

CAHIER DES CHARGES TECHNIQUE

Projet gestion de bibliothèque



Dossier rédigé par OUAKRIM Hanane

Sommaire

| | |
|---|---|
| 1. Contexte du projet | 3 |
| 1.1. Présentation du projet | 3 |
| 1.2. Date de rendu du projet | 3 |
| 2. Besoins fonctionnels | 3 |
| 3. Ressources matérielles nécessaires à la réalisation du projet..... | 3 |
| 4. Ressources logicielles nécessaires à la réalisation du projet..... | 3 |
| 5. Gestion du projet | 4 |
| 6. Conception du projet | 4 |
| 6.1. Le front-end | 4 |
| 6.2. Le back-end..... | 4 |
| 6.2.1. Diagramme de cas d'utilisation | 5 |
| 6.2.2. Diagramme de classe | 5 |
| 6.2.3. Modèles Conceptuel de Données (MCD) | 6 |
| 6.2.4. Modèles Logique de Données (MLD) | 6 |
| 6.2.5. Modèle Physique de Données (MPD)..... | 6 |
| 7. Environnement technologique | 7 |
| 7.1. Pare feu | 7 |
| 7.2. Antivirus | 7 |
| 7.3. OS version | 7 |
| 7.4. IDE Version | 7 |
| 7.5. SGBD | 7 |
| 7.6. Navigateur..... | 8 |
| 7.7. Github | 8 |
| 7.8. Les type de tests unitaires | 8 |
| 7.9. Les type de tests fonctionnels | 8 |

1. Contexte du projet

1.1. Présentation du projet

La bibliothèque municipale actuelle gère manuellement le processus d'emprunt et de retour des livres par ses adhérents. Pour améliorer l'a capacité et réduire la charge de travail des bibliothécaires, la direction a décidé d'acquérir une borne automatique permettant aux adhérents de retirer ou de déposer des livres sans l'assistance du personnel. Cependant, la borne est livrée sans logiciel adéquat pour enregistrer les transactions.

Votre équipe et vous-même avez pour mission de proposer une solution temporaire qui répondra à la demande du client.

1.2. Date de rendu du projet

Le projet doit être rendu au plus tard le 15 mars 2024.

2. Besoins fonctionnels

Le logiciel devra permettre de gérer les données de la librairie.

Les données seront stockées dans une base de données relationnelle pour faciliter la gestion et la mise à jour des informations. Ces données peuvent être gérées directement depuis le logiciel.

3. Ressources matérielles nécessaires à la réalisation du projet

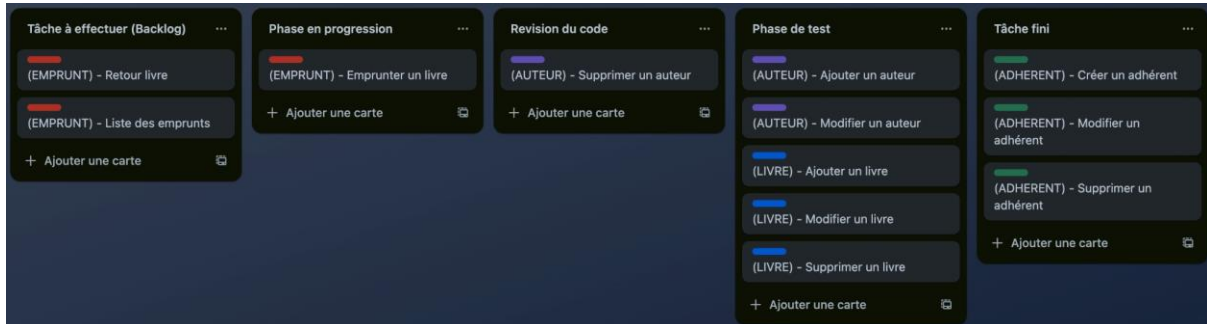
Pour mener à bien ce projet, nous aurons besoin de certains équipements matériels, notamment un ordinateur avec ses périphériques (souris, clavier et écran), ainsi qu'un ordinateur portable.

4. Ressources logicielles nécessaires à la réalisation du projet

Pour ce projet, nous aurons besoin des logiciels suivants : Visual Studio Code pour l'édition de texte, MAMP pour simuler un serveur de base de données, Figma et Trello pour la conception et la gestion de projet, ainsi que Mocodo et Visual Paradigm pour la modélisation.

5. Gestion du projet

Pour réaliser ce projet, j'ai utilisé la méthode Agile Kanban. J'ai utilisé l'outil de gestion de projet en ligne Trello.



De plus, la gestion de ce projet s'opère à travers un diagramme de Gantt, permettant ainsi de définir la durée nécessaire à sa réalisation.

[illegible]

6. Conception du projet

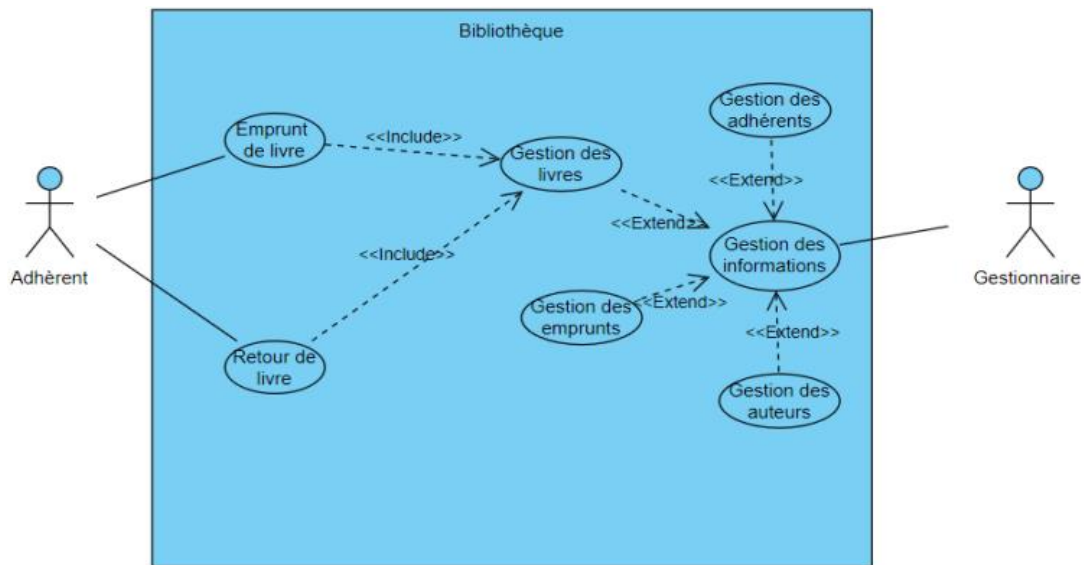
6.1. Le front-end

La conception du projet repose sur le développement d'une interface utilisateur dynamique et intuitive, élément crucial pour l'expérience utilisateur dans son ensemble.

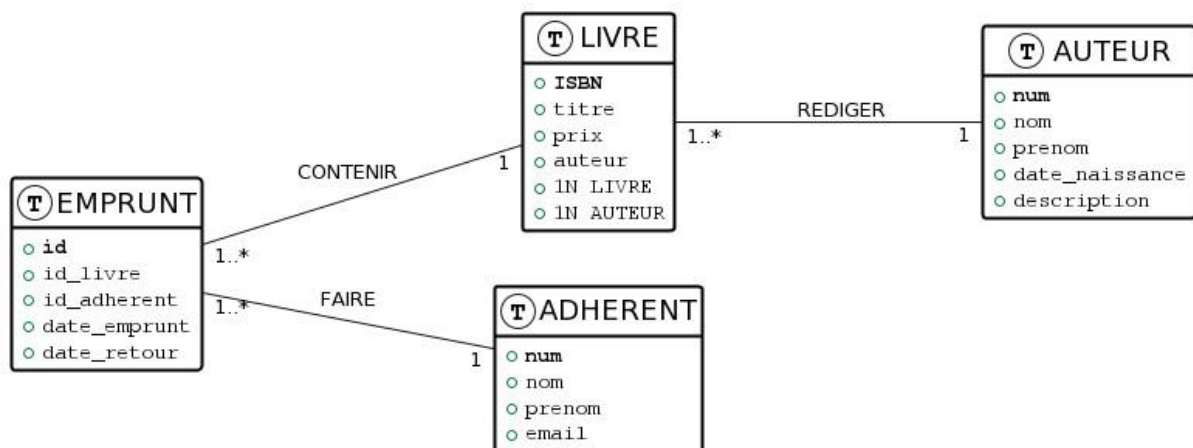
6.2. Le back-end

Le backend est responsable de la communication avec la base de données, de l'exécution des opérations côté serveur, et de la génération des réponses nécessaires aux requêtes provenant du frontend.

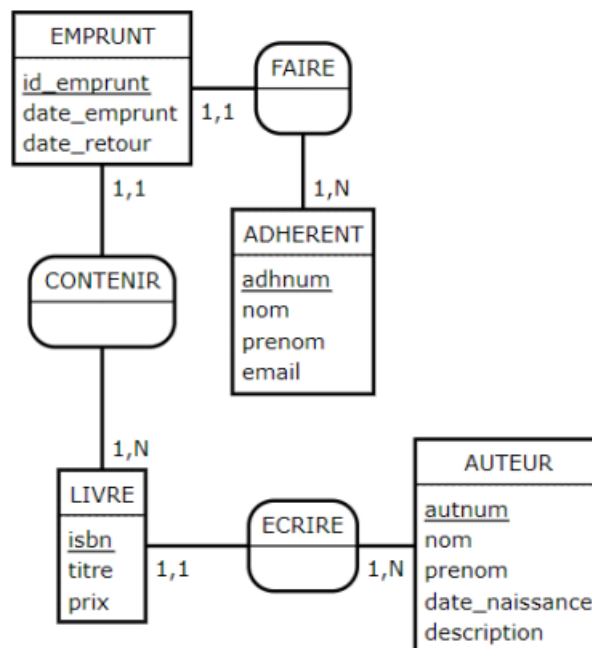
6.2.1. Diagramme de cas d'utilisation



6.2.2. Diagramme de classe



6.2.3. Modèles Conceptuel de Données (MCD)



6.2.4. Modèles Logique de Données (MLD)

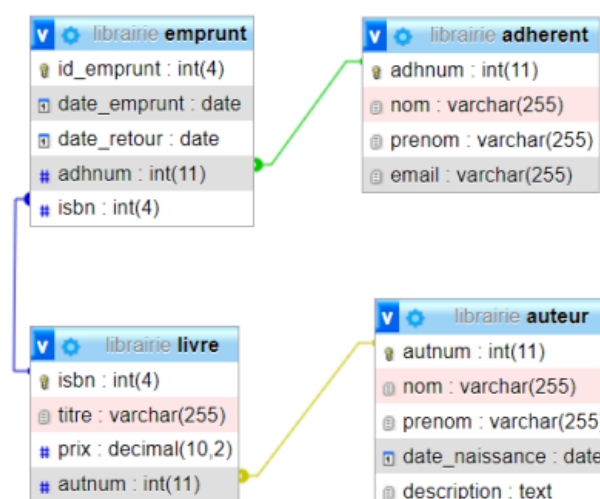
ADHERENT (adhnum, nom, prenom, email)

AUTEUR (autnum, nom, prenom, date_naissance, description)

EMPRUNT (id_emprunt, date_emprunt, date_retour, #adhnum, #isbn)

LIVRE (isbn, titre, prix, #autnum)

6.2.5. Modèle Physique de Données (MPD)



7. Environnement technologique

Le choix de l'environnement technologique pour le développement et le test du système de gestion de bibliothèque est crucial pour le succès du projet et son déploiement au sein de la bibliothèque. Par conséquent, les spécifications technologiques sont essentielles pour garantir que le processus de développement se déroule de manière efficace et que le produit final réponde aux exigences attendues.

7.1. Pare feu

Pour sécuriser le développement du projet, nous utiliserons Windows Defender comme pare-feu pour les systèmes Windows.

7.2. Antivirus

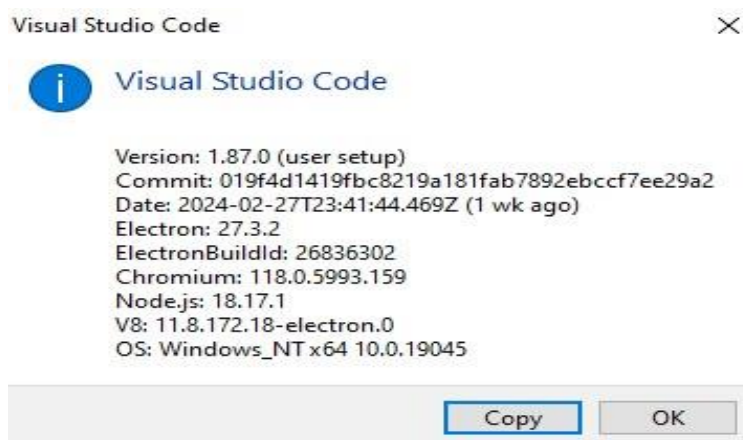
Nous installerons un logiciel antivirus sur toutes les machines utilisées pour le développement et le déploiement du système de gestion de bibliothèque. Nous choisirons une solution reconnue telle que Kaspersky, qui offre une protection en temps réel contre les virus, les logiciels malveillants, etc.

7.3. OS version

Le système sera développé et testé sur plusieurs versions de systèmes d'exploitation, y compris Windows 10 et 11, pour assurer une compatibilité maximale. Nous ciblerons les dernières versions stables de Windows afin de garantir une expérience utilisateur cohérente sur les différentes machines.

7.4. IDE Version

L'utilisation de la dernière version stable d'un environnement de développement intégré (IDE) tel que Visual Studio Code offrira une productivité maximale et proposera des fonctionnalités avancées pour le développement du programme de gestion de bibliothèque.



7.5. SGBD

Le choix d'un système de gestion de base de données fiable tel que MySQL permettra de stocker et de gérer efficacement les données de la bibliothèque, assurant ainsi une performance optimale du système. Pour cette conception en local, l'utilisation du logiciel MAMP sera nécessaire pour générer un serveur. Nous possédons ainsi les versions suivantes :

- Version du client de base de données : libmysql - mysqlnd 7.4.33
- Version de phpMyAdmin : 5.2.0

7.6. Navigateur

Nous utilisons Google Chrome Version 122.0.6261.112 (Build officiel) (64 bits) pour visualiser la base de données relationnelle présente en local.

7.7. Github

Pour assurer le bon déroulement du développement du projet de gestion de bibliothèque, le code sera déployé sur GitHub. Cette plateforme offre une gamme complète d'outils de gestion de versions basés sur Git, ce qui permettra de suivre l'évolution du code source, de gérer les branches de développement, et de faciliter la résolution des conflits.

7.8. Les type de tests unitaires

Nous procéderons à des tests unitaires pour vérifier le bon fonctionnement de chaque élément du programme. Ces tests comprendront des validations des entrées ainsi que des vérifications des fonctions internes du programme.

7.9. Les type de tests fonctionnels

Nous procéderons à des tests fonctionnels pour évaluer le système dans son ensemble en simulant les interactions utilisateur. Ces tests incluront des scénarios d'utilisation typiques ainsi que des tests de compatibilité avec différents systèmes d'exploitation. L'objectif est de garantir la stabilité et les performances du système dans des conditions réelles d'utilisation.