échec de son travail.

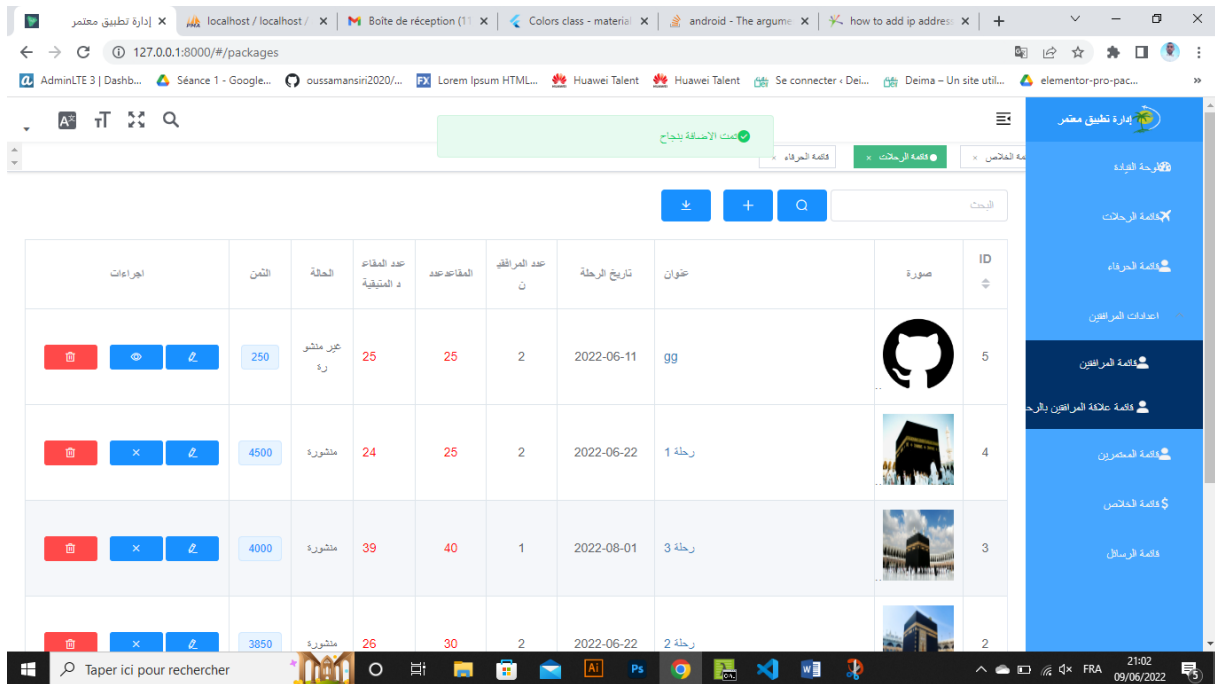


Figure 38 : Affichage d'un toast de succès

3.2. Application Mobile Pèlerin

Dans cette partie, je vais vous présenter les différentes interfaces de l'application Mobile.

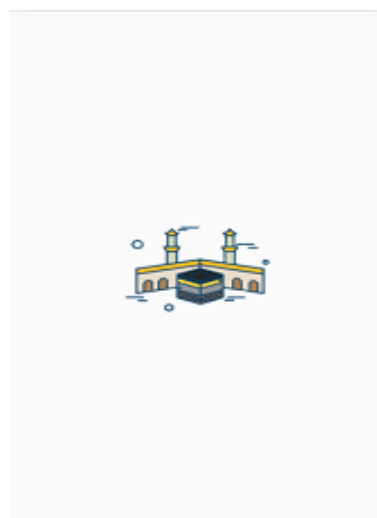


Figure 39 : Splash screen

3.2.1. Authentification

Le client s'authentifie tout d'abord, pour accéder à la page d'accueil, sinon, il doit s'inscrire

Figure 40 : Interface d'authentification

Figure 41 : Interface d'inscription

3.2.2. Accueil

Si un client est authentifié ou il a créé un nouveau compte, il sera redigé automatiquement vers la page d'accueil dans laquelle on s'affiche la liste de service de notre application.



Figure 42 : Interface d'accueil

3.2.3. Interface de liste packages et ses détails

Lorsqu'un utilisateur appuie sur l'icône avion dans le menu qui se trouve en bas de l'écran de son appareil. Ce dernier va être redirigé pour l'affichage de la liste des packages qui sont en état de publication et avec une clique sur l'un des packages, il va afficher son détail.

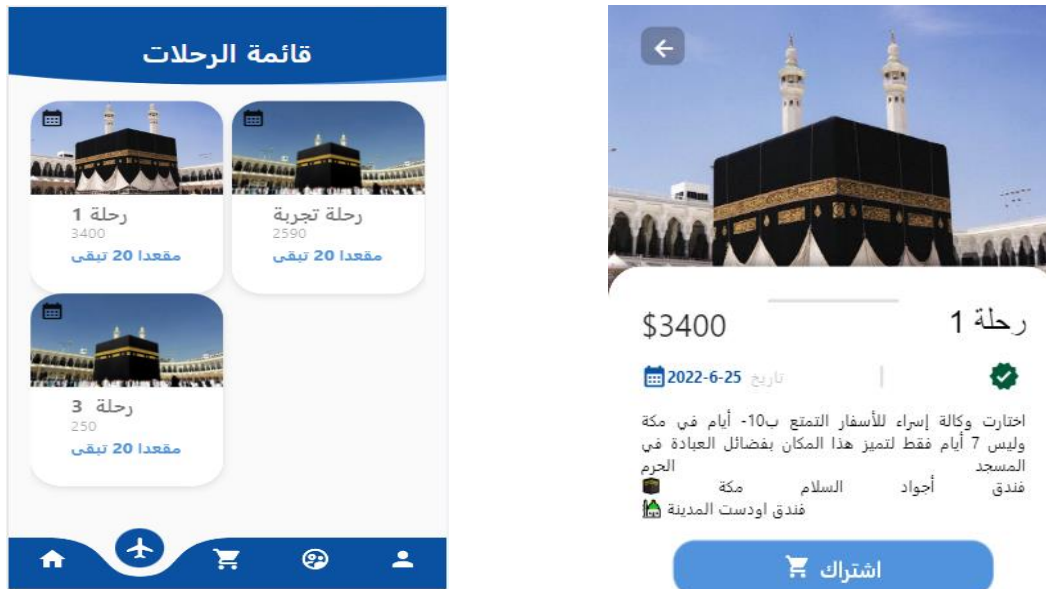


Figure 43 : Interface de Liste package de vols et ses details

3.2.4. Interface de formulaire de participation à un package.

Si le utilisateur veut participer à un package qui lui convient, il est demandé alors de cliquer sur le bouton 'اشترك' et remplir le formulaire qui va apparaitre avec des détails de son passeport.

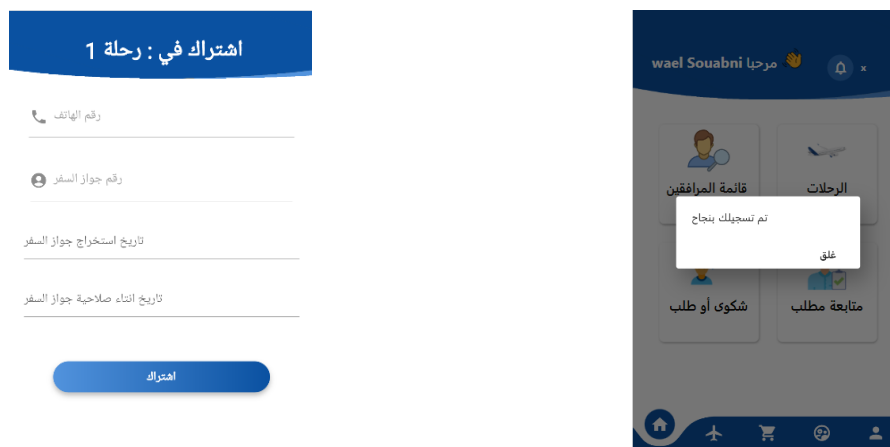


Figure 44 : Formulaire d'inscription au package Omra

3.2.5. Interface de Historique de participation

Si l'utilisateur veut consulter la liste de ses participations, il n'a que de cliquer sur l'icône de milieu de son écran. Et la liste s'affichera avec la date de sa participation.

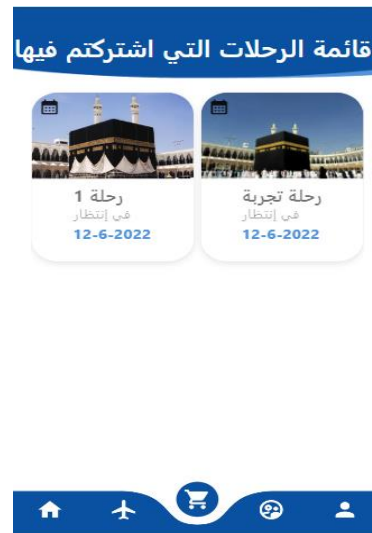


Figure 45: liste d'historique de participations

3.2.6. Interface de Liste des accompagnateurs

Lorsqu'un utilisateur appuie sur l'icône de groupe d'utilisateurs dans le menu qui, une liste de tous d'accompagnateurs va s'afficher avec leur cordonnée, leur numéro de contact et leur numéro de WhatsApp s'il dispose.

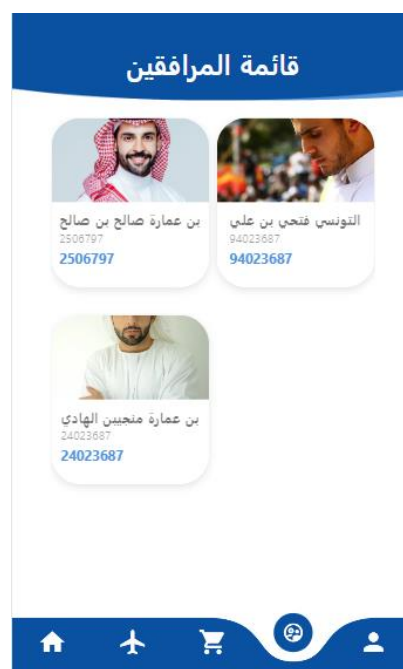


Figure 46 : Interface liste d'accompagnateur

3.2.7. Interface d'envoyer un message

On cas de besoin d'annoncer une réclamation, de donner un feedback ou d'apprendre un renseignement, cette application permet à ce client de le faire. Il suffit seulement d'appuyer sur le dernier élément de son menu et puis choisir 'ترك رسالة'.



Figure 47 : Interface profil et envoyer un message

3.3. Application Mobile Accompagnateur

L'accompagnateur peut participer à un package ou d'ajouter un pèlerin. Son authentification, affichage de liste de packages ou liste de son historique, c'est identique à celle de pèlerin.



Figure 48 : Interface d'authentification

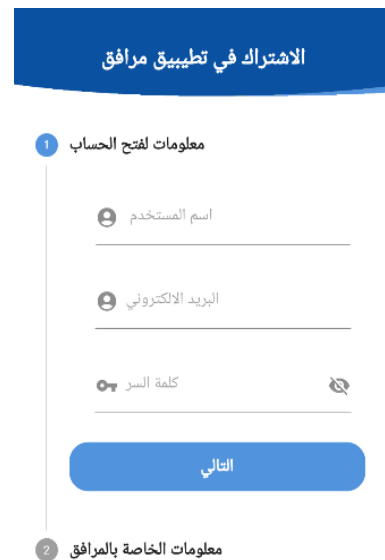


Figure 49 : Interface d'inscription

3.3.1. Interface de détail package et ajouter un pèlerin

A partir de l’affichage du détail d’un package, l’accompagnateur peut se participer ou s’inscrire à un package.

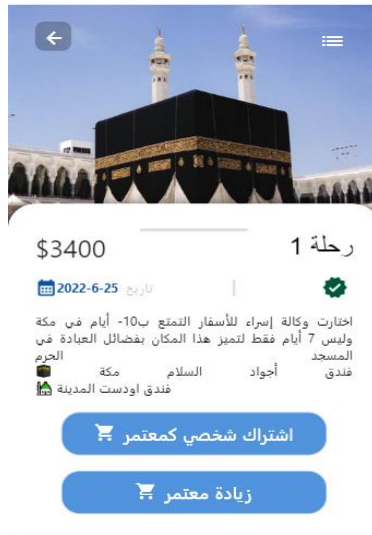


Figure 50 : Interface de détail du package

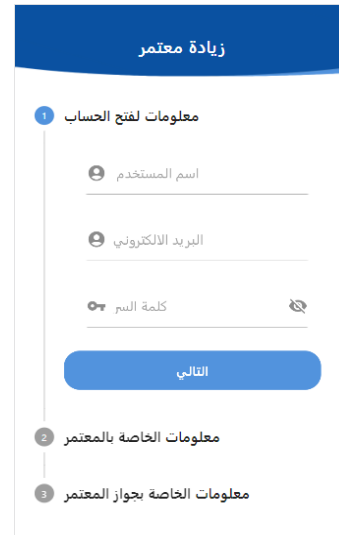


Figure 51 : Interface d'ajout de pèlerin

Et comme d'autres détails qu'un accompagnateur peu les voir la liste des pèlerins .Ainsi que cette application mobile permet à leur utilisateur de voir le détail de package sur un calendrier comme montres les interfaces suivant :

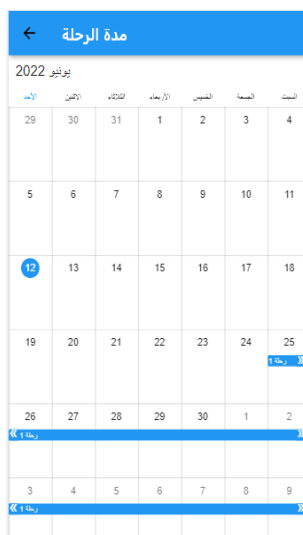


Figure 52 : Interface de calendrier



Figure 53 : interface de liste de pèlerins par vol

3.4. Api rest

Dans le but de créer une relation entre l'application web et mobile, j'ai utilisé le Framework laravel pour créer mes api par la création des méthodes dans des contrôleurs et assigner une URL pour eux dans Api.php sous le dossier route. Puis j'ai les consommer avec la bibliothèque http de flutter.

Conclusion

Dans ce chapitre, j'ai présenté le projet réalisé durant ce stage .Ainsi que j'ai présente les technologies et les outils qui ont m'aider à cette réalisation.

Conclusion

Le travail que j'ai présenté dans ce rapport, j'ai réalisé dans le cadre du mon stage de fin d'étude au sein de l'entreprise Israa Travel Agency a été le fruit d'une longue réflexion, pendant laquelle, il a fallu répondre à la problématique qui a été trouvée.

Il faut avouer que la tâche n'a pas été facile, mais avec ma formation en développement web et mobile, dans le cadre de mastère TICIT, j'avais tous les outils nécessaires pour répondre au problème qui a été rencontré. C'était l'occasion pour moi de consolider mes connaissances théoriques en développement. La résolution du problème est passée par les étapes suivantes : l'analyse des besoins, la conception de mon système et la réalisation.

L'application que j'ai réalisée a comme but de faciliter la tâche pour les agents de l'agence. Ainsi permet au client de découvrir les offre de package de l'Omra.

Comme perspective, je continuerai à travailler sur la simplicité d'utilisation de l'application, en ajoutant le module de communication inter-application pour que je rende la possibilité que la famille de pèlerins peut contacter directement l'accompagnateur et avoir les nouvelle de leur famille.

Bibliographie & Néographie

- 1 : <https://www.commentcamarche.net/contents/1141-introduction-a-uml>
- 2 : https://fr.wikipedia.org/wiki/Diagramme_de_cas_d'utilisation
- 3 : <https://www.ibm.com/docs/fr/rsm/7.5.0?topic=uml-sequence-diagrams>
- 4: <https://laurent-audibert.developpez.com/Cours-UML/?page=diagramme-classes>
- 5 : https://fr.wikipedia.org/wiki/Diagramme_de_d%C3%A9ploiement
- 6 : <https://fr.wikipedia.org/wiki/Laravel>
- 7 : <https://doc.laravue.dev/guide/#features>
- 8 : https://www.frandroid.com/android/535194_quest-ce-que-flutter-loutil-permettant-decreer-des-applications-android-et-ios
- 9 : <https://ismvsectioninfo.wordpress.com/2018/11/17/laragon-environnement-dedeveloppement-web/>
- 10 : https://fr.wikipedia.org/wiki/Visual_Studio_Code
- 11 : <https://fr.wikipedia.org/wiki/PhpMyAdmin>
- 12 : <https://progsoft.net/fr/software/advanced-rest-client>
- 13 : <https://kinsta.com/fr/base-de-connaissances/base-de-connaissances-github/>