
Rapport de Projet de Fin de formation

Filière Techniques Développement Informatique

Institut spécialisé dans les métiers de l'offshoring - Tétouan

Promotion 2019 – 2020

**Développement d'une application web
pour la gestion d'un magasin des livres**

Réalisé par : Oweis Youssef

Encadre par :

M. RAHMOUNI Oussama

Professeur à l'ISMO

Dédicaces

A mes chers parents,

Aucune dédicace ne serait témoin de mon profond amour et ma grande gratitude, car je ne pourrais jamais oublier votre soutien continu, votre tendresse et votre amour, vous étiez toujours présents avec vos sacrifices, afin de m'offrir un grand secours tout au long de toutes ces années, je vous dédie ce modeste travail comme ma reconnaissance pour tous vos efforts.

A toute la famille,

Je vous remercie pour votre soutien, et pour vos encouragements sincères que vous m'avez pas manqué de m'offrir pendant les moments où j'avais besoin.

A tous mes amis,

Je vous remercie pour vos encouragements, pour les bons moments et pour le soutien que vous m'avez offert.

A mes chers professeurs,

Je remercie tous mes professeurs tout au long de mon parcours à L'institut Spécialisé dans les Métiers de l'Offshoring, pour leur générosité, et leur soutien, et je remercie énormément tous mes professeurs.

A tous les membres de l'administration,

J'offre ce modeste travail également à tous les membres de l'administration de L'institut Spécialisé dans les Métiers de l'Offshoring de Tétouan.

Remerciements

Au terme du projet de fin d'étude, je tiens à exprimer mes sincères remerciements à tous ceux qui m'ont aidé dans l'élaboration de ce projet.

Je remercie infiniment mon encadrant Monsieur RAHMOUNI Oussama, mon professeur à

L'institut Spécialisé dans les Métiers de l'Offshoring de Tétouan, pour en raison de son dévouement à l'enseignement de la programmation et de toutes les applications qu'il a construites avec nous au cours de cette année, j'ai beaucoup appris sur la programmation orientée objet et la gestion de bases de données tout en utilisant C Sharp pour créer des applications réelles.

J'exprime aussi ma profonde gratitude à tout le cadre administratif et professoral de L'institut Spécialisé dans les Métiers de l'Offshoring de Tétouan, pour leurs efforts considérables et la formation prestigieuse qu'ils nous ont prodiguée.

Résumé

Le présent rapport constitue une synthèse du projet de fin d'études, pour obtenir mon diplôme de technicien spécialisé en développement informatique et en raison de la pandémie actuelle, j'ai eu la possibilité de créer l'application par moi-même, et aussi de choisir les technologies utilisées.

L'objectif de ce projet est de créer une application basée sur Spring et React JS pour un projet de vente des livres, cette application doit faciliter la gestion des ventes et location des livres aussi gérer les revenus.

Le projet a été développé avec le langage Java sous le Framework Spring, la librairie Lombok, ainsi React JS pour le Front-End.

Abstract

This report is a synthesis of the final project, to obtain my diplomat of specialized technician, in software development, and in light of the current situation of the global pandemic, I had the chance to create the whole application by myself and choose the technologies to use

The objective of this mission is to create an application based on Spring and React JS for a bookstore manager application, this application must simplifies book store management and also offer the ability to handle sells, lends and manage the store budget.

The project was developed using Java language with the Spring framework, Lombok library also React JS for the front-end part.

Liste des figures

Figure 1: Diagramme de cas d'utilisation.....	5
Figure 2 : Diagramme de classe.....	5
Figure 3 : Interface Home.....	13
Figure 4 : Interface Book.....	14
Figure 5 : Interface Book – Lend.....	15
Figure 6 : Interface Book - Show Lended.....	15
Figure 7 : Interface Revenue.....	16

Table des matières

INTRODUCTION GENERALE	1
CHAPITRE 1 : SPECIFICATION DES EXIGENCES	2
I. Spécification des exigences fonctionnelles	3
1. Analyse et identification des exigences	3
2. Spécification des exigences	3
II. Validation des exigences	4
1 Cas d'utilisation : Gestion de la liste d'envies	4
3. Diagramme de classe	5
CHAPITRE 2 : ENVIRONNEMENT DE DEVELOPPEMENT	7
1. Le langage de développement	9
2. Front End	9
3. Back End	10
4. Base de données	10
5. Outil de développement	11
CHAPITRE 3 : APPLICATION	12
I. Interface Home	13
II. Interface Book	14
1. Interface Book : Lend	15
2 Interface Book : Show Lended	15
III. Interface Revenue	16
CONCLUSION GENERALE	17
REFERENCES	18

Introduction générale

Durant nos jours, Un projet commercial ne peut pas survivre dans un marché sans exploiter la puissance de domaine informatique. Un magasin des livres n'est pas l'exception, du coup, améliorer son process et comment gérer ses opérations doit être une priorité.

Vu que le web a imposé son rôle dans le monde des affaires, il fallait donc l'intégrer avec des plateformes et des solutions révolutionnaires, garantissant une distinction au niveau du web. Une approche pareille ne va pas juste moderniser la stratégie de notre magasin, mais aussi minimiser le temps nécessaire pour tous nos activités, ce qui nous permet de grandir comme un projet commercial.

Afin de bien cerner le projet dans son contexte et mieux comprendre les différentes phases de sa réalisation, nous avons découpé ce rapport en trois chapitres.

Le première chapitre invoque les spécifications des besoins fonctionnelles et non fonctionnelles, et introduire les capacités de notre application. Et l'étude conceptuelle du projet via les différents diagrammes UML modélisant de notre système. Nous verrons le comportement statique et dynamique de tous les composants de notre système.

Le deuxième chapitre présente l'environnement techniques et les différents outils utilisés durant ce projet.

Le dernier chapitre présent une description pour chaque interface et une simulation pour chaque opération possible.

Chapitre 1 : Spécification des exigences

Ce chapitre invoque la spécification des besoins fonctionnels et non fonctionnels, et présente une modélisation globale du notre projet.

Introduction

Avant de se lancer dans la discussion des aspects techniques, la définition, les outils logiciels utilisés, la réalisation ainsi que les autres axes, il faut tout d'abord introduire et invoquer les spécifications des besoins fonctionnelles du projet afin d'être capable de réagir de la manière la plus pertinente. Le processus de compréhension et de formalisation des besoins et des contraintes de développement demeure l'un des points les plus essentiels, qui requiert un esprit d'analyse et de description élevé ; la qualité des spécifications fonctionnelles décide le succès ou l'échec d'un produit informatique.

Tout au long de ce chapitre, nous allons essayer d'être le plus clair possible et introduire en détails les différentes spécifications fonctionnelles.

I. Spécification des exigences fonctionnelles

1. Analyse et identification des exigences

Il s'agit d'une application web dynamique de gestion avec des interfaces qui respecte les normes UX/UI. Le site offre les fonctionnalités suivantes :

- Permet de présenter les livres, ainsi leurs caractéristiques
- Permet la possibilité d'ajouter un nouveau livre
- Permet à l'utilisateur de marquer la vente d'un livre
- Permet à l'utilisateur de marquer la location d'un livre
- Permet à l'utilisateur de marquer quand un client retourne un livre
- Permet de suivre les revenus avec des détails pour chaque opération (type, gain, date)

2. Spécification des exigences

Après une étude des exigences qui contribue à la compréhension des buts de l'application, on passe vers l'élaboration d'un cahier des charges fonctionnelles, où nous présentons les détails des exigences, et les règles de gestion.

Pour gérer un projet de vente des livres, initialement, il faut avoir une interface de gestion des livres - gestion de stock - et un autre pour gérer la vente et la location des livres. Pour faciliter

l'UX pour nos utilisateurs, on a décidé de créer une seule interface qui nous permet de gérer les deux aspects.

D'un autre part, on a décidé de créer une deuxième interface pour visualiser les revenus de notre magasin des livres, offrant plus de détails sur chaque opération.

Interface book : Présentation des livres ainsi les actions : "Sell", "Lend", "Return a book"

Interface Revenue : Présentation des revenus avec leurs détails.

II. Validation des exigences

Afin de valider les exigences de notre projet, nous avons choisi d'adopter la technique des cas d'utilisation pour assimiler les besoins fonctionnels, vu que ce dernier est clair et va faciliter le développement de notre application. Un cas d'utilisation représente un ensemble de séquences d'actions réalisées par le système et produisant un résultat observable intéressant pour un acteur particulier.

Pour constituer les cas d'utilisation, il faut considérer l'intention fonctionnelle de l'acteur par rapport au système dans le cadre de l'émission ou de la réception de chaque message. Les cas d'utilisation sont obtenus, en regroupant les intentions fonctionnelles en unités cohérentes.

Une vue globale de notre donnée sera primordiale, pour cela, on a utilisé la méthode diagramme de classes. Ce dernier présente la structure et les relations entre nos données, et ça sera un outil nécessaire pour bien décider si on a besoin d'ajouter/enlever des informations durant la phase de développement.

Ainsi, il faut considérer seulement les données nécessaires pour notre système, sans duplication des informations entre les tableaux, pour optimiser notre base de données et le temps du traitement nécessaires pour nos opérations CRUD.

Dans notre cas, on va présenter les cas d'utilisation d'un gérant d'un projet de vente et allocation des livres.

1. Cas d'utilisation : Gestion des livres

Dans les diagrammes suivants, nous allons illustrer l'interaction entre l'utilisateur et notre application.

La figure suivante présente le diagramme de cas d'utilisation de la gestion des livres :

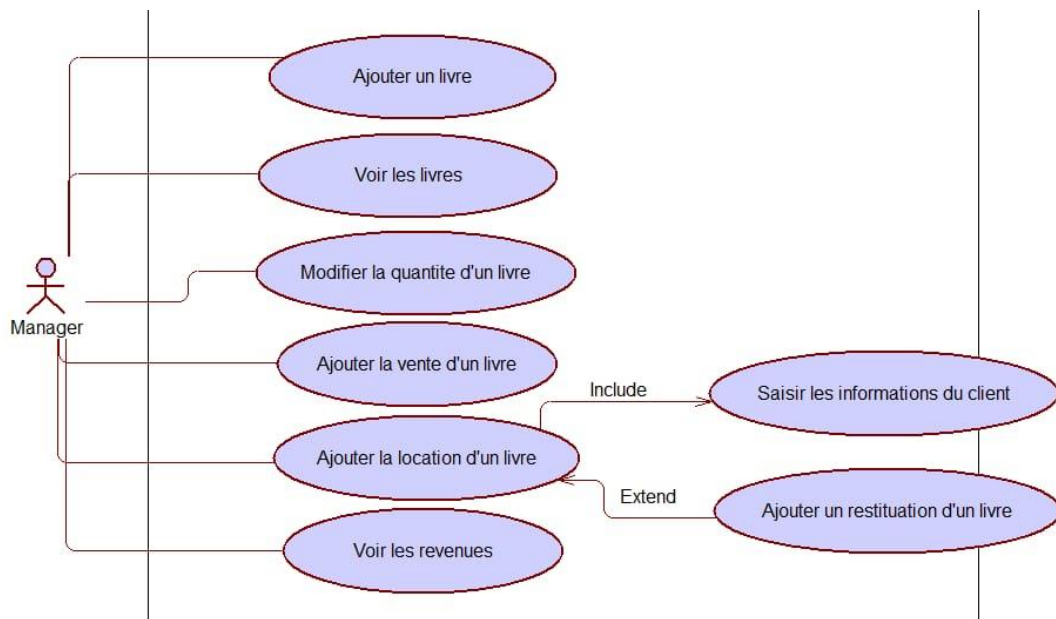


Figure 1 : Diagramme de cas d'utilisation

2. Diagramme de classes : Modélisation des données

On a suivi une approche pratique pour notre diagramme de classe, qui va répondre à tous les cas d'utilisation qu'on a déjà défini, en même temps, assurer un temps de traitement minimale.

Le tableau "Book" : contient les détails de notre livre : nom, description, prix et nombre.

Le tableau "Revenue" : contient le nom du client, son CIN, la date de l'opération, et le type.

Un seul "Book" peut avoir plusieurs "Revenue", vu qu'on peut le louer aux plusieurs clients.

PS : Ce n'est pas nécessaire d'avoir un tableau "Client" sur notre système, parce qu'on s'intéresse aux changements du notre inventaire des livres et nos revenus, plutôt que les informations des clients.

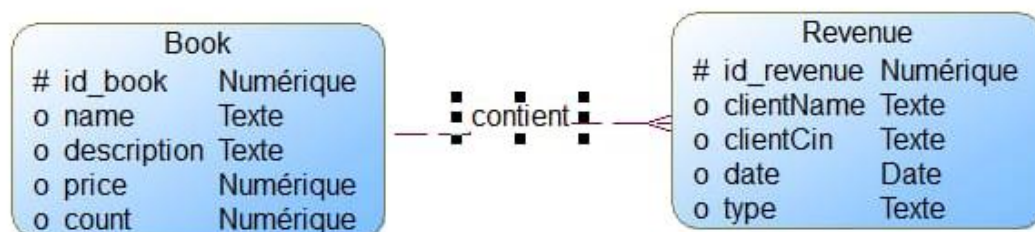


Figure 2 : Diagramme de classe

Conclusion

L'étude fonctionnelle était le premier pilier dans le projet. En effet, à travers cette étude, nous avons pu cerner la structure et les différents scénarios de l'application.

Nous avons présenté le diagramme de cas d'utilisation qui met en clair les fonctionnalités de système, ainsi le diagramme de classe qui donne une vue globale de nos données et les relations entre eux.

Nous pouvons conclure que les modélisations réalisées sont alignées avec les spécifications définies, l'environnement technique sera le sujet du chapitre suivant.

Chapitre 2 : Environnement de développement

Ce chapitre présente l'environnement de développement et la justification de l'implémentation de chaque technologie.







Introduction




Durant ce chapitre, nous allons justifier les raisons pour laquelle nous avons choisi les technologies utilisées, après nous allons présenter les interfaces de notre application.

Une fois les besoins sont définis et analysés, la phase suivante a pour rôle de la concrétisation de ces besoins en procédant au développement et l'implémentation de l'application.

Avant de commencer la réalisation du projet, il est primordial de mettre en évidence les technologies et les outils avec lesquels nous avons entamé cette réalisation.

Pour réaliser ce projet, Initialement, on a décidé d'utiliser Spring Boot pour la partie Backend et React JS pour la partie frontend, suivant cette décision on a utilisé des autres libraires et une base de données Embedded (Embedded Database H2) :

Couche/Utilite	Technologies	Logo
Front End	React Js	
	CSS	
	JS	
Back End	Java	
	Spring	
Base de données	H2	

Outils	Intellij	
	Visual Studio Code	
	Postman	

1. Le langage de développement

On a décidé d'utiliser le langage JAVA, pour la richesse de l'API de cette langue, la grande communauté des développeurs, et les Frameworks puissantes disponibles - Spring dans notre cas.

Notamment, il y a plusieurs langages de développement avec les mêmes avantages, mais j'ai choisi Java et React Js pour me lancer sur un parcours du développeur Full Stack Java/React, cette application m'a offert la chance d'entraîner sur ces technologies et ces concepts.

2. Front END

Parallèlement avec le langage Java, on a besoin des autres langages essentiels pour la construction des sites web. Aujourd'hui, la plupart des sites web professionnels utilisent les technologies React js, Angular, ou Vue js. Pour cela on a décidé d'utiliser la librairie (ou framework) React js.

On a utilisé React Js, et pas un autre Framework FrontEnd, à cause de sa flexibilité et ses performances, grâce au travail sur un DOM virtuel et en ne mettant à jour le rendu dans le navigateur qu'en cas de besoin. Comme les navigateurs sont plutôt lents à réagir face aux changements du DOM, React a l'avantage de limiter les interactions avec ce dernier. Il effectue donc les opérations sur le DOM virtuel et le compare au vrai DOM pour réaliser les changements à effectuer.

Aujourd'hui, rarement on trouve un site web qui ne contient pas des codes CSS et Javascript. Ces langages, nous les avons utilisés dans notre projet, et ci-dessus, vous trouvez une description de ces langages et ses utilités :

CSS, Cascading Style Sheets, c'est un langage informatique qui est utilisé pour formater la mise en page des pages Web. Ils peuvent être utilisés pour définir des styles de texte, des tailles de tableaux et d'autres aspects des pages Web, aide les développeurs Front à créer une présentation du contenu uniforme sur les pages d'un site web.

JavaScript est un langage de programmation de scripts principalement employé dans les pages web interactives mais aussi pour les serveurs. Au contraire du HTML, le Javascript permet un "interactivité" entre le site et l'internaute.

3. Back END

On a utilisé Spring qui facilite le développement d'applications fondées sur Java, Spring boot propose 2 fonctionnalités principale et très intéressantes, l'auto-configuration, les starters, ce qui nous permet de concentrer sur nos cas d'utilisation plutôt que les configurations et la structure basique d'un projet Java.

Ainsi, on a utilisé Lombok pour faciliter le process de développement et éviter de créer des constructeur, getters, setters chaque fois (pas primordial dans notre cas, mais juste pour entrainer avec le plus des technologies possible).

4. Base de données

Notre application est adaptable avec n'importe quel Base de données. Avec Spring, il suffit juste de modifier la configuration sur le fichier *properties* pour changer d'un Base de données à un autre. Pour avoir une application facilement testable, on a utilisé une base de données embedded, comme ça il faut que lancer les deux applications backend et frontend pour utiliser et tester l'application.

H2 est un système de gestion de base de données relationnelles. Il peut être intégré à une application Spring facilement. Les tables peuvent être créées en mémoire vive ou sur disque. Elles peuvent être persistantes ou temporaires. H2 est quasiment sécurise, elle se protège des injections SQL avec l'utilisation d'expressions avec paramètres formels.

5. Outil de développement :

Pour la phase de développement, il faut choisir les outils convenables pour notre projet. Une tranche des tâches redondantes, seront être plus facile avec IntelliJ pour Java, Visual Studio Code pour React, ainsi Postman sera notre compagnon pour tester nos APIs.

IntelliJ IDEA est un environnement de développement intégré (IDE) développé par JetBrains et disponible en plusieurs éditions, on a utilisé l'édition communauté. La génération des classes, la mise à jour des dépendances Maven et tous les tâches liées, sont automatique avec IntelliJ.

Prenant en considération que notre Front End sera un couch React Js, on a décidé d'utiliser Visual Studio Code, avec tous les raccourcis que les plugins VSC offre, c'est qu'une question de quelques touches pour créer un composant React, vérifier la configuration package.json ou parcourir et afficher une liste des objets.

Pour tester nos APIs calls, on peut utiliser un navigateur normal, chrome ou mozilla, mais avec Postman, on peut construire et d'exécuter des requêtes HTTP avec des interfaces graphiques agréable, qui nous propose plus d'informations, et minimiser le temps nécessaire pour tester tous les scénarios possibles avec notre API, afin de traiter les bugs et éviter tout faille de sécurité.

Conclusion

Au cours de ce chapitre nous avons fait le point sur les différentes technologies et les outils de développement et la raison de choisir ces outils.

Chapitre 3 : Application

Ce chapitre présente la description des interfaces de l'application, et une simulation de création de contenu.

Introduction

L'objectif de cette partie est de donner un aperçu sur les interfaces graphiques de l'application. En premier lieu, nous allons présenter les interfaces composant le front office. Après, une simulation de la création du livre et manipulation d'allocation et vente de notre livres.

I. Interface : Home

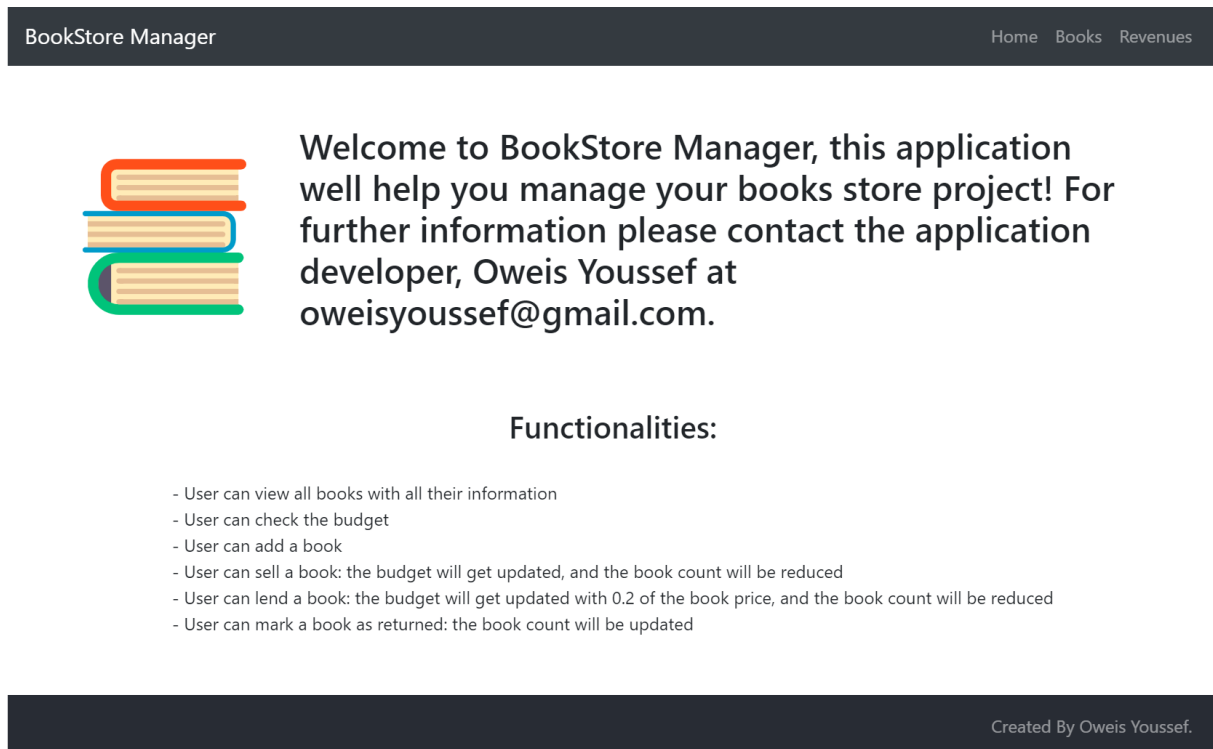


Figure 3 : Interface Home

Cette figure présente la page d'accueil. Elle contient une description des fonctionnalités que l'application offre, et les informations du développeur de l'application.

II. Interface Book

BookStore Manager

HomeBooksRevenues

Add Book

Name

Description

Price

Count

Save

Cancel

Books

Midnight Sun	Midnight Sun When Edward Cullen and Bella Swan met in Twilight, an iconic love story was born. But until now, fans have heard only Bellas side of the story.	170 MAD	-10+	<div>Sell</div>	<div>Lend</div>	<div>Show Lended</div>
Where the Crawdads Sing	Where the Crawdads Sing Where the Crawdads Sing is at once an exquisite ode to the natural world, a heartbreaking coming-of-age story, and a surprising tale of possible murder.	130 MAD	-5+	<div>Sell</div>	<div>Lend</div>	<div>Show Lended</div>
The Vanishing half	The Vanishing Half: A Novel A story of absolute, universal timelessness.	160 MAD	-6+	<div>Sell</div>	<div>Lend</div>	<div>Show Lended</div>

Created By Oweis Youssef.

Figure 4 : Interface Book

Cette figure présente l'interface Book, elle contient deux sections :

- Section 1 : **Add book**, ou on peut ajouter un nouveau livre avec sa quantité
- Section 2 : **Books**, qui affiche les détails des livres, et les boutons pour changer la quantité actuelle, enregistre une vente, allocation ou quand un client retourne un livre.

2.1 Interface Book : Lend

Si on clique sur le button "Lend", un formulaire va être affiché, où on doit ajouter les informations du client :

Books

Midnight Sun	Midnight Sun When Edward Cullen and Bella Swan met in Twilight, an iconic love story was born. But until now, fans have heard only Bellas side of the story.	170 MAD	- 9 +	Sell	Lend	Show Lended
<hr/>						
Name						
<input type="text"/>						
CIN						
<input type="text"/>						
<div>Save Cancel</div>						

Figure 5 : Interface Book - Lend

2.2 Interface Book : Show Lended

Si on clique sur le button "Show Lended", une liste des emprunteurs va être affichée, où on peut signaler le retour d'un livre à notre stock :

Books

Midnight Sun	Midnight Sun When Edward Cullen and Bella Swan met in Twilight, an iconic love story was born. But until now, fans have heard only Bellas side of the story.	170 MAD	<div><div>-</div><div>9</div><div>+</div></div>	<div>Sell</div>	<div>Lend</div>	<div>Show Lended</div>
Where the Crawdads Sing	Where the Crawdads Sing Where the Crawdads Sing is at once an exquisite ode to the natural world, a heartbreaking coming-of-age story, and a surprising tale of possible murder.	130 MAD	<div><div>-</div><div>4</div><div>+</div></div>	<div>Sell</div>	<div>Lend</div>	<div>Show Lended</div>
Name: Oweis Youssef		Cin: L54556	<div>Return Book</div>			
The Vanishing half	The Vanishing Half: A Novel A story of absolute, universal timelessness.	160 MAD	<div><div>-</div><div>5</div><div>+</div></div>	<div>Sell</div>	<div>Lend</div>	<div>Show Lended</div>

Figure 6 : Interface Book - Show Lended

1. Interface Revenue

BookStore Manager

HomeBooksRevenues

Revenues: 262 MAD

RETURNED	Midnight Sun	+34.00 MAD	Client: Mohammed Younssi/L12345	9/11/2020, 4:49:01 PM
LEND	The Vanishing half	+32.00 MAD	Client: Said ahmed/L487687	9/11/2020, 4:48:22 PM
LEND	Where the Crawdads Sing	+26.00 MAD	Client: Oweis Youssef/L54556	9/11/2020, 4:48:09 PM
SELL	Midnight Sun	+170.00 MAD		9/11/2020, 4:47:56 PM

Created By Oweis Youssef.

Figure 7 : Interface Revenue

L'interface Revenue, résume toutes les revenus de notre projet du Vente/Allocation du livres, avec les détails de chaque opération :

- **Sell** signifie la vente d'un livre
- **Lend** signifie location d'un livre
- **Returned** signifie retour d'un livre

Conclusion

Durant ce chapitre, on a présente les interfaces propose par notre application, l'interface Book englobe les fonctionnalités nécessaires pour gérer notre projet de vente, et l'interface Revenues nous permettons de suivre toutes les opérations réalisées.

Conclusion générale

Durant ce projet, on a essayé d'élaborer une solution informatique qui respect les exigences professionnels, et qui répondre au besoin d'un magasin des livres, et assiste la croissance de ce dernier.

Cette expérience, m'a permis d'apprendre plusieurs technologies, et enrichir mon profil comme étant un développeur informatique, et mes compétences de gestion et modélisation du projet informatique.

En perspective pour ce projet, on compte d'ajouter des autres fonctionnalités, et le partager comme une solution open source pour aider les entrepreneurs à créer des magasins des livres.

Références

Bibliographie

Kathy Sierra, Bert Bates, **Head First Java**, 2015 O'Reilly Media

Webographie

React Js documentation : <https://reactjs.org/tutorial/tutorial.html>

Java 8 documentation : <https://docs.oracle.com/javase/8/docs/>

Spring Boot documentation : <https://spring.io/guides>