



RAPPORT DE PROJET DE FIN D'ETUDES
POUR L'OBTENTION D'UN LICENCE PROFESSIONNEL
LIAM

Etude et réalisation d'une plateforme de guide touristique

Réalisé Par :

Amellouk Amine

Labiz Omar

Encadré Par :

Pr. Soussi Abdelmoughit

Devant le jury :

Pr. Soussi Abdelmoughit

encadrant

Année Universitaire 2019-2020

Dédicace

Du profond du cœur, nous dédions ce modeste travail à :

Nos parents,

Aucune dédicace ne saurait exprimer le respect, l'amour éternel et la considération pour les sacrifices que vous avez consentis pour notre instruction et notre bien-être. Nous vous remercions pour tout le soutien et l'amour que vous nous portez depuis notre enfance et nous espérons que votre bénédiction nous accompagne toujours que ce modeste travail soit l'exaucement de vos vœux tant formulés, le fruit de vos innombrables sacrifices. Puisse Dieu, le très haut, vous accorder santé, bonheur et longue vie.

Notre famille et nos chers amis,

Qui nous ont accordé leur soutien dans les instants les plus difficiles.

Tous nos formateurs et l'équipe pédagogique et administrative de
IT-Learning,

Pour l'aide qu'ils ont toujours portée aux étudiants

Toute personne qui de près ou de loin a participé à notre formation.

Et à vos chers lecteurs.

Remerciement

On remercie Dieu, le tout-puissant, notre créateur pour l'intelligence, le courage, la santé et la force qu'il nous a donnée, puisse-t-il toujours nous garder.

On tient à remercier également :

On exprime nos sincères gratitude envers notre encadreur Mr. SOUISSI Abdelmoughit, pour les conseils qu'il n'a cessés de nous prodiguer, sa compréhension et la confiance qu'il a toujours témoignée à notre égard.

Nous remercions aussi Mr. LGADI Taoufik responsable formation et développement pour ces conseils et ces écoutes de nous problèmes tout au long de notre formation au sein du campus d'IT-Learning.

Nous remercions aussi aux membres du jury, vous faites un grand honneur en acceptant de juger ce travail, veuillez trouver ici l'expression de notre grand respect et nos vifs remerciements.

Nous associons à ces remerciements tous les enseignants qui ont contribué à notre formation ainsi qu'à toutes les personnes qui travaillent au sein du IT-Learning à la FST de Settat.

Résumé

L'objectif principal de ce travail de projet de fin d'étude consiste à concevoir et développer une application mobile.

Cette application permet aux touristes de trouver des endroits d'hébergement, restauration, historique et des événements pendant leurs séjours au Maroc. L'application inclue aussi un service web qui permet au personnel administratif de gérer les comptes, les endroits et les réclamations des clients.

La partie mobile est développée avec Android SDK. Tandis que la partie web est développée sous le Framework Angular, en utilisant Firebase.

À travers ce document, nous allons décrire en détail les différentes étapes de réalisation de ce projet.

Mots clés: Android, Angular, Firebase.

Abstract

The main objective of this graduation project work is to design and develop a mobile application.

This application allows tourists to find accommodation, catering, history and events during their stays in Morocco. The application also includes a web service that allows administrative staff to manage customer accounts, locations and complaints.

The mobile part is developed with the Android SDK. While the web part is developed under the Angular Framework, both using Firebase.

Through this document, we describe in detail the different stages of realization of this project.

Keywords: Android, Angular, Firebase.

Sommaire

Liste de figures.....	3
Liste des tableaux	4
Liste des abréviations.....	5
Introduction générale	6
Chapitre I : Cadre général.....	8
Introduction.....	9
1. Positionnement et problématique	9
1.1. Contexte	9
1.2. Problématique.....	9
2. Etude de l'existant	9
2.1. Visit Rabat	9
3. Critique de l'existant.....	11
4. Présentation de la solution Traveliva.....	11
5. Méthodologie de conception	12
6. Conclusion.....	12
Chapitre II : Capture des besoins.....	13
Introduction.....	14
1. Identification des acteurs.....	14
2. Besoins fonctionnels.....	14
3. Besoins non fonctionnels	15
Conclusion	15
Chapitre III : Analyse et conception	16
Introduction.....	17
1. Conception générique.....	17
1.1. Le langage UML	17
1.2. Les diagrammes d'UML.....	17
1.3. Modèles d'UML utilisés.....	19
2. Conception détaillée.....	19
2.1.1. Diagrammes de cas d'utilisation générale	19
2.1.2. Description des cas d'utilisation	20
2.1.3. Diagrammes de séquences.....	22
2.1.4. Diagramme de classes.....	33

3. Conception préliminaire des interfaces - maquettes	33
Conclusion	35
Chapitre IV : Implémentation	36
Introduction.....	37
1. Environnement de développement.....	37
1.1. Environnement matériel	37
1.2. Environnement logiciel	38
2. Technologies.....	40
2.1. Développement mobile.....	40
2.2. Développement web.....	43
3. Outils de développement.....	46
3.1. Outils de Google	46
3.2. Contrôle de version.....	47
4. Présentation des interfaces de l'application.....	48
4.1. Les interfaces de l'application mobile.....	48
4.2. Les interfaces de l'application web.....	50
Conclusion	51
Conclusion générale.....	52
Bibliographie.....	53

Liste de figures

Figure 1: interface d'application Visit Rabat	10
Figure 2: Interface d'application Visit Morocco	11
Figure 3: Diagramme de cas d'utilisation générale	20
Figure 4: Diagramme de séquence – inscription	22
Figure 5: Diagramme de séquence - authentification	23
Figure 6: Diagramme de séquence – modification du profil	23
Figure 7: Diagramme de séquence – recherche d'un lieu.....	24
Figure 8: Diagramme de séquence – affichage d'un lieu sur la cartographie	24
Figure 9: Diagramme de séquence – affichage d'un lieu sur la cartographie	25
Figure 10: Diagramme de séquence – ajouter un lieu	26
Figure 11: Diagramme de séquence – ajouter une note.....	27
Figure 12: Diagramme de séquence – soumission d'une réclamation	28
Figure 13: Diagramme de séquence – suppression d'un utilisateur.....	29
Figure 14: Diagramme de séquence – gestion des réclamations	30
Figure 15: Diagramme de séquence – suppression d'un lieu.....	31
Figure 16: Diagramme de séquence – modification d'un lieu.....	32
Figure 17: Diagramme de classes	33
Figure 18: Maquette – interface d'accueil	34
Figure 19: Maquettes – interface de détails	34
Figure 20: Maquettes – interface de soumission des avis	35
Figure 21: Architecture générale d'Android.....	40
Figure 22: Le mode de fonctionnement du MVC	43
Figure 23: Architecture générale d'Angular	44
Figure 24: Interface d'accueil - mobile.....	48
Figure 25: Interface détails - mobile	49
Figure 26: Interface d'authentification -web.....	50
Figure 27: Interface de gestion des lieux – web	50

Liste des tableaux

Table 1: Identification des acteurs.....	14
Table 2: Cas d'utilisation d'authentification.....	21
Table 3: Gestion des utilisateurs.....	21
Table 4: Gestion des soumissions	21
Table 5: Gestion des réclamations.....	22
Table 6: Environnement matériel – ordinateurs	37
Table 7: Environnement matériel - téléphones.....	37

Liste des abréviations

CPT : Conseil Préfectoral du Tourisme

UML : Unified Modeling Language

MVC : Model-View-Controller

Introduction générale

Le tourisme est accepté comme l'une des plus grandes industries de services au monde, ayant connu une expansion et une diversification continues. Selon l'Organisation Mondiale du Tourisme (OMT), les arrivées de touristes internationaux ont augmenté de 4% en 2019 pour atteindre un total de 1,5 milliard, confirmant le tourisme comme un secteur économique leader et résilient

Au Maroc l'activité touristique a clôturé l'année avec un total des arrivées aux postes frontières d'environ 13 millions de touristes, ce qui correspond à une évolution positive de +5,2% en 2019, comparativement à l'année 2018, selon l'Observatoire du tourisme.

De nos jours, l'internet domine le monde en lançant de nouvelles technologies qui ont imposé à l'utilisateur de suivre le rythme de son développement. En plus, le mobile fait une partie intégrante de notre vie

Le développement rapide du secteur du tourisme au Maroc doit être équilibré par l'évolution de la technologie et des installations soutenues. Maintenant, l'utilisation d'une technologie de smartphone est l'une des choses importantes et fait partie des activités quotidiennes des gens modernes

Sur la base du nombre d'utilisateurs de smartphones dans le monde, en 2020 il y a 3,5 milliard utilisateurs au monde, ce qui fait 45.15% de la population mondiale selon Statista, le chiffre pour le Maroc est à 75.7% des marocains selon l'Agence Nationale de Réglementation des Télécommunications (ANRT) (chiffre 2019)

La transformation culturelle imposée par la digitalisation croissante des activités change les habitudes des voyageurs et des touristes. Le smartphone. Que l'on soit sur iOS ou Android, les smartphones sont omniprésents et il est aujourd'hui impossible de voyager sans son précieux téléphone. Or, ce changement de comportement a un impact sur les professionnels du tourisme, car les nouvelles générations voyagent différemment par rapport à leurs aînés. Entre liberté, flexibilité et mobilité, les applications mobiles ont révolutionné le monde du tourisme.

Plusieurs organisations et sociétés ont déjà pris l'initiative de se lancer dans ce tout nouveau secteur qu'est l'e-tourisme. Par exemple le Conseil préfectoral du tourisme (CPT) de Meknès à lancer une application mobile "VISITMEKNES" qui permet que visiteur d'organiser leur séjour, ainsi que d'autre ville dispose maintenant d'une application mobile

Au début de nos recherches et réflexions sur le sujet, il avait été question de réaliser une plateforme touristique en version mobile qui englobe toutes les régions du royaume afin que les touristes n'ont pas un besoin de télécharger une application pour chaque région, Après discussion lors la réalisation du cahier des charges, nous sommes arrivés à comprendre l'ampleur de l'aspect de ce projet, certes une idée audacieuse que nous aimions, mais en commençant par se concentrer sur une ville, puis l'expansion serait plus gérable pour un groupe de deux. Mais nous ne pouvions pas refuser ce défi, c'est pourquoi permettre aux utilisateurs de contribuer serait d'une grande aide pour faire évoluer cette application.

De là, avons décidé afin d'approfondir nos connaissances vers la création des applications mobiles et web de réaliser une application qui soit riche et réalisable à l'aide de techniques de développement tels que Android, iOS et Angular, et de faire une application qui permet aux touristes de découvrir les endroits et zones d'intérêt.

Ce présent rapport sera structuré en chapitres suivants :

Dans le premier chapitre <<Cadre général>>, nous définissons notre projet avec la problématique qu'il fait face, étudions d'autres solutions existantes et les critiquons afin de proposer notre solution.

Dans le chapitre <<Capture des besoins>>, nous identifions les acteurs du futur système, les besoins fonctionnels et non fonctionnels ainsi que le diagramme de cas d'utilisation générale.

Dans le chapitre <<Analyse et conception>>, nous traitons l'aspect conceptuel de notre application.

Dans le chapitre <<Implémentation>>, nous commencerons par une description de l'environnement travail, les technologies et outils de développements utilisés. Ensuite, nous présenterons des interfaces de l'application mobile et web.

En fin, nous clôturons ce mémoire par une conclusion dans laquelle nous résumons notre solution et exposons quelques perspectives futures.

Chapitre I : Cadre général

Introduction

Au cours de ce chapitre, nous allons introduire notre projet en étudiant son cadre général et la problématique en face à réaliser cette application. Nous allons critiquer les solutions existantes pour dégager leurs insuffisances et proposer les orientations de notre solution future.

La présentation de la solution souhaitée nous mène à bien analyser les différentes méthodologies de travail existantes afin de dégager celle la plus adéquate à notre projet.

1. Positionnement et problématique

1.1. Contexte

Le projet entre dans le cadre de préparation de projet de fin d'études pour l'obtention de la licence professionnelle en ingénierie des applications mobiles de l'IT-Learning Campus du FST Settat. Il nous a été demandé de concevoir et réaliser une application mobile qui va aider les touristes l'hors de leur séjour au royaume.

1.2. Problématique

Pour créer une application mobile qui fournira des services à un touriste. Un touriste a besoin de technologies modernes qui peuvent lui servir. L'application mobile doit donc être facile à utiliser et efficace pour gérer les activités de voyage. En raison de l'indisponibilité d'une application disponible pour l'ensemble du royaume, les touristes sont confrontés à des problèmes lors de leurs déplacements au Maroc. Ils doivent payer un beau budget de voyage aux guides et agents locaux pour obtenir des informations et doivent utiliser différentes applications pour chaque région.

Le guide touristique approprié n'est pas disponible, ce qui pourrait détecter un emplacement actuel, calculer la distance et fournir des lignes directrices, de nombreux endroits et emplacement locales ne sont pas connues des touristes

2. Etude de l'existant

La réalisation de tout projet doit être précédée par une étude de l'existant qui détermine les points faibles et les points forts des systèmes actuels et les besoins du client en vue de les prendre en considération lors de la conception et de la réalisation. Dans cette section nous mène alors à une étude basée sur les observations de différentes applications web spécialisées dans le domaine de l'e-tourisme. Cette étude nous permet de dégager et donner leurs atouts et leurs faiblesses afin de déterminer les besoins et les traiter.

2.1. Visit Rabat

Visit Rabat est une application mobile disponible sur Android et iOS lancer par le CPT de Rabat, qui offre la possibilité de planifier votre voyage à Rabat ou dans sa région, ce guide de voyage est une répertoire de tous les sites et monuments, et sert à vous informer de toutes

manifestations et évènements qui s'y déroulent, la localisation et accès à un descriptif complet. Mais quelques endroits et emplacements peuvent ne pas être listés et cette application est utile seulement à la région de Rabat.

Visit Rabat ressemble aux plusieurs applications mobiles lancées par les CPT des autres villes portant le nom Visit « Nom de la ville » et qui servent la même fonction.



Figure 1: interface d'application Visit Rabat

2.2. Visit Morocco

VisitMorocco est l'application officielle du tourisme au Maroc, disponible seulement sur Android, qui permet de découvrir de près nos destinations, télécharger des brochures, avoir des informations nécessaires et accéder à une galerie. Une simple recherche sur le Play Store résulte en une autre application portant le même nom mais développée par une personne différente, ainsi que la version d'application met en question sur l'actualité des informations fournies.



Figure 2: Interface d'application Visit Morocco

3. Critique de l'existant

Après l'aperçu précédent, il est maintenant possible de comparer les fonctionnalités fournies des applications étudiées et de conclure qu'il y a une réponse à cela, mais les nouveaux touristes peuvent être confus en devant télécharger une nouvelle application dans chaque ville qu'ils visitent, et également pour mentionner que tout le monde n'est pas impatient de télécharger une application pour une ville qu'il pourrait juste passer et y passer une journée pour se reposer avant de passer à l'autre, une autre chose à mentionner est que certaines villes ne fournissent pas d'application similaire. Le besoin d'une application à jour est une clé pour améliorer le secteur du tourisme au Maroc, nous envisageons même les locaux et les touristes de manière égale pour aider à l'améliorer en soumettant leurs activités ou d'autres attractions touristiques locales.

4. Présentation de la solution Traveliva

Après avoir détaillé notre problématique et bien analysé les différentes solution existante au Maroc, la question suivant doit se poser naturellement :

Est-ce que les applications mobiles existantes peuvent représenter une solution qui répond à notre problème ?

Les applications disponibles sur le marché sont limitées et parfois incomplètes. Cependant, nous avons besoin d'une solution spécifique, pour tout le royaume et qui répond parfaitement aux besoins attendus par les touristes. Par conséquent, notre solution consiste à développer un système informatique composée de :

- ✓ Application mobile permettant aux touristes d'accéder à un ensemble des données relatif aux différents endroits
- ✓ Application web concernant la gestion centralisée de toutes les données du système à développer.

5. Méthodologie de conception

Une méthode de conception est une démarche générale reflétant une philosophie de présentation et de suivi du système. Elle propose des outils spécifiques permettant un suivi efficace de l'information relative au système.

Notre choix se porte sur le langage UML (Unified Modeling Language) qui facilite l'interactivité avec la base de données à l'aide des diagrammes de cas d'utilisation, diagrammes de séquence et diagrammes de classes.

6. Conclusion

Dans ce premier chapitre nous avons spécifié le contexte, fait une étude de l'existant et discuter la méthodologie à suivre que nous permettant de bien entamer notre projet.

Chapitre II : Capture des besoins

Introduction

Le présent chapitre présente une analyse qui joue rôle important dans la démarche ergonomique du développement d'une application.

Il a donc comme objectif de permettre de formaliser les étapes préliminaires de développement d'un système afin de rendre ce développement plus fidèle aux besoins pré définis sur le cahier de charges qui sont divisés en besoins fonctionnels, besoins non fonctionnels, identification des acteurs et leur cas d'utilisations.

1. Identification des acteurs

Un acteur est une entité externe qui interagit avec un système. En réponse à l'action d'un acteur, le système fournit un service qui correspond à son besoin.

Notre solution a 2 acteurs principaux, le tableau ci-dessous présente les acteurs et leurs rôles :

Acteur	Rôle
Administrateurs	<ul style="list-style-type: none">- S'authentifier- Gérer les utilisateurs- Valider les soumissions- Répondre aux réclamations- Affichage des statistiques
Touristes	<ul style="list-style-type: none">- S'authentifier- Consulter les endroits et zones- Proposer des endroits et zones- Soumettre un avis- Contacter le support

Table 1: Identification des acteurs

2. Besoins fonctionnels

Le système doit permettre :

Besoin fonctionnels pour la partie mobile

- ✓ Présenter les informations sur les endroits des villes marocains
- ✓ Spécifier les localisations des endroits
- ✓ Consulter la liste des endroits favoris
- ✓ Poser un nouvel endroit touristique
- ✓ Poser un avis
- ✓ Tracer des chemins selon l'emplacement et les besoins du touriste

Besoin fonctionnels pour la partie web

- ✓ Gérer les endroits
- ✓ Gérer les utilisateurs
- ✓ Répondre aux réclamations

3. Besoins non fonctionnels

Les besoins non fonctionnels décrivent les objectifs liés aux performances du système et aux contraintes de son environnement. Ses exigences techniques sont souvent exprimées sous forme d'objectifs spécifiques que doit atteindre le système

Les besoins non fonctionnels de cette application sont les suivants :

- ✓ Assurer une cohérence et une crédibilité aux informations stockées dans la base de données.
- ✓ L'ergonomie des interfaces de l'application mobile et web.
- ✓ Sécurité
- ✓ Qualité
- ✓ Accessibilité et performance
- ✓ Renouvellement
- ✓ Fiabilité

Conclusion

À travers ce chapitre, nous avons définis les acteurs qu'on considère principales et indiquer les fonctionnalités que notre système doit permettre et celles qu'ils permettent d'augmenter la performance de l'application pour répondre à nos besoins.

Chapitre III : Analyse et conception

Introduction

Dans cette partie nous traitons l'aspect conceptuel de notre application. Pour la conception et la réalisation de cette dernière, nous nous appuyons sur le formalisme UML basé sur les diagrammes et offrant une flexibilité marquante.

1. Conception générique

La phase de conception permet de décrire de manière non ambiguë, le plus souvent en utilisant un langage de modélisation, le fonctionnement futur du système, afin d'en faciliter la réalisation.

1.1. Le langage UML



Pour faciliter notre tâche nous avons recours au langage de modélisation unifié UML, c'est une notation qui permet de modéliser un problème de façon standard. Ce langage qui est né de la fusion de plusieurs méthodes existantes auparavant est devenu une référence en terme de modélisation objet, UML est caractérisé par :

- C'est langage formel et normalisé.
- Il permet le gain de précision, encourage l'utilisation d'outils et constitue à cet effet un gage de stabilité.
- Est un support de communication performant.
- Il cadre l'analyser et facilite la compréhension de représentations abstraites complexes.
- Son caractère polyvalent et sa souplesse en font un langage universel.

1.2. Les diagrammes d'UML

UML 2.3 s'articule autour de 14 types de diagrammes, chacun d'eux étant dédié à la représentation des concepts particuliers d'un système logiciel. Ces types de diagrammes sont répartis en trois grands groupes :

– Diagrammes de structure

- Diagramme de classes : représentation des classes intervenant dans le système.
- Diagramme d'objets : représentation des instances de classes (objets) utilisées dans le système.
- Diagramme de composants : représentation des composants du système d'un point de vue physique, tels qu'ils sont mis en œuvre
- Diagramme de déploiement : représentation des éléments matériels (ordinateur, périphériques, réseaux, systèmes de stockage...) et la manière dont les composants du système sont répartis sur ces éléments matériels et interagissent entre eux.
- Diagramme des paquets : représentation des dépendances entre les paquets (un paquet étant un conteneur logique permettant de regrouper et d'organiser les éléments dans le modèle UML), c'est-à-dire entre les ensembles de définitions.
- Diagramme de structure composite : représentation sous forme de boîte blanche des relations entre composants d'une classe
- Diagramme de profils : spécialisation et personnalisation pour un domaine particulier d'un meta-modèle de référence d'UML

– Diagrammes de comportement

- Diagramme des cas d'utilisation : représentation des possibilités d'interaction entre le système et les acteurs (intervenants extérieurs au système), c'est-à-dire de toutes les fonctionnalités que doit fournir le système.
- Diagramme états-transitions: représentation sous forme de machine à états finis du comportement du système ou de ses composants.
- Diagramme d'activité : représentation sous forme de flux ou d'enchaînement d'activités du comportement du système ou de ses composants.

– Diagrammes d'interaction

- Diagramme de séquence : représentation de façon séquentielle du déroulement des traitements et des interactions entre les éléments du système et/ou de ses acteurs.
- Diagramme de communication : représentation de façon simplifiée d'un diagramme de séquence se concentrant sur les échanges de messages entre les objets
- Diagramme global d'interaction : représentation des enchaînements possibles entre les scénarios préalablement identifiés sous forme de diagrammes de séquences.

- Diagramme de temps : représentation des variations d'une donnée au cours du temps).

1.3. Modèles d'UML utilisés

Les diagrammes qu'on a montrés dans la partie précédente sont des moyens de description des objets ainsi que des liens qui les relient. Nous n'utilisons pas les 14 types de diagrammes proposés par UML 2.3, mais seulement un tiers parmi eux en insistant particulièrement sur les diagrammes suivants :

- Diagramme de cas d'utilisation
- Diagramme de séquence
- Diagramme de classes

Cette limitation est largement suffisante pour la plupart des projets et particulièrement pour notre travail.

2. Conception détaillée

2.1.1. Diagrammes de cas d'utilisation générale

Le cas d'utilisation décrit le comportement du système du point de vue de l'utilisateur sous la forme d'actions et de réactions. Un cas d'utilisation indique une fonctionnalité du système déclenché par un acteur externe au système.

Le diagramme ci-dessous montre l'ensemble des cas d'utilisation et décrit les exigences fonctionnelles du système. Chaque cas d'utilisation correspond donc à une fonction métier du système, selon le point de vue de l'un de ses acteurs.

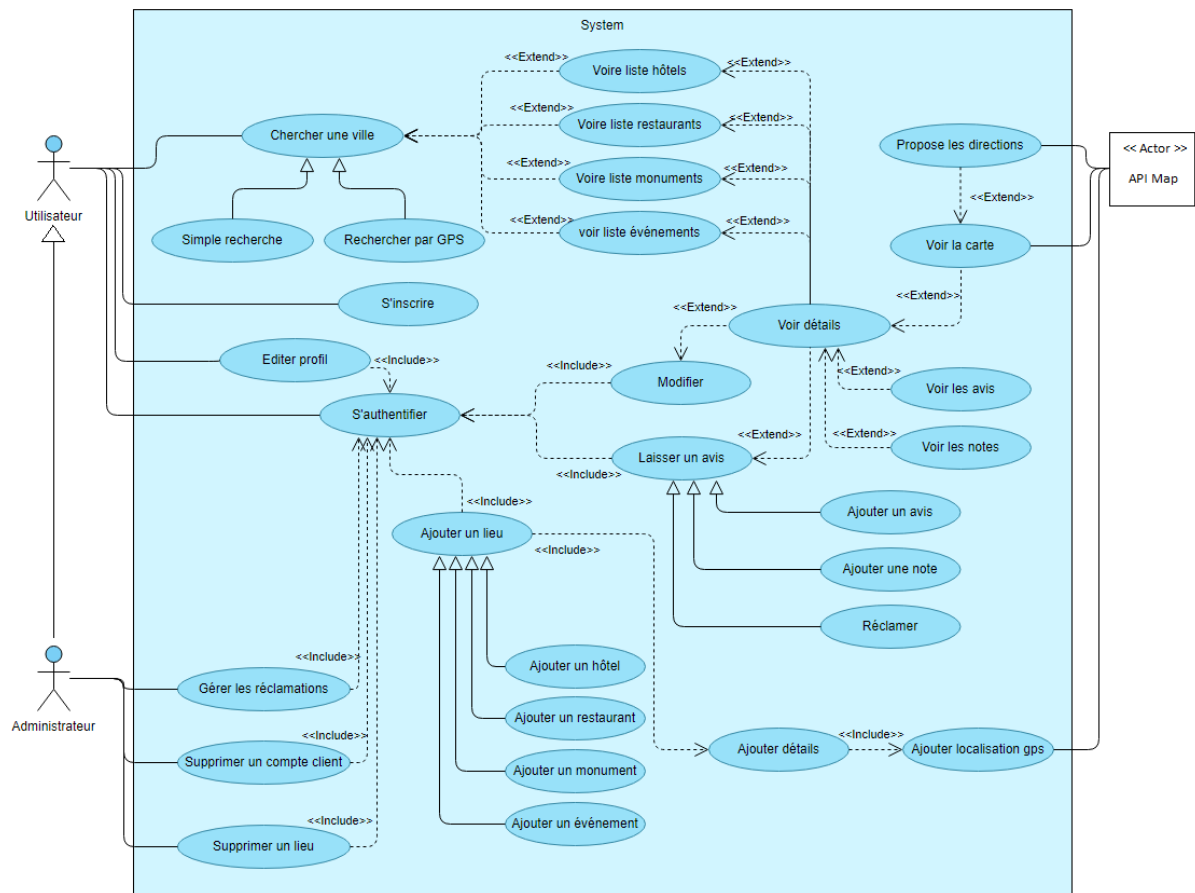


Figure 3: Diagramme de cas d'utilisation générale

2.1.2. Description des cas d'utilisation

Dans cette section, nous représenterons une description détaillée de chaque cas d'utilisation. Il s'agit de définir les scénarios possibles.

1.1.1.1. Cas d'utilisation <<authentification>>

Les acteurs de notre application doivent s'authentifier afin d'accéder aux fonctionnalités de l'application. Le tableau ci-dessous décrit en détail les scénarios possibles de l'authentification.

Nom de cas	Authentification
Brève description	-Permet à un acteur de s'authentifier avant d'accéder à l'application
Enchaînement principal	-S'authentifier par un login et un mot de passe -Le login et le mot de passe sont corrects. -Accès à l'application.
Enchaînement alternatif 1	-S'authentifier par un login et un mot de

	passe -Le login et le mot de passe sont corrects. -Ressaisie du login et le mot de passe.
--	---

Table 2: Cas d'utilisation d'authentification

1.1.1.1. Gestion les utilisateurs

Nom de cas	Création d'une proposition
Objectifs	Permettre aux administrateurs de modifier le rôle des utilisateurs
Acteur concernés	Administrateur
Pré condition	-Authentification
Scénario alternatif 1	-Touriste crée un nouveau compte -Administrateur peut lui modifier son rôle qu'est par default touriste

Table 3: Gestion des utilisateurs

1.1.1.2. Gestion des soumissions

Nom de cas	Soumettre une soumission
Objectifs	Permettre au touriste de soumettre une attraction touristique
Acteur concernés	Touriste, administrateur
Pré condition	-Authentification
Scénario alternatif 1	-Le touriste remplis les informations requis pour la soumission et valide -L'administrateur peut voir la liste des soumissions, et éventuellement valider la soumission ou la refuser -En cas de validation la soumission est ajoutée à la liste des activités touristique que les touristes peuvent consulter -En cas de refus, la soumission n'est pas accepter et supprimer

Table 4: Gestion des soumissions

1.1.1.3. Gestion des réclamations

Nom de cas	Soumettre une réclamation
Objectifs	Permettre au touriste de signaler des erreurs, demander des réponses et poser des questions
Acteur concernés	Touriste, administrateur
Pré condition	-Authentification
Scénario alternatif 1	-Le touriste remplis les informations requis par le formulaire pour la soumission et valide

	-L'administrateur se charge de résoudre son problème
--	--

Table 5: Gestion des réclamations

2.1.3. Diagrammes de séquences

Le diagramme de séquence montre les interactions entre les objets, agencées en séquence dans le temps, il montre en particulier les objets participant à l'interaction par lignes de vie et les messages qu'ils s'échangent ordonnancés dans le temps, mais il ne montre pas les relations entre les objets. Il peut servir à illustrer un cas d'utilisation.

Nous présentions, ci-après, les diagrammes de séquences que nous avons jugés importants.

2.1.3.1. Diagramme de séquence relative à l'inscription

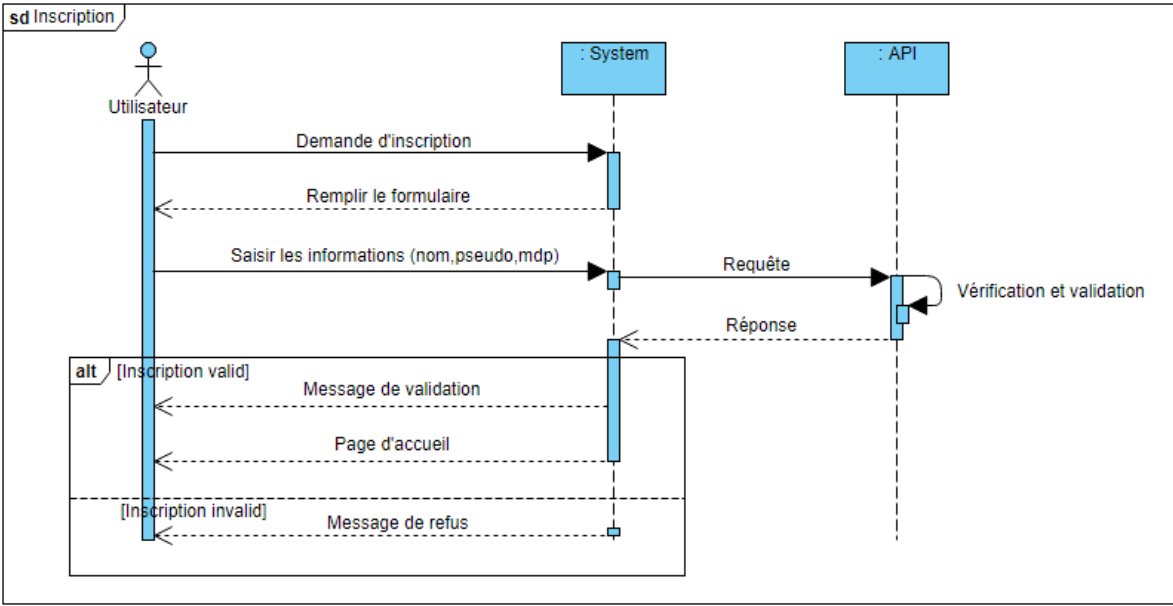


Figure 4: Diagramme de séquence – inscription

2.1.3.2. Diagramme de séquence relative à l'authentification

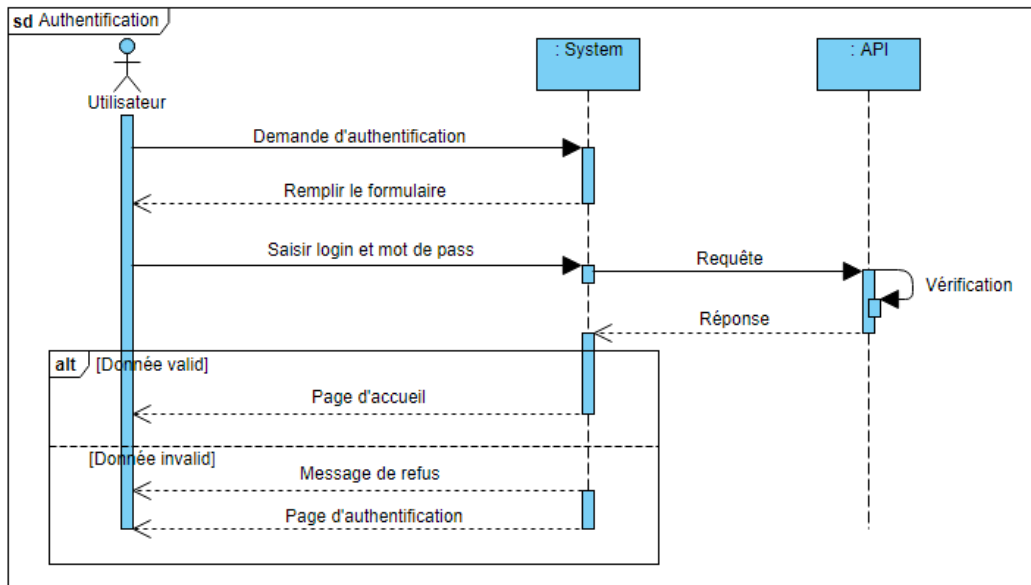


Figure 5: Diagramme de séquence - authentification

2.1.3.3. Diagramme de séquence relative à la modification du profil

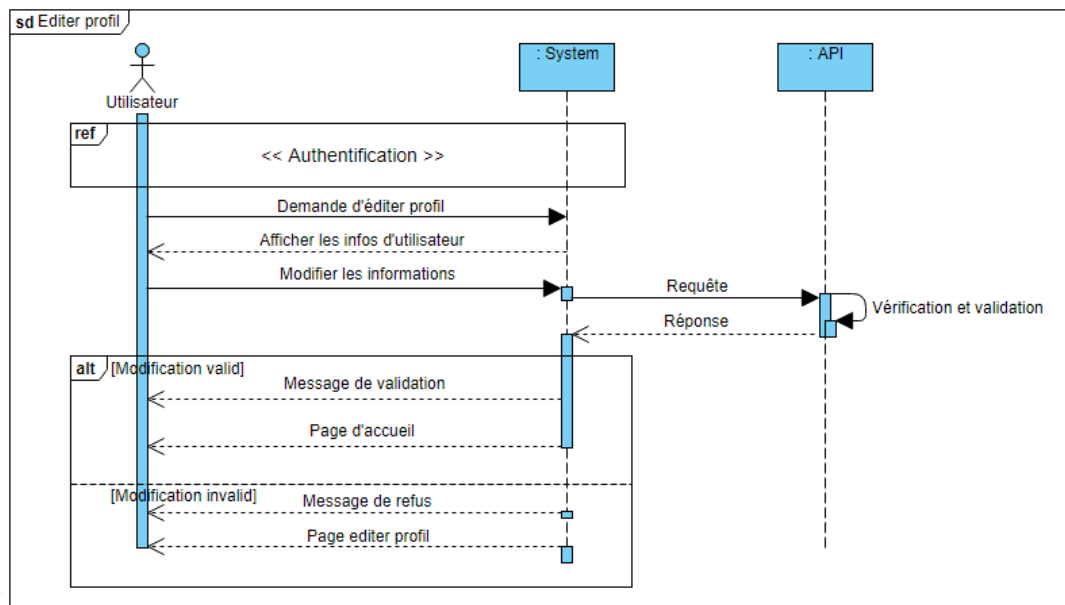


Figure 6: Diagramme de séquence – modification du profil

2.1.3.4. Diagramme de séquence relative à la recherche d'un lieu

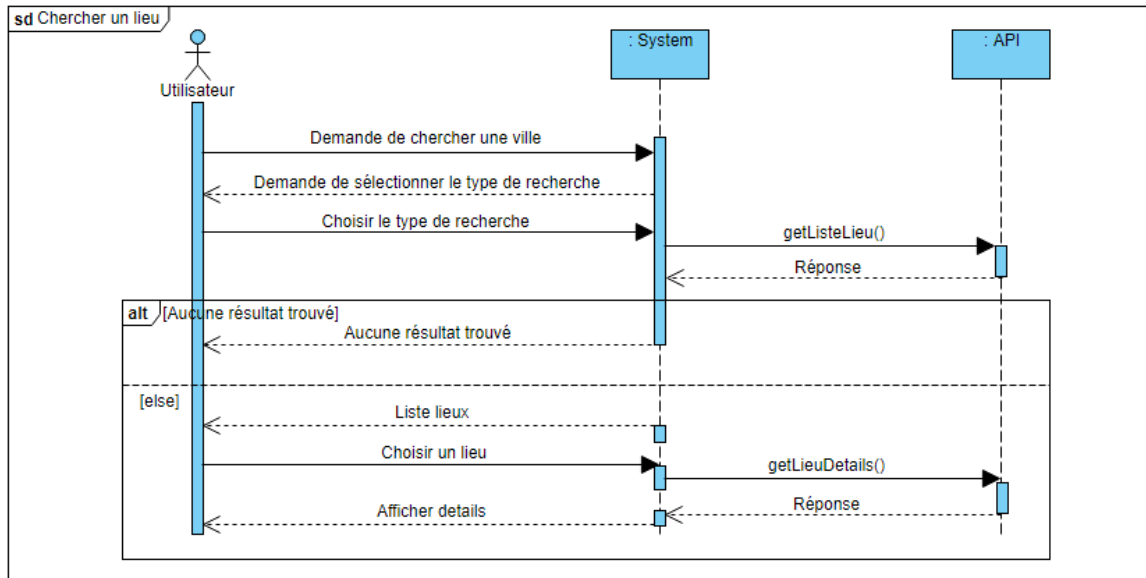


Figure 7: Diagramme de séquence – recherche d'un lieu

2.1.3.5. Diagramme de séquence relative à l'affichage d'un lieu sur la cartographie

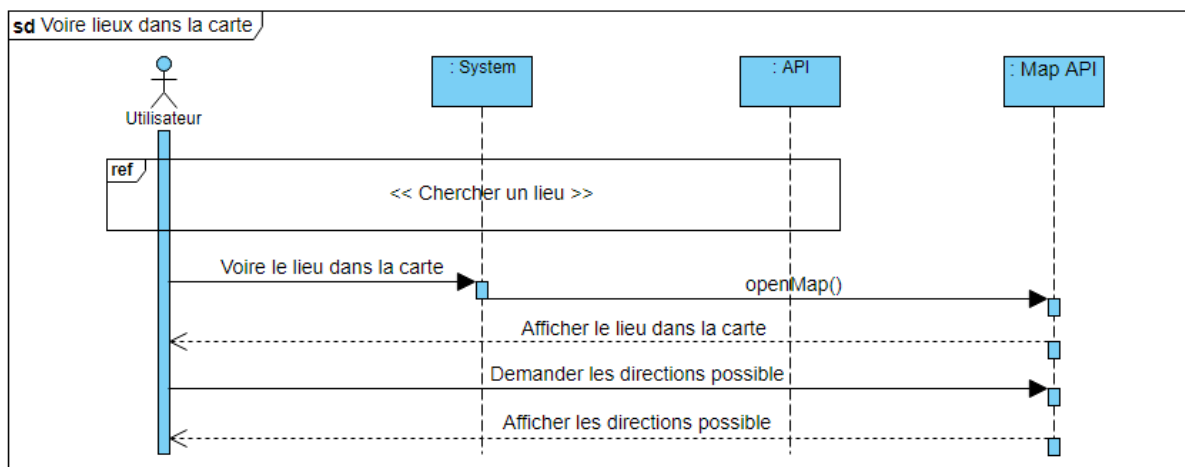


Figure 8: Diagramme de séquence – affichage d'un lieu sur la cartographie

2.1.3.6. Diagramme de séquence relative à l'ajout d'un lieu

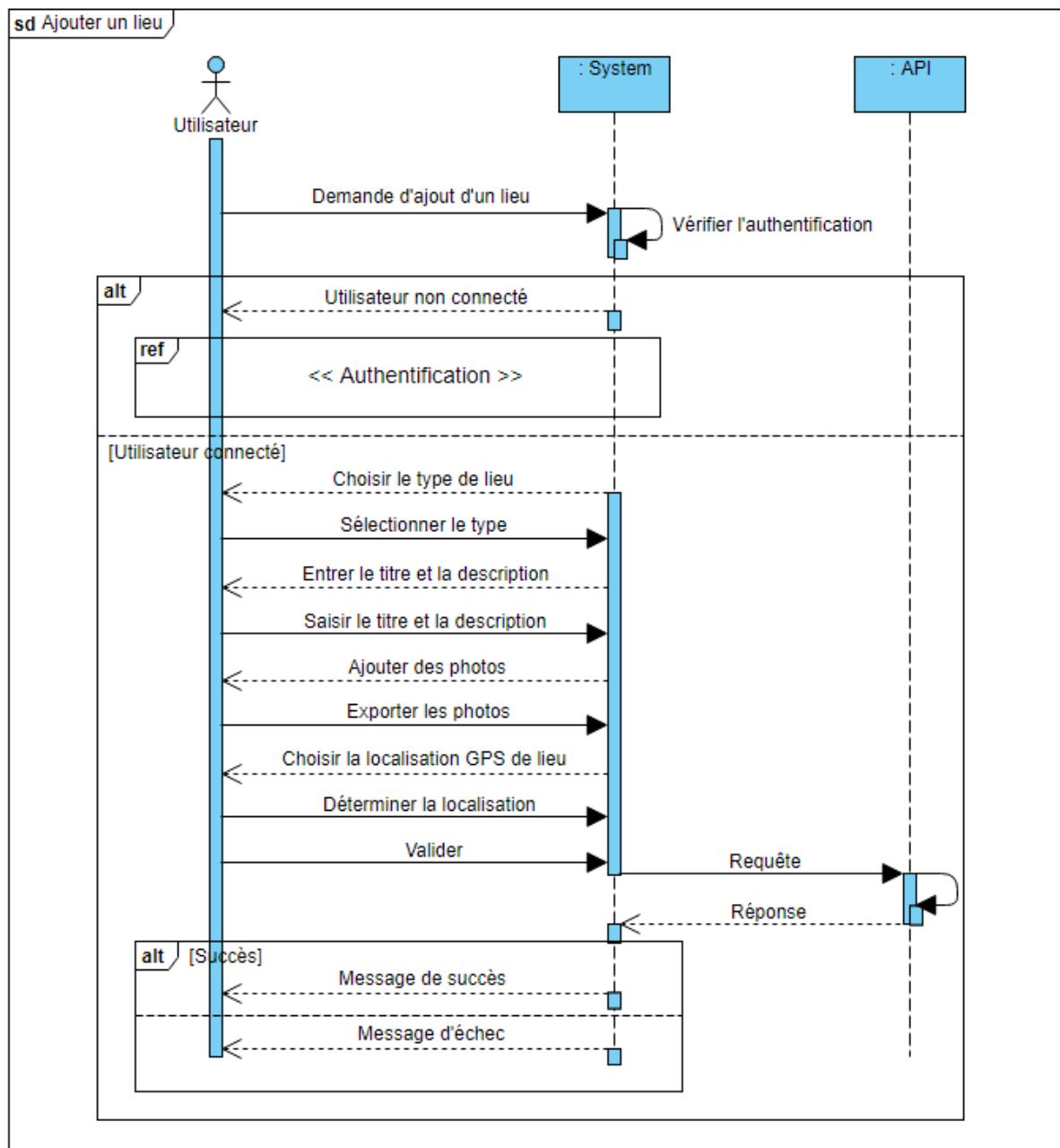


Figure 9: Diagramme de séquence – affichage d'un lieu sur la cartographie

2.1.3.7. Diagramme de séquence relative à l'ajout d'un avis

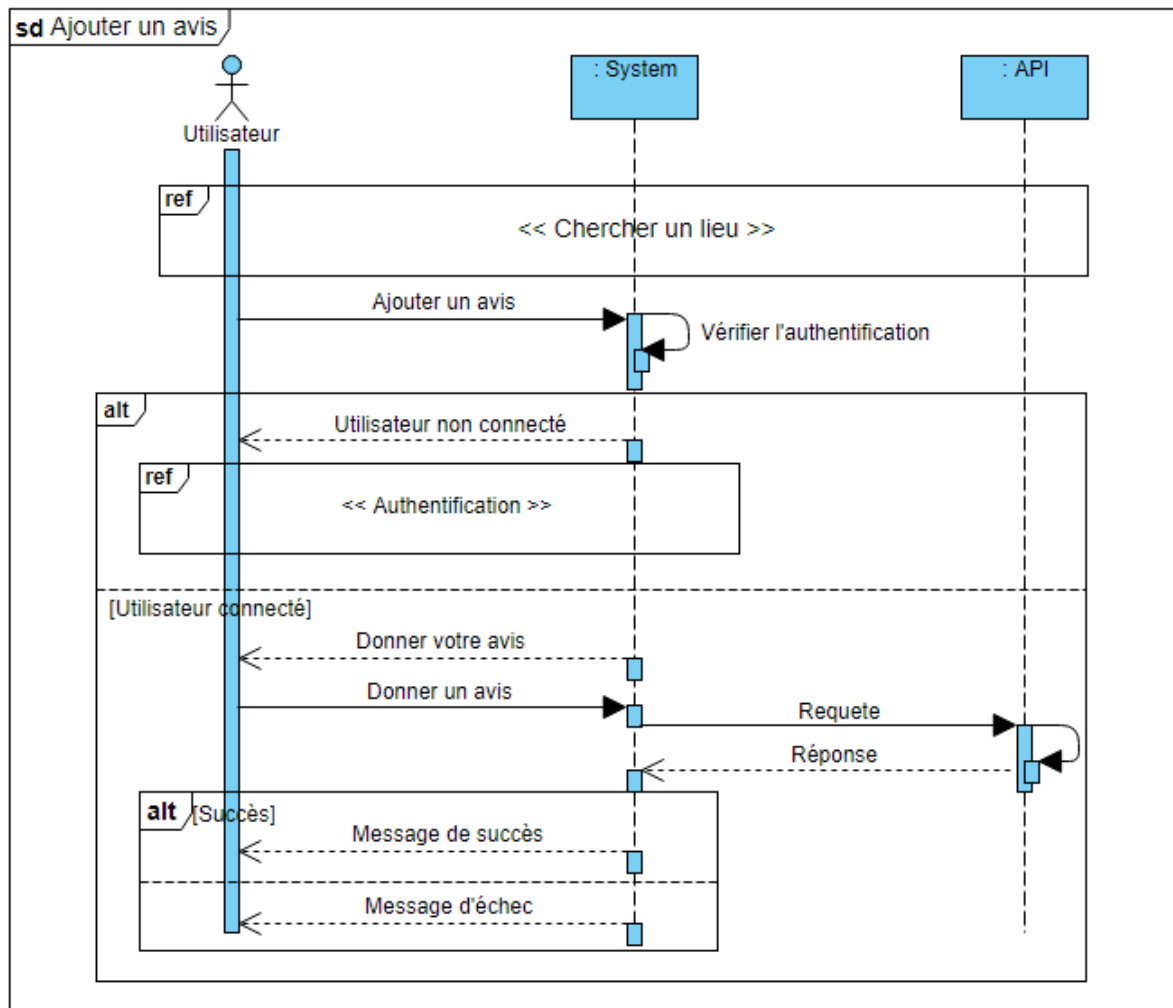


Figure 10: Diagramme de séquence – ajouter un lieu

2.1.3.8. Diagramme de séquence relative à l'ajout d'une note

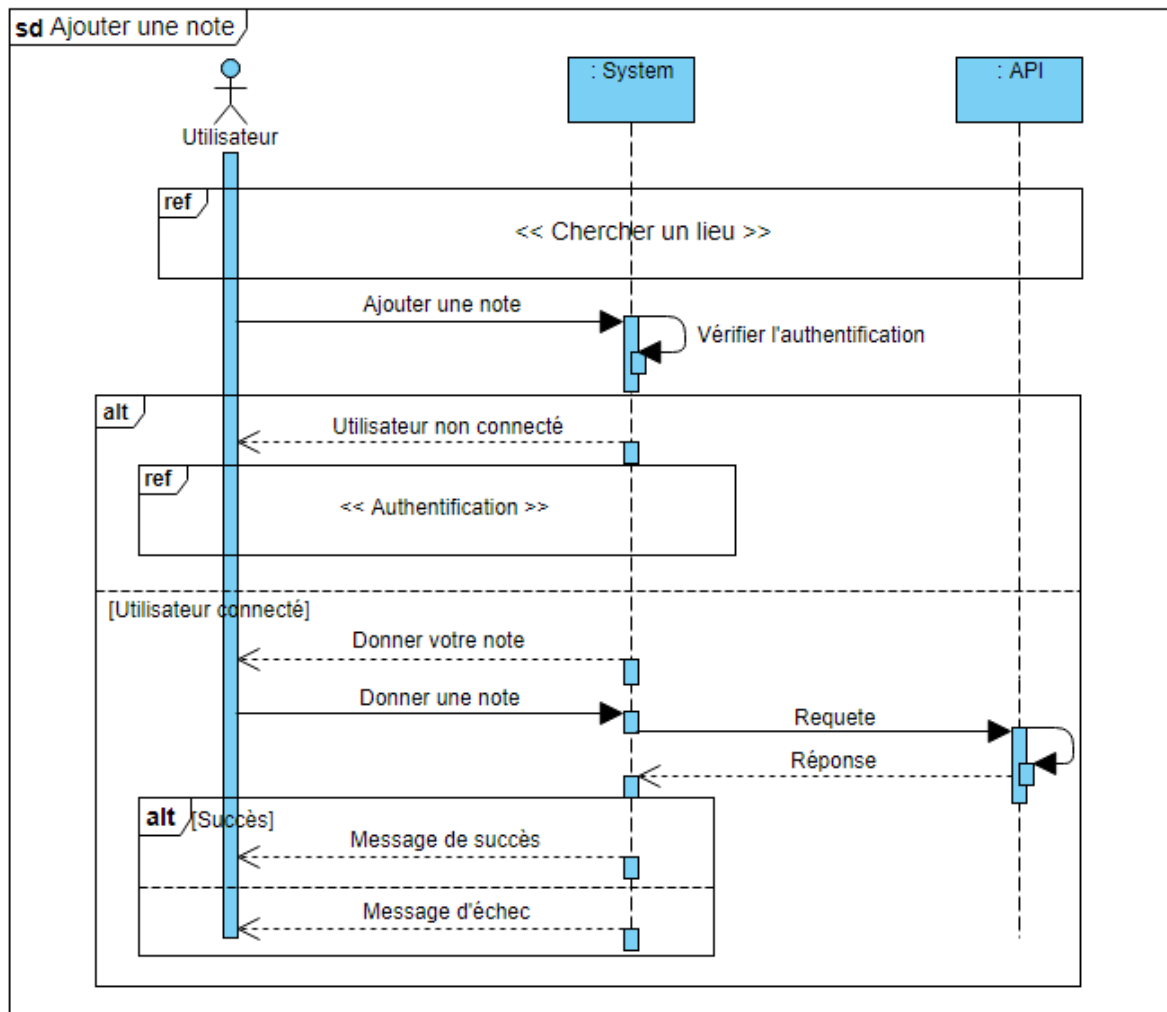


Figure 11: Diagramme de séquence – ajouter une note

2.1.3.9. Diagramme de séquence relative à une soumission d'une réclamation

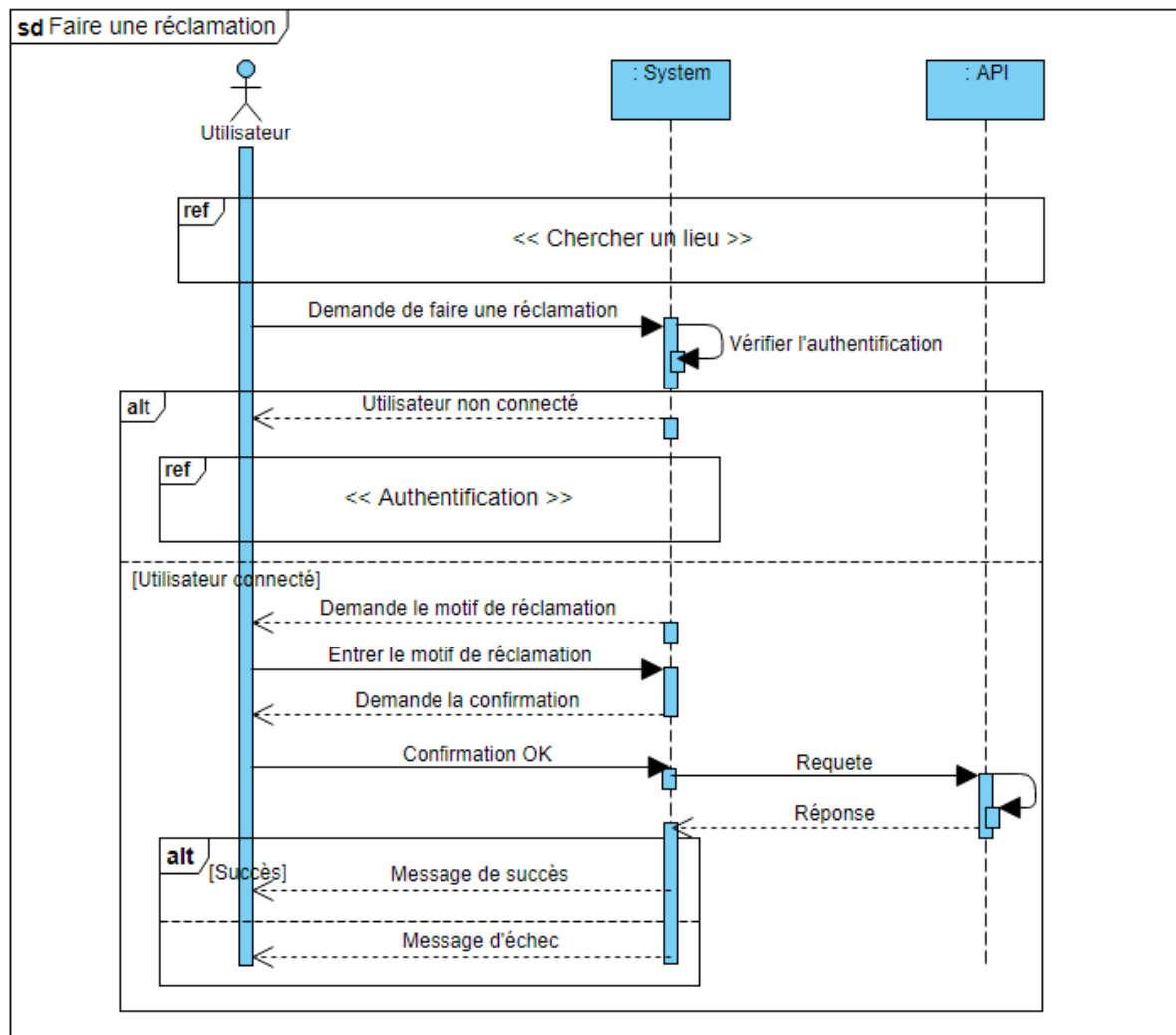


Figure 12: Diagramme de séquence – soumission d'une réclamation

2.1.3.10. Diagramme de séquence relative à la suppression d'un utilisateur

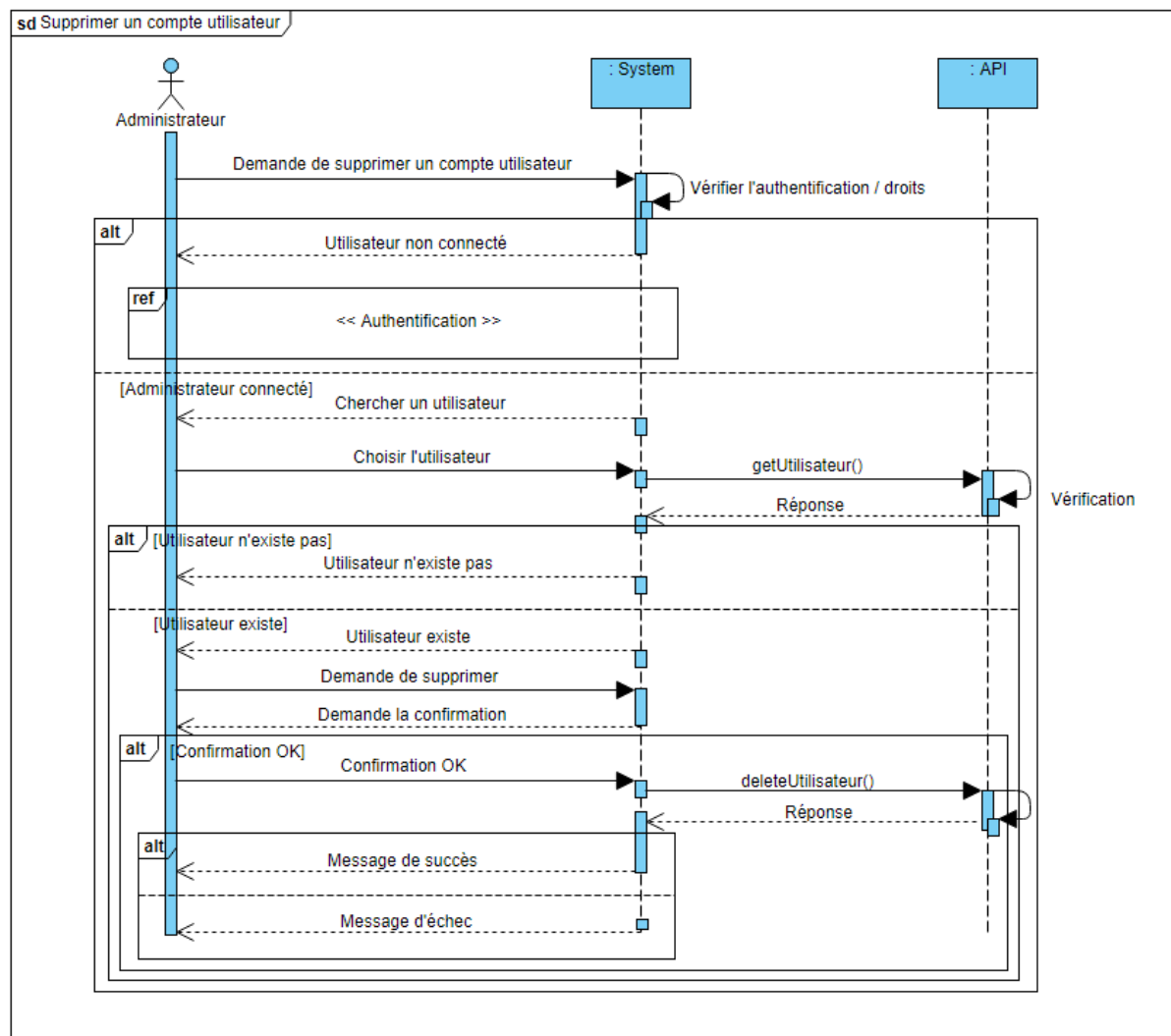


Figure 13: Diagramme de séquence – suppression d'un utilisateur

2.1.3.11. Diagramme de séquence relative à la gestion des réclamations

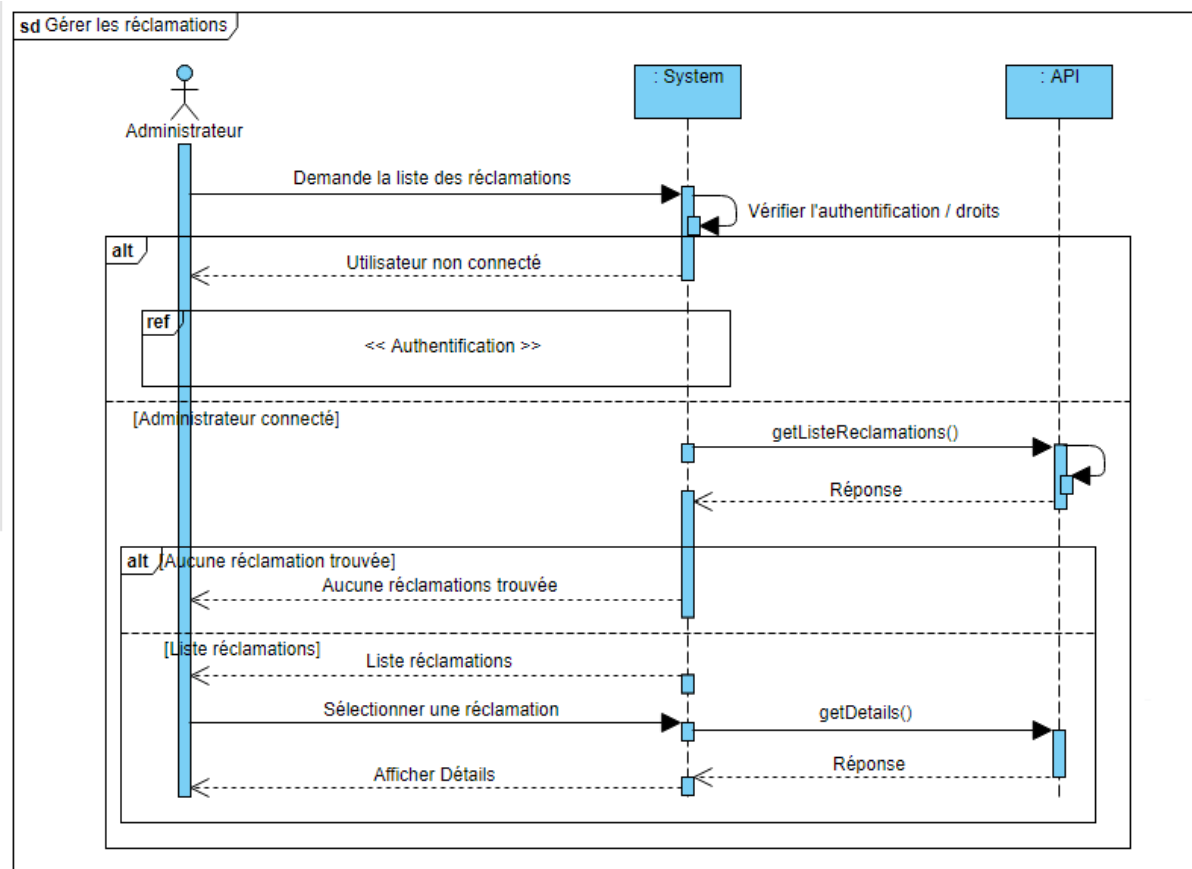


Figure 14: Diagramme de séquence – gestion des réclamations

2.1.3.12. Diagramme de séquence relative à suppression d'un lieu

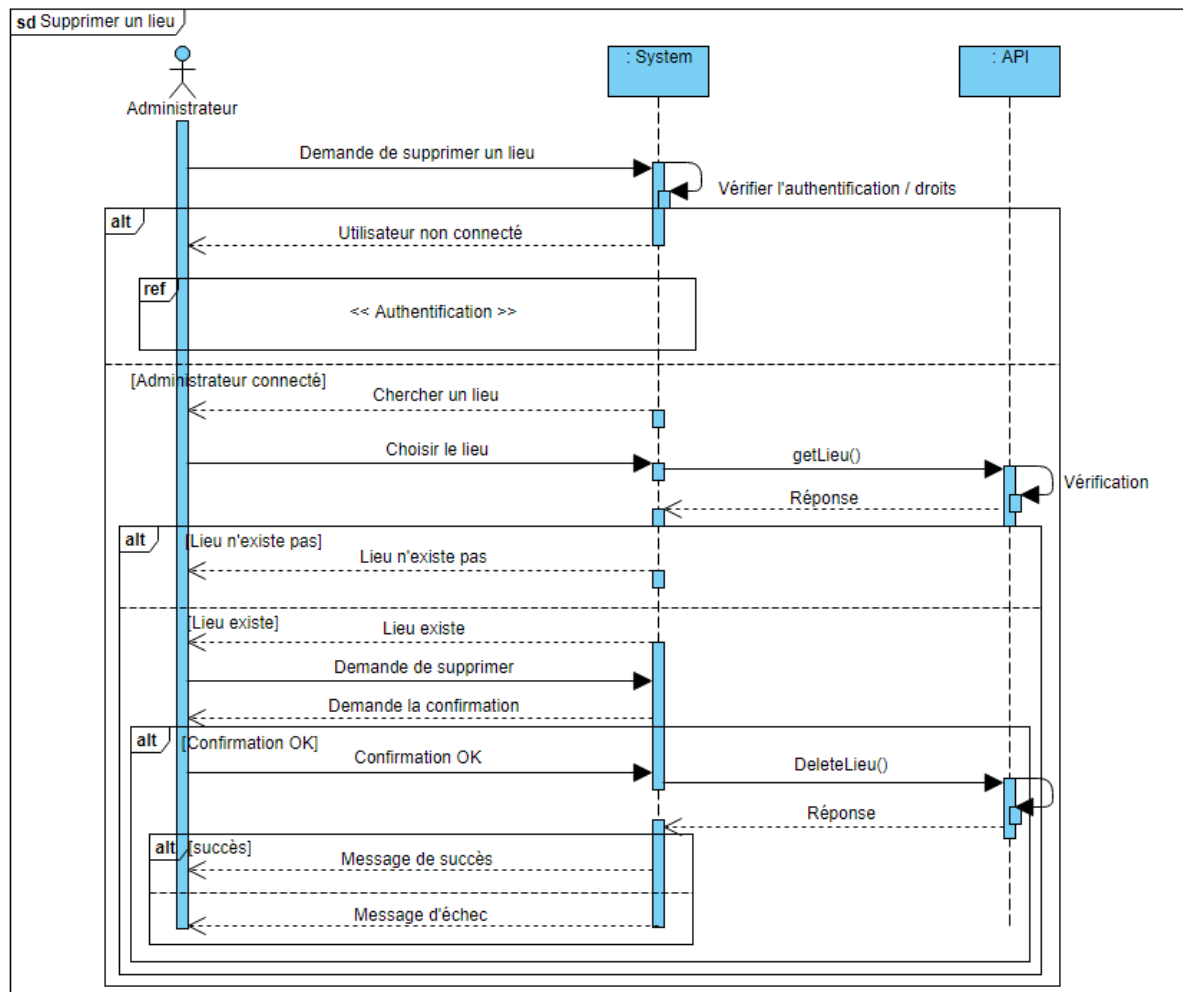


Figure 15: Diagramme de séquence – suppression d'un lieu

2.1.3.13. Diagramme de séquence relative à la modification d'un lieu

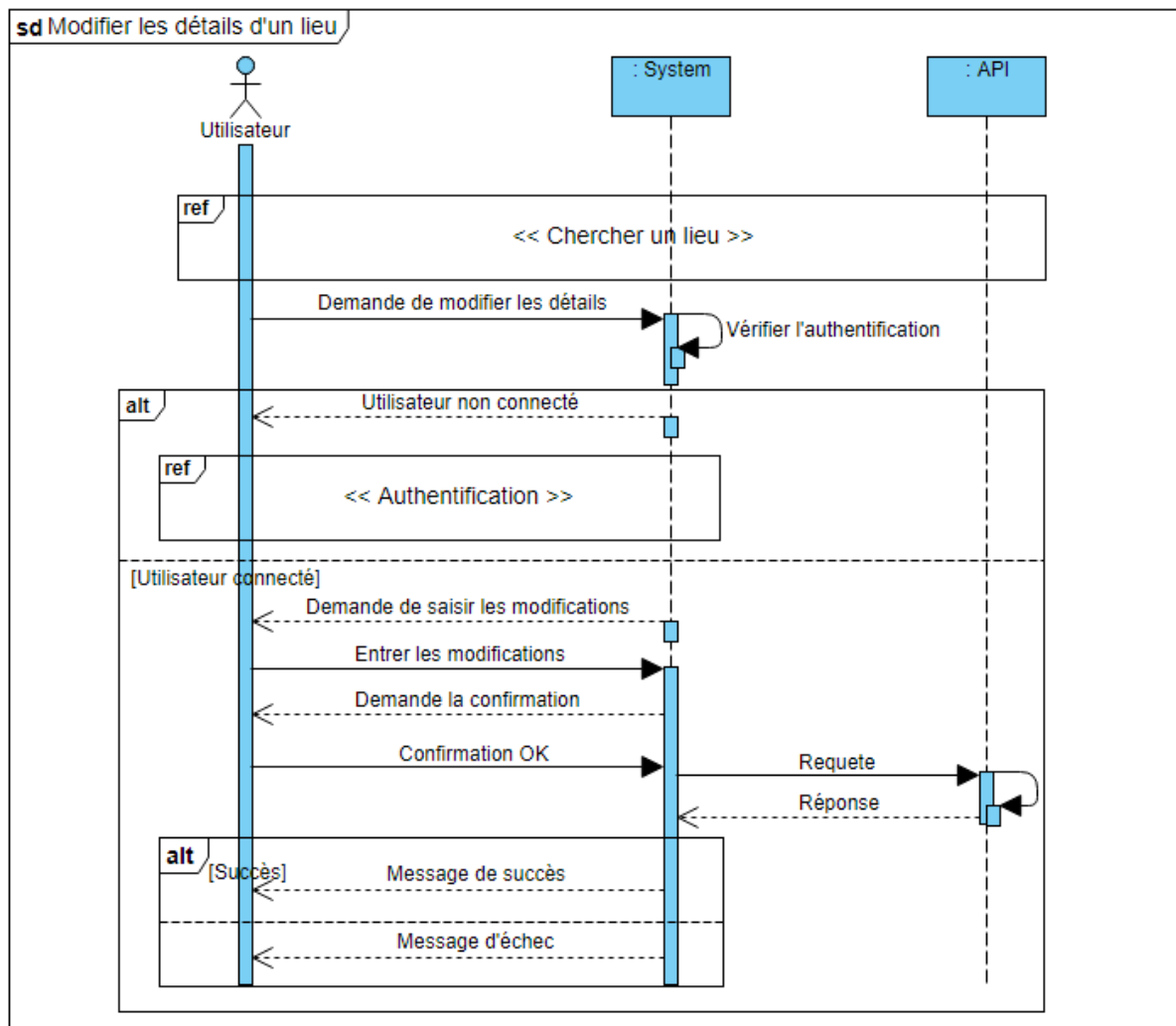


Figure 16: Diagramme de séquence – modification d'un lieu

2.1.4. Diagramme de classes

Le diagramme de classe est une représentation statique des éléments qui composent un système et leurs relations. La figure suivante représente les classes intervenant dans le système.

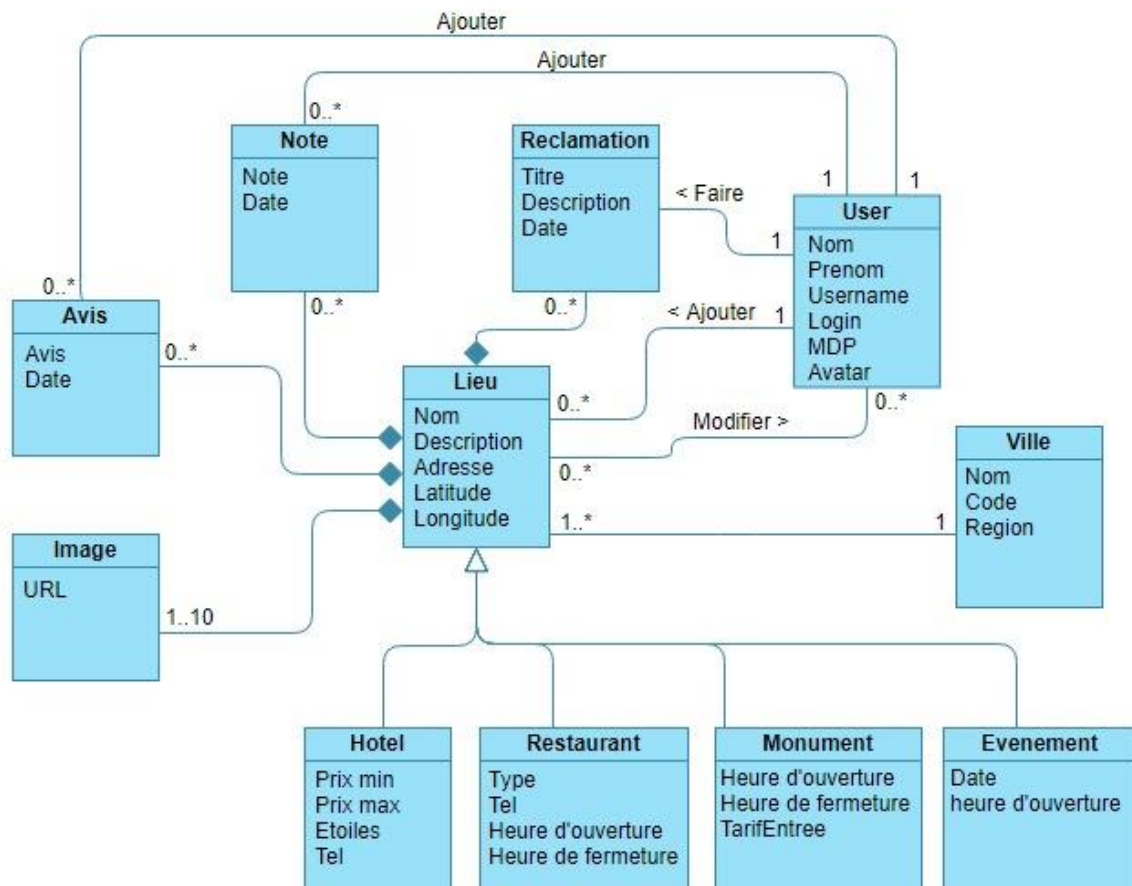


Figure 17: Diagramme de classes

3. Conception préliminaire des interfaces - maquettes

Une maquette est un produit jetable donnant aux utilisateurs une vue concrète mais non définitive de la future interface de l'application. Elle est développée rapidement afin d'améliorer la relation développeur-client. Les maquettes de notre application sont les suivantes :

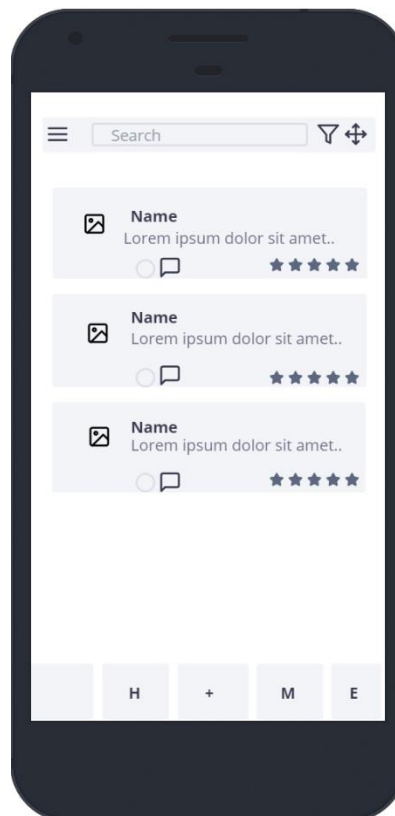


Figure 18: Maquette – interface d'accueil

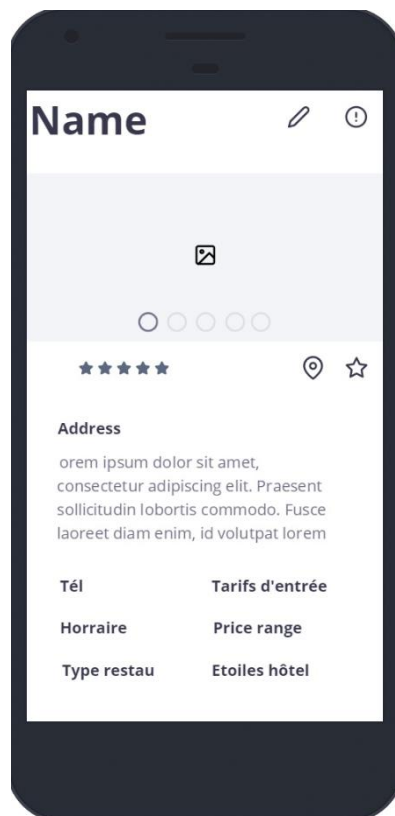


Figure 19: Maquettes – interface de détails

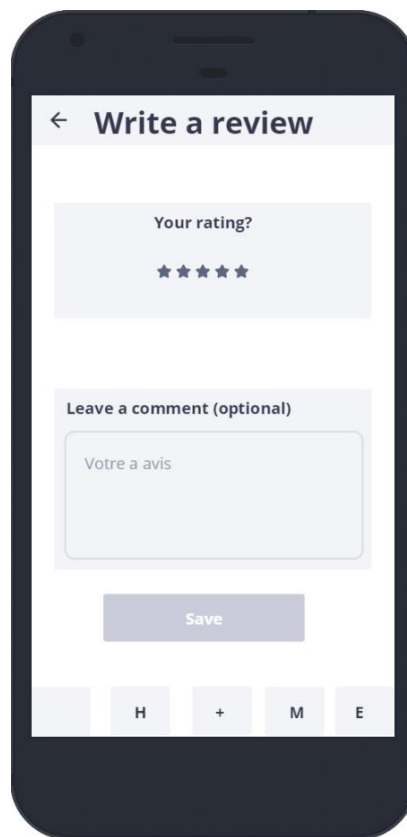


Figure 20: Maquettes – interface de soumission des avis

Conclusion

Dans ce chapitre, nous avons présenté les différents diagrammes définis par UML qui ont permis de bien comprendre les besoins du système à développer ainsi que les différentes interactions entre les objets participant à son fonctionnement, chose qui facilitera la phase d'implémentation et de codage.

Chapitre IV : Implémentation

Introduction

En détaillant ce chapitre, nous sommes déjà passé par les étapes nécessaires à fin d'entraîner la phase de la réalisation. Le problème a été profondément analysé, nous avons défini une conception complète à notre jugement. Une conception qui comporte et décrit tous les besoins de l'application.

Dans ce chapitre nous commencerons par la description de l'environnement de développement ainsi que les différentes technologies utilisées et outils de développement. Ensuite, nous présenterons les différents aspects de fonctionnement de notre application.

1. Environnement de développement

Dans cette partie, nous présentons les différents outils matériels et logiciels nécessaires pour le développement de notre application.

1.1. Environnement matériel

Pour développer l'application, nous avons utilisé comme environnement matériel deux ordinateurs portables et deux téléphones portables qui possèdent les caractéristiques suivant :

Marque	HP	HP
Processeur	Intel Core i5	Intel Core i3
Mémoire	8 GO	12 GB
Système d'exploitation	Windows 10 Professionnel	Windows 10 Professionnel

Table 6: Environnement matériel – ordinateurs

Marque	Nokia	Infinix
Processeur	Octa-core 2.0 GHz Cortex-A53	Octa core 1.3 GHz Cortex-A53
Mémoire	3 GB	4 GB
Disque dur	32 GB	32 GB
Système d'exploitation	Android 10	Android 8.1

Table 7: Environnement matériel - téléphones

1.2. Environnement logiciel

– Ora



Ora est un logiciel de gestion de projet et de suivi du temps pour les développeurs de produits, les startups et les équipes logicielles où l'agilité rencontre la convivialité et l'expérience intuitive.

– Visual Paradigm



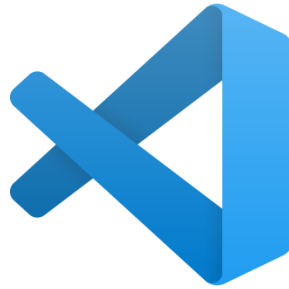
Visual Paradigm est un logiciel de création de diagrammes dans le cadre d'une programmation. Tout en un, il possède plusieurs options permettant une large possibilité de modélisation en UML.

– Android Studio



Android Studio est un environnement de développement pour développer des applications Android. Il est basé sur IntelliJ IDEA, mis gratuitement à la disposition des développeurs Android accompagné de sa documentation complète. Il est créé par Google pour remplacer l'ancien IDE eclipse

— Visual Studio Code



Visual Studio Code est un éditeur de code développé par Microsoft en 2015. Il est un produit open source et gratuit, et surtout disponible sur les systèmes d'exploitations Windows, Linux et Mac. Développé avec le framework Electron et conçu principalement pour développer des projets avec Javascript, Node.js ou encore TypeScript.

— Gradle



Gradle est un moteur de production fonctionnant sur la plateforme Java. Il permet de construire des projets en Java, Scala, Groovy voire C++.

— NPM



NPM, est l'abréviation de Node Package Manager, qui est un outil gérant les bibliothèques de programmation JavaScript pour Node.js.

2. Technologies

2.1. Développement mobile

2.1.1. Présentation de la plateforme Android



Android doit son nom à la startup éponyme spécialisée dans le développement d'applications mobiles rachetée par Google en août 2005, nom venant lui-même d'«androïde» qui désigne un robot construit à l'image d'un être humain.

2.1.2. Architecture Android

Android est basé sur un noyau linux 2.6, au-dessus l'architecture générale :

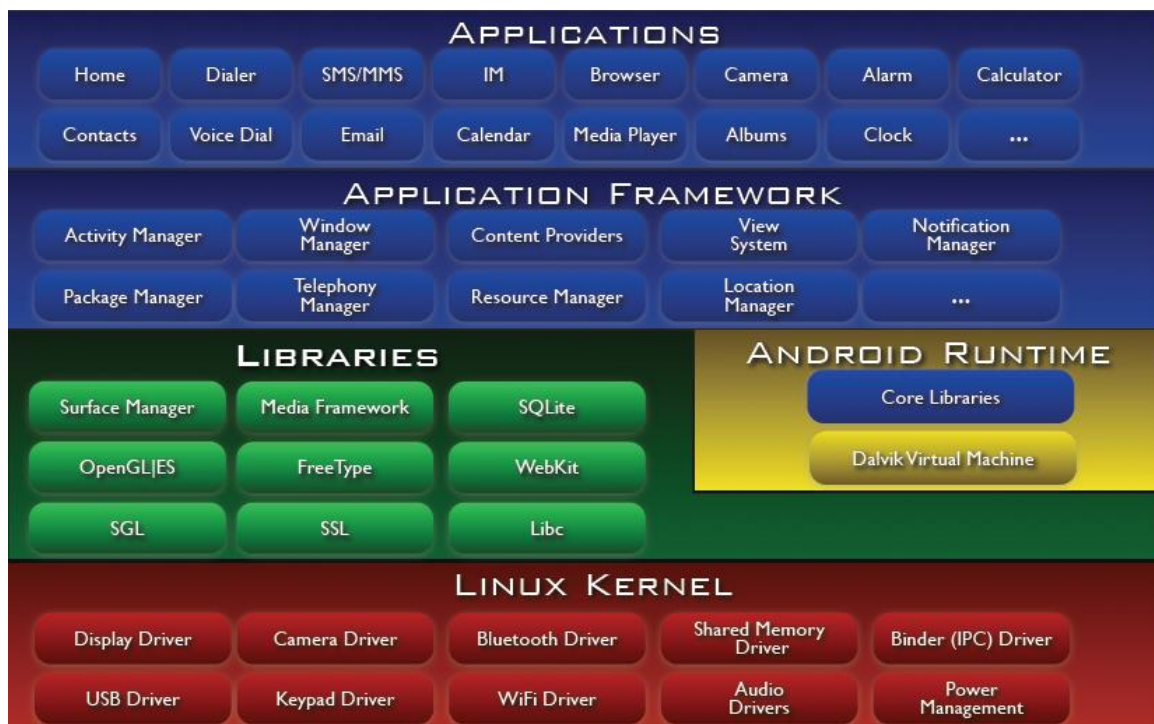


Figure 21: Architecture générale d'Android

– Applications

Android est fourni avec un ensemble de programmes de base permettant d'accéder à des fonctionnalités comme les courriels, les SMS, les photos, pour ne citer que quelques exemples. Ces applications sont développées à l'aide du langage de programmation Java. Pour l'utilisateur final, c'est la seule couche accessible et visible.

– Application Framework

En fournissant une plateforme de développement ouverte, Android offre aux développeurs la possibilité de créer des applications extrêmement riches et innovants. Les développeurs sont libres de profiter du matériel périphérique, les informations de localisation d'accès, exécuter les services d'arrière-plan, définir des alarmes, et d'autres de plus.

– Librairies

En interne, Android inclut un ensemble de bibliothèques C et C++ utilisées par de nombreux composants de la plateforme Android. Ces bibliothèques sont en réalité accessibles au développeur par l'intermédiaire du framework Android. En effet, le framework Android effectue, de façon interne, des appels à des fonctions C/C++ plus rapides à exécuter que des méthodes Java standard.

– Android Runtime

Android inclut un ensemble de bibliothèques qui fournit la plupart des fonctionnalités disponibles dans les bibliothèques de base du langage de programmation Java.

– Linux Kernel

Android repose sur un noyau Linux qui gère les services du système, comme la sécurité, la gestion de la mémoire et des processus, la pile réseau et les pilotes. Il agit également comme une d'abstraction entre le matériel et la pile logicielle.

2.1.3. Langage Java



Java est un langage de programmation et une plate-forme informatique qui ont été créés par Sun Microsystems en 1998. Java est rapide, sécurisé et fiable. Java est présente sur plusieurs fronts : des ordinateurs portables aux centres de données, des consoles de jeux aux superordinateurs scientifiques, des téléphones portables à internet.

2.1.4. Méthodologie de développement : la démarche MVC

MVC : Model-View-Controller (Modèle-Vue-Contrôleur en français), C'est un design pattern (patron de conception), c'est-à-dire un concept d'architecture logicielle pour son application. Il permet d'avoir un code plus structuré, plus évolutif, plus maintenable, permettant de profiter de plusieurs mécanismes, d'avoir de la persistance de données, et bien d'autres choses encore.

Le Model est la représentation interne des données. Il permet comme son nom l'indique de modéliser les données que l'on va manipuler dans l'application. Représente les véritables données avec toutes les informations qu'elles véhiculent.

Le Vue quant à elle est la représentation visuelle de ces données à l'écran.

Le Contrôleur enfin, sert à faire l'interface entre le modèle et la vue. En effet, puisque le modèle et la vue sont censés être au maximum indépendants, le contrôleur sert à faire le lien pour faire communiquer l'un (Modèle) avec l'autre (Vue).

Ci-dessous, le mode de fonctionnement du MVC :

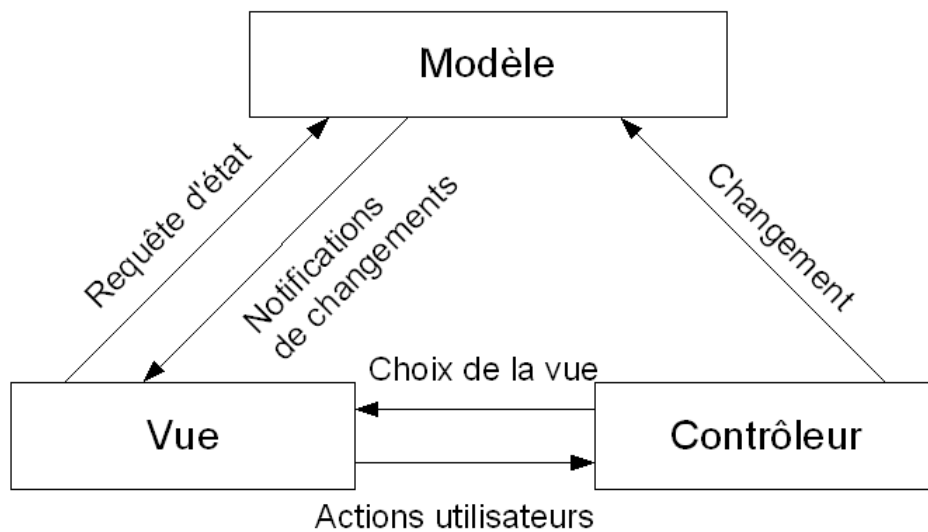


Figure 22: Le mode de fonctionnement du MVC

2.2. Développement web

2.2.1. Présentation du framework Angular



Angular est un framework JavaScript qui permet de créer des *single page applications*, ou SPA.

La toute première version d'Angular, qui s'appelait alors AngularJS, est sortie en 2010. Elle a été créée par Miško Hevery, qui a décidé de publier le code du framework en open source. Peu de temps après, Google a embauché Miško et investi de grosses ressources derrière le framework, dont la popularité a explosé à partir de 2012.

Les applications Angular sont en grande majorité codées en TypeScript, même si d'autres langages sont supportés, comme Dart. Ces langages sont convertis en JavaScript avant

l'exécution – le mot officiel est transpilé, pour permettre aux applications Angular de s'exécuter dans tous les navigateurs internet.

2.2.2. Architecture Angular

Voici ci-dessous l'architecture générale :

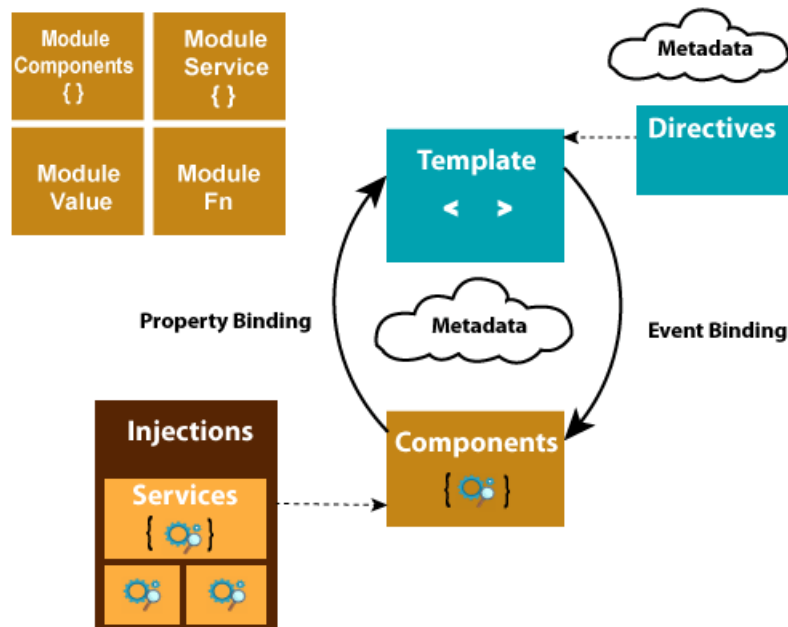


Figure 23: Architecture générale d'Angular

2.2.3. HTML5



HTML, l'acronyme de «HyperText Markup Language» (langage de balises pour l'hypertexte), est utilisé pour créer et représenter visuellement une page web. C'est ce qui détermine le contenu et la présentation de base d'une page web mais pas ses fonctionnalités.

2.2.4. SASS



SASS, l'acronyme de «Syntactically Awesome Style Sheets», est un préprocesseur pour le langage CSS, il a été créé il y a quelques années par Hampton Catlin et Nathan Weizenbaum. Il permet de générer dynamiquement du code CSS tout en offrant une syntaxe simple et un code facilement réutilisable et maintenable.

2.2.5. TYPESCRIPT



TypeScript est un langage de programmation développé par Microsoft en 2012. Son ambition principale est d'améliorer la productivité de développement d'applications complexes. C'est un langage open source, développé comme un sur-ensemble de JavaScript. Ce qu'il faut comprendre par-là, c'est que tout code valide en JavaScript l'est également en TypeScript.

2.2.6. Angular Material



Angular Material, est un module d'Angular d'intégration qui permet d'obtenir facilement une interface responsive harmonieuse et unie dans le style 'flat' de Google selon le Material Design.

3. Outils de développement

3.1. Outils de Google

– Firebase



- Authentication

Firebase Authentication, est un outil fournit des SDK faciles à exploiter, des services back-end ou encore des bibliothèques d'interface utilisateur. Ces bibliothèques permet d'authentifier les utilisateur

- Cloud Firestore

Cloud Firestore, est une base de données de documents NoSQL qui vous permet de facilement stocker, synchroniser et interroger des données pour vos applications mobiles et web – l'échelle mondiale

- Cloud Storage

Cloud Storage, permet de partager ou encore de stocker du contenu produit par les utilisateurs comme les images, les vidéos ou encore les fichiers audio. C'est une solution de stockage d'objets puissante qui se démarque par sa simplicité et son caractère économique.

– Google Maps Platform



- Google Maps API

Google Maps API, permet de créer des cartes personnalisées et évolutives, qui répondent concrètement aux besoins des utilisateurs avec des plans statiques et dynamiques, des images Street View et des vues à 360°.

3.2. Contrôle de version

– GitHub



GitHub, 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

– Git



Git, est un système de contrôle de version distribué pour suivre les changements de code source pendant le développement logiciel. Il conçu pour coordonner le travail des programmeurs, mais il peut être utilisé pour suivre les changements dans n'importe quel ensemble de fichiers. Ses objectifs incluent la vitesse, l'intégrité des données et la prise en charge des flux de travail distribués et non linéaires.

4. Présentation des interfaces de l'application

Cette section comporte des captures d'écran de quelques interfaces de l'application réalisé accompagnée par une brève description.

4.1. Les interfaces de l'application mobile

– Interface d'accueil

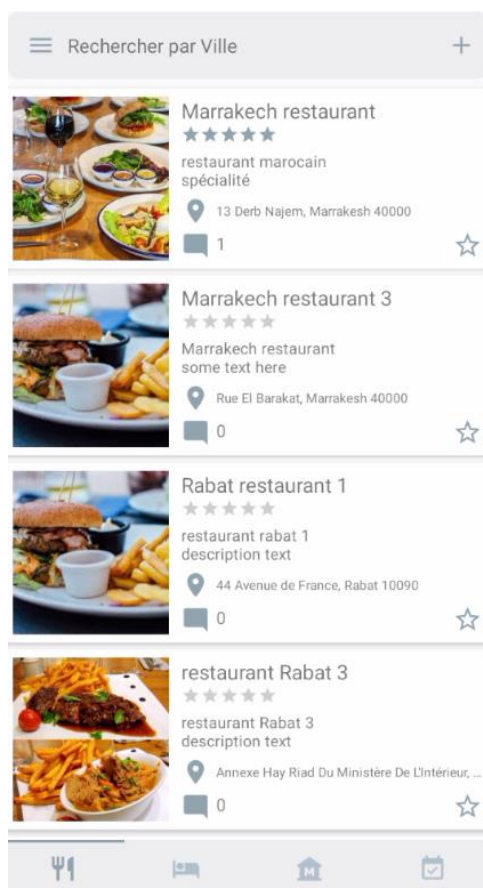


Figure 24: Interface d'accueil - mobile

L'interface affiche une liste des différentes activités touristique et évènements que l'utilisateur peut faire glisser entre et taper pour plus de détails.

– Interface détails



Figure 25: Interface détails - mobile

L'interface représente toute les informations nécessaire savoir l'endroit concerné, avec des photos, possibilité d'y ajouter un avis et aussi de signaler par exemple des erreurs. L'utilisateur peut aussi voir l'endroit sur la cartographie et/ou l'ajouter à ses favoris

4.2. Les interfaces de l'application web

Cette application permet à l'administrateur de gérer les endroits, les utilisateurs et les réclamations

– Interface d'authentification

Your email

Your password

Login

Sign-In with Google

Figure 26: Interface d'authentification -web

L'accès à l'application web est protégé par un système d'authentification. L'administrateur doit saisir son login et mot de passe à fin d'y accéder.

– Interface lieux

Dashboard

Réclamations

Utilisateurs

Lieux

Gérer réclamations

Restaurants à approuver

Filter

Nom	Added by user	Description	Adresse	Ville	TypeManage
Dar Zaki	Omar labiz	Que vous résidiez à Rabat ou que vous 2 Avenue Abdelkrim El Khattabi, soyez en v...	Témara, Morocco	Témara	Accepter Refuser Voir
Yamal Acham	Omar labiz	Réservation en ligne ou par téléphone de votre r...	51 Rue BAHRAIN, Témara, Morocco	Témara	Accepter Refuser Voir

Items per page: 5 1 - 2 of 2

Hôtels à approuver

Filter

Nom	Added by user	Description	Adresse	Ville	EtilesManage
Dina	Omar labiz		n1 Avenue Hassan II, Témara 12020, Morocco	Témara3	Accepter Refuser

Figure 27: Interface de gestion des lieux – web

L'interface représente la liste des endroits qui contient générale information sur les endroits soumis à l'ajout que l'administrateur peut accepter d'ajouter à l'application ou les refuser, ainsi il peut aussi voir plus d'informations.

Conclusion

Dans ce chapitre on a mis l'accent sur la description des caractéristiques de l'environnement du travail et décrit les plateformes matérielles et logiciel sur lesquelles nous avons construit notre application d'une part, d'autre part par la représentation des différents interfaces développées tout au long de l'application.

Conclusion générale

Au bout de notre cursus en licence professionnel en ingénierie des applications mobiles, nous avons été chargés de réaliser un projet de fin d'étude. Notre travail s'est basé sur le développement d'une application sur les technologies mobiles. Ceci nous a amené à enrichir nos compétences et découvrir la plateforme développement Android, ainsi que notre savoir.

Ce projet se dirige dans le cadre de notre licence professionnel en ingénierie des applications mobiles au sein d'IT-Learning de FST de Settat.

Au cours de la phase de réalisation de notre application, nous avons élaboré une étude préalable sur les smart phones et son importance sur le tourisme afin de préciser le but principal pour la future application. Cette phase a constitué le point de départ pour l'étape d'analyse et de spécification des besoins. Une fois nos objectifs sont fixés nous avons enchaîné avec la conception afin de mener à bien notre projet. Nous avons procédé à la phase de réalisation au cours de laquelle nous nous sommes familiarisés avec les langages et technologies utilisés.

Pour conclure, notre travail peut être sujet à des extensions. En effet, nous envisageons d'ajouter de plus de fonctionnalité, améliorer l'accessibilité et faire une version iOS.

Bibliographie

<https://www.uml.org/>

<https://mtataes.gov.ma/fr/tourisme/>

<https://firebase.google.com/>

<https://github.com/>

<https://cloud.google.com/maps-platform>