# **Bookstore : Spécification des besoins :**

Dans cette partie nous allons identifier les acteurs du système réagissant de près ou de loin avec le système, les besoins fonctionnels et non fonctionnels de notre système afin de mieux comprendre le contexte de notre projet.

## **1. Les besoins fonctionnels :**

**Créer un compte :** Chaque visiteur du Bookstore peut accéder à un espace de création de compte afin de devenir membre.

**S'authentifier :** Un utilisateur peut s'authentifier et accéder à son espace personnel avec un nom d'utilisateur ou une adresse mail et un mot de passe.

**Gérer son compte :** Un membre peut accéder à son espace privé pour gérer son compte.

**Chercher un livre :** Chaque type d’utilisateur doit pouvoir effectuer une recherche sur les livres disponibles dans le Bookstore et consulter leurs informations.

**Sélectionner 3 genres de livres :** Après la création d’un compte, le membre choisit 3 genres de livres préférés.

**Consulter la liste de recommandation :** Un membre peut consulter des livres recommandés par l’application selon le choix de genre choisis lors de l’inscription.

**Consulter la liste des livres populaires** : Un membre peut consulter les livres les plus notés par les autres membres de l’application.

**Filtrer la liste des livres selon des critères** : Un membre peut filtrer la liste des livres disponibles dans le Bookstore selon : leur type, leur genre, leur nombre de pages et leurs maisons d’édition.

**Passer une commande :** Un membre peut passer une commande pour un livre.

**Consulter historique des achats:** Un membre peut consulter la liste des livres commandés.

**Ajouter un feedback :** Un membre peut donner des avis concernant les livres et peut attribuer une notre /5 à un livre.

**Envoyer une recommandation :** Un membre peut réclamer l’ajout d’un livre non disponible dans le Bookstore.

**Gérer les produits :** Les gérants et propriétaire peuvent ajouter, supprimer ou modifier les produits du Bookstore.

**Gérer les commandes :** Les gérants et propriétaire peuvent consulter, valider ou refuser les commandes.

**Gérer les feedbacks :** Le propriétaire et ses gérants peuvent consulter ou supprimer les feedbacks.

**Consulter les statistiques des bookstores :** Le propriétaire peuvent consulter les statistiques générales de son Bookstore. (Vente, disponibilité et livraisons des articles).

**Gérer les comptes utilisateur gérant:** Le propriétaire peut gérer la liste des utilisateurs.

## **2. Les besoins non-fonctionnels :**

Toutes les applications à un certain point dans leurs cycles de vie doivent considérer des besoins non-fonctionnels. Pour notre cas on note :

**Fiabilité** : L’application doit être fiable afin d’assurer sa mission dans les conditions d’environnement donnés et pendant une durée donnée.

**Sécurité :** L’application doit être sécurisée pour protéger ses données et les données de ses utilisateurs.(spring security)

**Disponibilité :** L’application doit être disponible à tout instant, utilisable et facilement accessible grâce à l’architecture 4-tier choisie.

**Maintenabilité** : L’application doit être maintenable afin de facilité l’analyse et la modification de l’application ainsi qu’avoir une stabilité et testabilité continue.

## **3. Identification des acteurs :**

Un acteur est une entité externe et autonome qui interagit directement avec le système.

Notre projet contient 4 acteurs :

* **Propriétaire** **ou Admin** : C’est l’acteur qui gère les produits, assigne et gère les gérants et consulte les statistiques et les données de son Bookstore.
* **Gérant** : C’est l’acteur qui gère les produits.
* **Membre** : C’est l’acteur clé de tout Bookstore, ce dernier peut consulter (naviguer entre les différents services et pages, recherche …), passer une commande, gérer son compte ou donner un feedback.
* **Visiteur** : C’est l’acteur qui visite l’application pour s’inscrire ou effectuer une recherche sur les livres disponibles.

## **Bookstore : Backlog du produit :**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Id** | **En tant que …** | **je veux …**  **(Fonctionnalité)** | **Description** | **Priorité** | **Tâche attribué à** |
| **1** | Propriétaire | Ajouter des gérants à l’application | Développer une interface de création, des comptes utilisateurs gérants. |  |  |
| **2** | Propriétaire | Consulter la liste de ses gérants. | L’interface du propriétaire doit afficher la liste de ses gérants. |  |  |
| **3** | Propriétaire | Supprimer un gérant de la liste. | L’interface du propriétaire doit contenir un bouton pour supprimer un gérant à partir de la liste. |  |  |
| **4** | Propriétaire | Consulter mon chiffre d’affaire mensuel. | L’interface du propriétaire doit contenir un dashboard ou une courbe qui faite le calcul total des ventes par mois. |  |  |
| **5** | Propriétaire …  Gérant | Consulter les statistiques du Bookstore. | L’interface de l’administrateur et du gérant doit contenir des statistiques du type : nombre de vente des livres, nombre d’exemplaire disponible. |  |  |
| **6** | Propriétaire …  Gérant | Consulter les feedback des utilisateurs. | L’application doit notifier l’administrateur ou le gérant lors de la publication d’un feedback à propos d’un livre de la part d’un membre |  |  |
| **7** | Propriétaire …  Gérant | Supprimer les feedbacks des utilisateurs. | Lors de la consultation des feedback, l’administrateur ou le gérant ont la possibilité de supprimer un feedback inapproprié. |  |  |
| **8** | Gérant | Consulter la liste des commandes sur les livres. | L’application doit notifier les gérants lors de la réception d’une nouvelle commande de la part d’un membre. |  |  |
| **9** | Gérant | Valider les commandes des membres sur les livres. | L’application doit contenir une interface pour consulter la liste des livraisons qui ne sont pas encore validées. |  |  |
| **10** | Gérant | Refuser les commandes des membres sur les livres. | Lors de la réception d’une notification d’une nouvelle commande. |  |  |
| **11** |  |  |  |  |  |
| **12** |  |  | . |  |  |
| **13** |  |  |  |  |  |

US:

* en tant que “ role” j’ai besoin “description” afin de “ fonctionnalité”
* en tant que “ role” je veux “ fonctionnalité”

Priorité

ROI = BV/CX

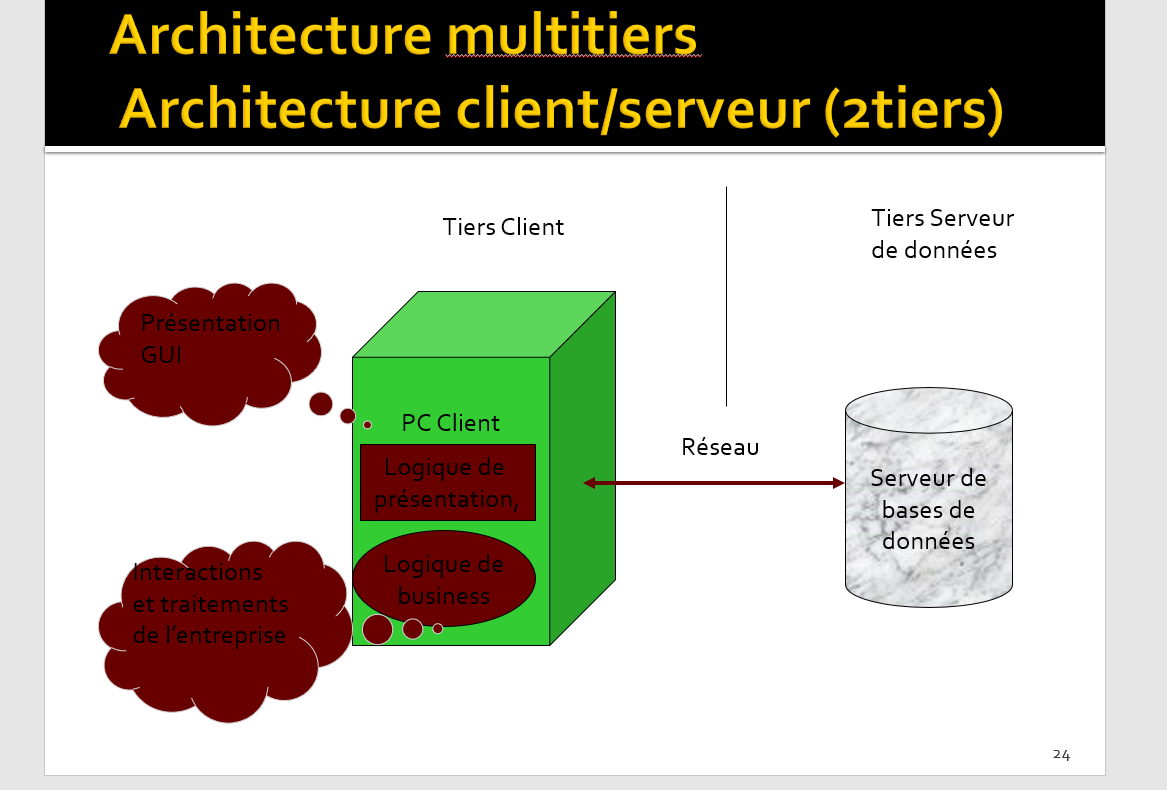
+BV 0--100

+CX 0--100

## Bookstore : architecture physique choisie :

Pour l’application Desktop :

Architecture client/serveur: Un serveur est un module logiciel qui répond aux requêtes d’autres modules appelés clients; Généralement, les services et les clients sont hébergés dans des machines différentes et communiquent via le réseau (Intranet/Internet) Ex: le service http répond aux requêtes des clients qui sont les navigateurs Web



Pour l’application web :

Pour améliorer la performance de notre application, nous utilisons une architecture 3 tiers, le choix de cette architecture a pour objectif de bien structurer et organiser les informations. Elle permet la séparation entre les différents composants du système. L’architecture trois tiers, également appelée architecture à trois niveaux ou à trois couches, est une architecture client-serveur :

— La présentation des données (Application) : correspondant à l’affichage, l’interaction homme machine.

— Le traitement métier des données (Serveur) : correspondant à la mise en oeuvre de l’ensemble des traitements.

— L’accès aux données persistante (Base de données) : correspondant aux données qui sont destinées à être conservées.

