

MINISTERE DE L'ENSEIGNEMENT SUPERIEUR
ET DE LA RECHERCHE SCIENTIFIQUE
UNIVERSITE DE SOUSSE
المعهد العالي للإعلامية و تقنيات الاتصال بحمام سوسة



INSTITUT SUPERIEUR D'INFORMATIQUE
ET DES TECHNIQUES DE COMMUNICATION – HAMMAM SOUSSE

Département Télécommunications

MEMOIRE DE STAGE FIN D'ETUDES

Présenté en vue de l'obtention du diplôme de Licence Appliquée
Technologies des Communications

Développement d'une plateforme de Crawling

Réalisé par :

Mayssa LOUDHAIEF et Najib BELHADJ

Encadré(e)(s) par :

Mr. Karim SAHEB ETTABAA

Société d'accueil

IPRECISION

Année Universitaire 2019 – 2020

MINISTERE DE L'ENSEIGNEMENT SUPERIEUR
ET DE LA RECHERCHE SCIENTIFIQUE

UNIVERSITE DE SOUSSE

المعهد العالي للإعلامية و تقنيات الاتصال بحمام سوسة

INSTITUT SUPERIEUR D'INFORMATIQUE
ET DES TECHNIQUES DE COMMUNICATION – HAMMAM SOUSSE

Département Télécommunications

MEMOIRE DE STAGE DE FIN D'ETUDES

Présenté en vue de l'obtention du diplôme de Licence Appliquée en
Technologies des Communications

Réalisé par :

Mayssa LOUDHAIEF

Najib BELHADJ

Encadrant : Mr. Karim SAHEB ETTABAA Date :

Signature :.....

Superviseur : Mr. Karim SAHEB ETTABAA Date :

Signature :.....

Année Universitaire 2019 – 2020

Dédicaces

C'est avec grande émotion que je dédie ce travail :

A ma mère Naaima, en signe d'amour, et de gratitude pour tous les soutiens et les sacrifices dont elle a fait preuve à mon égard que Dieu l'accorde santé, longue vie et bonheur.

À mon frère « Mohamed » et sa femme « Fatma » qui m'ont beaucoup aidé tout au long de ce projet

A mon Neveu Rayssen mon petit homme que j'adore.

À mes chères sœurs « Chayma » et « Dorra », pour leurs conseils et encouragements à chaque moment de ma vie.

A tous mes amis qui m'ont beaucoup encouragé et m'ont chaleureusement supporté au long de mon parcours.

A mon cher binôme « Mayssa » pour tout ce qu'elle fait pour la réussite de ce projet.

 ... Najib

Dédicaces

*Je dédie ce modeste projet de fin d'étude en témoignage de mon
profond respect et mon grand amour à :*

A ma chère mère

*Pour ses sacrifices demeurés, son amour infini, ses conseils et sa
patience.*

A mon cher père

*Qui était présent à mes côtés dans les moments difficiles, pour son
encouragement, sa générosité sans limites et pour ses sacrifices
durant toute sa vie.*

J'espère pouvoir réaliser aujourd'hui l'un de leurs rêves et les honorer.

A mes frères Firas et Dhia, A mes sœurs Oumayma et Ghofrane

Qui ont été toujours prêts à me renforcer et me donner l'espoir.

*A mes chères amies Oumayma et Noura, je vous dédie ce travail en
témoignage de l'amitié sincère qui nous lie et des bons moments
passés ensemble, je vous souhaite un avenir radieux.*

 ... Mayssa

Remerciements

C'est avec grand plaisir que nous réservons cette page en signe de gratitude et de profonde reconnaissance envers les personnes qui ont contribué de près ou de loin à la réalisation de ce travail.

Monsieur **Karim SAHEB ETTABAA** notre encadrant de l'Institut Supérieur d'Informatique et des Techniques de Communication de Hammam Sousse et le directeur technique de la société IPRECISION pour la confiance qu'il nous a accordée, pour le temps consacré aux réunions qui ont rythmé les différentes étapes de notre projet et pour ses conseils en or tout au long de la rédaction de ce rapport.

Les membres du jury pour avoir accepté de nous accorder leur attention et pour l'évaluation de ce travail.

Notre Institut qui nous a donné l'occasion d'acquérir une formation professionnelle.

✍ ... Mayssa & Najib

Liste des figures

Figure 1 : Structure de la société	5
Figure 2:Architecture informatique	6
Figure 3:Processus Scrum	11
Figure 4:les types de vues UML 2.0.....	14
Figure 5: Diagramme de Gantt	14
Figure 6: diagramme de cas d'utilisation général	18
Figure 7 : Logo MYSQLWORKBENCH	21
Figure 8: Logo XAMPP	22
Figure 9:Logo Symfony	22
Figure 10:Logo Visual Studio Code.....	23
Figure 11: Logo Selenium	23
Figure 12: Logo Android Studio	23
Figure 13:logo Star UML	24
Figure 14 : Architecture trois tiers du projet	24
Figure 15: Le modèle MVC	26
Figure 16: diagramme des classes	27
Figure 17: diagramme de cas d'utilisation sprint1	31
Figure 18: diagramme de cas d'utilisation « gérer hôtel ».....	31
Figure 19 : diagramme de séquence « ajouter un hôtel »	34
Figure 20: diagramme de séquence « éditer un hôtel »	35
Figure 21:Interface d'inscription.....	36
Figure 22:Interface de connexion.....	36
Figure 23:interface de la liste des agences	37
Figure 24:Interface de la liste des hôtels	38
Figure 25:Interface ajouter un hôtel	38
Figure 26:Interface modifier un hôtel.....	39
Figure 27:diagramme de cas d'utilisation de sprint 2	40
Figure 28:diagramme de cas d'utilisation « récupérer les prix des hôtels »	41
Figure 29:diagramme de séquence « récupérer les prix des hôtels »	44
Figure 30:diagramme de séquence « mise à jour des prix existants ».....	45
Figure 31:Interface du crawling	46
Figure 32 : diagramme de cas d'utilisation sprint 3.....	47
Figure 33:diagramme de cas d'utilisation « lancer mapping »	48
Figure 34 : diagramme de cas d'utilisation « consulter statistiques »	49
Figure 35: diagramme de séquence « lancer mapping »	50
Figure 36 : diagramme de séquence « consulter statistiques »	51
Figure 37:Interface de mapping.....	52
Figure 38:Interface de statistiques 1	53
Figure 39:Interface de statistiques 2	53
Figure 40 : Interface Mobile de recherche	54
Figure 41: Interface Mobile de comparaison de prix d'un hôtel	54

Liste des tableaux

Tableau 1:Product Backlog	20
Tableau 2:Les caractéristiques des Pc utilisé	21
Tableau 3 :Backlog sprint1	30
Tableau 4: Description du cas d'utilisation « ajouter un hôtel »	32
Tableau 5: Description du cas d'utilisation « Editer un hôtel »	33
Tableau 6 :Test sprint1	39
Tableau 7 :Backlog sprint2	40
Tableau 8:Description textuelle du cas d'utilisation « récupérer les prix des hôtels »	42
Tableau 9:Description textuelle du cas d'utilisation « Mise à jour des prix existants »	43
Tableau 10 :Backlog sprint 3	47
Tableau 11:Description textuelle du cas d'utilisation «lancer mapping »	48
Tableau 12:Description du cas d'utilisation « consulter statistiques »	49
Tableau 13 :Test sprint 3	55

Liste des Acronymes

SCRUM : Systematic Customer Resolution Unraveling Meeting

MVC : Model View Controller / Modèle Vue Contrôleur.

UML : Unified Modeling Language / Langage de Modélisation Unifié.

PHP : Hypertext Preprocessor.

API : Application Programmable Interface / Interface Applicative de Programmation.

Présentation de la société d'accueil

Société : IPRECISION

Nom commercial : IPRECISION

Adresse : Imm Ezzitouna, App C-10, Monpalisir, Tunis 1002, Tunisie

Téléphone : 00 216 71 906 298

Fax : 00 216 71 906 297

Iprecision est une société privée Tunisienne, située à Tunis, d'ingénierie et de conseil en services informatiques, spécialisée dans le développement, la gestion de prestations Internet et l'intégration des solutions d'entreprise.

Table des matières

Liste des Acronymes	vii
Présentation de la société d'accueil	viii
Introduction générale	1
 Chapitre I : Contexte général du projet	 3
1 Introduction.....	4
2 Présentation de l'organisme d'accueil	4
2.1 Présentation de la société.....	4
2.2 Objectifs de travail.....	4
2.3 Nature du secteur d'activité.....	4
2.4 Nouvelle structure de la société « Iprecision »	5
3 Présentation du sujet	6
3.1 Problématique	6
3.2 Analyse de l'existant.....	6
3.2.1 Etude de l'existant	7
3.2.2 Critique de l'existant	7
3.3 Solution proposée	7
3.4 Travail demandé	7
4 Conclusion.....	8
 Chapitre II : Etude préalable	 9
1 Introduction.....	10
2 Définition d'un crawler	10
3 Fonctionnement d'un crawler	10
4 Choix de la méthodologie de gestion de projet	10
4.1 Le framework scrum.....	11
4.2 L'équipe de Scrum	11
4.3 Pilotage de notre projet avec Scrum	12
4.4 Les événements de Scrum.....	12
4.5 Les artefacts de Scrum.....	13
5 Méthodologie d'analyse et conception UML.....	13
6 Planification du projet	14
7 Conclusion.....	14
 Chapitre III : Analyse et spécification des besoins.....	 15
1 Introduction.....	16
2 Analyse de besoins	16
2.1 Identifications des acteurs.....	16
2.2 Les besoins fonctionnels	16

2.3	Les besoins non fonctionnels	17
2.4	Diagramme de cas d'utilisation générale.....	17
3	Product backlog.....	18
4	Technologies, environnements et outils de développement.....	21
4.1	Outils matériels	21
4.2.1	Outils de développement	21
4.2.2	Outils de modélisation.....	24
5	Architecture.....	24
5.1	Architecture du système trois tiers	24
5.2	Architecture de l'application MVC.....	25
6	Conception générale du projet	26
6.1	Diagramme des classes	26
7	Conclusion.....	27
Chapitre IV : Réalisation des sprints.....		28
1	Introduction.....	29
2	Sprint 1: Authentification , gestion agence, gestion hôtel	29
2.1	Sprint backlog	29
2.2	Diagramme de cas d'utilisation sprint1	31
2.2.1	Raffinement du cas d'utilisation « gérer hôtel »	31
2.2.2	Tableaux textuels.....	32
2.3	Diagramme de séquence	33
2.4	Réalisation	35
2.4.1	Les interfaces développées	35
2.4.2	Test et Validation	39
3	Sprint 2 : module de crawling	40
3.1	Sprint Backlog	40
3.2	Diagramme de cas d'utilisation du sprint2	40
3.2.1	Raffinement du cas d'utilisation « récupérer les prix des hôtels ».....	41
3.2.2	Tableaux textuels.....	41
3.3	Diagramme de séquence	43
3.4	Réalisation	45
3.4.1	Les interfaces développées	45
3.4.2	Test et Validation	46
4	Sprint 3: mapping et comparaison des prix	46
4.1	Sprint Backlog :	47
4.2	Diagramme de cas d'utilisation sprint3	47
4.2.1	Raffinement du cas d'utilisation « lancer mapping »	48
4.2.2	Tableau textuel	48
4.2.3	Raffinement du cas d'utilisation « consulter statistiques »	49
4.2.4	Tableau textuel	49

4.3	Diagramme de séquence	50
4.4	Réalisation	52
4.4.1	Les interfaces développées	52
4.4.2	Test et validation	55
5	Conclusion.....	55
Conclusion générale et perspectives.....		56
Bibliographie.....		57
Webographie.....		58
Résumé		59

Introduction générale

Depuis quelques années, l'industrie du voyage a connu un grand bouleversement sous l'impulsion du digital. L'internet est devenu la plus grande agence touristique, étant elle est accessible tous les jours et en toute heure. Autour d'elle, une multitude de start-ups ont vu le jour et se sont développées en proposant des alternatives aux offres classiques. Désormais, toutes les prestations touristiques sont à portée de clic.

La Tunisie dispose de plusieurs atouts dans le domaine touristique, ce qui lui permet d'attirer des touristes de tous les pays du monde avec ses sites pittoresques, ses monuments historiques et culturels. Au cours de la haute saison, des millions de touristes ont pu visiter notre pays et profiter des différents circuits touristiques aussi bien dans les anciennes médinas que dans le désert et les sites environnementaux. Avec cette concurrence de plus en plus forte et avec l'influence des prestataires de voyages sur internet, organiser un voyage ou un séjour sur mesure n'est pas plus simple.

En effet, dans une société moderne caractérisée par la rapidité et l'efficacité, les clients sont confrontés à des problèmes majeurs lors de la réservation de chambres d'hôtel, telles que la collecte d'informations et la comparaison des prix avec différentes agences de voyage, ce qui signifie perdre beaucoup de temps à utiliser plusieurs services.

C'est pourquoi, il est nécessaire de concevoir une solution satisfaisante pour de point de vue temps et séduction des clients pour le meilleur prix pour un hôtel sélectionné.

Dans ce cadre, la société « IPRECISION » nous a proposé de focaliser notre travail sur les technologies Web et mobile pour développer une solution qui permettrait à tout utilisateur de comparer les prix des hôtels de son choix en utilisant la méthode de crawling.

Ce rapport présente la démarche suivie pour développer la solution. Il comporte quatre chapitres organisés comme suit :

Le premier chapitre est essentiellement consacré à la présentation du cadre du travail. En effet, ce chapitre est divisé en trois parties : la première présente l'organisme d'accueil, la deuxième : aborde avec plus de détails la présentation de notre solution et la dernière explique la solution proposée et le travail demandé.

Le deuxième chapitre, quant à lui, est consacré tout d'abord à introduire le concept de base du crawling utilisé ensuite nous étudions la méthode de gestion de projet adopté et enfin nous exposons la planification de notre projet.

Dans le troisième chapitre nous présentons dans la première partie les différents besoins fonctionnels et non fonctionnels, les futurs acteurs de l'application, le backlog de notre

produit puis nous allons décrire l'architecture choisie pour réaliser notre solution et la conception générale du projet.

Le dernier chapitre est consacré à son tour, à la conception dynamique, l'implémentation et la réalisation des sprints tout en respectant les exigences de la méthodologie utilisée.

Finalement, nous clôturons notre travail par une conclusion générale tout en mentionnant les différents atouts de ce projet et les perspectives d'améliorations possibles.

Chapitre I : Contexte général du projet

1	Introduction	4
2	Présentation de l'organisme d'accueil	4
3	Présentation du sujet.....	6
4	Conclusion	8

1 Introduction

L'étude du cadre du projet constitue une étape primordiale à sa réalisation puisqu'elle permet de mettre le travail dans son contexte général. Ce chapitre est consacré tout d'abord à la présentation de l'organisme d'accueil qui nous a pris en charge durant toute la période de notre stage par la suite la problématique, l'étude de l'existant et nous finissons par proposer notre solution et exposer le travail demandé.

2 Présentation de l'organisme d'accueil

2.1 Présentation de la société

Iprecision est une société privée d'ingénierie et de conseil qui offre les services informatiques suivants :

Solutions Web :

- Création de Sites web statiques et/ou dynamiques
- Création de boutiques en ligne avec paiement en ligne
- Webdesign et conseil à l'Hébergement

Conseil, Développement et Assistance :

- Management Consulting, IT Stratégie
- Développement : Composite Application Framework : J2EE, .Net, Web services, Jboss, C#...
- Conseil et Assistance technique

2.2 Objectifs de travail

La société « Iprecision » aura comme principaux objectifs :
Support et la maintenance de l'application « Istep Société d'ingénierie et de conseil en services informatiques, spécialisée dans :

- Le domaine des TICs.
- Le développement et la gestion de prestations Internet, Intégration des Solutions d'entreprise (Portail, ERP, Applications métiers)
- Le développement et intégrations de solutions informatiques métiers.

2.3 Nature du secteur d'activité

Développement et maintenance de solutions informatiques.
Les prestations offertes sont des solutions entreprises :

- Création et intégration de portails d'entreprise
- Intégration du système d'archivage électronique
- Import des données issues des différents Tours Opérateurs ainsi que la maintenance des modules d'import déjà développés
- Développement de solutions d'intégration au profit du marché international
- Administration de la base de données

2.4 Nouvelle structure de la société « Iprecision »

Afin d'atteindre les objectifs cités ci-dessus, la société s'organisera en créant trois sections : (voir figure 1)

- Data Import : Cette section assurera le développement des modules d'import de « web services » issues de différents tours opérateurs, le traitement, le nettoyage et la diffusion des fichiers résultants d'un façon cyclique au sein de l'application Istep.
- Software Support : Cette section assurera la maintenance et le développement de nouveaux modules de l'application « Istep ».
- Analysais and Décision : Cette section assurera le développement de modules décisionnels au profit de la société « Istep ».



Figure 1 : Structure de la société

La nouvelle structure sera supportée par l'architecture informatique présentée dans la figure 2.

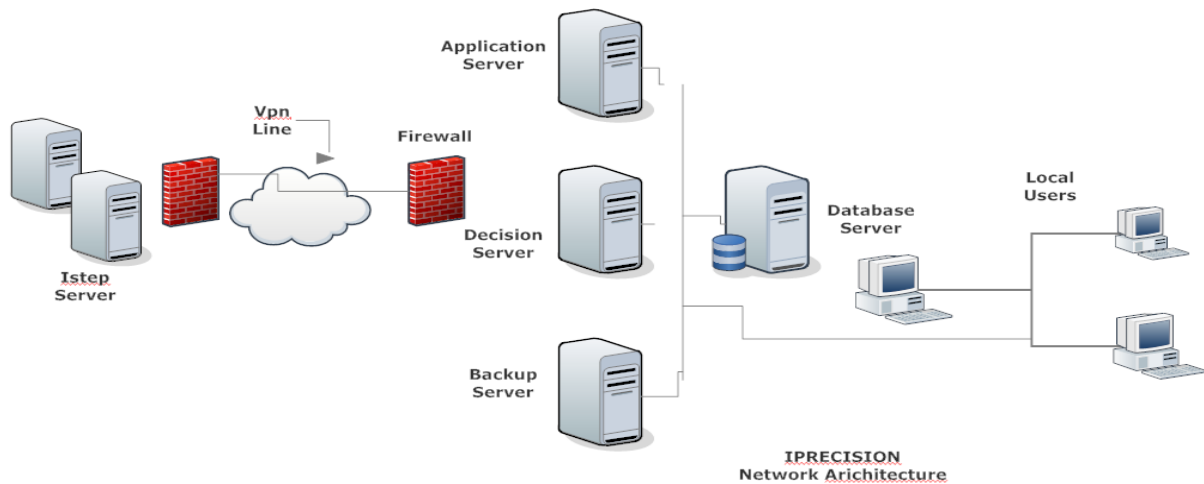


Figure 2:Architecture informatique

3 Présentation du sujet

Notre projet s'inscrit dans le cadre de l'obtention du diplôme de licence appliquée en Technologies des Communications.

3.1 Problématique

Plusieurs vacanciers s'informent et surfent sur Internet pour préparer leurs voyages ou leurs séjours mais l'opération de réservation d'une chambre d'hôtel est devenue très compliquée puisqu'on peut avoir plusieurs tarifs pour un même hôtel proposés par des agences différentes : la cause de cela se résume aux nombreux concurrents présents dans le secteur des agences de voyages .Le client perd donc beaucoup de temps à déterminer la meilleure agence garantissant un prix sur mesure pour un hôtel sélectionné .C'est dans ce cadre que nous sommes ramenés à développer une solution pour remédier à ces problèmes en utilisant une méthode de crawling.

3.2 Analyse de l'existant

L'analyse de l'existant est l'une des étapes les plus importantes pour réaliser tout type de projet, car elle vise à mettre en évidence les lacunes des solutions actuelles, ce qui nous permet de concevoir une solution plus fluide, plus développée en offrant des fonctionnalités plus efficaces et à utilisation simple.

Dans cette section on va présenter des exemples de sites de réservation par la suite nous passerons à une étude critique pour dégager les points faibles.

3.2.1 Etude de l'existant

Nombreux sont les sites de réservation en ligne en Tunisie on peut en citer :

- TahaVoyages
- TunisieBooking
- Traveltodo

Ces différents sites permettent au client de réserver une ou plusieurs chambres d'hôtel qui répond à ses exigences en quelques clic (après avoir rempli un formulaire et le choisir le mode de paiement) en proposant des tarifs différents et variés selon la date d'arrivée.

3.2.2 Critique de l'existant

La variété des sites de réservation avec des listes d'hôtel différentes impose des problèmes majeurs au client lors de la réservation tels que :

- Le prix du même hôtel sélectionné est différent (l'agence augmente ou diminue le prix par rapport aux autres concurrents).
- Perte de temps et choix difficiles (les contraintes budgétaires obligent les clients à rechercher des prix raisonnables).

3.3 Solution proposée

Notre projet provient d'un besoin réel dans le but de remédier les carences citées dans la critique de l'existant. La solution proposée consiste à développer une application web et mobile sous forme d'un comparateur de prix de réservation d'hôtels à partir de la création des scripts puissants d'analyse et de récupération automatique des données d'hôtels offerts par des agences sur le web. Cette solution a pour but de :

- Comparer les tarifs d'un hôtel choisi par rapport aux différentes agences du site donc augmenter la chance du visiteur à trouver une réservation à meilleure prix.
- Gain du temps et simplification du processus de recherche d'un prix raisonnable.

3.4 Travail demandé

Le travail demandé dans ce projet est l'analyse, la conception et la réalisation d'une plateforme web administrateur et une plateforme mobile dans le but de comparer les prix de réservation d'hôtels .Nous sommes menés à accomplir les tâches suivantes :

- Analyser et spécifier les besoins
- Développer des symboles de base pour la fondation de la base de données

- Développer une plateforme web administrateur
- Développer un module de crawling pour récupérer et mettre à jour les prix
- Développer une application mobile pour le visiteur

4 Conclusion

Au cours de ce chapitre introductif nous avons présenté l'environnement dans lequel s'inscrit notre projet de fin d'études. Nous avons commencé par la société d'accueil, par la suite nous avons présenté une brève description de notre projet commençant par la problématique, l'étude de l'existant pour arriver à la proposition de la solution et le travail demandé. Le chapitre suivant sera consacré à l'étude préalable.

Chapitre II : Etude préalable

<u>1</u>	<u>Introduction</u>	10
<u>2</u>	<u>Définition d'un crawler</u>	10
<u>3</u>	<u>Fonctionnement d'un crawler</u>	10
<u>4</u>	<u>Choix de la méthodologie de gestion de projet</u>	10
<u>5</u>	<u>Méthodologie d'analyse et conception UML</u>	13
<u>6</u>	<u>Planification du projet</u>	14
<u>7</u>	<u>Conclusion</u>	14

1 Introduction

L'objectif final de ce projet est de développer un comparateur de prix de réservation d'hôtels en se basant sur un module de crawling des sites web des agences de réservation, pour ce faire il faut tout d'abord définir le concept du crawling, par la suite fixer d'avance les technologies et les méthodologies à utiliser. Dans ce chapitre nous donnons une définition du crawler et son fonctionnement nous présentons après la méthodologie de gestion de projet adoptée et nous clôturerons ce chapitre par la planification du projet.

2 Définition d'un crawler

Crawler ou robot de crawl désignent dans le monde de l'informatique un robot d'indexation. Concrètement, il s'agit d'un logiciel qui a pour principale mission d'explorer le Web afin d'analyser le contenu des sites visités et les stocker de manière organisée dans un index.[W1]

3 Fonctionnement d'un crawler

Les crawlers peuvent être programmés pour parcourir le Web avec des objectifs déterminés. Ils sont actifs en permanence et visitent les pages selon les instructions qui leur sont données. Le crawler parcourt donc, en permanence, de façon autonome et automatique, les différents sites et pages Internet à la recherche de contenus nouveaux ou d'éventuelles mises à jour de contenus déjà explorés par le passé.[W1]

4 Choix de la méthodologie de gestion de projet

Lorsque nous démarrons un projet, il est nécessaire de s'interroger sur l'organisation et les méthodes que nous utiliserons pour la mise en œuvre et la réalisation afin d'organiser correctement notre travail. Dans le domaine de la gestion de projet, il existe deux méthodes principales, la méthode classique et la méthode agile.[W2]

Pour la méthode classique: elle se caractérise par la collecte des exigences, la définition des produits, le développement et le test des produits avant de livrer le produit, et plus le projet progresse, plus l'impact du risque est important, le retour au produit sera toujours plus difficile et coûteux. Retourne lorsqu'une exception est trouvée.

Contrairement à l'approche classique, la gestion de projet agile fournit un développement itératif, qui divise le projet en plusieurs étapes. Chaque étape doit conduire à

la création d'un élément fonctionnel global qui correspond aux objectifs du projet. Après une étude comparative des méthodes agiles les plus connues, nous avons choisi la méthode "Scrum" pour gérer le processus de développement

4.1 Le framework scrum

Scrum est un cadre de travail (framework) au sein duquel les acteurs peuvent aborder des problèmes complexes et adaptatifs, en livrant de manière efficace et créative des produits de la plus grande valeur possible. Il met en évidence l'efficacité relative à la gestion de produit et aux techniques de travail afin que vous puissiez continuellement améliorer le produit, l'équipe et l'environnement de travail.

Le cadre Scrum est constitué d'équipes Scrum et leurs rôles, événements, artefacts et règles associés. Chaque composante de ce cadre a un but précis et essentiel au succès et à l'utilisation de Scrum.[B1] (voir figure 3)

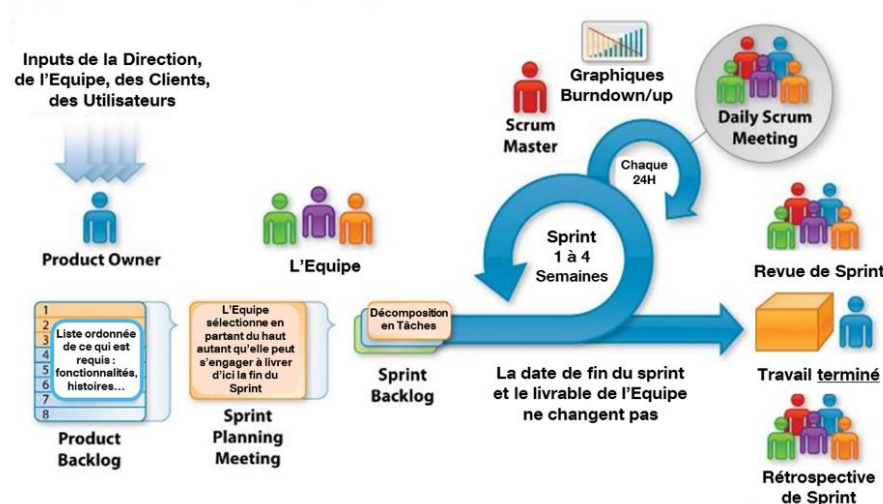


Figure 3:Processus Scrum

4.2 L'équipe de Scrum

Scrum définit un modèle d'équipe optimisant la flexibilité, la créativité et la productivité.

L'équipe Scrum renferme 3 rôles :

- **Product Owner** : Le Product Owner est responsable de maximiser la valeur du produit généré par le travail de l'équipe de développement. Il est seul responsable de la gestion du backlog produit.

- L'équipe de Développement : qui est chargée de transformer les besoins exprimés par le Product Owner en fonctionnalités utilisables. Les membres de cette équipe n'ont pas de rôle bien déterminé : développeur, architecte logiciel, testeur...
- L'équipe peut être de taille de 3 à 9 personnes.
- Scrum Master : qui doit maîtriser Scrum et s'assurer que ce dernier est correctement appliqué. Il a donc un rôle de coach à la fois auprès du Product Owner et auprès de l'équipe de développement. Il facilite la communication au sein de l'équipe et cherche à améliorer la productivité et le savoir-faire de son équipe.

4.3 Pilotage de notre projet avec Scrum

Dans notre projet, les rôles sont repartis comme suivant :

Product owner : Mr Karim SAHEB ETTABAA

Scrum Master : Mr Karim SAHEB ETTABAA

Equipe de développement : Najib BELHADJ et Mayssa LOUDHAIEF

4.4 Les événements de Scrum

Scrum utilise des événements prescrits pour créer de la cohérence et minimiser le besoin des réunions non définies. Tous les événements sont limités par le temps, donc chaque événement a une durée maximale.

- **Sprint** : La durée de Sprint est d'un mois ou moins et pendant cette période, la version "finale", utilisable et éventuellement livrable du logiciel sera créée. Chaque sprint a un objectif, on lui associe une liste d'éléments du carnet du produit (fonctionnalités) à réaliser.
- **Daily Scrum** : le daily scrum est une réunion de planification de 15 minutes pour l'équipe de développement. Chaque membre discute tour à tour trois sujets : ce qu'il a fait la veille, il prévoit de réaliser l'objectif de sprint le même jour, et les obstacles qui empêchent l'équipe d'atteindre le but du sprint.
- **Sprint planning Meeting** : Il s'agit d'une réunion qui dure plus de 4 heures. Avant chaque Sprint, toute l'équipe Scrum participe à cette réunion pour planifier les user stories du backlog produit qu'elle a décidé de traiter lors de la prochaine itération.
- **Sprint review** : À la fin du sprint, l'équipe Scrum organise une réunion pour effectuer une revue de sprint, dont le but est d'échanger le travail effectué pendant le sprint, de vérifier les incréments obtenus et d'ajuster la liste des tâches du produit si nécessaire. Cette réunion est limitée à un délai de quatre heures pour un sprint d'un

mois. Pour les Sprints plus courts, la durée de Sprint review est généralement plus courte.

- **Sprint Retrospective** : La réunion rétrospective Sprint est l'occasion pour l'équipe Scrum de s'auto-examiner et d'élaborer un plan d'amélioration, qui sera mis en œuvre lors du prochain Sprint. La rétrospective Sprint aura lieu après la revue Sprint et avant la prochaine réunion de planification Sprint. Ceci est une réunion, limitée à trois heures par mois.

4.5 Les artefacts de Scrum

Le processus Scrum définit les artefacts suivants :

- **Le Product Backlog** : une liste ordonnée de tous les éléments jugés nécessaires pour le produit. Il possède toutes les fonctions, fonctionnalités, exigences, améliorations et tous correctifs qui constituent des modifications des futures versions du produit. Les éléments du backlog produit se composent d'une description, d'un ordre, d'une estimation et d'une valeur.
- **Le Sprint Backlog** : Le Backlog Sprint est l'ensemble des éléments sélectionnés pour le Sprint plus un plan pour livrer l'incrément du produit et réaliser l'objectif du Sprint.

5 Méthodologie d'analyse et conception UML

Pour simplifier nos tâches et modéliser nos idées, nous adoptons le langage de modélisation unifié UML .

UML est un langage de modélisation graphique normalisé par l'OMG (Object Management Group). Il s'agit d'une méthode de visualisation standard pour spécifier, concevoir et enregistrer des applications orientées objet. Il existe deux types de vues :vue statique et vue dynamique .[B2]

L'ensemble des treize types de diagrammes UML peut ainsi être résumé sur la figure 4 :

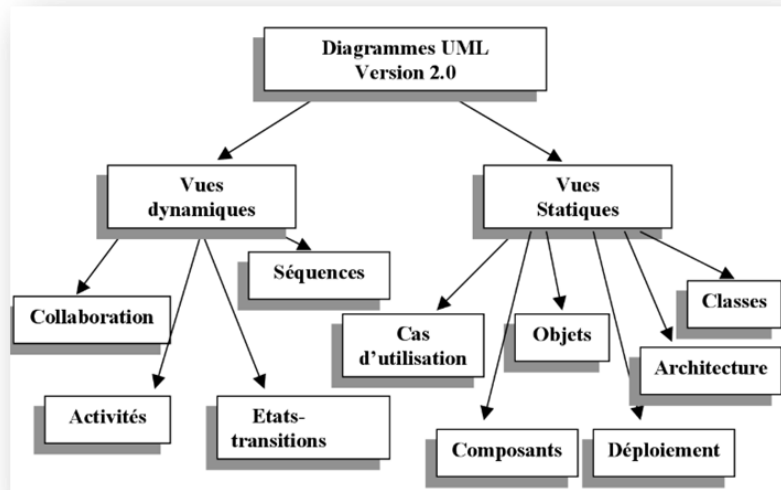


Figure 4:les types de vues UML 2.0

6 Planification du projet

Avant de commencer chaque projet, il est indispensable de faire une planification des tâches afin de bien organiser le travail, faciliter sa réalisation et garantir son accomplissement dans les meilleurs délais fixés d'avance.

Dans notre cas, la planification des tâches de notre projet est présentée sous forme d'un diagramme de Gantt dans la figure 5.



Figure 5: Diagramme de Gantt

7 Conclusion

Après avoir défini la méthodologie Scrum et énuméré les outils techniques essentiels pour accomplir le travail demandé, le chapitre suivant est consacré au traitement de l'analyse et à la spécification des besoins dans le chapitre suivant pour avoir une vision profonde du projet.

Chapitre III : Analyse et spécification des besoins

<u>1</u>	<u>Introduction</u>	16
<u>2</u>	<u>Analyse de besoins</u>	16
<u>3</u>	<u>Product backlog</u>	18
<u>4</u>	<u>Technologies, environnements et outils de développement</u>	20
<u>5</u>	<u>Architecture</u>	24
<u>6</u>	<u>Conception générale du projet</u>	25
<u>7</u>	<u>Conclusion</u>	27

1 Introduction

Dans ce chapitre nous entamons la phase préparatrice du projet. Nous commençons par présenter l'analyse de besoins par la suite, nous détaillons le backlog produit et le diagramme de cas d'utilisation globale. Enfin nous allons énumérer les outils nécessaires à la réalisation et nous exposons l'architecture proposée de notre application ainsi que la conception générale.

2 Analyse de besoins

L'analyse de besoins est une démarche qui consiste à déterminer les besoins fonctionnels qui seront exécutés par le système et les besoins non fonctionnels qui amélioreront la qualité logicielle du système.

2.1 *Identifications des acteurs*

Un acteur est une entité dont le rôle est d'interagir avec notre système. Elle peut être soit une machine, un autre système ou un utilisateur. Cette section a pour but de présenter les acteurs ainsi que leurs fonctionnalités auxquelles doit répondre notre application. Cette dernière fait intervenir à savoir :

- **Le crawler** : c'est le système responsable de l'extraction automatique des prix des hôtels , la mise à jour et l'enregistrement des prix récupérées dans notre base de données.
- **L'administrateur** : Il peut consulter et gérer toutes les interfaces de l'application.
- **Le Visiteur** : Il peut consulter les statistiques des prix de réservation d'hôtels y compris : comparaison des prix, moyenne des prix, minimum des prix, maximum des prix.

2.2 *Les besoins fonctionnels*

Les besoins fonctionnels sont les fonctionnalités que doit contenir notre module. Pour assurer que notre système soit opérationnel il doit satisfaire les besoins suivants :

- Gérer les agences
- Consulter la liste des agences
- Gérer les hôtels
- Consulter la liste des hôtels
- Lancer automatiquement l'algorithme approprié de chaque agence pour récupérer les prix des hôtels désirés.

- Lancer mapping
- Consulter statistiques

2.3 Les besoins non fonctionnels

Les besoins non fonctionnels expriment l'ensemble des contraintes qu'on doit respecter lors de la réalisation pour assurer le bon fonctionnement du système. Notre futur système doit satisfaire les exigences suivantes :

- L'Ergonomie et Convivialité : Développer des interfaces conviviales, simples et faciles à manipuler par les utilisateurs
- La Sécurité et la Confidentialité : La sécurité est assurée par un module d'authentification qui gère l'accès entre l'administrateur et l'application
- L'évolutivité : La possibilité d'ajouter d'autres fonctionnalités et des nouvelles options dans l'avenir
- L'efficacité : L'application doit fournir des résultats performants qui répondent aux besoins de l'utilisateur

2.4 Diagramme de cas d'utilisation générale

Le diagramme de cas d'utilisation générale permet d'identifier les possibilités d'interaction entre le système et les acteurs (intervenants extérieurs au système), c'est-à-dire toutes les fonctionnalités que doit fournir le système [W2].

Le diagramme de cas d'utilisation globale de notre application est représenté par la figure ci-dessous.

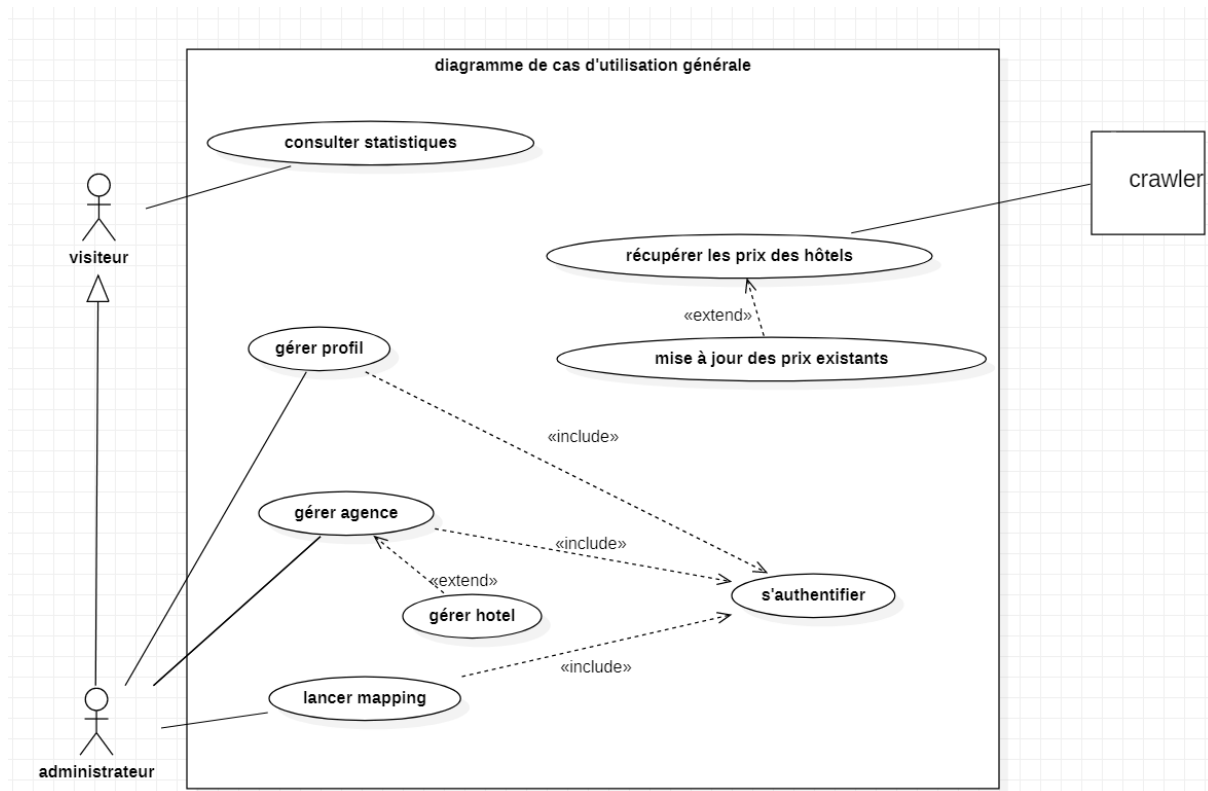


Figure 6: diagramme de cas d'utilisation général

Dans le chapitre suivant nous allons détailler les cas d'utilisation réalisés.

3 Product backlog

Le backlog produit renferme toutes les fonctionnalités qui composent le futur produit. Les caractéristiques fonctionnelles sont appelées histoires d'utilisateurs (user stories).

Les user stories sont caractérisés par :

- Identifiant : il détermine l'identifiant unique de l'histoire en question.
- Description : elle décrit le besoin d'un acteur.
- Critères d'acceptation : à chaque user story sont associés des critères d'acceptation qui permettent au client tester l'histoire. Ces critères d'acceptation fournissent une grande aide à l'équipe.
- Estimation : est une estimation, c'est une valeur entière appartenant à la suite de Fibonacci.
- Priorité : les priorités sont utilisées pour définir l'ordre de mise en œuvre, elles permettent de créer un flux d'histoires qui va aider l'équipe.

Le tableau 2 présente le backlog de notre solution.

ID	User story	Critères d'acceptation	Priorité	Estimation
U1	En tant qu'administrateur je veux créer un compte en entrant mon nom, mon prénom mon adresse e-mail, un mot de passe et une confirmation de mot de passe pour protéger mon système.	Compte crée suite à la saisie d'un nom, d'un prénom, d'une adresse email et un mot de passe et une confirmation de mot de passe.	1	3
U2	En tant qu'administrateur dans la plateforme, je veux connecter en utilisant mon nom de compte et mon mot de passe afin de pouvoir commencer à gérer l'application.			
U3	En tant qu'administrateur, je veux ajouter une agence.	Le formulaire d'ajout doit contenir : -nom de l'agence -description -lien -numéro de téléphone -email -agence ref	2	2
U4	En tant qu'administrateur, je veux modifier une agence.	Le formulaire doit avoir les mêmes caractéristiques du formulaire de l'ajout	3	2
U5	En tant qu'administrateur, je veux supprimer une agence	Chaque agence doit avoir un bouton supprimer	4	2
U6	En tant qu'administrateur je veux consulter les hôtels d'une agence	Chaque agence doit avoir un bouton Hôtels	5	2
U7	En tant qu'administrateur, je veux ajouter un hôtel.	Le formulaire d'ajout doit contenir : -nom de l'hôtel -description -lien -catégorie	6	2

U8	En tant qu'administrateur, je veux éditer un hôtel	Le formulaire doit avoir les mêmes caractéristiques du formulaire de l'ajout	7	2
U9	En tant qu'administrateur, je veux supprimer un hôtel	Chaque hôtel doit avoir un bouton supprimer.	8	2
U10	En tant qu'administrateur, je veux consulter un hôtel	Chaque hôtel doit avoir un bouton show.	9	2
U01	En tant crawler, je peux aspirer les prix des hôtels des différents agences	Extraction des prix dans un intervalle bien déterminée	10	8
U02	En tant crawler, je peux mettre à jour les prix existants par les données récupérées.	Mise à jour de la base de données	11	8
U12	En tant qu'administrateur, je veux lancer le mapping	L'accès doit être à partir de la barre de menu (mapping). Une fois une nouvelle agence est ajoutée elle doit être présente dans la liste de mapping (interface dynamique)	12	5
U13	En tant que visiteur, je veux choisir un hôtel ou tous les hôtels d'une ville	Une liste déroulante qui contient les noms des hôtels disponibles .	13	3
U14	En tant que visiteur, je veux choisir la date pour consulter les statistiques	Deux champs de dates (date d'arrivée et date de départ) et un bouton pour la confirmer et passer à la comparaison des prix	14	5
U15	En tant que visiteur, je veux consulter les statistiques	La comparaison des prix sera affichée sous forme des courbes (prix en fonction de la date choisit) avec un affichage des statistiques suivantes pour chaque hôtel ou pour : moyenne des prix , maximum des prix et minimum des prix .	15	8

Tableau 1: Product Backlog

4 Technologies, environnements et outils de développement

Dans cette partie nous allons énumérer les principaux outils pour le développement et la conception :

4.1 Outils matériels

Nous avons utilisé deux PC qui permettent d'assurer aux exigences à accomplir.

Le tableau ci-dessous montre les détails des caractéristiques de ces deux PC :

Machine	Machine de développement 1	Machine de développement 2
RAM	8 Go	16Go
Processeur	Intel®Core™i7-8550U CPU@2.2Ghz	Intel®Core™i7-7500U CPU@2.7Ghz
Disque dur	1 To	1 To
Système d'exploitation	Windows 10 64bits	Windows 10 64bits

Tableau 2: les caractéristiques des Pc utilisé

4.2 Outils logiciels

Dans cette partie nous allons énumérer les principaux outils pour le développement et la conception :

4.2.1 Outils de développement

- **MySQL Workbench : outil de modélisation de base de données**

MySQL Workbench est un logiciel de gestion et d'administration de bases de données MySQL. Il permet de créer, modifier ou supprimer des tables, des comptes utilisateurs, et d'effectuer toutes les opérations inhérentes à la gestion d'une base de données via une interface graphique intuitive. Pour ce faire, il doit être connecté à un serveur MySQL. [W4]



Figure 7 : Logo MYSQLWORKBENCH

- **XAMPP : Serveur de gestion des bases des données**

XAMPP est un ensemble de logiciels qui permet de configurer un serveur Web local. Il s'agit d'une distribution de logiciel libre (X (cross) Apache MariaDB Perl PHP), qui offre une bonne flexibilité d'utilisation et est connue pour sa facilité d'installation et sa rapidité. Par conséquent, comme il ne nécessite aucune connaissance spécifique et peut fonctionner sur le système d'exploitation le plus large, il est à la portée de nombreuses personnes. [W5]



Figure 8: Logo XAMPP

- **Symfony 4.4 : framework PHP**

Symfony est un ensemble de composants PHP et un framework MVC libre écrit en PHP. Il fournit des fonctionnalités modulaires et adaptables, facilitant et accélérant le développement de sites Web. Symfony 4.4 est l'une des dernières versions du framework de développement Web Symfony [W6].



Figure 9: Logo Symfony

- **Visual Studio Code : éditeur de code open-source**

Visual Studio Code est un éditeur de code extensible développé par Microsoft pour Windows, Linux et MacOS. Cet outil destiné aux développeurs supporte plusieurs dizaines de langages de programmation comme le HTML, C++, PHP, Javascript, CSS, etc... [W7]



Figure 10: Logo Visual Studio Code

- **Selenium : outil d'automatisation de navigateur web**

Selenium est un framework portable pour tester les applications web. Selenium fournit un outil de lecture pour créer des tests fonctionnels sans avoir besoin d'apprendre un langage de script de test (Selenium IDE). Il fournit également un langage de domaine de test (sélénien) pour écrire des tests dans un certain nombre de langages de programmation populaires, notamment C #, Java, Perl, PHP, Python, Ruby et Scala. Les tests peuvent ensuite s'exécuter sur la plupart des navigateurs Web modernes [W8].



Figure 11: Logo Selenium

Nous avons utilisé sélénium pour faire des tests sur les sites des agences et extraire les prix .

- **Android Studio : Environnement de développement Mobile**

Android Studio est un environnement de développement intégré, couramment abrégé en IDE, qui permet de développer et tester des applications, des outils et services dédiés au système d'exploitation Android. Principalement destiné aux développeurs, cette plateforme intègre une pléthore d'outils et fonctionnalités pour concevoir des applications de qualité [W9]



Figure 12: Logo Android Studio

4.2.2 Outils de modélisation

Nous avons utilisé StarUML pour faire une conception détaillée de notre projet

- **StarUML**

« StarUML » est un logiciel de modélisation UML sophistiqué pour la modélisation agile et concise. Il est ciblé aux équipes de développement, aux professionnelles et aux instituts d'enseignement. [W10]



Figure 13: logo Star UML

5 Architecture

5.1 Architecture du système trois tiers

Cette section est réservée à introduire l'architecture globale du projet qui est l'architecture trois tiers. L'architecture 3-tiers est composée de trois couches :

- Couche présentation (interface homme-machine)
- Couche métier (règles de gestion & logique applicative)
- Couche données (accès aux données)

La figure 14 illustre l'architecture trois tiers de notre projet :

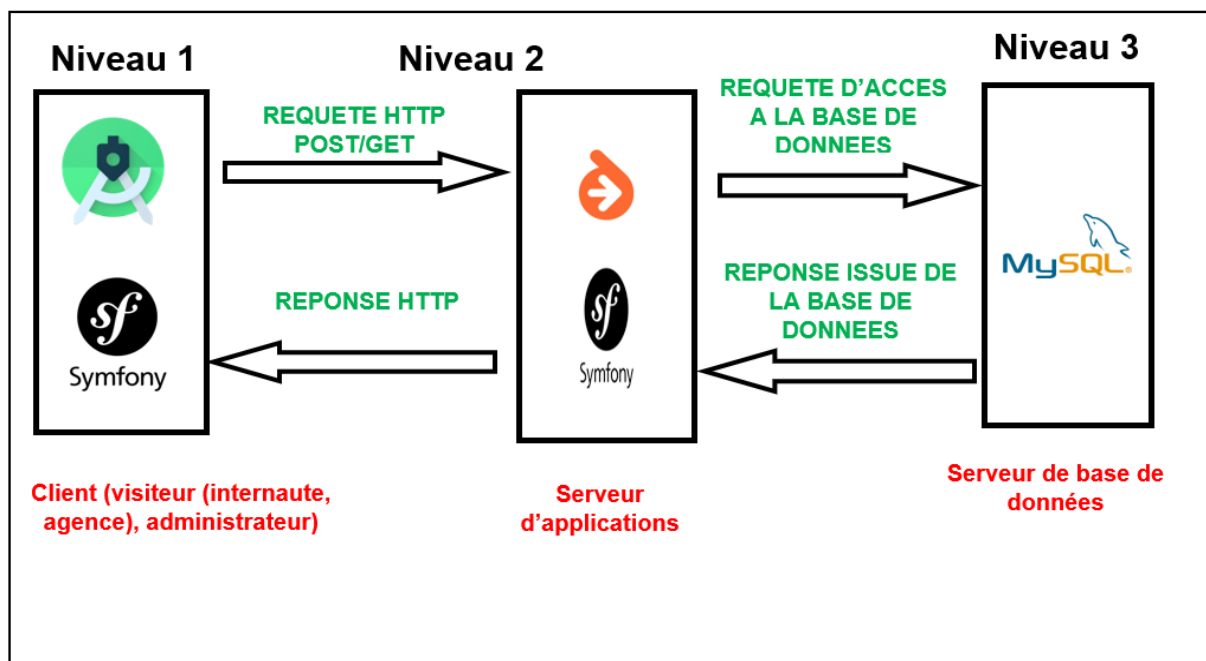


Figure 14 : Architecture trois tiers du projet

Ce type d'architecture est conçu pour séparer très clairement les trois couches logicielles de la même application ou le même système pour modéliser et présenter l'application comme un empilement à trois niveaux, étapes, ou couches chaque une avec un rôle clairement défini :

Couche de présentation : c'est la partie visible et interactive de l'application sur une interface graphique ou textuelle qui peut être représentée en plusieurs langages afin d'être utilisée par un téléphone portable ou exploité par un navigateur web.

Couche de traitement : c'est la partie fonctionnelle de l'application qui présente les opérations et les traitements qu'elle opère en fonction des requêtes des utilisateurs.

Couche d'accès aux données : la couche d'accès et au stockage des données qui assure la persistance d'une quantité importante de données de façon structurée. Ces données peuvent être propres à une à l'application, ou gérées par une autre application [W11].

5.2 Architecture de l'application MVC

Notre application est conforme au modèle MVC (Modèle/View/Contrôleur), c'est une architecture qui consiste à séparer les données (modèle), l'affichage (vue) et les actions (contrôleur) c'est une façon d'organiser une interface graphique d'un programme.

Ce modèle est composé de trois types d'entités ayant chacun un rôle précis dans l'interface. Ces rôles sont définis comme suivant :

▪ **Model (Modèle) :** La couche Modèle contenant le traitement logique des données (les accès à la base de données se trouvent dans cette couche) : validation, lecture et enregistrement.

▪ **View (vue) :** Il s'agit de la partie visible de l'interface graphique (graphiques, formulaires, boutons, etc.) qui contient des éléments visuels et la logique nécessaire pour afficher les données du modèle, la vue capture donc les événements effectués par l'utilisateur et les envoie au contrôleur.

▪ **Controller (contrôleur) :** Le contrôleur est chargé d'assurer l'échange entre le modèle et la vue. Il reçoit les événements des utilisateurs, les analyse, puis déclenche les actions à effectuer et modifie les données du modèle et de la vue.[W12].

La figure 15 montre le modèle MVC

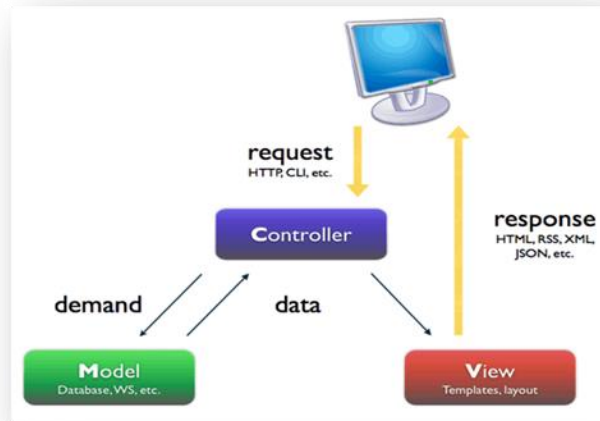


Figure 15: Le modèle MVC

6 Conception générale du projet

Cette partie sera dédiée à exposer la conception générale du projet.

6.1 Diagramme des classes

Le diagramme de classes est considéré comme le plus important de la modélisation orientée objet. Il permet de fournir une représentation abstraite des objets du système qui vont interagir pour réaliser les cas d'utilisation. Il s'agit d'une vue statique, car on ne tient pas compte du facteur temporel dans le comportement du système.

Le diagramme de classe de notre application comporte 4 classes et une classe d'association :

- Classe administrateur : elle définit celui qui gère l'application.
- Classe provider : elle renferme les données d'une agence appartenant à notre application et qui est manipulée par l'administrateur.
- Classe providerHotel : elle contient la liste d'hôtels convenant à une agence. Ainsi que leurs informations.
- Classe providerHotelImg : elle possède les images de chaque hôtel.
- Class-association providerHotelPrix : elle contient les prix de chaque hôtel correspondants à des dates bien déterminés.

La figure 16 montre le diagramme des classes de notre projet.

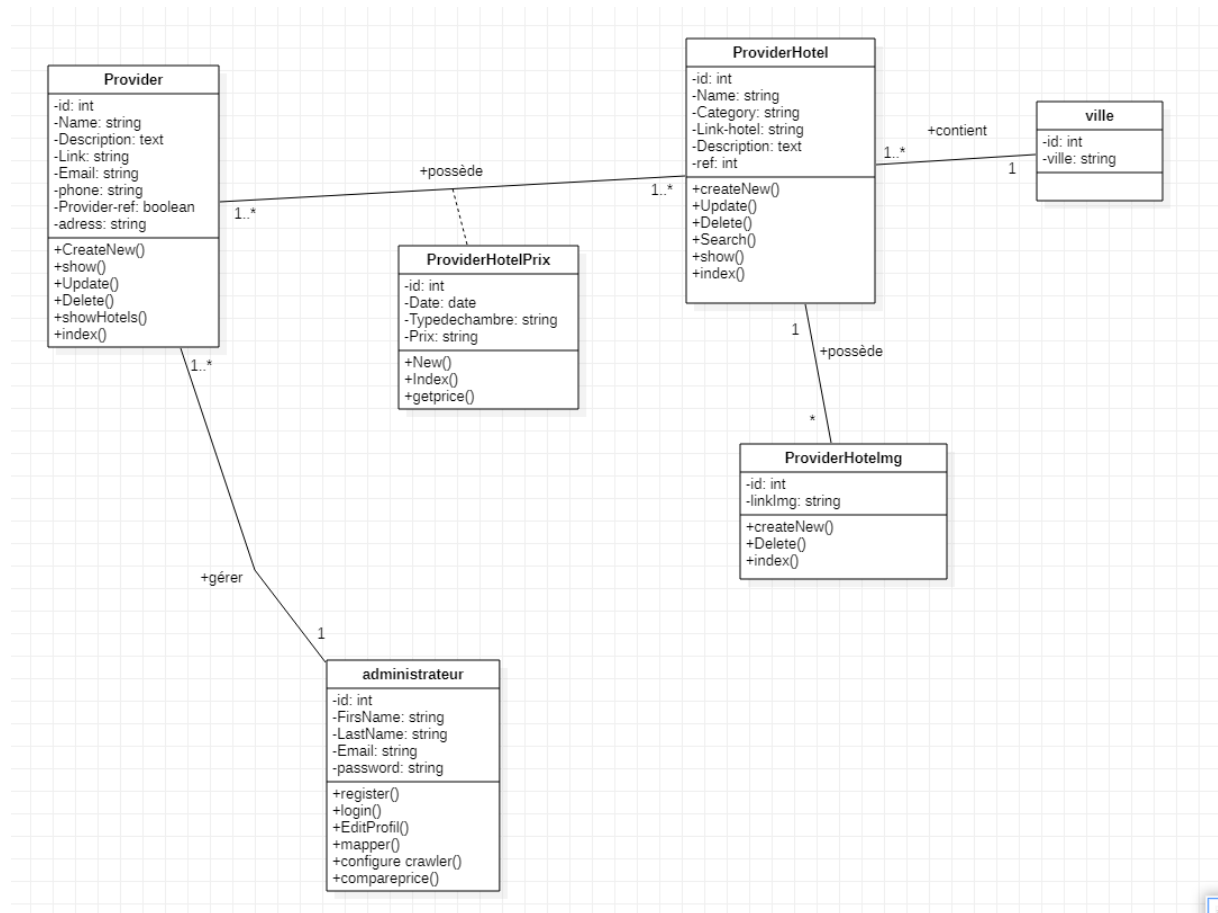


Figure 16: diagramme des classes

7 Conclusion

À partir de ce chapitre, nous avons une vision plus approfondie de ce sujet ceci nous permettre de comprendre le travail à faire. Le chapitre suivant est le chapitre le plus important du rapport, dans lequel on va entamer, en détails, nos sprints en se basant sur la méthode Scrum.

Chapitre IV : Réalisation des Sprints

<u>1</u>	<u>Introduction</u>	29
<u>2</u>	<u>Sprint 1: Authentification , gestion agence, gestion hôtel</u>	29
<u>3</u>	<u>Sprint 2 : module de crawling</u>	39
<u>4</u>	<u>Sprint 3: mapping et comparaison des prix</u>	45
<u>5</u>	<u>Conclusion</u>	54

1 Introduction

Comme il est mentionné précédemment nous avons choisi Scrum comme méthodologie de gestion de projet, ce chapitre va être repartit selon les exigences de Scrum, en effet notre travail est divisé en Sprints, pour chacun d'eux nous présentons en premier lieu le but et le Sprint Backlog, ensuite nous exposons la conception. Enfin nous clôturons chaque sprint par sa réalisation et sa validation.

2 Sprint 1: Authentification , gestion agence, gestion hôtel

L'objectif du Sprint (Sprint Goal) est un but fixé pour le Sprint et peut être réalisé par l'implémentation d'une partie du Backlog Produit. Il est défini lors de la réunion de planification du Sprint (Sprint Planning). Suite à une étude sur la division du projet basée sur ce que nous avons présenté dans le Backlog du produit, nous avons choisi le but suivant : développer le module d'authentification, de gestion d'agences et de gestion d'hôtels.

2.1 *Sprint backlog*

Après avoir défini l'objectif, il est temps de spécifier les user stories à inclure dans ce Sprint pour atteindre cet objectif. Le tableau 3 présente le Sprint Backlog contenant les items choisis à partir du Product Backlog ainsi que les tâches à faire pour les accomplir :

ID	User story	Taches
U1	En tant qu'administrateur, je veux créer un compte en entrant mon nom, mon prénom mon adresse e-mail et un mot de passe pour protéger mon système	-Création du formulaire d'inscription
U2	En tant qu'administrateur dans la plateforme je veux me connecter en utilisant mon nom de compte et mon mot de passe afin de pouvoir commencer à gérer l'application.	-Création du formulaire de connexion -Ajout de l'option « remember me »
U3	En tant qu'administrateur, je veux ajouter une agence.	-Création CRUD agence -Ajout du lien showhôtels dans la page gestion agence qui redirige l'administrateur vers la page gestion hôtel de l'agence choisi. -Configuration de la page gestion agence
U4	En tant qu'administrateur, je veux modifier une agence.	
U5	En tant qu'administrateur, je veux supprimer une agence	
U6	En tant qu'administrateur, je veux lister les hôtels d'une agence	
U7	En tant qu'administrateur, je veux ajouter un hôtel.	-Création CRUD hôtel -Ajout des liens show dans la page gestion hôtel pour chaque hôtel pour consulter les images, les prix et la description. -Configuration de la page gestion hôtel.
U8	En tant qu'administrateur, je veux éditer un hôtel	
U9	En tant qu'administrateur, je veux supprimer un hôtel	
U10	En tant qu'administrateur, je veux consulter un hôtel (description , prix images).	

Tableau 3 : backlog sprint1

Afin d'avoir une compréhension globale du comportement fonctionnel des user stories dans ce sprint, nous avons proposé de présenter le diagramme de cas d'utilisation.

2.2 Diagramme de cas d'utilisation sprint1

La figure 17 présente le diagramme de cas d'utilisation de notre sprint courant

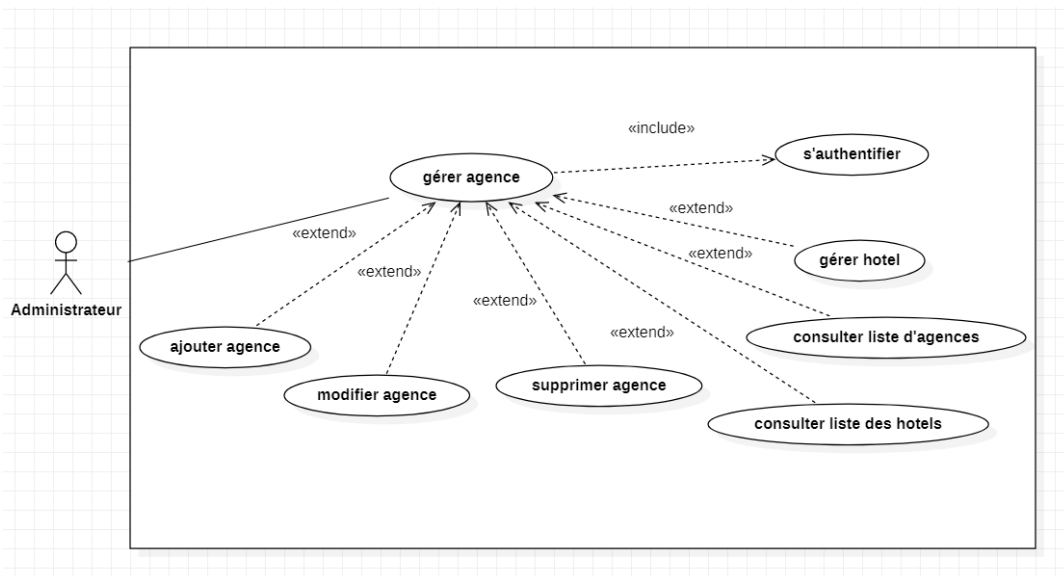


Figure 17: diagramme de cas d'utilisation sprint1

2.2.1 Raffinement du cas d'utilisation « gérer hôtel »

La figure 18 illustre le raffinement de l'item « gérer hôtel » (user story U7, U8, U9, U10)

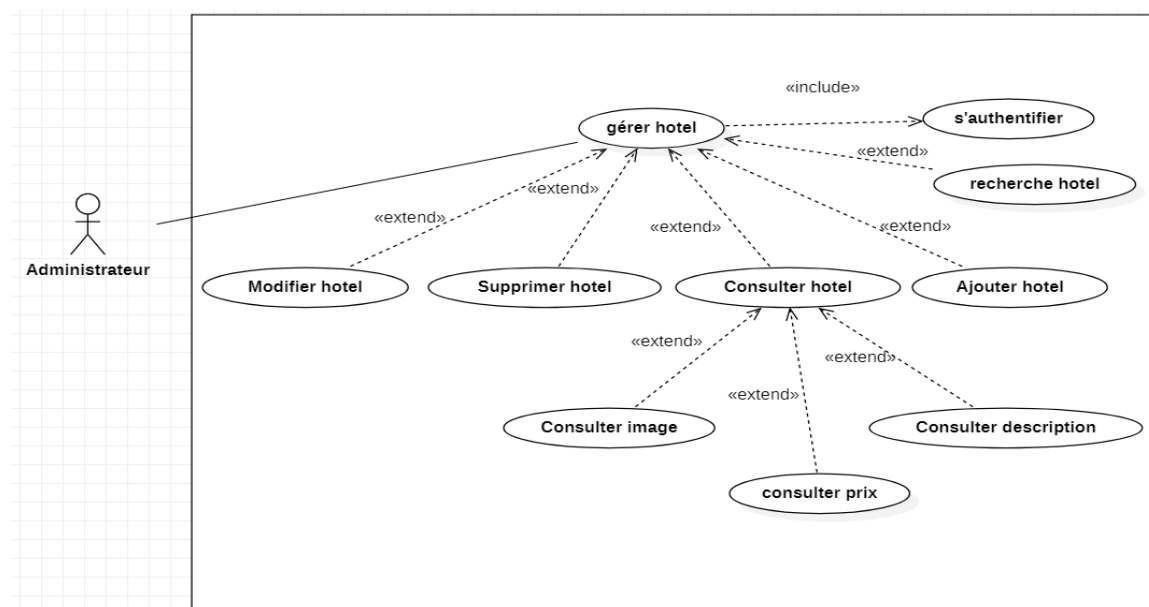


Figure 18: diagramme de cas d'utilisation « gérer hôtel »

2.2.2 Tableaux textuels

Les tableaux 5 et 6 exposent les tableaux textuels des cas d'utilisation les plus essentiels dans le sprint 1

- **Description textuelle du cas d'utilisation « ajouter un hôtel » (user story U7) :**

Titre	Ajouter hôtel
Résumé	L'administrateur ajoute un nouvel hôtel
Acteur	Administrateur
Précondition	Administrateur déjà authentifié et consulte la liste des hôtels
Scénario principal	1- L'administrateur clique sur « create new ». 2- Le système affiche le formulaire d'ajout 3- L'administrateur remplit les champs du formulaire puis il valide les données entrées 4- Le système vérifie ces données puis ajoute le nouvel hôtel à notre base et affiche la nouvelle liste d'hôtels
Scénario alternatif	Mal saisie des champs du formulaire Le formulaire d'ajout est affiché de nouveau
Postcondition	Hôtel ajouté

Tableau 4: Description du cas d'utilisation « ajouter un hôtel »

- **Description du cas d'utilisation « éditer un hôtel» (user story U8) :**

Titre	Editer un hôtel
Résumé	L'administrateur modifie un hôtel choisi
Acteur	Administrateur
Précondition	Administrateur déjà authentifié et consulte la liste des hôtels
Scénario principal	1-L'administrateur clique sur « Edit » situé sur chaque ligne d'hôtel 2-L'administrateur modifie les champs du formulaire 3- L'administrateur valide les modifications en cliquant sur update 4-Le système vérifie les données entrées 5-Le système sauvegarde les modifications dans la base et affiche la liste d'hôtels
Scénario alternatif	Mal saisie des champs du formulaire Le formulaire est affiché de nouveau
Postcondition	Hôtel modifié

Tableau 5:Description du cas d'utilisation « Editer un hôtel »

Afin d'avoir une vision plus claire et plus dynamique des cas d'utilisation cités précédemment, nous utiliserons le diagramme de séquence

2.3 Diagramme de séquence

Un diagramme de séquence est un type de diagramme d'interaction, car il décrit comment et dans quel ordre plusieurs objets fonctionnent ensemble. Les diagrammes de séquences sont une représentation en détails d'un cas d'utilisation UML.[w13]

Nous avons proposé de présenter dans cette section les diagrammes de séquence du système de classification les plus importants dans ce premier sprint tel que l'ajout d'un hôtel et la modification d'un hôtel.

On suppose que l'administrateur est déjà authentifié

Pour ajouter un nouvel hôtel l'administrateur clique sur « create new » via l'interface de gestion d'hôtel puis il remplit les champs du formulaire et enfin il valide les données qu'il a saisies en cliquant sur le bouton « save ». Le système vérifie tout d'abord si l'administrateur a bien saisi les champs du formulaire. Cela est modélisé par une étiquette alt. Si les données sont valides le contrôleur envoie une requête insert afin d'ajouter le nouvel hôtel dans la Table ProviderHotel. L'hôtel est par la suite ajouté et le système affiche la nouvelle liste d'hôtels. Si non, un message d'erreur est renvoyé et le formulaire d'ajout est affiché de nouveau.

La figure 19 présente le diagramme de séquence d'ajout d'un hôtel (user story 7) :

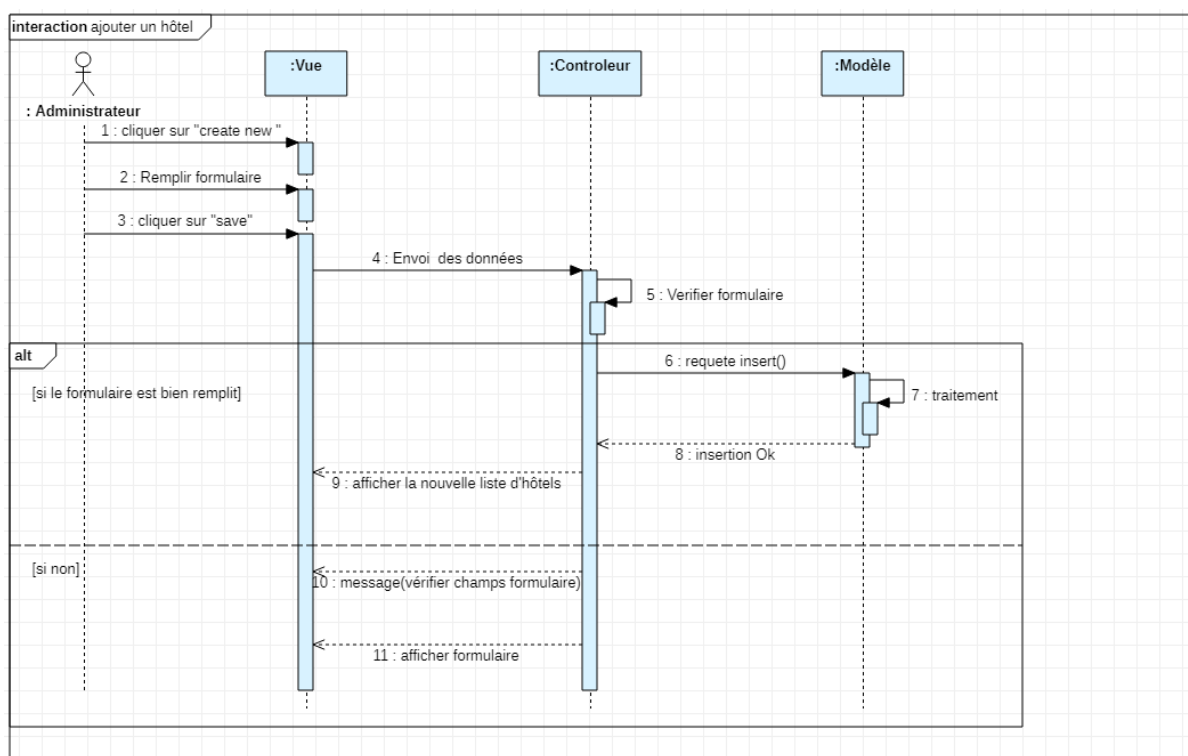


Figure 19 : diagramme de séquence « ajouter un hôtel »

Par la suite pour que l'administrateur puisse éditer un hôtel, il clique sur « Edit » qui se trouve sur chaque ligne d'hôtel. Le système génère le formulaire associé à l'hôtel pour que l'administrateur saisi les nouvelles informations. L'administrateur finit cette tâche par enregistrer en cliquant sur « update ». Le système vérifie tout d'abord si l'administrateur a bien saisi les champs du formulaire. Cela est modélisé par une étiquette alt. Si les données sont valides le contrôleur envoie une requête update afin d'enregistrer les modifications dans la table ProviderHotel. L'hôtel est par la suite modifié et le système affiche la nouvelle liste d'hôtels. Si non un message d'erreur est renvoyé et le formulaire est affiché de nouveau.

La figure 20 présente le diagramme de séquence d'édition d'un hôtel qui s'appuie sur le même principe d'ajout d'un hotel (user story 8) :

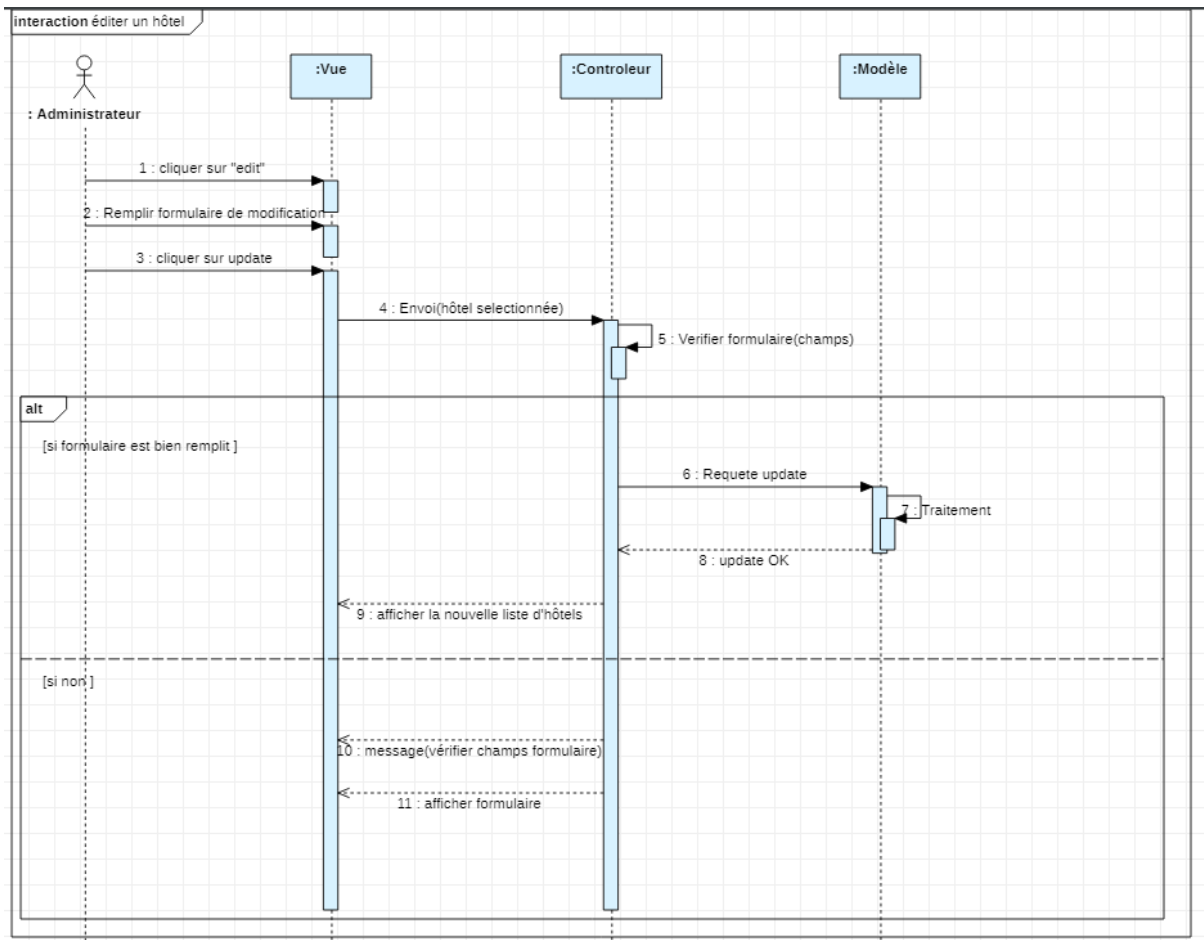


Figure 20: diagramme de séquence « éditer un hôtel »

2.4 Réalisation

Dans cette section nous présentons les interfaces et les fonctionnalités fournies pour l'administrateur. Nous avons commencé par développer les interfaces d'authentification (inscription , connexion) par la suite nous avons développé les interfaces liés à la gestion d'agence tel que consulter la liste d'agences, ajout-agence ,éditer-agence , consulter la liste contenant tous les hôtels d'une agence .

2.4.1 Les interfaces développées

Avant de procéder à la page de gestion d'agence, l'administrateur doit s'authentifier. Pour ce faire, il commence par la création d'un compte. La figure 21 présente l'interface d'inscription :

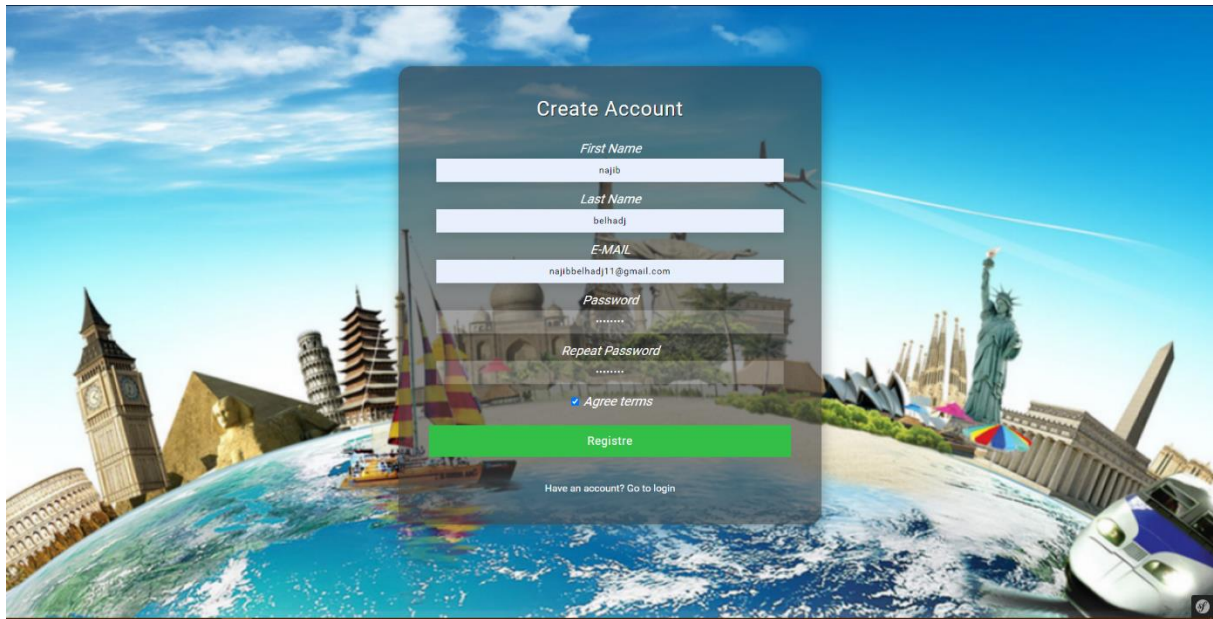


Figure 21:Interface d'inscription

Après la validation de l'inscription l'administrateur est redirigé vers la page de connexion pour s'authentifier et commencer à utiliser l'application. La figure 22 présente l'interface de connexion :

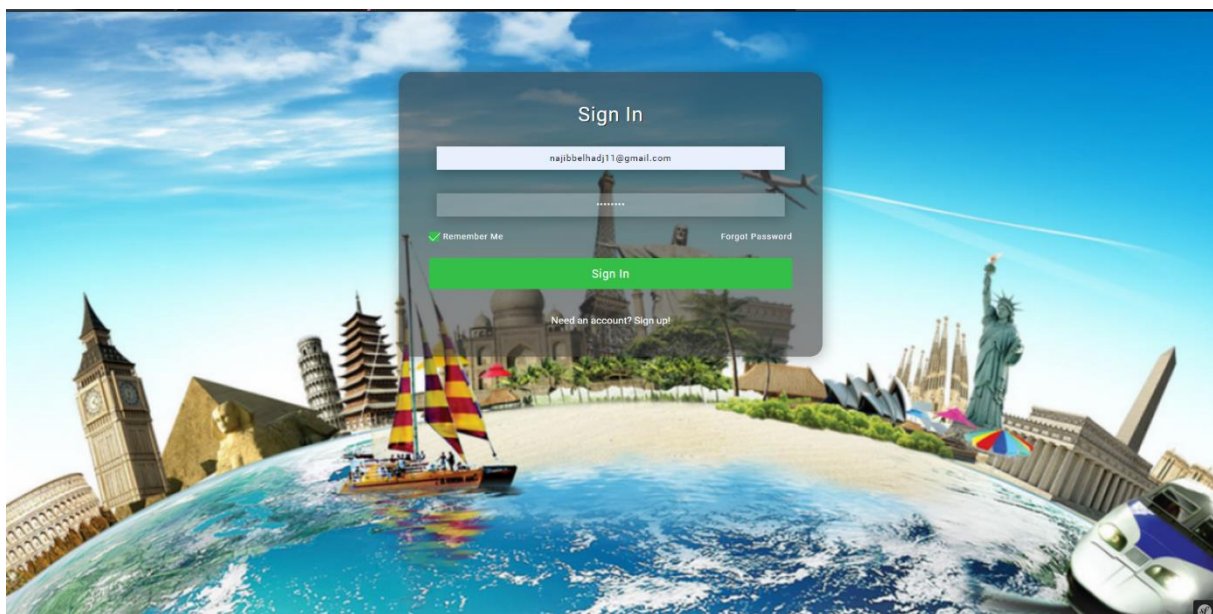


Figure 22:Interface de connexion

L'administrateur consulte l'interface de liste des agences à partir de laquelle il peut commencer à gérer les agences. Cette liste comporte sur chaque ligne 4 boutons :un bouton pour visualiser les détails d'une agence , un bouton pour modifier ses données , un bouton pour la suppression d'une agence et l'autre pour afficher les hôtels d'une agence . S'ajoute à

cela un autre bouton au-dessus de la liste d'agences qui permet d'ajouter une nouvelle agence. La figure 23 montre cette interface :

Admin

Providers Mapping Crawling Statistics

Provider index

Create new

Search for...

Show 10 entries

Id	Name	phone	Email	Link	ProviderRef	Hotels	actions
1	Main Provider				Yes	show	show edit Delete
2	Tahavoyage	+216 71 900 900	info@tahavoyage.com	https://www.tahavoyages.com/	No	show	show edit Delete
3	Tunisiebooking	+216 71 124 124	contact@tunisiebooking.com	https://tn.tunisiebooking.com/	No	show	show edit Delete
14	Traveltodo	+(216) 70 103 103	client@traveltodo.com	https://www.traveltodo.com/	No	show	show edit Delete

Showing 1 to 4 of 4 entries

Previous 1 Next

Logged in as: najib bulhadj

Copyright © 2020

Privacy Policy - Terms & Conditions

Figure 23:interface de la liste des agences

Dès que l'administrateur consulte l'interface de gestion des hôtels d'une agence, il peut commencer à gérer ces hôtels. Dans cette interface on a 5 boutons sur chaque ligne d'hôtel:

un bouton pour consulter ses données, un bouton pour modifier l'hôtel, un bouton pour supprimer l'hôtel, un bouton pour visualiser ses images et un autre bouton pour voir ses prix. Au-dessus de cette liste il existe un bouton « create New » pour ajouter un nouvel hôtel.

Il existe également un champ de saisie qui permet de chercher un hôtel à partir des attributs affichés dans l'interface et un select box pour choisir le nombre souhaité des hôtels affichés (varie entre 10 et 100).La figure 24 présente interface :

Admin

Providers Mapping Crawling Statistics

Provider Hotel index

Create new

Search for...

Show 10 entries

Id	Name	Ville	Category	Link	image	prix	actions
961	paradise Djerba	DJERBA	4	https://tn.tunisiebooking.com/detail_hotel_6/	show	show	show edit Delete
962	Miramar Petit Palais Djerba	DJERBA	4	https://tn.tunisiebooking.com/detail_hotel_7/	show	show	show edit Delete
963	Club Marmara Djerba Mare Djerba	DJERBA	4	https://tn.tunisiebooking.com/detail_hotel_8/	show	show	show edit Delete
964	Vincci Helios Beach Djerba	DJERBA	4	https://tn.tunisiebooking.com/detail_hotel_9/	show	show	show edit Delete
965	Djerba Resort Djerba	DJERBA	4	https://tn.tunisiebooking.com/detail_hotel_10/	show	show	show edit Delete
966	Venice Beach Djerba	DJERBA	3	https://tn.tunisiebooking.com/detail_hotel_11/	show	show	show edit Delete
967	Haroun Djerba Djerba	DJERBA	3	https://tn.tunisiebooking.com/detail_hotel_12/	show	show	show edit Delete

Logged in as: najib belhadj

Figure 24:Interface de la liste des hôtels

La figure 25 montre l'interface d'ajout d'un hôtel :

Admin

Providers Mapping Crawling Statistics

Create new Hotel

Name

Ville

Category

Link

Description

Provide

Save

back to list

Logged in as: najib belhadj

Figure 25:Interface ajouter un hôtel

La figure 26 montre l'interface d'édition d'un hôtel :

Figure 26:Interface modifier un hôtel

2.4.2 Test et Validation

Dans cette section nous présentons dans le tableau 6 les tests de validation de notre premier sprint

Fonctionnalités	Etat	Validation
Module d'authentification	Fait	Satisfait
Ajouter agence	Fait	Satisfait
Modifier agence	Fait	Satisfait
Supprimer agence	Fait	Satisfait
Ajouter hôtel	Fait	Satisfait
Consulter hôtel	Fait	Satisfait
Modifier hôtel	Fait	Satisfait
Supprimer hôtel	Fait	Satisfait
Recherche hôtel	Fait	Satisfait

Tableau 6 : Test sprint1

Dans l'ensemble, tout fonctionne bien dans le premier Sprint, et nous avons terminé le travail dans les délais.

3 Sprint 2 : module de crawling

Partant du même principe que le sprint précédent, nous définissons d'abord l'objectif notre deuxième sprint. En se basant sur les priorités des user stories restantes, nous avons décidé de travailler sur le module de crawling. Ce sprint a comme but le développement du module du crawling.

3.1 Sprint Backlog

Le tableau 7 illustre le sprint backlog du sprint courant

ID	User Story	Tâches
U01	En tant crawler, je peux aspirer les prix des hôtels des différents agences	-Configuration du crawler : La date et l'heure de déclenchement La date de début et de fin de l'extraction Développement des algorithmes d'extraction des prix des hôtels approprié pour chaque agence.
U02	En tant crawler, je peux mettre à jour les prix existants	-Parsing des données récupérées et les insérer dans la base de données

Tableau 7 : backlog sprint2

3.2 Diagramme de cas d'utilisation du sprint2

La figure 27 présente le diagramme de cas d'utilisation de sprint courant :

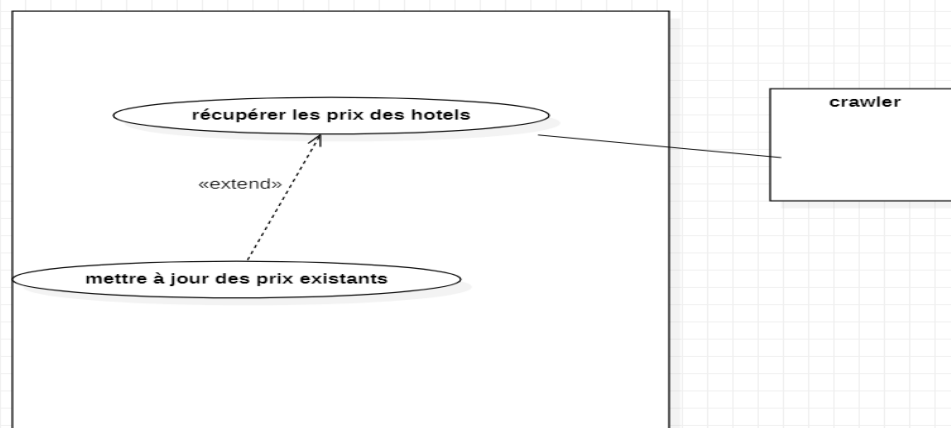


Figure 27:diagramme de cas d'utilisation de sprint 2

3.2.1 Raffinement du cas d'utilisation « récupérer les prix des hôtels »

La figure 28 illustre le raffinement de l'item « récupérer les prix des hôtels »

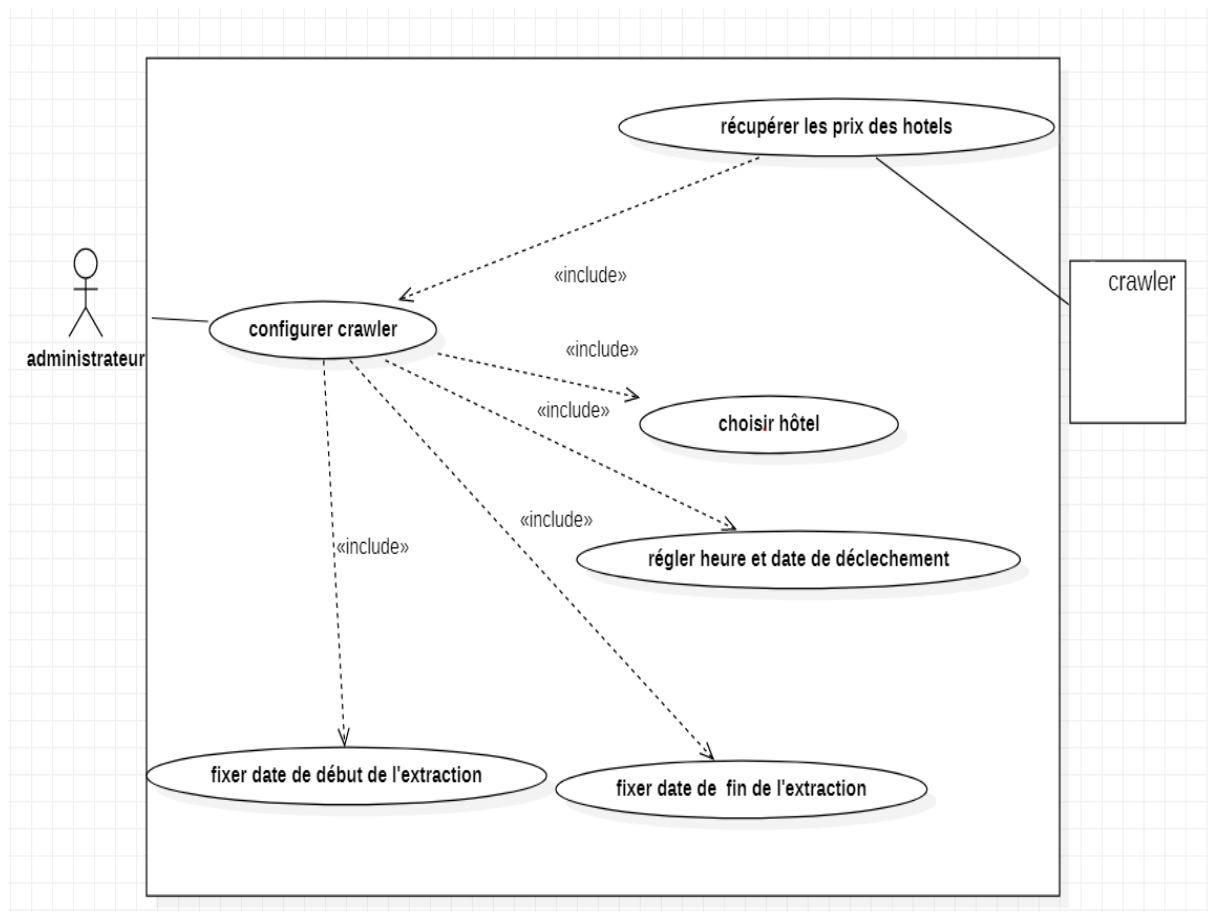


Figure 28:diagramme de cas d'utilisation « récupérer les prix des hôtels »

3.2.2 Tableaux textuels

Les tableaux 8 et 9 exposent les tableaux textuels des cas d'utilisation les plus essentiels dans le sprint courant.

• **Description textuelle du cas d'utilisation « récupérer les prix des hôtels »**

Titre	Récupérer les prix des hôtels
Résumé	Aspirer les prix des hôtels sélectionnés dans un intervalle bien déterminé (date de début et de fin de l'extraction)
Acteur	Le crawler
Précondition	Administrateur authentifié, consulte l'interface de crawling
Scénario principal	<ol style="list-style-type: none"> 1- L'administrateur configure le crawler : choisit l'hôtel, fixe les dates de l'extraction et règle l'heure et la date de déclenchement du crawler. 2- L'administrateur clique sur le bouton « crawling » pour valider ses choix 3- Le système cherche le crawler correspondant (cherche l'agence de l'hôtel choisi pour identifier le crawler correspondant) 4- Le système exécute le crawler concernée et lui envoie les données de l'hôtel choisit (id, lien et l'intervalle de l'extraction) 5- Le crawler est lancé et ouvre le lien de l'hôtel 4- Le crawler effectue des tests automatiques sur le lien de l'hôtel et aspire les prix dès la date de début de l'extraction jusqu'à la date de fin .
Scénario alternatif	<p>L'heure de déclenchement n'est pas arrivée</p> <p>Problème de connexion Internet</p>
Postcondition	Les Prix sont récupérés

Tableau 8: Description textuelle du cas d'utilisation « récupérer les prix des hôtels »

- **Description textuelle du cas d'utilisation « Mise à jour des prix existants »**

Titre	Mise à jour des prix existants
Résumé	Mise à jour des prix existants dans la base de données et insertion des nouveaux prix
Acteur	Le crawler
Précondition	Récupération des prix
Scénario principal	1- Le crawler envoie chaque date vers la base de données 2- Le crawler envoie le prix correspondant à la date envoyée. 3- Sauvegarde des prix selon date dans la base de données
Scénario alternatif	La date envoyée existe dans la base de données et possède un prix Connexion avec la base de données échouée
Postcondition	Mise à jour de la table des prix

Tableau 9: Description textuelle du cas d'utilisation « Mise à jour des prix existants »

3.3 Digramme de séquence

Dans cette section, nous présentons les diagrammes de séquence des cas d'utilisation classés les plus importants dans ce sprint tel que « récupérer les prix des hôtels » et « mise à jour des prix existants ».

Pour récupérer les prix des hôtels l'administrateur doit commencer à choisir la liste des hôtels dont il veut récupérer les prix.

On suppose que l'administrateur a choisi un hôtel

Par la suite, il choisit l'intervalle de l'extraction (date de début et de fin de l'extraction) et enfin il fixe l'heure et la date de déclenchement du crawler et valide des choix en cliquant sur le bouton crawling. Si ce n'est pas l'heure du crawling, rien ne se passe. Si non le contrôleur lance la recherche de l'agence de l'hôtel choisi pour trouver le crawler correspondant. Dès que ce dernier est trouvé, il est automatiquement lancé avec les paramètres suivants (id de l'hôtel, l'intervalle de l'extraction et lien de l'hôtel). Cela est

modélisé par une étiquette alt. Si la connexion Internet a échoué, il y'a échec de l'extraction des prix. Si non le crawler effectue des tests sur le lien de l'hôtel et aspire pour chaque date de l'intervalle de l'extraction le prix correspondant. Cela est modélisé par une deuxième étiquette alt.

La figure 29 montre le scénario de récupération des prix des hôtels :

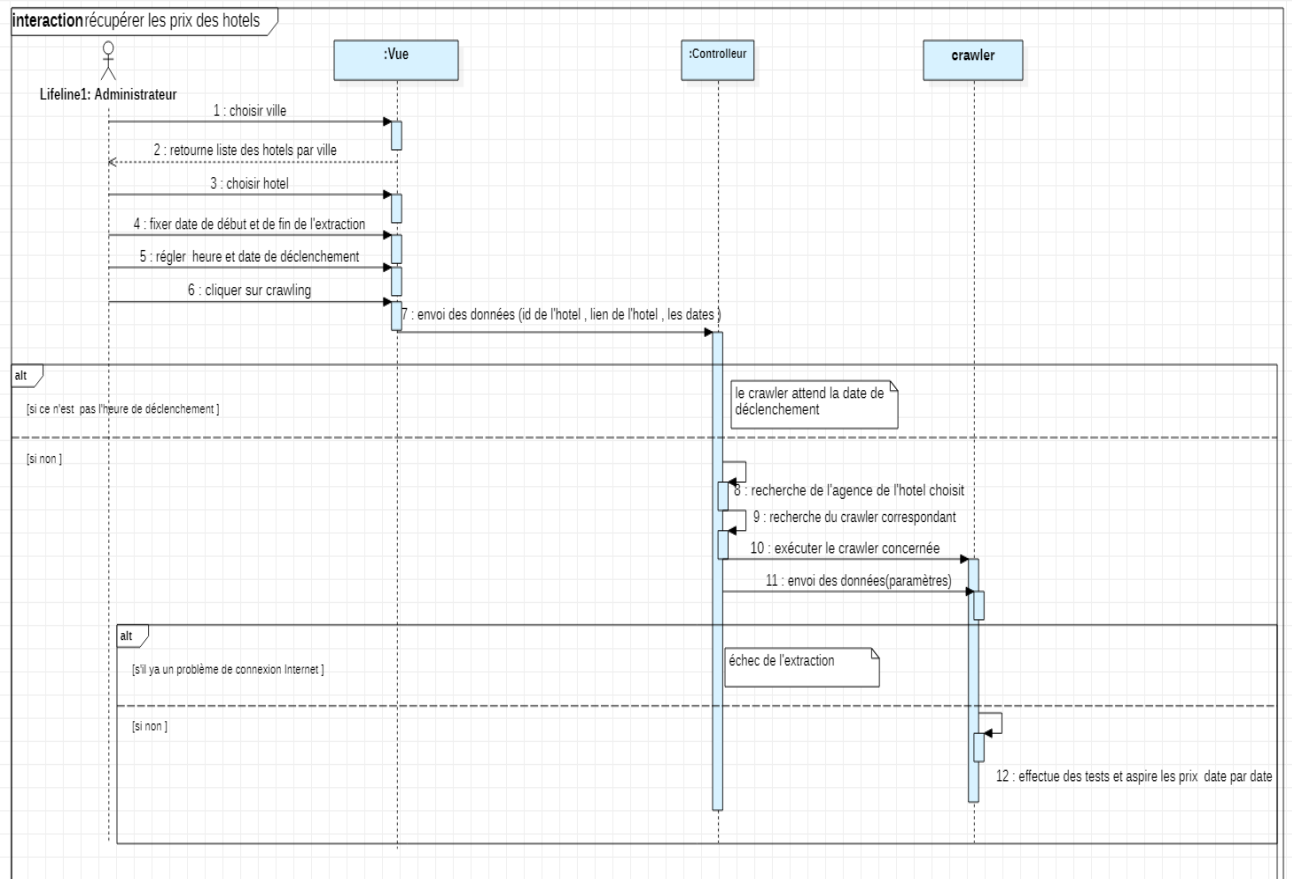


Figure 29:diagramme de séquence « récupérer les prix des hôtels »

Le scénario de récupération de données ne s'arrête pas à ce niveau : une fois le crawler a récupéré le prix de la première date, il doit l'insérer dans la base de données. Pour ce faire, d'abord le crawler envoie la date à la base de données. Si la connexion avec la base de données a échoué, il y'a retour d'un échec. Si non le crawler envoie une requête select pour vérifier l'existence de cette date dans la base de données et si elle possède un prix. Cela est modélisé par une étiquette alt. Si la date est trouvée, le crawler met à jour le prix correspondant à cette date en envoyant une requête update. Si non, le crawler envoie une requête insert pour ajouter le prix récupéré avec la date correspondante dans la base de données. Ceci est modélisée par une autre étiquette alt. La figure 30 expose le diagramme de séquence de mise à jour des prix existants.

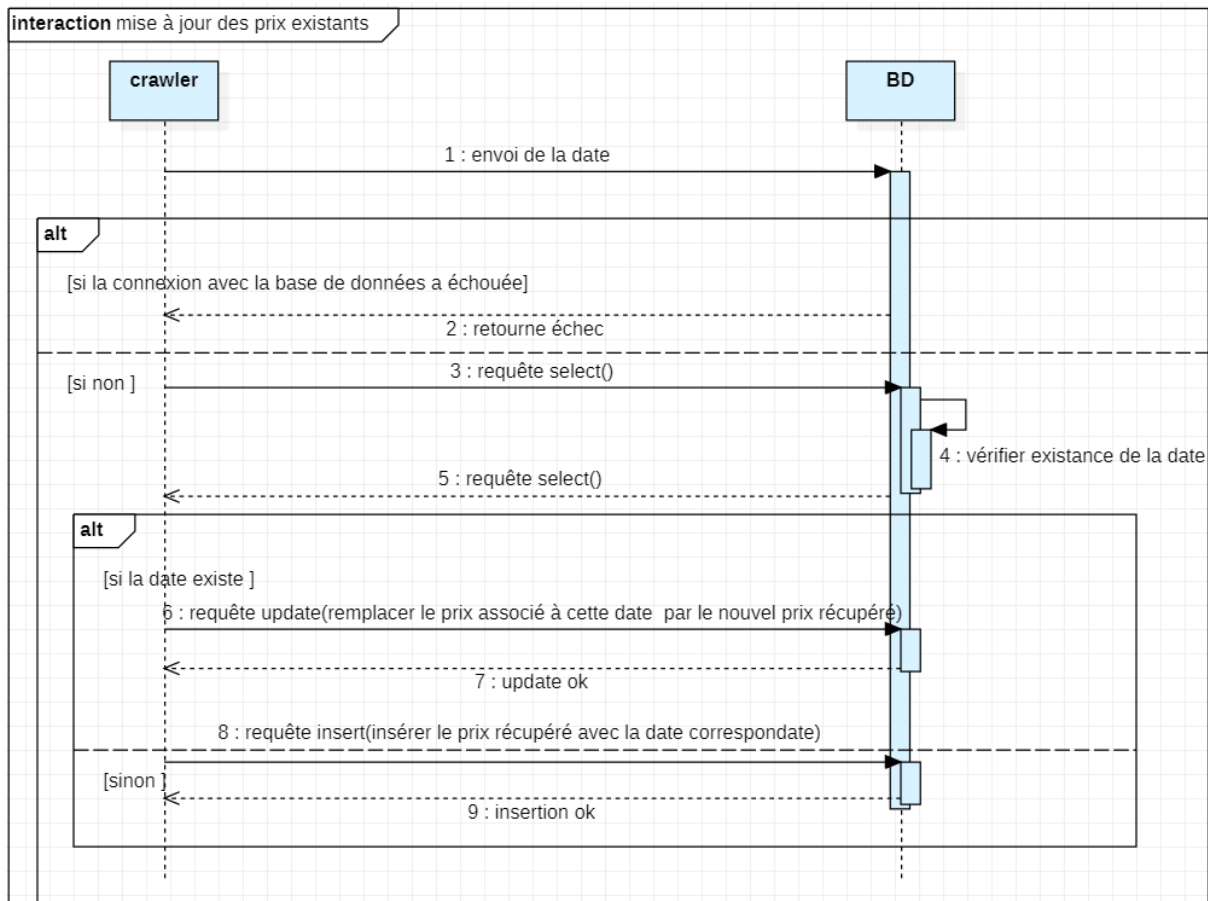


Figure 30:diagramme de séquence « mise à jour des prix existants »

3.4 Réalisation

3.4.1 Les interfaces développées

Dans cette section nous allons présenter l'interface de crawling qui englobe le travail effectué durant ce sprint :

L'administrateur consulte l'interface de crawling, à partir de laquelle il peut choisir les hôtels dont il veut mettre à jour les prix. Pour ce faire, il commence à choisir la ville puis les hôtels. Il existe trois calendriers : deux pour choisir la date de début et de fin de l'extraction et une dernière pour régler la date déclenchement et un bouton crawling pour valider. La figure 31 présente cette interface :

Admin

Search for...

Providers

Mapping

Crawling

Statistics

City:

☐ SOUSSE ☐ JERBA ☐ MAHDIA ☐ HAMMAMET ☐ TABARKA ☐ MONASTIR ☐ TUNIS ☐ KEBELI ☐ GAMMARTH ☐ TOZEUR ☐ DOUZ ☐ AIN DRAHAM ☐ ZARZIS ☐ NABEUL

Hotel:

Select city

Choose date:

From:

To:

Trigger date:

Crawling

Logged in as: najib belhadj

Copyright © 2020

Gestionnaire des tâches

Privacy Policy - Terms & Conditions

Figure 31:Interface du crawling

3.4.2 Test et Validation

Le tableau 9 expose la validation du deuxième sprint :

Fonctionnalité	Etat	Validation
Choisir hôtel par ville	Fait	Satisfait
Fixer l'intervalle de l'extraction	Fait	Satisfait
Régler l'heure et la date de déclenchement	Fait	Satisfait
Extraction des prix dans un intervalle bien déterminé	Fait	Satisfait
Mise à jour des prix dans la base de données	Fait	Satisfait

Tableau 9 : Test sprint2

Dans l'ensemble tout fonctionne bien dans ce sprint à l'exception de l'extraction des prix qui été très difficile et qui nous a engendré un peu de retard pour l'accomplir.

4 Sprint 3: mapping et comparaison des prix

Ce sprint a pour but de présenter le développement de la partie mapping et le développement de la partie web et mobile de la comparaison des prix. En suivant la même démarche suivit dans les sprints précédents, nous débutons par élaborer le backlog du sprint.

Par la suite nous présentons la partie conceptuelle et nous terminons par l'exposition de la réalisation et la validation.

4.1 Sprint Backlog :

Le tableau 10 représente le sprint backlog de ce sprint :

ID	User story	Taches
U12	En tant qu'administrateur, je veux lancer le mapping.	-Ajout interface de mapping -Configuration mapping
U13	En tant que visiteur, je veux choisir un hôtel ou tous les hôtels d'une ville	-Ajout du script de récupération des prix à partir de la date choisit par l'utilisateur -Installation et Configuration du
U14	En tant que visiteur, je veux choisir la date pour consulter la comparaison des prix	composant highchartsbundle pour afficher la comparaison des prix. -Développement d'une API mobile
U15	En tant que visiteur, je veux consulter les statistiques	-Développement d'une Interface mobile pour le visiteur peut voir les statistiques des prix (y compris le minium des prix , la moyenne, le maximum)

Tableau 10 : backlog sprint 3

4.2 Diagramme de cas d'utilisation sprint3

La figure 32 présente le diagramme de cas d'utilisation du sprint3

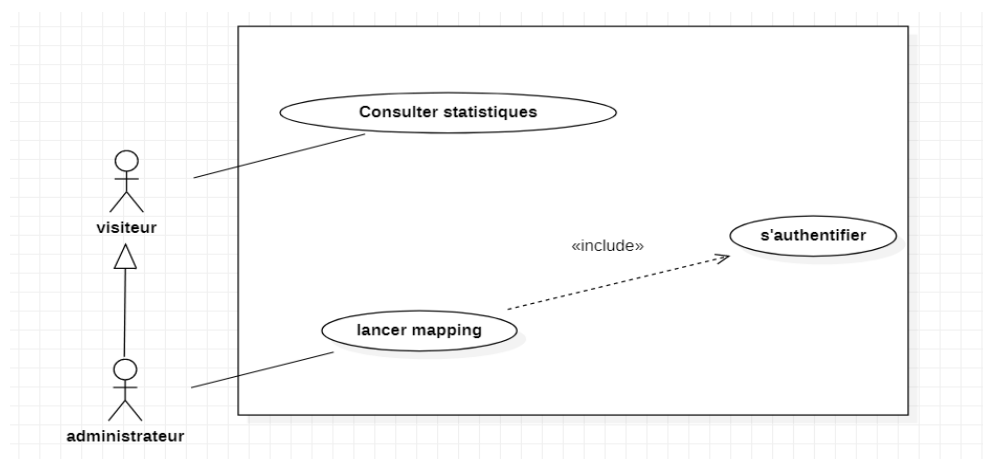


Figure 32 : diagramme de cas d'utilisation sprint 3

4.2.1 Raffinement du cas d'utilisation « lancer mapping »

La figure ci-dessous illustre le raffinement de l'item « lancer mapping » (user story U12) :

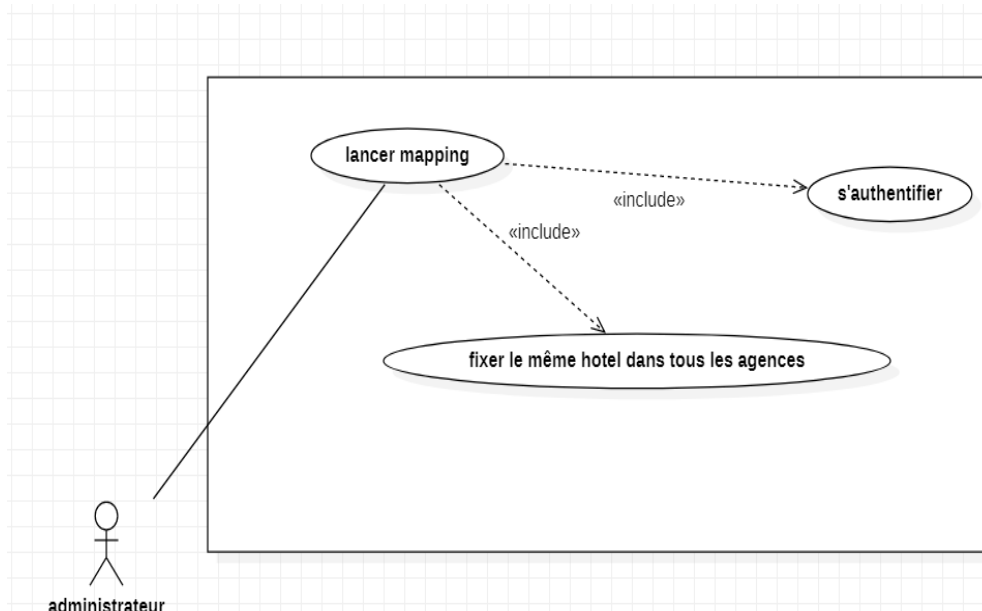


Figure 33:diagramme de cas d'utilisation « lancer mapping »

4.2.2 Tableau textuel

- Description du cas d'utilisation « lancer mapping »

Titre	Lancer mapping
Résumé	L'administrateur relie un hôtel de l'agence principale avec l'hôtel qui convient dans les autres agences du site s'il existe
Acteur	Administrateur
Précondition	Administrateur authentifié et consulte l'interface de mapping
Scénario principal	1- L'administrateur choisit un hôtel de la liste hôtel de l'agence principal et son correspondant dans les autres agences du site. 2- L'administrateur clique sur mapper 3- Le système relie les hôtels (remplit la case RefHotel de l'hôtel qui convient dans les autres agences par l'id de l'hôtel de l'agence principale)
Scénario alternatif	Transaction impossible avec la base de données
Postcondition	RefHotel changé (les hôtels deviennent reliés)

Tableau 11:Description textuelle du cas d'utilisation «lancer mapping »

4.2.3 Raffinement du cas d'utilisation « consulter statistiques »

La figure 34 illustre le raffinement de l'item « consulter statistiques» (user stories U12, U13,U14)

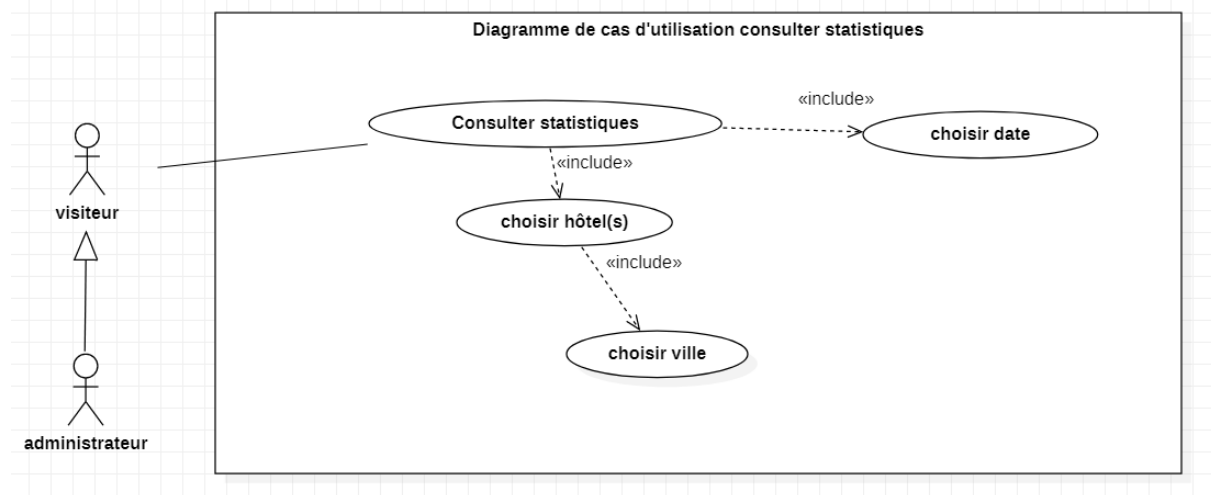


Figure 34 : diagramme de cas d'utilisation « consulter statistiques »

4.2.4 Tableau textuel

- Description du cas d'utilisation « consulter statistiques »

Titre	Consulter statistiques
Résumé	Le visiteur consulte les statistiques de prix des hôtels d'une ville choisi
Acteur	Visiteur /Administrateur
Précondition	Consulter l'interface de statistiques
Scénario principal	<p>1-Le visiteur choisit la ville par la suite il choisit un hotel ou tous les hôtels de cette ville</p> <p>2-Le visiteur saisit la date d'arrivé et la date de départ puis il clique sur le bouton « show statistics »</p> <p>3-Le système affiche la variation du prix de chaque hôtel de la ville chez les différentes agences du site sous forme des courbes et affiche les statistiques suivantes pour chaque hôtel ainsi que dans la région choisi (minimum des prix, maximum des prix, moyenne des prix)</p>
Scénario alternatif	Connexion avec la base de données échouée
Postcondition	Statistiques affichée

Tableau 12:Description du cas d'utilisation « consulter statistiques »

4.3 Diagramme de séquence

Dans ce paragraphe nous présentons les diagrammes de séquence les plus essentiels dans le sprint3 tel que lancer le mapping et consulter comparaison prix.

Afin de relier un hôtel de l'agence principale à l'hôtel qui convient dans les autres agences s'il existe, l'administrateur commence par choisir un hôtel de l'agence principale. Par la suite, il choisit l'hôtel correspondant dans les autres agences du site. En cliquant sur le bouton mapper la vue envoie les id des hotels choisis vers le contrôleur. Ce dernier exécute une requête insert afin d'insérer l'id de l'hotel de l'agence principale dans la case RefHôtel des hôtels correspondants. Une fois l'insertion est effectuée avec succès les hôtels deviennent mappés et page de mapping est affiché de nouveau .

La figure 35 présente le diagramme de séquence du lancement du mapping (user story U12) :

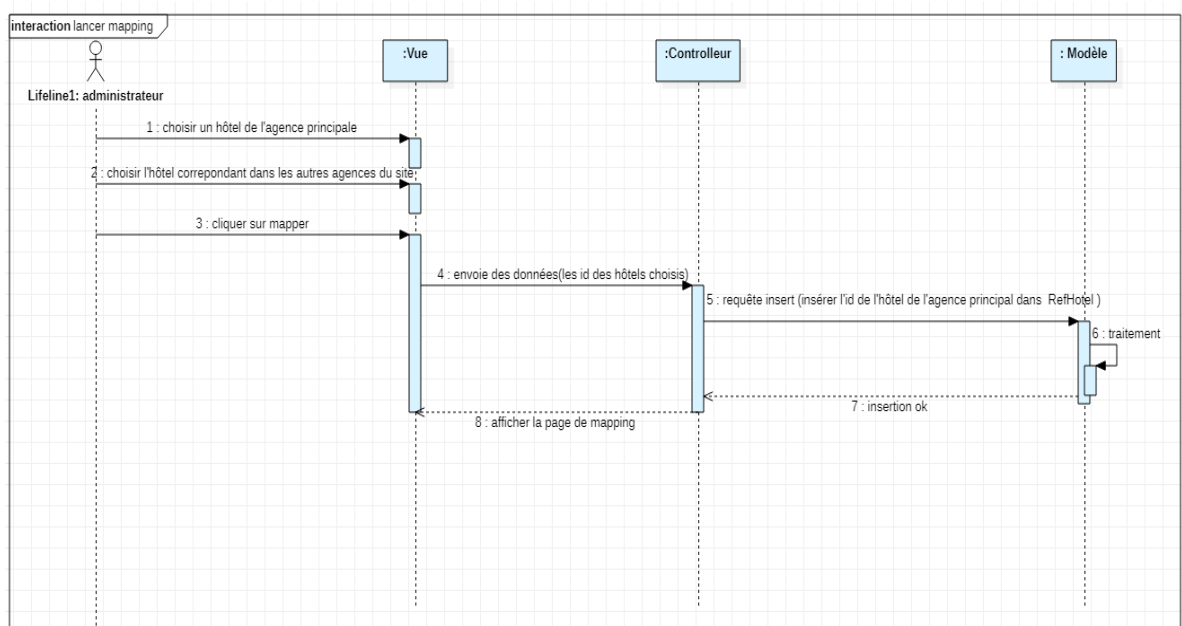


Figure 35: diagramme de séquence « lancer mapping »

Pour voir les statistiques des prix des hôtels l'administrateur doit choisir un hôtel ou tous les hôtels d'une ville par la suite il choisit la date d'arrivé et la date de départ.

On suppose que l'administrateur a choisi tous les hôtels d'une ville.

En cliquant sur le bouton « show statistics » la vue envoie l'id de la ville et les dates choisies vers le contrôleur. Ce dernier envoie une requête « select » pour chercher les hôtels de référence de la ville choisit et dès qu'il reçoit le résultat de la requête il est invité à plusieurs reprises à chercher les hôtels associés à chaque hôtel de reference dans tous les agences. Cela est modélisé par une étiquette « Loop ». Pour chaque hôtel trouvée le contrôleur envoie une

requête « rechercheprix » pour récupérer ses prix. Cette notion est modélisé par une deuxième étiquette « Loop ». Si la requête est vide aucun prix ne sera ajouté à la courbe de l'hôtel. Si non les prix récupérés seront filtrés selon les dates choisis et par la suite il seront ajoutés à la courbe et enfin le contrôleur effectue le calcul des statistiques de l'hôtel et les envoi vers la vue. Cela est modélisé par une étiquette « alt ».Après le succès de la dernière étape, le contrôleur effectue les calculs des statistiques des prix dans la ville et les envoi à la vue.

La figure 36 représente le diagramme de séquence correspondant au cas d'utilisation consulter statistiques (user story U13 , U14, U15) :

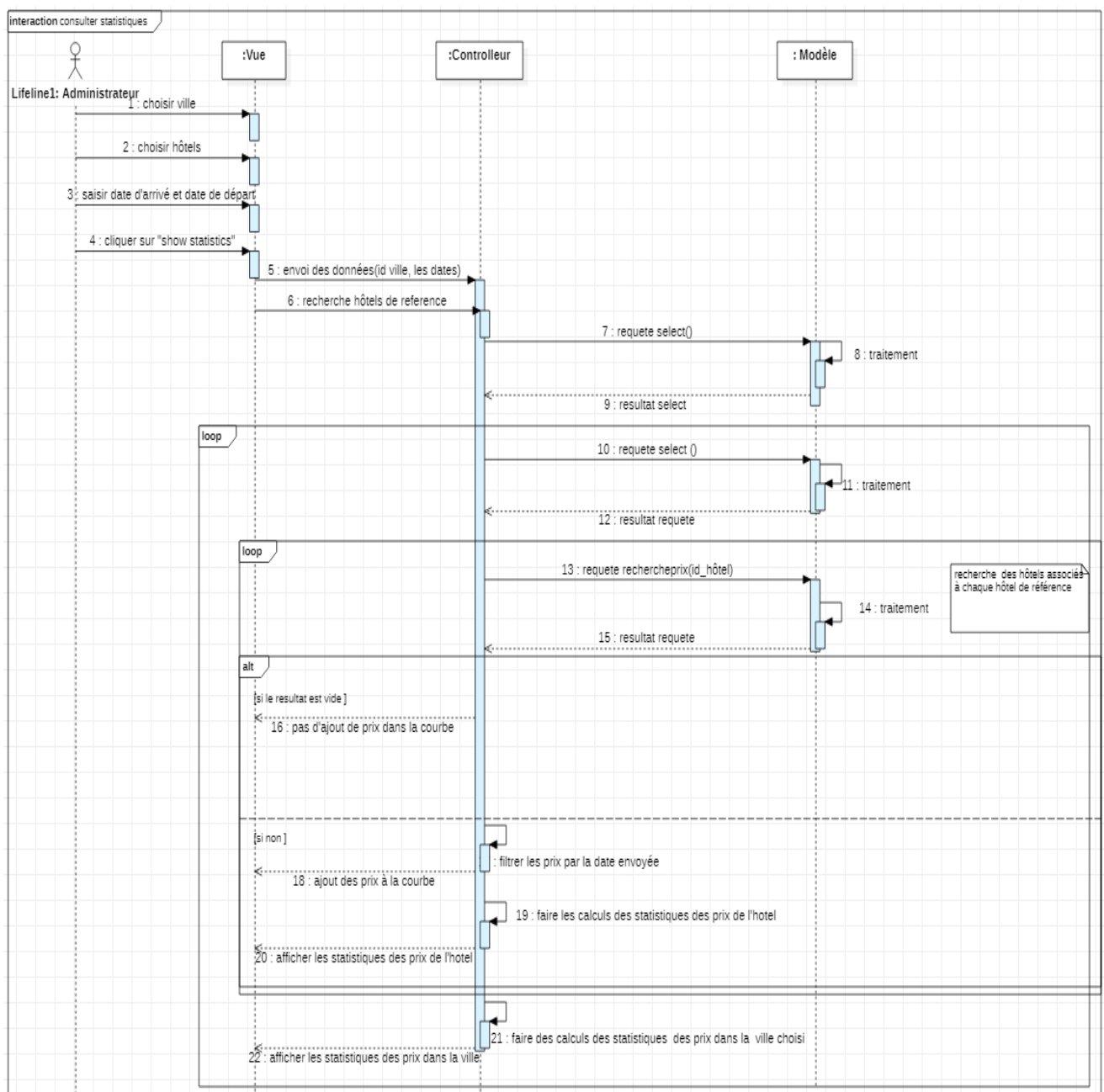


Figure 36 : diagramme de séquence « consulter statistiques »

4.4 Réalisation

Dans cette partie nous présentons les interfaces et les fonctionnalités fournies à l'administrateur et à l'utilisateur telles que l'interface de mapping et l'interface de recherche pour la comparaison des prix.

4.4.1 Les interfaces développées

L'administrateur consulte l'interface de mapping. Pour lancer le mapping, il doit choisir un hôtel de la liste déroulante des hôtels du main provider et son correspondant dans les autres agences. Il existe un bouton « mapper » au-dessous des listes déroulantes permet du lancement du mapping. L'interface de mapping est présentée dans la figure 37 :

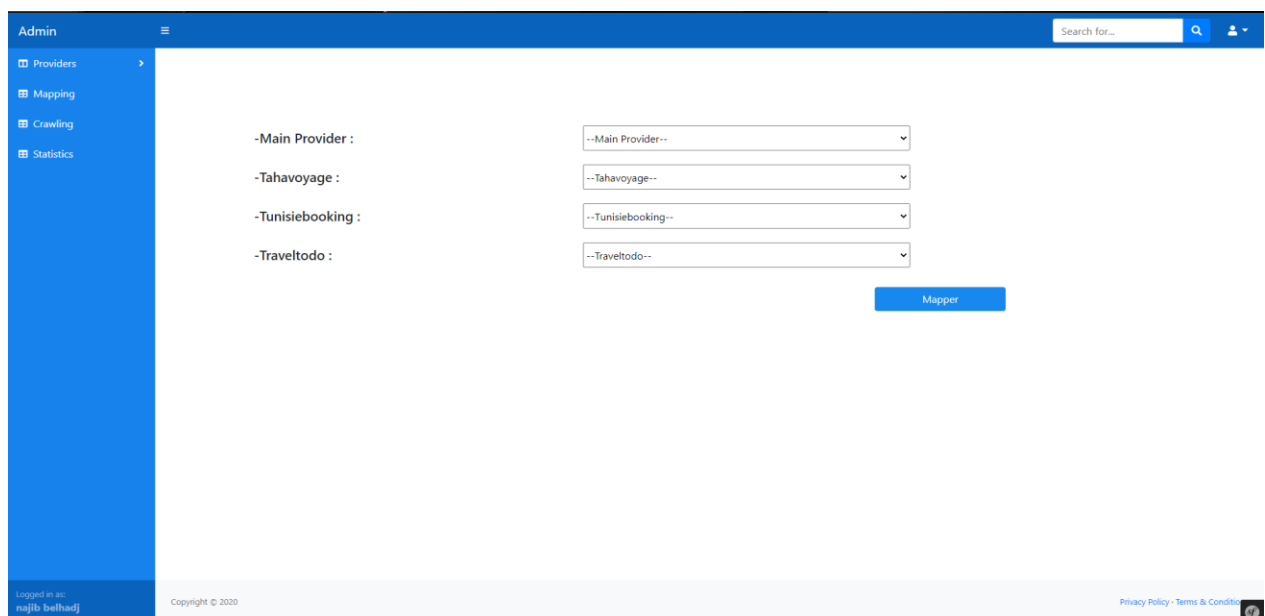


Figure 37:Interface de mapping

Une fois les hôtels sont mappés l'administrateur peut consulter l'interface de statistiques afin de voir la comparaison des prix. Pour commencer, il choisit un hôtel ou tous les hôtels d'une ville. Il existe deux calendriers pour choisir la date de départ et la date d'arrivée et un bouton « show statistics » pour accéder aux statistiques. Dans l'exemple ci-dessous on a choisi de voir les statistiques de tous hôtels de la région de Tunis du 20/06/20 jusqu'à 30/06/2020. Les figures 38 et 39 présentent le modal de statistiques qui sera affiché :

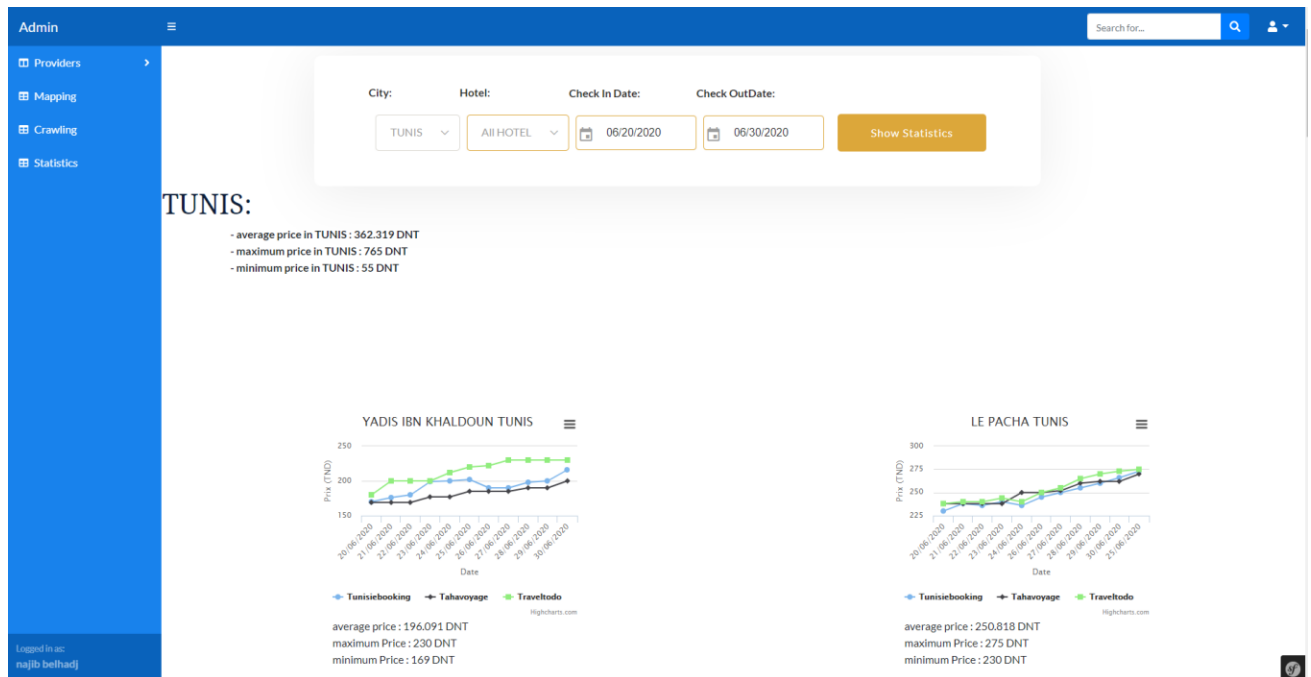


Figure 38:Interface de statistiques 1



Figure 39:Interface de statistiques 2

Le visiteur consulte l'application mobile, il peut choisir un hôtel de la liste hôtels de référence et par la suite il fixe la date de début et de fin de la comparaison et valide en cliquant sur le bouton « compare price ». Par exemple, on choisit l'hôtel RADISSON BLU RESSORT & SPA Hammamet et on choisit la date du 23/06/2020 jusqu'à 07/07/2020.

La figure 40 présente cette interface.

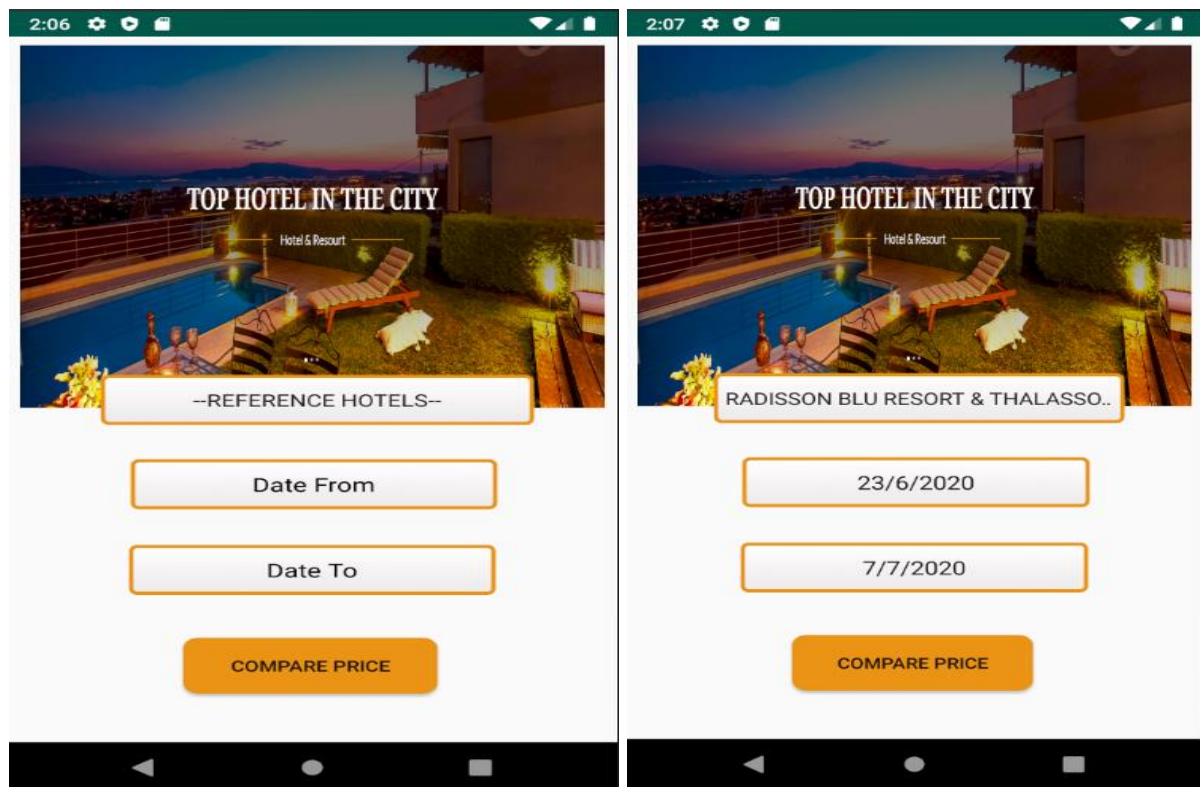


Figure 40 : Interface Mobile de recherche

Dès que le visiteur valide ses choix, une courbe de comparaison sera affichée contenant les prix de tous les agences du site. La figure 41 montre cette interface.

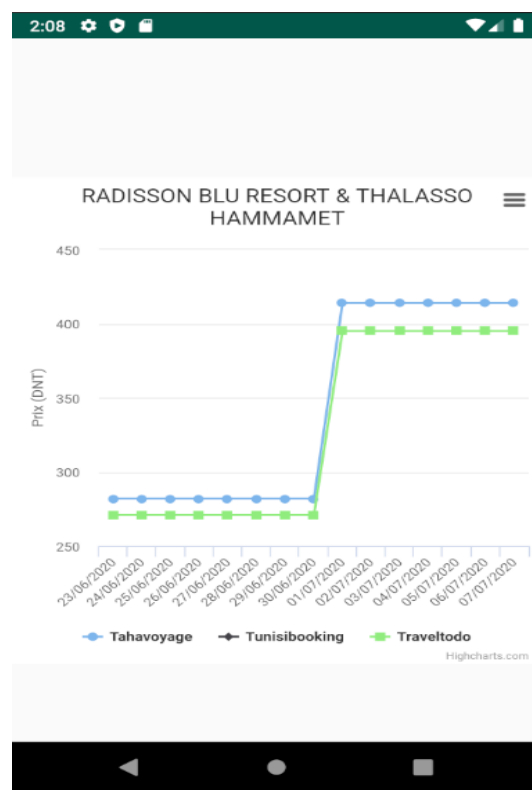


Figure 41: Interface Mobile de comparaison de prix d'un hôtel

4.4.2 Test et validation

Fonctionnalité	Etat	Validation
Visualiser les hôtels de référence	Fait	Satisfait
Visualiser les hôtels de chaque agence	Fait	Satisfait
Mise à jour reference	Fait	Satisfait
Choisir les dates	Fait	Satisfait
Courbe comparaison des prix	Fait	Satisfait
Statistiques des prix pour chaque hôtel	Fait	Satisfait
Statistiques des prix dans une région	Fait	Satisfait

Tableau 13 : test sprint 3

5 Conclusion

Dans ce chapitre, nous avons présenté l'avancement et la mise en œuvre du projet avec Scrum. Le projet a été divisé en trois Sprints : « authentification, gestion agence , gestion hôtel » , « module de crawling » « mapping et comparaison prix ». Nous terminons ce rapport par une conclusion générale.

Conclusion générale et perspectives

À la fin de ce rapport, nous pouvons conclure que le stage de fin d'études nous a procuré l'opportunité de confronter nos connaissances théoriques à l'environnement pratique et d'améliorer nos acquis.

Le présent travail a consisté à mettre en place une application web et mobile ayant pour but de comparer les prix de réservation d'hôtels par rapport aux concurrents qui sera très bénéfique pour le visiteur de point de vue gain de temps et facilitation du processus de recherche d'un prix raisonnable. De plus, nous avons intégré des statistiques pour un seul hôtel et pour les hôtels d'une région choisi qui sont très utiles pour aider le visiteur à choisir la bonne destination avec le meilleur prix. Nous avons commencé la réalisation de notre projet par la collecte des informations nécessaires qui nous ont permis de répondre à notre problématique, tout en spécifiant les besoins non fonctionnels et les besoins fonctionnels.

Nous avons essayé de développer notre solution incrément par incrément en utilisant la méthodologie Scrum.

La réalisation de ce projet nous a également apporté de grands apports à plusieurs niveaux. En termes de technologies, nous avons appris à manipuler le framework Symfony 4.4, Android Studio et de nombreux autres outils. Nous avons en également la possibilité de maîtriser la conception en utilisant la méthode UML.

Toutefois, il est indéniable d'affirmer que la réalisation de ce projet n'était pas sans difficultés. En fait, ces difficultés sont principalement résumées dans la réalisation de la partie crawling. Mais ce ne sont pas de grands obstacles, au contraire ils ont stimulé notre curiosité et notre volonté pour arriver à atteindre nos objectifs.

Finalement, les principaux objectifs de la solution ont été atteints et le système que nous avons mis en œuvre aurait être enrichi par d'autres fonctionnalités plus avancées selon les besoins des clients tel que l'intégration des nouvelles statistiques, un module de réservation en ligne après la consultation des statistiques et l'ajout des autres agences dans notre site.

Bibliographie

[B1] (en ligne) Ken Schwaber et Jeff Sutherland, « Le Guide Scrum »

https://www.scrumguides.org/docs/scrumguide/v2017/2017-Scrum-Guide-French.pdf?fbclid=IwAR0LlGhvKA9LH7-gqQgyq0Dw-6jyi_g-HPKWWbxa0I7lPzXtctrv166j420, consulté le 2 Mars 2020.

[B2] (en ligne) Pascal Roques, “UML2 - Modéliser une application web « Les cahiers du programmeur »» http://www.academiepro.com/uploads/cours/2015_10_07_%5BEYROLLES%5D_UML2_-_Modeliser_une_application_web.pdf, consulté le 2 Avril 2020.

Webographie

[W1]:<https://www.journaldunet.fr/web-tech/dictionnaire-du-webmastering/1203499-crawler-definition-traduction-et-acteurs/> (consulté le 5 février 2020)

[W2]:<https://finobuzz.com/2015/08/09/gestion-de-projet-methodes-classiques-vs-methodes-agiles/> (consulté le 7 mars 2020).

[W3]:https://www.pmconsultant.fr/blog_des_ti/uml/faq_ea/modeliser_un_acteur_non_humain.html (consulté le 4 avril 2020)

[W4]:<https://www.mysql.com/fr/products/workbench/> (consulté le 5 avril 2020)

[w5] :<https://fr.wikipedia.org/wiki/XAMPP> (consulté le 5 avril 2020)

[W6]:<https://fr.wikipedia.org/wiki/Symfony> (consulté le 5 avril 2020)

[W7]:<https://www.01net.com/telecharger/windows/Programmation/creation/fiches/130819.html> (consulté le 5 avril 2020)

[W8]:[https://en.wikipedia.org/wiki/Selenium_\(software\)](https://en.wikipedia.org/wiki/Selenium_(software)) (consulté le 5 avril 2020)

[W9]:<https://www.clubic.com/telecharger-fiche433537-android-studio-2.html> (consulté le 5 avril 2020)

[W10]:<https://docs.staruml.io/> (consulté le 17 avril 2020)

[W11]:https://fr.wikipedia.org/wiki/Architecture_trois_tiers (consulté le 5 mai 2020)

[W12]: <https://fr.wikipedia.org/wiki/Mod%C3%A8le-vue-contr%C3%B4leur> (consulté le 5 mai 2020)

[W13]:<https://lipn.univ-paris13.fr/~gerard/uml-s2/uml-cours05.html> (consulté le 15 mai 2020)

Résumé

Le présent rapport synthétise notre stage de fin d'études en vue de l'obtention du diplôme de licence appliquée en Technologies des Communications à l'Institut Supérieur d'Informatique et de Techniques de Communication à Hammam Sousse, effectué au sein de la société IPRECISION. Dans ce contexte, notre projet consiste à développer une plateforme web et mobile de comparaison de prix de réservation d'hôtels à partir d'un module de crawling des sites WEB des agences. La gestion de ce projet a été avec Scrum. Pour la partie Web nous avons utilisé Symfony 4.4 comme framework de développement qui est basé que le modèle MVC, nous avons également opté pour Android Studio afin de développer la partie mobile et nous avons MYSQL coté Backend.

Mots clés :Crawling, scrum, Symfony 4.4, MVC, MYSQL, Android Studio

Abstract

This report summarizes our final period of study for the degree in Communications Technologies at the Higher Institute of Computer Science and Communication Techniques in Hammam Sousse, carried out within the company IPRECISION. In this context, our project consists in developing a web and mobile platform to compare hotel booking prices based on the crawling of agencies' websites. The management of this project was with Scrum. For the Web part we used Symfony 4.4 as the development framework which is based on the MVC model, also we used Android Studio to develop the mobile part and we have MYSQL in the Backend side.

Key words : Crawling, scrum, Symfony 4.4, MVC, MYSQL, Android Studio