

# DOSSIER DE PROJET

Présenté en vue de l'obtention de  
la certification professionnelle  
**"Développeur Web et Web Mobile"**  
par  
Helder MORAIS

CONCEPTION ET DÉVELOPPEMENT  
D'UN LOGICIEL DE GESTION DE LIBRAIRIE  
ET DE PETITE RESTAURATION

Structure d'accueil :



le **MIGOU** BD café

Année 2018 / 2019

## Remerciements

Je souhaite, par le biais de ces quelques lignes de remerciements, exprimer ma gratitude envers toutes les personnes qui m'ont apportées leur soutien et leur encouragement.

Je tiens à remercier tout particulièrement **Aurélien CHIROT**, mon référent technique, pour sa disponibilité, ses conseils et son implication malgré son emploi du temps surchargé.

Je ne peux que remercier profondément **Sébastien CARRÈRE** et **Alexandre RUIZ DELVAL** pour leur accueil au sein de leur entreprise, leur patience et leur confiance envers mon travail. Vous avez tout mis en œuvre pour rendre mon stage possible et pour qu'il se passe dans les meilleures conditions.

Un grand merci à tous mes camarades de classe pour leur écoute, leur bienveillance et leur aide face à mes difficultés.

Mes remerciements s'adressent aussi à tout le personnel de **SIMPLON** qui m'ont soutenu et fait confiance lors des sélections d'entrée en formation.

# Table des matières

<b>1 Compétences couvertes par le projet .....</b>	<b>12</b>
1.1 Front-end .....	12
1.2 Back-end .....	12
<b>2 Contexte général du projet .....</b>	<b>34</b>
2.1 Organisme d'accueil .....	34
2.1.1 Présentation .....	43
2.1.2 Activités .....	22
2.2 Introduction à l'analyse de l'existant .....	2
2.2.1 Étude de l'existant .....	1
2.2.2 Axes d'améliorations .....	10
<b>3 Cahier des charges et restitution des besoins .....</b>	<b>10</b>
3.1 Cahier des charges .....	12
3.2 Identification des acteurs .....	12
3.3 Restitution des besoins .....	22
3.3.1 Users Stories .....	22
3.3.2 Diagramme d'état .....	22
<b>4 Solution proposé et conception .....</b>	<b>22</b>
4.1 Logiciels, outils et langages utilisés .....	22
4.1.1 Modèle Vue Contrôleur .....	2
4.2 Méthode agile .....	22
4.3 Bases de données .....	22
4.3.1 Modèle Conceptuel de Données .....	22
4.4 Maquettes / Charte graphique .....	22
4.5 Conceptions .....	22
4.5.1 Authentification des utilisateurs .....	33
4.5.2 Création d'un nouvel utilisateur .....	33
4.5.3 Modification / suppression d'un utilisateur .....	3
4.5.4 Vente librairie .....	33
4.5.5 Vente restauration .....	33
4.5.6 Gestion des fiches clients .....	32
4.5.7 Gestion des fiches articles .....	22
4.5.8 Gestion des commandes .....	11

4.5.9 Consulter le stock .....	12
4.5.10 Tableau de bord .....	12
4.6 Présentation du jeu d'essai .....	34
<b>5 Veille sur les vulnérabilités de sécurité .....</b>	<b>22</b>
<b>6 Situation de travail et site anglophone .....</b>	<b>2</b>
<b>8 Conclusion .....</b>	<b>10</b>
<b>9 Annexes .....</b>	<b>10</b>
9.1 Diagramme d'état .....	12
9.2 Maquettes réalisées en début de projet .....	12
9.3 Site-Map .....	12
9.4 Extraits de mon code .....	12

---

## Chapitre 1

---

### Compétences couvertes par le projet

#### Plan

1.1 Front-end .....	12
1.2 Back-end .....	12



## 1.1 FRONT-END



Maquetter une application

Réaliser une interface utilisateur web statique et adaptable

Développer une interface utilisateur web dynamique

Réaliser une interface utilisateur avec une solution de gestion de contenu ( ou e-commerce )

Développer les composants d'accès aux données

Créer une base données

Développer la partie back-end d'une application web ( ou web mobile )

Élaborer et mettre en œuvre des composants dans une application de gestion de contenu ( ou e-commerce )

## 1.2 BACK-END



## Chapitre 2

### Contexte général du projet

#### Plan

2.1 Organisme d'accueil .....	12
2.1.1 Présentation .....	12
2.1.2 Activités .....	
2.2 Introduction à l'analyse de l'existant .....	12
2.2.1 Étude de l'existant .....	2
2.2.2 Axes d'améliorations .....	10

## 2.1 Organisme d'accueil



Le Migou

### 2.1.1 Présentation

Crée en 2005 par deux passionnés de bandes dessinées, **Le Migou** est la seule librairie spécialisée BD dans le Gers.

Dans ce magasin atypique, il est difficile de ne pas trouver son bonheur quand on est fan de bulles. Manga, BD, comics, jeunesse, roman graphique, alternatif, .... peu importe le titre que l'on cherche, **Sébastien CARRÈRE** et **Alexandre RUIZ DELVAL** l'on certainement.

La librairie propose également aux clients un espace restauration avec un plat du jour qui est cuisiné tous les matins par Alexandre.

En plus des services proposés aux particuliers, Le Migou travaille en étroite collaboration avec des organismes tel que les mairies, bibliothèques, centres culturels, CDI (Centre de Documentation et d'Information), salons spécialisés, ....

### 2.1.2 Activités

Les activités de la librairie sont les suivantes :

- Vente en librairie
- Vente par correspondance
- Vente aux collectivités
- Restauration tous les midis
- Café, thé et espace lecture toute la journée
- Animations en librairie ( dédicaces d'auteurs, concerts, spectacles vivant et expositions )

## 2.2 Introduction à l'analyse de l'existant

Dans cette partie, je vais vous exposer une analyse de l'existant et faire apparaître les points sur lesquels une remédiation est possible en vue de l'utilisation qui en est faite par la librairie.

### 2.2.1 Étude de l'existant

Actuellement, Le Migou travaille avec le logiciel " Ellipses ".

Ce logiciel de gestion a été conçu pour apporter des solutions aux libraires qui ont besoins d'un outil qui puisse répondre aux problématiques spécifiques à leur métier.

Parmi les nombreuses fonctionnalités de Ellipses, on peut en énumérer les plus essentielles :

- les ventes
- les commandes clients et fournisseurs
- la gestion du stock
- la facturation et la gestion financière
- gestion des fichiers clients
- ....

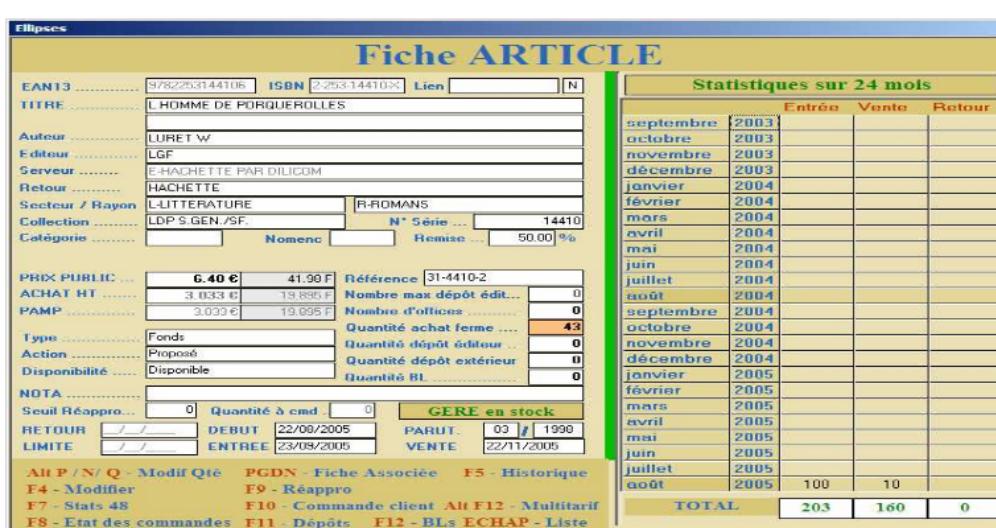
### Exemples de captures d'écrans du logiciel Ellipses :



Page d'accueil



Page vente caisse



Page fiche article

En plus des fonctionnalités de Ellipses, Le Migou à également souscrit un abonnement avec la société " **DILICOM** " qui est un service interprofessionnel destiné à faciliter le développement des échanges de données informatisés ( EDI ) dans le secteur commercial du livre.

### Mais comment fonctionne DILICOM ?

Pour commencer, les commandes sont saisies par les libraires sur leur système informatique, quel qu'il soit, puis émises vers le serveur au moment de leur choix, et immédiatement déposées dans les boîtes aux lettres de chacun des distributeurs concernés. Elles seront ensuite lues par ceux-ci, et injectées dans leur propre ordinateur, de façon que les bordereaux de préparation ou les factures soient automatiquement préparés, sans nouvelle saisie manuelle.

DILICOM est aussi en charge des catalogues informatisés.

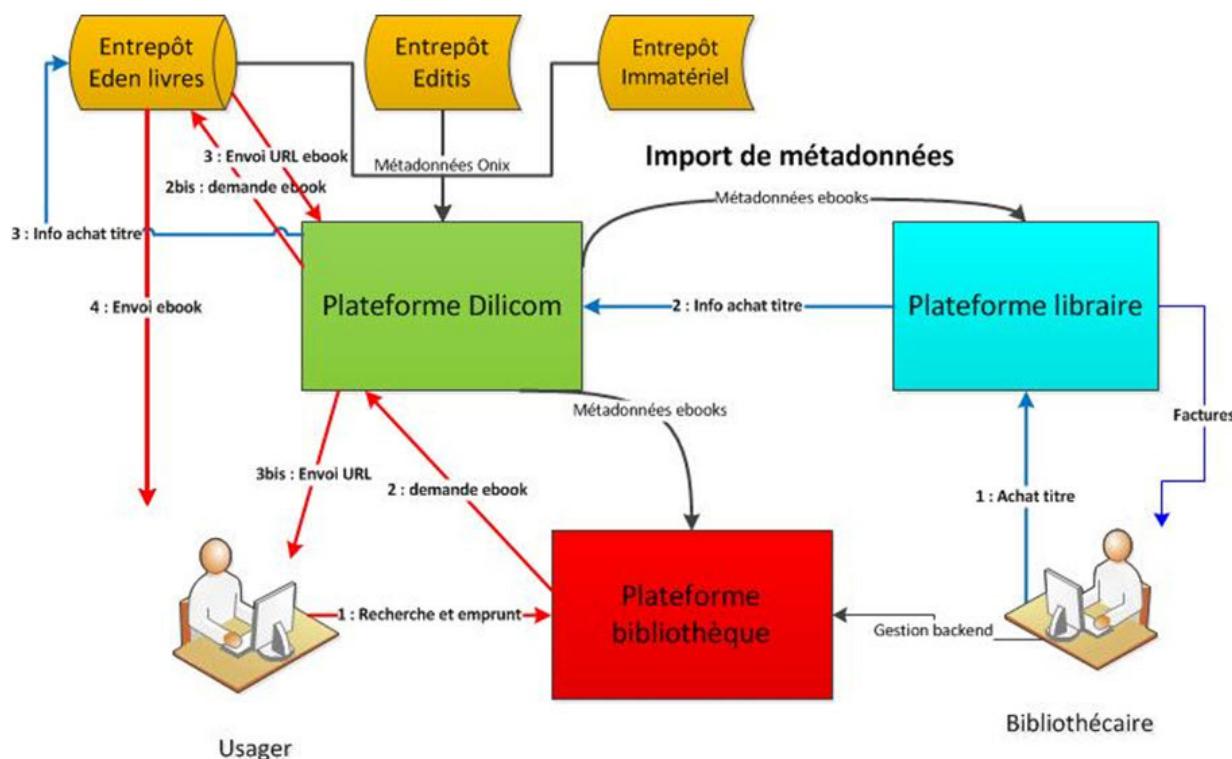
Des centaines de milliers de titres sont disponibles en France, et, chaque jour, des milliers de mises à jour sont nécessaires. Compte tenu du caractère très hétérogène de l'édition et de la distribution, il est difficile pour un libraire d'être certain de mettre à jour correctement ses fichiers.

DILICOM rassemble puis diffuse donc des catalogues électroniques, à vocation commerciale : cela permet aux libraires de bien gérer leurs commandes.

Le fichier de DILICOM, dénommé FEL pour Fichier Exhaustif du Livre, est mis à jour quotidiennement par les distributeurs, directement à partir de leurs systèmes informatiques, puis aussitôt diffusé vers les abonnés libraires.

DILICOM gère aussi les avis d'expédition. Il s'agit des copies électroniques du bon de livraison : ouvrages fournis ou manquants, prix, remise. Ces avis permettent aux libraires de préparer la réception des colis.

### Exemple de fonctionnement de Dilicom lors de l'achat d'un livre numérique



Concernant la partie **Restauration**, Le Migou travaille avec **LIBRE OFFICE** et un simple tableur qui leur permet de gérer la gestion financière du restaurant et du café.

Ce tableur ne leur permet pas de gérer les achats, commandes et la gestion du stock alimentaire.

### 2.2.2 Axes d'améliorations

C'est en écoutant et en discutant avec les responsables du Migou que l'on peut se rendre compte de quelques points négatifs concernant leurs installations.

Ci dessous, les points essentiels qui exigent une amélioration.

#### **Logiciel Ellipses :**

- logiciel non adapté au secteur de la BD
- logiciel beaucoup trop "évolué" et avec des fonctionnalités inutiles pour une petite librairie.
- logiciel pas assez intuitif.
- interface graphique désagréable et complexe
- pas de visuel couverture pour voir rapidement les éditions sorties
- pas de visuel rapide pour voir si le titre est encore disponible chez l'éditeur
- prise en main parfois compliquée
- recherche bibliographique très compliquées et pas du tout intuitive
- titre du livre souvent tronqué lors de la recherche historique client
- mode de gestion des retours fastidieux et manquant de souplesse

#### **Dilicom :**

- pas de critique

#### **Libre office :**

- c'est un simple tableur qui gère les entrées financières concernant la restauration et les boissons
- il n'y pas de base de données
- pas de possibilités pour faire des statistiques de ventes

## Chapitre 3

# Cahier des charges et restitution des besoins

### Plan

Introduction .....	12
3.1 Cahier des charges .....	43
3.2 Identification des acteurs .....	12
3.3 Restitution des besoins .....	2
3.3.1 Users stories .....	10
3.3.2 Diagramme d'état .....	2

## Introduction

Dans ce chapitre, je vais vous présenter le cahier des charges qui m'a été transmis au démarrage du projet. Je vais, par la suite, identifier quels seront les utilisateurs du logiciel et lister leurs besoins.

### 3.1 Cahier des charges

Voici donc le cahier des charges qui m'a été transmis par le client :



# CAHIER DES CHARGES

*logiciel de gestion de bd*

Présentation de l'entreprise

LE MIGOU est une librairie spécialisée dans la vente de bandes-déssinées neuves et d'occasions. Nous avons également un service de restauration qui est ouvert seulement le midi.

Les besoins

Actuellement, nous avons un logiciel de gestion qui ne nous convient plus car il n'est pas intuitif, agréable à utiliser et comporte de nombreuses fonctionnalités inutiles ou non fonctionnelles. De plus, il nous manque une base de données pour pouvoir gérer nos BD d'occasions et une autre pour pouvoir gérer la vente de la restauration. Nous avons donc besoin d'un logiciel avec lequel nous pourrons :  
-> Interagir avec notre base de données existante pour gérer le stock de BD neuves  
-> Interagir avec une nouvelle base de données pour gérer le stock de BD d'occasions  
-> Interagir avec une nouvelle base de données pour gérer la vente des produits du restaurant  
-> Pouvoir avoir des statistiques de ventes concernant ces trois points.

Nos attentes

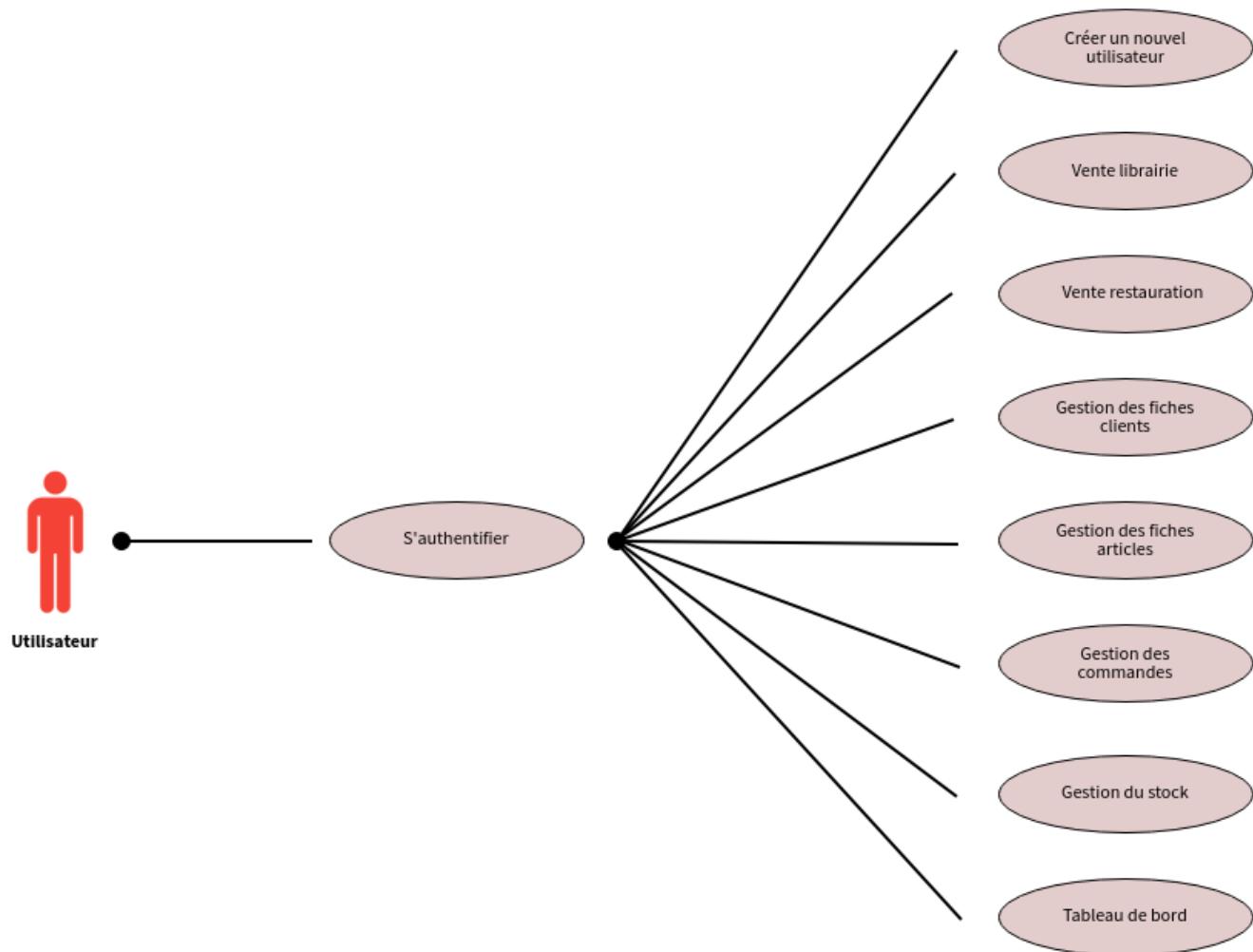
Nous souhaitons avoir un logiciel entièrement en français, qui soit simple d'utilisation, agréable à regarder et avec un graphique et une interface sobre. Nous attendons de la part du développeur qu'il nous fasse une proposition graphique et des fonctionnalités attendues. Nous lui laissons carte blanche sur la première proposition et discuterons ensemble sur les points qu'il nous semble à améliorer.

On ne peut que se rendre compte que le cahier des charges reste très succinct en rapport à l'ampleur du projet. C'est donc en faisant une synthèse entre les discussions avec le client et les éléments qui m'ont été fournis que j'ai pu analyser et comprendre les besoins.

### 3.2 Identification des acteurs

En ce qui concerne les acteurs, le client souhaite qu'ils puissent tous avoir les mêmes priviléges. En d'autres termes, tout le personnel pourra avoir accès à toutes les données et fonctionnalités du logiciel dès lors que l'utilisateur est authentifié.

**Diagramme de cas d'utilisation global**



### 3.3 Restitution des besoins

Toujours en me basant sur la synthèse entre les discussions avec le client et les éléments qui m'ont été fournis, j'en ai déduit que le Migou avait besoin d'un logiciel simplifié et adapté aux activités de ventes librairie spécialisée BD et de pouvoir également enregistrer, dans une base de données, les ventes restauration.

Ci-dessous, un listing énumérant les besoins détectés en début de projet.

#### Restitution des besoins

- Pouvoir gérer les ventes librairie et consulter des statistiques sur celles-ci
- Pouvoir gérer les ventes restauration et consulter des statistiques sur celles-ci
- Pouvoir consulter le stock librairie ( articles d'occasions et neufs )
- Pouvoir consulter le FEL ( Fichier Exhaustif du Livre ) avec DILICOM
- Pouvoir interagir avec la base de données librairie ( ajouter, modifier, supprimer )
- Pouvoir interagir avec la base de données restaurant ( ajouter, modifier, supprimer )
- Pouvoir interagir avec les comptes clients ( ajouter, modifier, supprimer )
- Pouvoir passer les commandes auprès des fournisseurs
- Pouvoir gérer les retours fournisseurs et consulter des statistiques sur ceux-ci

#### 3.3.1 Users Stories

Dans la page suivante, vous trouverez les Users stories que j'ai transmis au client en début de projet.

Au fur et à mesure de nos réunions, et comme nous travaillions en adoptant la méthode agile, ils ont évolués en prenant compte des souhaits du client et des améliorations apportés aux fonctionnalités à développer.

## Users Stories

<b>Je suis :</b>	<b>Je veux :</b>	<b>Afin de :</b>
Utilisateur	Accéder au stock du magasin	Pouvoir le gérer, consulter et avoir des informations sur les quantités
Utilisateur	Avoir un logiciel de vente pour mes produits en librairie et restauration	Pouvoir enregistrer mes ventes et avoir des statistiques sur celles-ci
Utilisateur	Avoir des comptes client	<ul style="list-style-type: none"> <li>- créer, modifier, supprimer un compte</li> <li>- voir l'historique d'achat d'un client ( par titre et par tome )</li> <li>- faire automatiquement un calcul dès la fin du dixième achat et faire un avoir de 5% sur la totalité des 10 achats.</li> <li>- pouvoir gérer les retours clients et leurs commandes</li> <li>- pouvoir envoyer automatiquement un mail au client au moment où sa commande est arrivée</li> </ul>
Utilisateur	Accéder à ma base de données	Pouvoir rechercher un produit par titre, auteur et maison d'édition
Utilisateur	Interagir avec mes fournisseurs	<ul style="list-style-type: none"> <li>- Pouvoir passer les commandes et gérer les retours produits auprès des fournisseurs</li> <li>- Pouvoir visualiser les ventes une par une et par titre et que l'on me propose automatiquement de les recommander</li> <li>- Pouvoir passer des commandes par envoi de mails automatiques vers les différents fournisseurs</li> <li>- Pouvoir consulter le taux de retour produit par éditeur ( en pourcentage d'achats )</li> <li>- Pouvoir afficher la date de parution d'un produit durant la phase " retour fournisseur "</li> <li>- Pouvoir supprimer ou modifier les lignes " retour fournisseur " avant la validation</li> </ul>
Utilisateur	Interagir avec mes fiches produits restaurant	Pouvoir créer, modifier ou supprimer une fiche produit restaurant

### 3.3.2 Diagramme d'état

En Annexes, vous trouverais le diagramme d'état qui m'a servis pour la conception du logiciel. Je l'ai conçus après l'analyse de la restitution des besoins et des Users Stories.

## Chapitre 4

# Solution proposée et conceptions

### Plan

4.1 Logiciels, outils et langages utilisés .....	12
4.1.1 Modèle Vue Contrôleur .....	12
4.2 Méthode agile .....	12
4.3 Bases de données .....	10
4.3.1 Modèle Conceptuel de Données .....	22
4.4 Maquettes / Charte graphique .....	22
4.5 Conceptions .....	22
4.5.1 Authentification des utilisateurs .....	2
4.5.2 Création d'un nouvel utilisateur .....	2
4.5.3 Modification / suppression d'un utilisateur .....	2
4.5.4 Vente librairie .....	2
4.5.5 Vente restauration .....	2
4.5.6 Gestion des fiches clients .....	1
4.5.7 Gestion des fiches articles .....	1
4.5.8 Gestion des commandes .....	1
4.5.9 Consulter le stock .....	1
4.5.10 Tableau de bord .....	1
4.6 Présentation du jeu d'essai .....	1

#### 4.1 Logiciels, outils et langages utilisés



GitHub est un service en ligne qui permet d'héberger ses repositories de code. Il m'a permis d'avoir un backup de mon code sur un serveur externe et, ainsi, avoir sous la main la dernière version de mes commits dans le cas où, par exemple, ma machine tombe en panne.



Trello est une application permettant de gérer des projets et tâches en équipe ou individuellement.

Je l'ai utilisé pour planifier et organiser l'ensemble de mon projet.



Basé sur la méthode Merise et très simple d'utilisation, JMerise est un logiciel qui m'a aidé et facilité le travail de modélisation et de création de mes bases de données.



MySQL est un système de base de données, souvent couplé au langage PHP. Ce système utilise le langage SQL (Structured Query Language, langage structuré de requêtes) ce qui m'a permis d'interroger, alimenter ou mettre à jour mes bases de données.



MockFlow est un logiciel en ligne qui m'a permis de réaliser les maquettes, les différents diagrammes et le site-map du projet.



Visual Studio Code est un éditeur de code source rapide et performant.



Laravel est un framework novateur et complet qui utilise les possibilités les plus récentes de PHP.

Il adopte une architecture MVC ( Modèle-Vue-Contrôleur ) qui permet d'organiser une interface graphique d'un programme.

Dans l'architecture MVC, les rôles des trois entités sont les suivants :

- modèle : données (accès et mise à jour)
- vue : interface utilisateur (entrées et sorties)
- contrôleur : gestion des événements et synchronisation



Bootstrap est un framework destiné à faciliter la création d'applications web. Il regroupe une collection d'outils fournis sous la forme de classes CSS et de librairies Javascript.



Javascript est un langage de script léger et orienté objet. Il consiste à créer des petits scripts sur une page HTML dans le but d'ajouter une petite animation ou un effet particulier sur la page. Cela permet en général d'améliorer l'ergonomie ou l'interface utilisateur.



Le framework jQuery est une bibliothèque logicielle, rapide et concise, conçue pour les programmeurs JavaScript. Elle facilite le développement rapide d'applications web et elle simplifie de nombreuses tâches en réduisant la quantité de code que nous avons besoin d'écrire.



PHP, (Personal Home Page) est un langage de programmation complet, simple à mettre en œuvre, documenté, sécurisé et fiable, permettant la création de pages Web dynamique.



Ajax représente un ensemble de technologies destinées à réaliser de rapides mises à jour du contenu d'une page Web, sans qu'elles nécessitent le moindre rechargeement visible par l'utilisateur de la page Web.



HTML ( Hypertext Markup Language ) est un langage de structuration des pages Web.



CSS3 ( Cascading Style Sheets : feuilles de style en cascade ) est un langage destiné à la présentation des documents HTML et XML .

#### 4.1.1 Modèle Vue Contrôleur

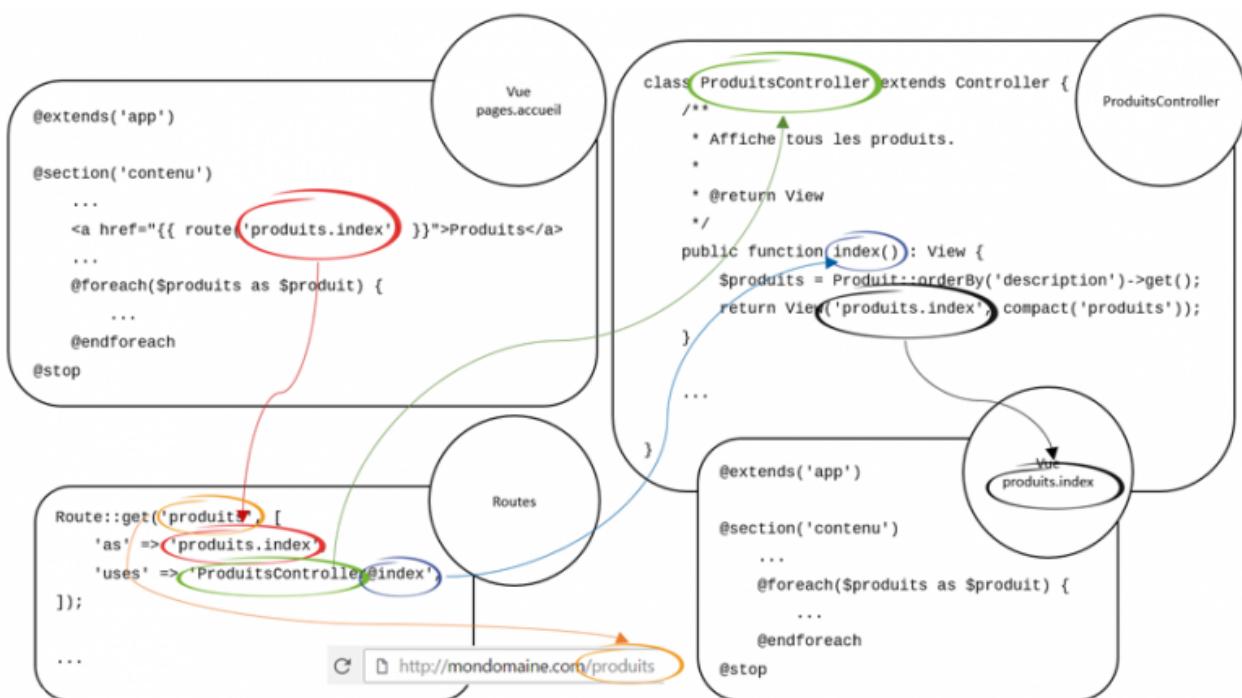
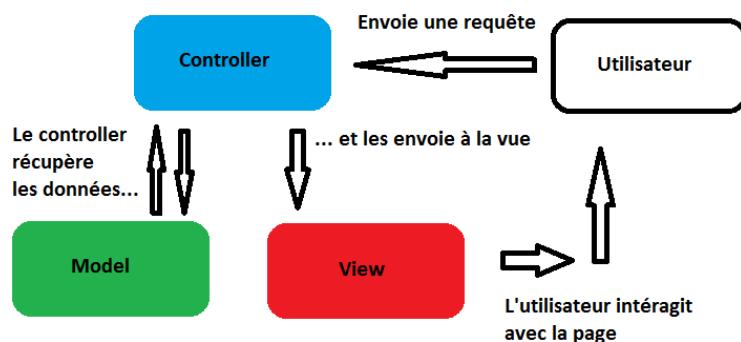
Pour réaliser le logiciel, j'ai choisi d'utiliser Laravel qui adopte une architecture MVC. Le modèle MVC se base sur la séparation des différentes couches d'une application Web en trois parties à savoir les données, les traitements et la présentation.

Ainsi, le logiciel se retrouve segmenté en trois composants essentiels qui sont :

- Le modèle : correspond à une table d'une base de données. C'est une classe qui étend la classe Model qui permet une gestion simple et efficace des manipulations de données et l'établissement automatisé de relations entre tables.

- La vue : correspond à l'interface avec laquelle l'utilisateur interagit. Elle est soit un simple fichier avec du code HTML, soit un fichier utilisant le système de template Blade de Laravel.

- Le contrôleur : Il prend en charge la gestion des événements de synchronisation pour mettre à jour la vue ou le modèle et les synchroniser. Il reçoit tous les événements de l'utilisateur et enclenche les actions à effectuer.



## 4.2 Méthode agile

Soucieux de voir les responsables du Migou satisfaits de mon travail durant le stage , je leur ai présenté et expliqué la méthodologie agile. Prenant conscience de l'intérêt global de cette méthode, ils ont accepté de coopérer et ce, malgré le temps qu'il a fallu consacrer aux nombreuses réponses à mes questions.

La méthodologie Agile m'a donc permis d'intégrer les besoins du client au centre des priorités du projet.

## 4.3 Base de données

Conscient de l'importance d'une BDD homogène et conforme aux attentes des utilisateurs, j'ai apporté une attention toute particulière au " cœur " de mon programme.

Tout d'abord, j'ai ré-analysé les users-Stories et la restitution des besoins, rediscuté avec les acteurs du logiciel, fait des schémas et brouillons afin de bien comprendre les éléments qui devaient être informatisés. Cela m'a également permis de bien comprendre quelles étaient les relations et les dépendances entre les différents acteurs ( clients, commandes, articles, fournisseur, ... )

Par la suite, j'ai opté pour la méthode Merise pour la modélisation de ma BDD. J'ai pu, ainsi, bien organiser les données analysées auparavant et de les transcrire informatiquement en flux de données et de traitements.

Une fois le travail terminé sur jMerise, j'ai exporté le SQL de ma base de données vers phpMyAdmin afin de pouvoir travailler en local sur ma machine.

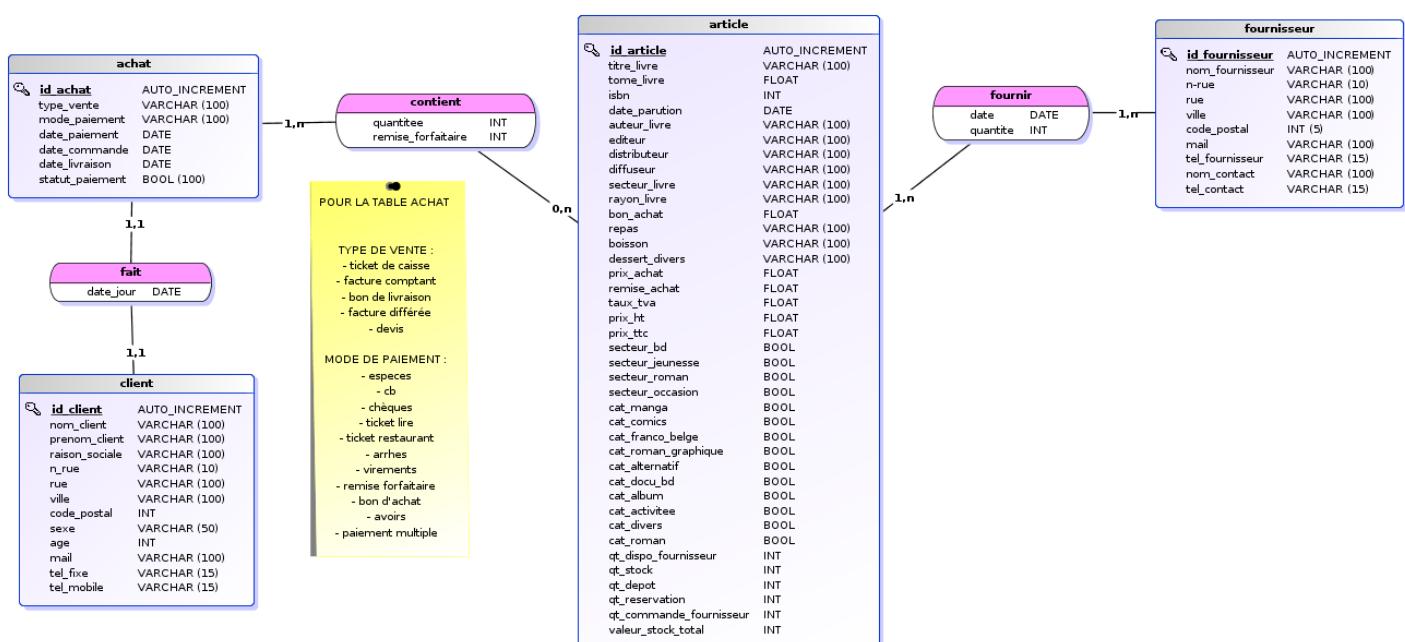
### 4.3.1 Modèle Conceptuel de Données

Le Modèle Conceptuel de Données ( MCD ) m'a permis d'avoir une représentation graphique simple à comprendre et avec les différents éléments qui sont liés entre eux à l'aide de diagrammes codifiés.

On peut donc y voir :

- Les entités ( 1 rectangle = 1 objet )
- Les propriétés ( la liste des données de l'entité )
- Les relations qui expliquent et précisent comment les entités sont reliées entre elles
- Les cardinalités ( 0:1 , 1:1 , 1:N , .... )

Ci-dessous, une photo des entités en cours de création :



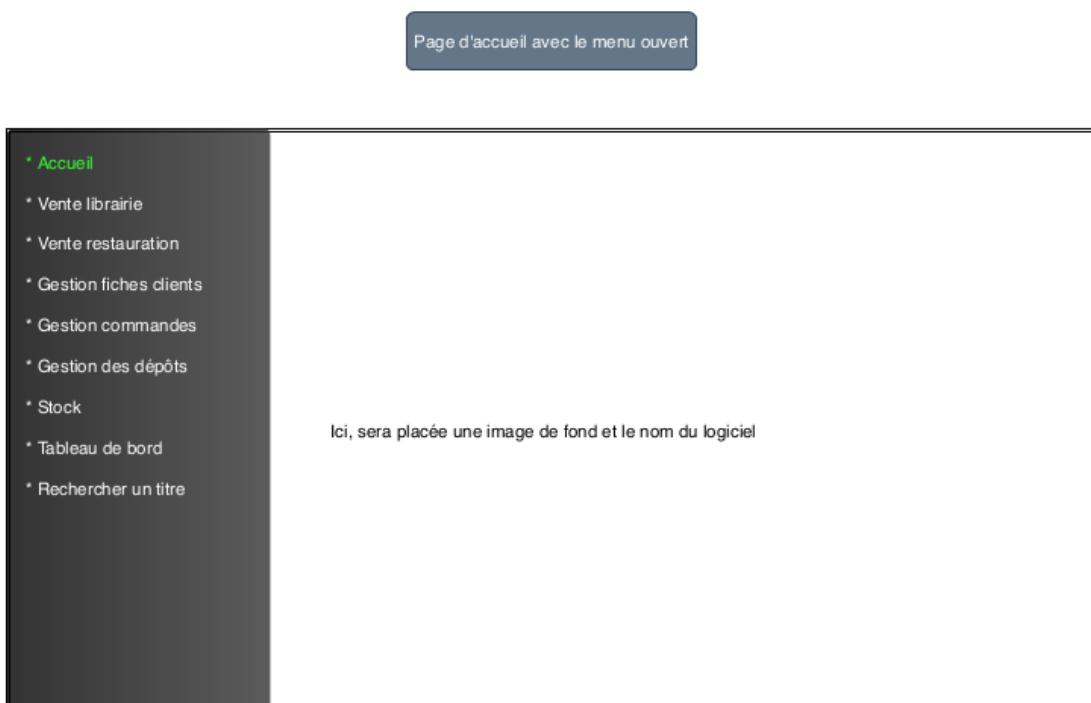
#### 4.4 Maquettes / Charte graphique

Même en sachant que le client m'avait donné " carte blanche " concernant la charte graphique et les maquettes, j'ai tout de même préféré étudié le sujet avec lui.

J'adore la méthode agile .... !!! Elle m'a permis de gagner beaucoup de temps et mieux comprendre les souhaits du client concernant la charte graphique.

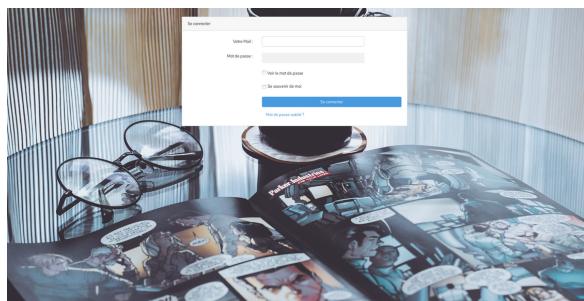
Ci-dessous, une photo de maquette qui à permis de suivre une charte tout au long du projet. Bien évidemment, il y a eu une évolution et des changements au fur et à mesure que le client me transmettait ses souhaits.

En annexes, vous trouverez les maquettes réalisés en début de projet.



## 4.5 Conceptions

### 4.5.1 Authentification des utilisateurs



Formulaire pour s'authentifier

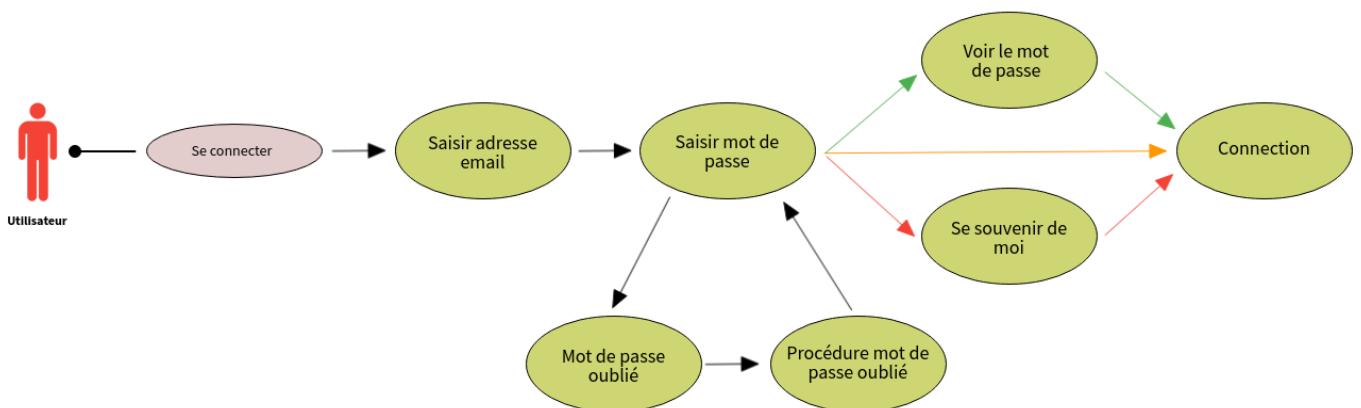
Pour pouvoir accéder à l'ensemble des fonctionnalités du logiciel, un utilisateur doit pouvoir s'authentifier avec son adresse mail et son mot de passe. Si ce n'est pas le cas, il reste bloqué sur la page d'accueil avec comme seule possibilité, se connecter.

Le client souhaite que tous les utilisateurs puissent avoir les mêmes privilèges et donc, avoir accès à toutes les fonctionnalités du logiciel. Dans ce cas là, on a pas besoin de différencier, par exemple, un administrateur d'un simple utilisateur.

Laravel nous propose un système intégré d'authentification des utilisateurs qui évite de longues heures de développement. Pour activer l'authentification, il suffit d'utiliser artisan avec " php artisan make:auth " et " php artisan migrate ".

Grace à ces deux commandes, les fonctions d'authentification de Laravel vont créer tout ce qui est nécessaires au bon fonctionnement de l'authentification des utilisateurs.

**Diagramme de cas d'utilisation " s'authentifier " :**

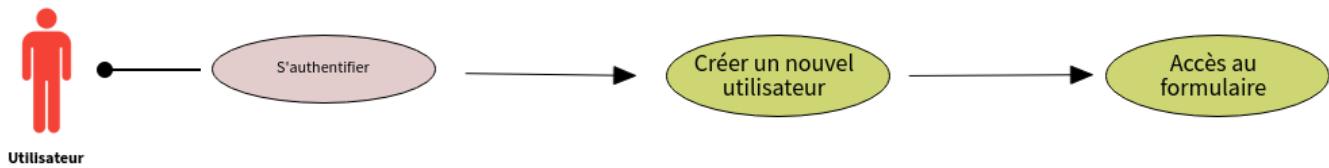


### 4.5.2 Crée un nouvel utilisateur

Formulaire création nouvel utilisateur

Après authentification, un utilisateur pourra accéder au formulaire de création d'un nouvel utilisateur.

### Diagramme de cas d'utilisation " créer un nouvel utilisateur " :



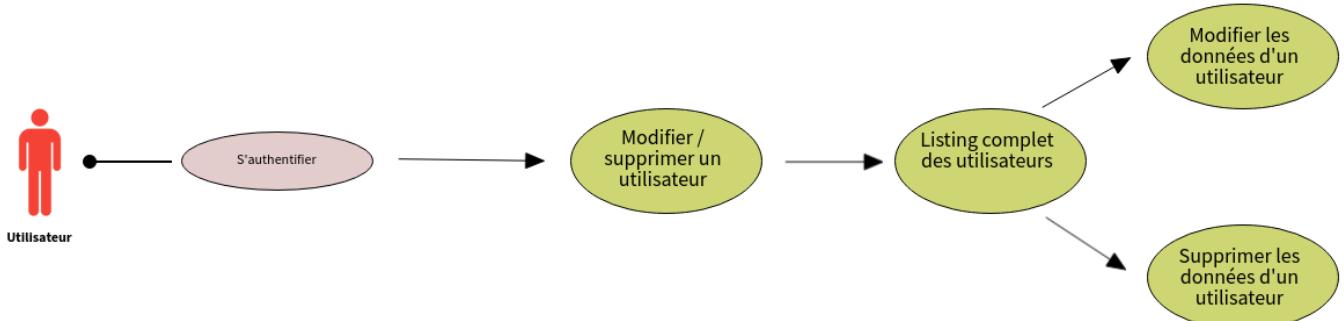
### 4.5.3 Modifier ou supprimer un utilisateur

Nom du client											
id.	Date de création	Nom	Prénom	Raison sociale	Adresse	age	sexé	mail	tel.fixe	tel.mobile	
1	2019-02-20 21:31:03	morais	helder	*HelDev*	9 rue boissy d'anglas 32000 Auch	45	male	morais.helder@hotmail.fr		06 15 77 07 42	
2	2019-02-20 21:31:56	lolo	lapinou	y a pas	12 rue des fleurs 56789 new villa	32	female	lala@hto.fr		1234567890	

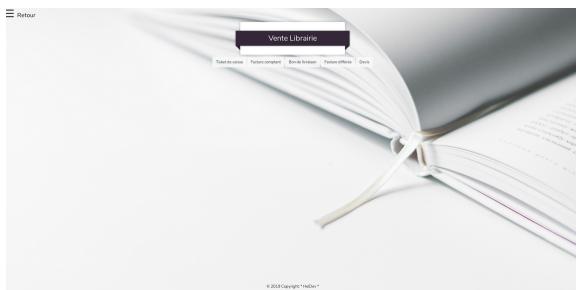
Listing et recherche client par nom

Après authentification, un utilisateur pourra modifier ou supprimer les données d'un utilisateur. L'utilisateur pourra retrouver un client grâce à une barre de recherche qui se trouvera au dessus du tableau de " listing client ". Une fois validée, une suppression sera définitive.

### Diagramme de cas d'utilisation " Modifier ou supprimer un utilisateur " :



#### 4.5.4 Vente librairie



Page d'accueil

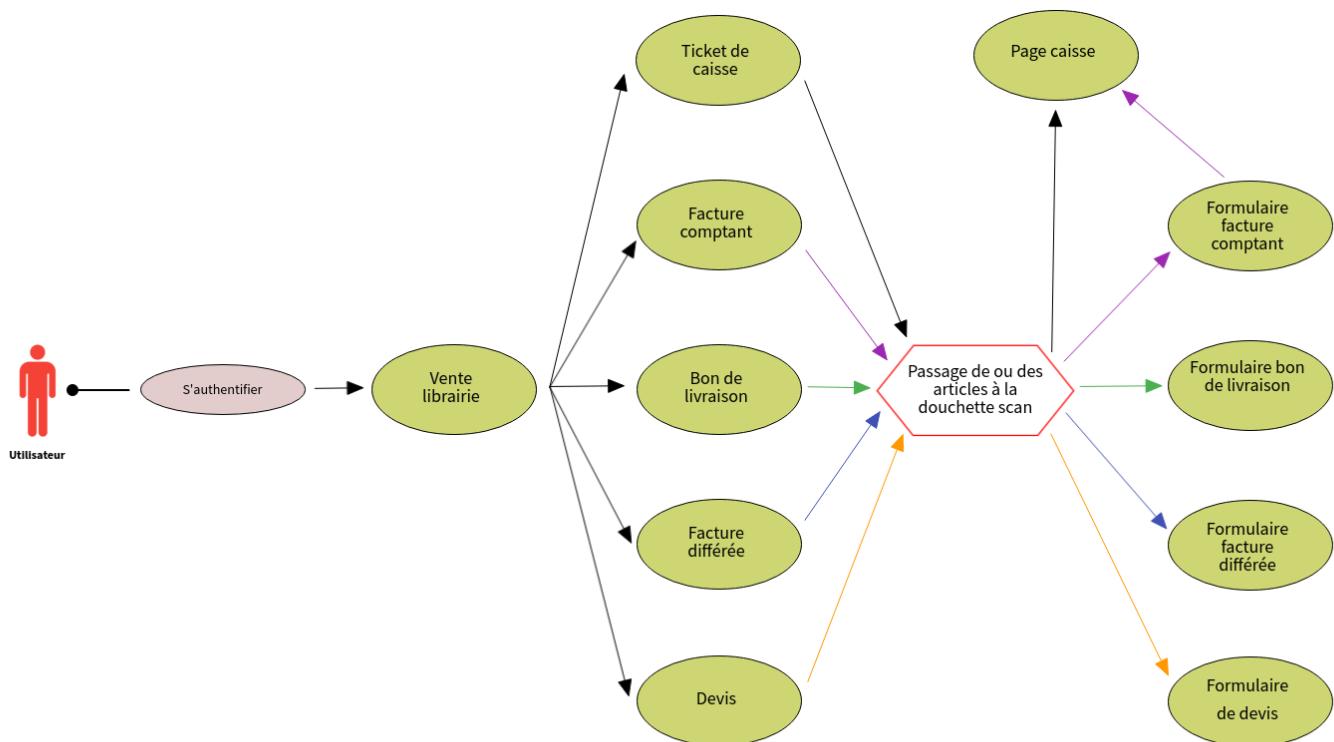
Après authentification, un utilisateur pourra faire une vente librairie.

Pour effectuer une vente, l'utilisateur devra se servir de la douchette pour entrée les données et les caractéristiques de ou des articles choisis par le client.

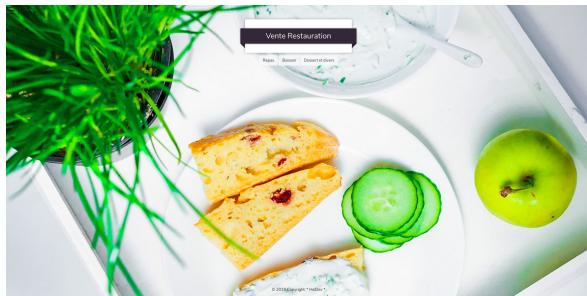
Il aura le choix entre plusieurs types de ventes :

- les ventes "**ticket de caisse**" qui sont de simple ventes avec, comme preuve d'achat, un ticket de caisse.
- les ventes "**facture comptant**" qui sont les mêmes que pour le ticket de caisse mais avec, comme preuve d'achat, une facture éditée au nom du client.
- les "**bon de livraison**" sont des formulaires édités par la librairie et qui servent de preuve lors de la livraison d'une commande ou d'un retour vers les fournisseurs. Ils sont accompagnés d'une facture différée dans le cas d'une commande client ou d'une commande d'un organisme partenaire de la librairie.
- les "**factures différées**" sont éditées par la librairie et transmises généralement aux organismes partenaires lors de la réception de la commande ou de l'achat direct en librairie.
- les "**devis**" qui servent surtout pour les organismes partenaires pour leur permettre de faire un appel à cotisation aux structures concernées.

**Diagramme de cas d'utilisation " Vente librairie " :**



### 4.5.5 Vente restauration



Page d'accueil

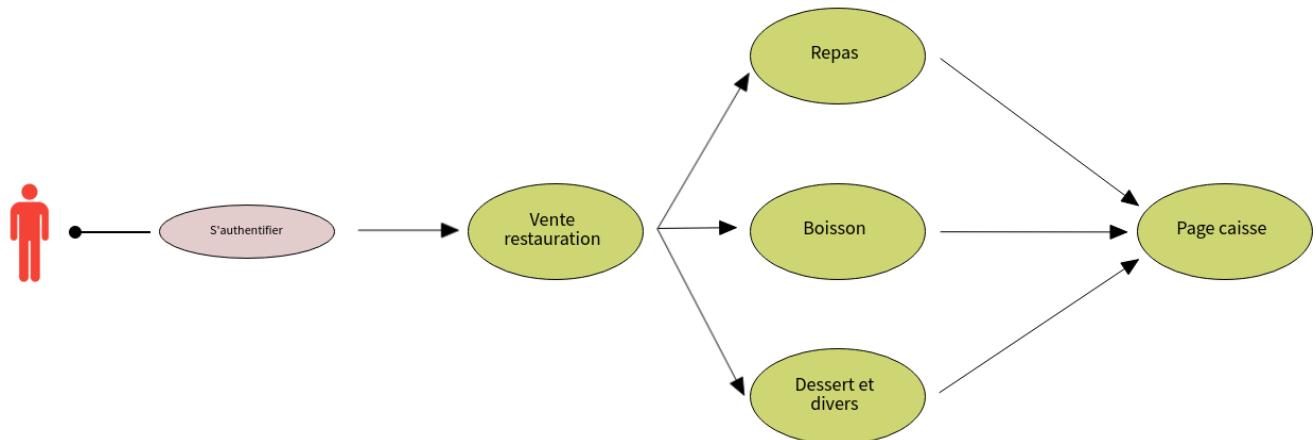
Capture d'écran page Caisse

Après authentification, un utilisateur pourra faire une vente restauration.

Les ventes restauration sont composées de seulement trois types :

- les repas ( assiettes, menus complet, tartines, salades)
- les boissons ( quand elles sont servies en dehors des heures d'ouverture du restaurant )
- les desserts et divers ( comme pour les ventes boissons, ce type de vente est à utiliser en dehors des heures d'ouverture du restaurant )

**Diagramme de cas d'utilisation " Vente restauration " :**



### 4.5.6 Gestion des fiches clients



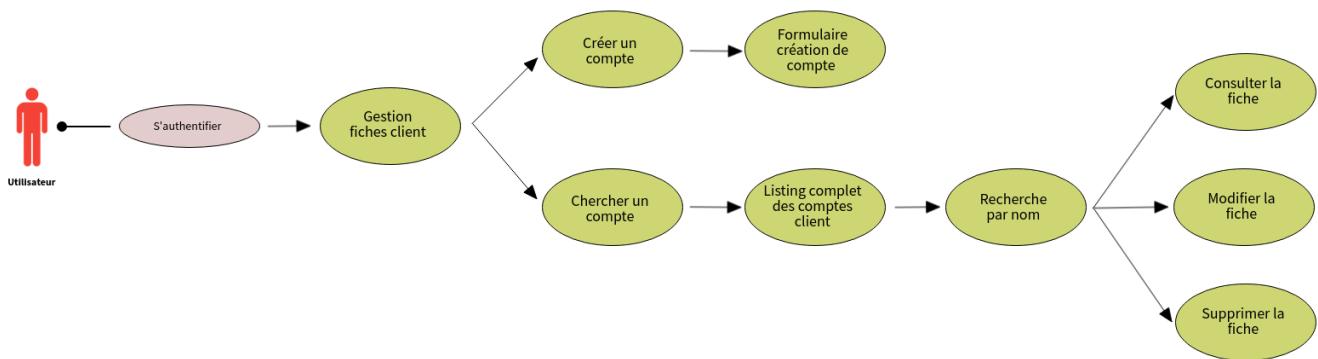
Page d'accueil

Après authentification, un utilisateur pourra accéder aux fichiers clients.

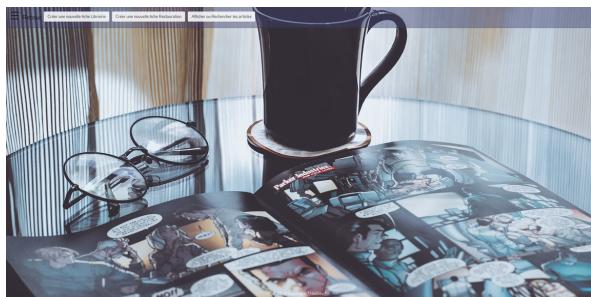
Il aura le droit de consulter, créer, modifier ou supprimer une fiche client sans aucune contrainte de priviléges.

L'utilisateur pourra trouver facilement un client grâce à une barre de recherche placée au dessus du listing client.

### Diagramme de cas d'utilisation " Gestion fiches clients " :



### 4.5.7 Gestion des fiches articles



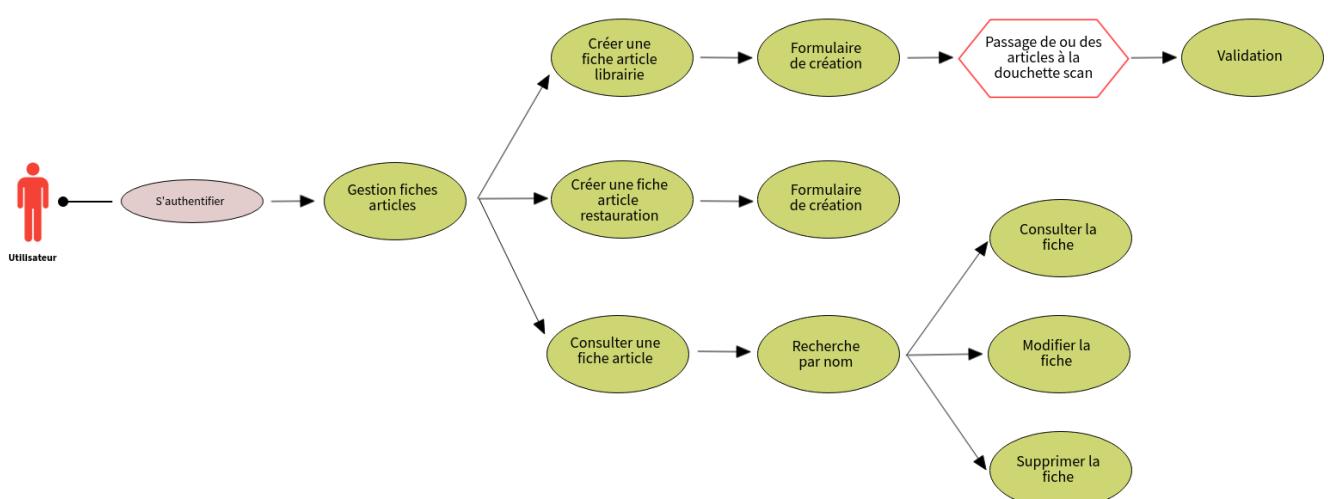
Page d'accueil

En ce qui concerne les articles de type librairie, les données et caractéristiques du produit devront être enregistrées en utilisant la douchette.

Après authentication, un utilisateur pourra accéder aux fichiers articles.

Tout comme les fichiers clients, l'utilisateur aura le droit de consulter, créer, modifier ou supprimer une fiche article sans aucune contrainte de priviléges.

### Diagramme de cas d'utilisation " Gestion fiches articles " :



#### 4.5.8 Gestion des commandes

Après authentification, un utilisateur pourra accéder à la gestion des commandes.

Il aura le droit de consulter, créer, modifier ou supprimer une commande sans aucune contrainte de privilèges.

Il pourra également avoir le droit de réceptionner les différentes commandes qui arrivent à la librairie. Pour cela, il devra utiliser la douchette pour entrer les caractéristiques de chaque titre réceptionné en base de données.

L'utilisateur pourra aussi faire une recherche de commande et avoir ainsi une vue d'ensemble concernant les informations de cette commande.

Il existe plusieurs sortes de commandes :

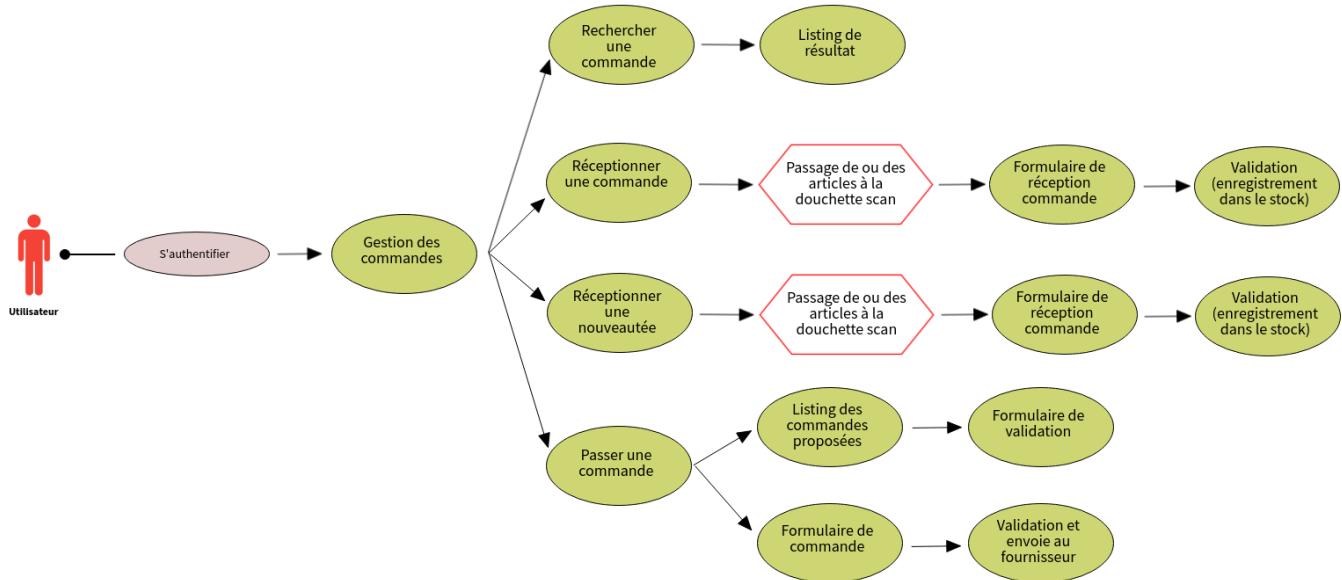
- **Les commandes clients** qui sont des demandes particulières des clients ou tous simplement établies à leur demande car le titre n'est plus en stock dans la librairie.

- Celles **des fournisseurs** qui sont les commandes du libraire pour approvisionner le stock.

- **Les nouveautés** qui sont des commandes passées directement avec les commerciaux qui se déplacent à la librairie.

- Et pour finir, **les commandes proposées**. Celles-ci sont un récapitulatif des ventes de la journée qui restent enregistrées en base de données. Ainsi, le libraire aura la possibilité de visualiser rapidement les manques dans le stock et valider ou annuler une nouvelle commande chez le fournisseur par un simple envoi de mail automatisé.

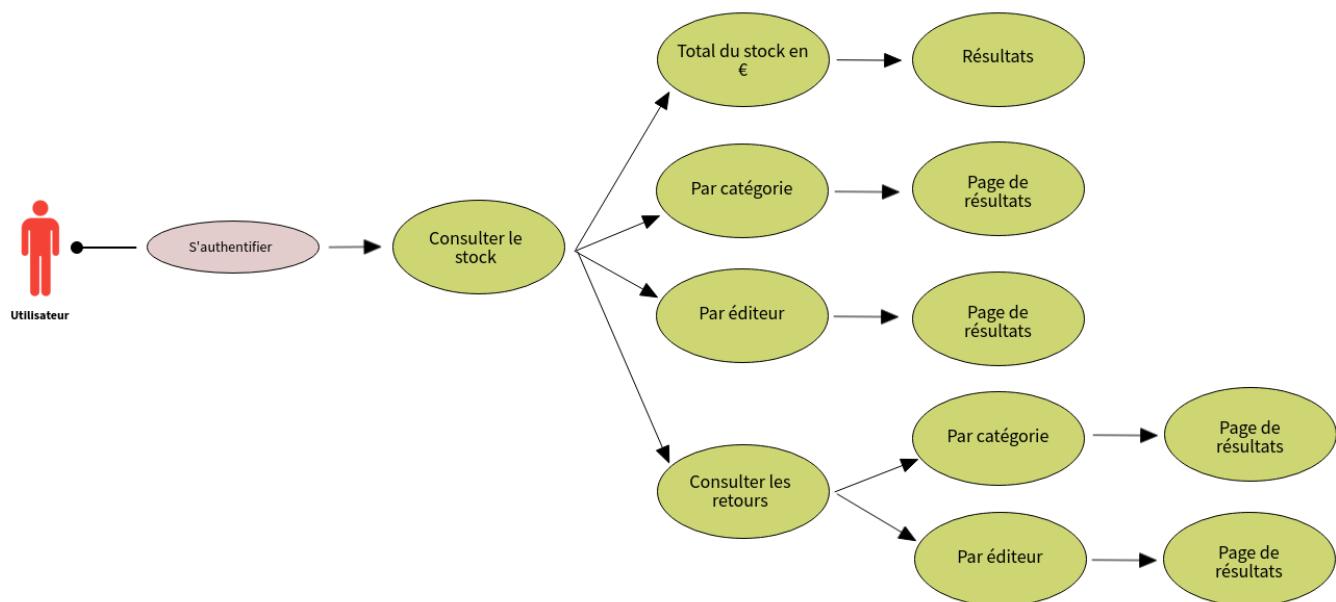
**Diagramme de cas d'utilisation " Gestion des commandes " :**



#### 4.5.9 Consulter le stock

Après authentification, un utilisateur pourra accéder aux données du stock de la librairie. Il aura le droit de consulter les données sans aucune contrainte de privilèges.

**Diagramme de cas d'utilisation " Consulter le stock " :**



#### 4.5.10 Tableau de bord

Après authentification, un utilisateur pourra accéder aux données du Tableau de bord.

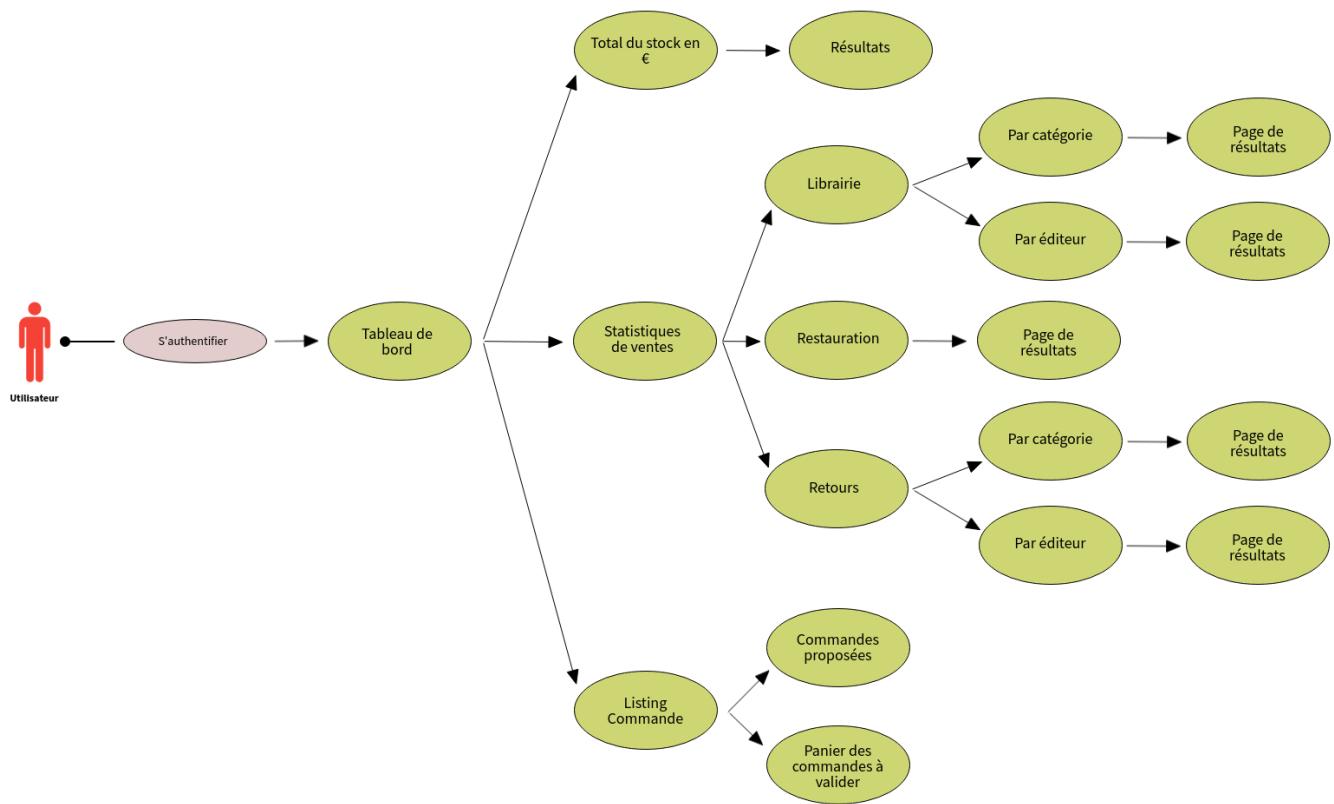
Il aura le droit de consulter les données sans aucune contrainte de privilèges.

Le tableau de bord va servir essentiellement à visualiser, par le biais de différents graphiques, toutes les données enregistrées en base données concernant les ventes librairie, les ventes restauration, les retours fournisseurs, les commandes, le total en euros du stock, ....

Les données pourront être consultées de différentes manières car elles seront classées par jours, semaines, mois et années.

Les données concernant les ventes librairie et les retours fournisseurs seront classées par éditeurs et par catégories ( BD, mangas, jeunesse, ... ). Cela permettra de visualiser, avec précision, toutes les informations nécessaires à l'utilisateur pour étudier et analyser les différences entre les achats de la librairie, les ventes et les retours fournisseurs qui sont très coûteux pour la librairie.

On retrouvera également dans le tableau de bord les informations sur les commandes proposées et un panier qui contiendra toutes les commandes clients, fournisseurs et nouveautés en attentes de validation par l'utilisateur.

**Diagramme de cas d'utilisation " Tableau de bord " :**


#### 4.6 Présentation du jeu d'essai

Référence de scénario : Rechercher un client par son nom

Action	Résultats attendus	Résultats obtenus
Cliquer sur " Gestion des fiches clients "	Ouverture de la page d'accueil " Clients "	Pas d'erreurs
Cliquer sur " Rechercher client"	Ouverture de la page " Vos clients " dans laquelle se trouve le tableau avec tous les clients et ses données.	Pas d'erreurs
Rechercher le client par son nom	Taper le nom du client dans la barre de recherche afin d'éliminer les autres données qui ne sont pas en rapport avec la recherche.	Pas d'erreurs
Afficher les données du client	Affichage des données du client.	Pas d'erreurs

**Extraits de code du jeu d'essai :**

➤ Extraits du code pour le tableau d'affichage :

```
<div class="listCustom_container">
{-- <table class="table table-sm table-dark table-striped" id="customerTable" cellspacing="0" width="90%"> --}
<table id="customerTable" class="table table-striped table-bordered table-sm" cellspacing="0" width="100%">
<label for="searchCust">Rechercher un client par son nom :</label>
<input type="text" id="searchCust" onkeyup="searchCustomer()" placeholder="Nom du client" onfocus="this.value=''">

<thead>
    <tr>
        <th class="th-sm">id.</th>
        <th class="th-sm">Date de création</th>
        <th class="th-sm">Nom</th>
        <th class="th-sm">Prénom</th>
        <th class="th-sm">Raison sociale</th>
        <th class="th-sm">Adresse</th>
        <th class="th-sm">age</th>
        <th class="th-sm">sexe</th>
        <th class="th-sm">mail</th>
        <th class="th-sm">tel.fixe</th>
        <th class="th-sm">tel.mobile</th>
    </tr>
</thead>
<tbody>

    @if(count($client) > 0)

        @foreach($client as $client)

            <tr>
                <td> {{ $client->id_client }} </td>
                <td> {{ $client->created_at }} </td>
                <td> {{ $client->nom }} </td>
                <td> {{ $client->prenom }} </td>
                <td> {{ $client->raison_sociale }} </td>
                <td> {{ $client->numero_rue.' '.$client->rue.' '.$client->code_postal.' '.$client->ville }} </td>
                <td> {{ $client->age }} </td>
                <td> {{ $client->sexe }} </td>
                <td> {{ $client->mail }} </td>
                <td> {{ $client->tel_fixe }} </td>
                <td> {{ $client->tel_mobile }} </td>
            </tr>
        @endforeach

        @else
            <p id='alertCust'>Vous n'avez pas de clients enregistrés</p>
        @endif
    @endif
</tbody>
```

➤ Code JS pour la recherche du client :

```
// code js for the customers search
function searchCustomer() {
    // Declare variables
    var input, filter, table, tr, td, i, txtValue;
    input = document.getElementById("searchCust");
    filter = input.value.toUpperCase();
    table = document.getElementById("customerTable");
    tr = table.getElementsByTagName("tr");

    // Loop through all table rows, and hide those who don't match the search query
    for (i = 0; i < tr.length; i++) {
        td = tr[i].getElementsByTagName("td")[2];
        if (td) {
            txtValue = td.textContent || td.innerText;
            if (txtValue.toUpperCase().indexOf(filter) > -1) {
                tr[i].style.display = "";
            } else {
                tr[i].style.display = "none";
            }
        }
    }
}
```

---

## Chapitre 5

---

### Veille sur les vulnérabilités de sécurité

## 5 Veille sur les vulnérabilités de sécurité

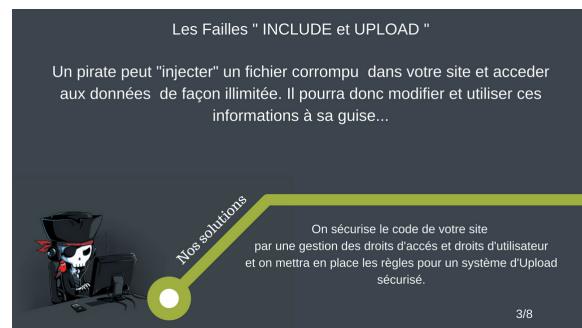
Dans le cadre de ma formation, nous avons eu l'occasion de préparer une journée " portes ouvertes " dans les locaux de Simplon.

Pour préparer cette journée, j'étais en charge, avec **Alexandra KUROWSKI**, une apprenante et collègue, de présenter au public une veille sur la sécurité et les précautions à prendre sur la toile. C'est donc ce même travail que j'ai présenté au personnel du Migou et à mon tuteur.

Lien vers la diapositive :

<https://drive.google.com/file/d/15YQsvTxizGOrtOENDu9DyfpCE-ZjJzKf/view?usp=sharing>

Quelques exemples de la diapositive :



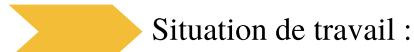
---

## Chapitre 6

---

### Situation de travail et site anglophone

## 6 Situation de travail et site anglophone



Situation de travail :

Ayant fait l'acquisition, en début de stage, du livre "**Laravel, un framework efficace pour développer vos applications PHP**" de Raphaël HUCHET, j'avoue avoir eu beaucoup moins besoin de faire mes recherches sur la toile.

Toutefois et en cours de conception des tables, j'ai eu besoin de retourner sur le site officiel de Laravel et de MySQL plusieurs fois pour le choix du type de variables.

Pour des raisons de sécurité, je voulais employer les bons types de variables et j'ai trouvé les informations nécessaires à la bonne pratique de conception sur ces deux sites.

Je ne vais pas énumérer toutes mes recherches mais je me souviens avoir hésité sur les "**Integer**".



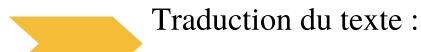
Recherche sur MySQL.com :

### 11.2.1 Integer Types (Exact Value) - INTEGER, INT, SMALLINT, TINYINT, MEDIUMINT, BIGINT

MySQL supports the SQL standard integer types **INTEGER** (or **INT**) and **SMALLINT**. As an extension to the standard, MySQL also supports the integer types **TINYINT**, **MEDIUMINT**, and **BIGINT**. The following table shows the required storage and range for each integer type.

Table 11.1 Required Storage and Range for Integer Types Supported by MySQL

Type	Storage (Bytes)	Minimum Value Signed	Minimum Value Unsigned	Maximum Value Signed	Maximum Value Unsigned
<b>TINYINT</b>	1	-128	0	127	255
<b>SMALLINT</b>	2	-32768	0	32767	65535
<b>MEDIUMINT</b>	3	-8388608	0	8388607	16777215
<b>INT</b>	4	-2147483648	0	2147483647	4294967295
<b>BIGINT</b>	8	-2 <sup>63</sup>	0	2 <sup>63</sup> -1	2 <sup>64</sup> -1



Traduction du texte :

" MySQL peut supporter les types entiers **INTEGER** (ou **INT**) et les **SMALLINT**. En prolongement de la norme, MySQL peut également supporter les integer de types **TINYINT**, **MEDIUMINT** et **BEGINT**. Le tableau suivant présente les besoins de mémoire et les types d'Integer ".

## Conclusion



Deux mois et demi.

Ça me semblais tellement bien adapté en temps pour concevoir et finaliser la demande de mon tuteur de stage.

Deux mois et demi .... oui, mais....

J'ai donc effectué mon stage, en tant que développeur web, dans une librairie spécialisée BD, pour répondre à leur besoin de création d'un logiciel de gestion de librairie et de petite restauration.

Lors de ce stage, j'ai pu mettre en pratique mes connaissances théoriques acquises durant ma formation à l'école régionale du numérique "Simplon.Co Auch", tout en étant confronté aux difficultés réelles du métier et à la gestion du temps.

Ce stage m'a surtout permis de comprendre que la création d'un logiciel demande du temps, de la réflexion autour de cette gestion et que ces deux facteurs sont essentiels à la réalisation de tout projet.

Cette expérience m'a également permis de me rendre compte de mes capacités d'adaptation face à un stress de "besoin de résultat" et de "conscience professionnelle". En effet, et toujours en lien avec le temps "qui passe", je me suis rendu compte que je devais mieux organiser mes sprints pour améliorer la gestion de conception et de résultat final.

C'est ce qui a été le plus enrichissant pour moi car j'ai ainsi découvert des axes d'améliorations pour mon avenir professionnel. Je suis conscient, aujourd'hui, des erreurs à éviter.

Pour conclure, ce stage m'a conforté dans mon avenir professionnel et m'a permis de réfléchir autrement sur ma méthode de travail. Je me suis également rendu compte qu'il a fait naître en moi une volonté d'évolution dans l'apprentissage de nouveaux langages.

## Chapitre 9

# Annexes

### Plan

9.1 Diagramme d'état .....	12
9.2 Maquettes réalisés en début de projet .....	12
9.3 Site-Map .....	4
9.4 Extraits de mon code .....	12