***Projet:***

*Gestion de livres* Dans une bibliothèque

**

***-Nom des participants :***

***Yassine AIT ALLA***

***Sofiane ZAION***

**TABLE DES MATIERES**

*[Section 1 Plan du document Projet 5](#_TOC_250005)*

* 1. [Liste des versions 5](#_TOC_250004)
  2. [But du document 5](#_TOC_250003)
  3. [Cahier des charges 5](#_TOC_250002)
  4. [Spécification 5](#_TOC_250001)
  5. Planning 5
  6. [Conception 6](#_TOC_250000)
  7. Réalisation des tests 6
  8. Traitement de la sécurité et RGPD 6
  9. Des annexes 6

*Section 2 Codage de l’application 7*

## 

## *Section 1 Plan du document projet*

### *Liste des versions*

|  |  |  |
| --- | --- | --- |
| **Version** | **Date de la version** | **Libellé** |
| 1.3  1.4  1.6 | 26/06/2022  25/07/2022  29/07/2022 | Cahier des charges Diagramme UML (use case) Diagramme de classe |

### *But du document*



* Le projet sur lequel nous avons décidé de nous baser mon camarade ainsi que moi est le suivant :
* **Gestion de livres dans une bibliothèque.**
* Nous avons ici pour but d’effectuer les objectifs cités par le client(nous). Cela va nous permettre de disposer d’une application autonome à laquelle il faut s’identifier pour accéder aux fonctions de l’outil.

***Charte graphique***

## *Logo :*

## 

## *Nb pages : 7 ;*

* ***Langue :* Française ;**

## *Couleur principale :*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Aperçu de l'orange** | **Code HEXA** |  |  |  |
|  | #ff7f00 |  |  |  |

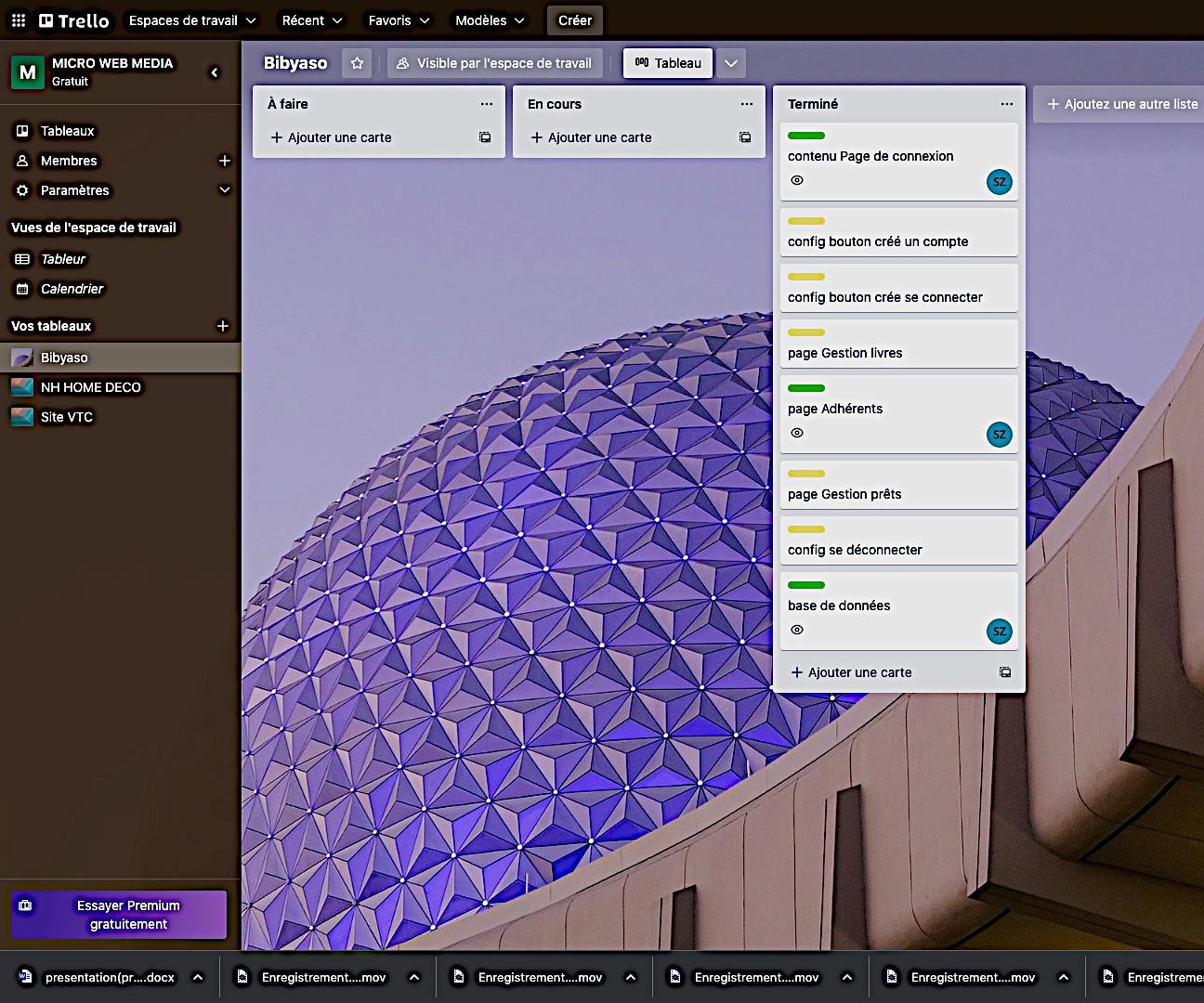
* ***Couleur secondaire :***

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Aperçu du vert** | **Code HEXA** |  |  |  |
|  | #00FF00 |  |  |  |

### *Cahier des charges*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ***Quel est notre objectif ?*** | ***Qui sont nos utilisateurs***  ***?*** | ***Qu’attend t’en-t-il de nous ?*** | ***Les objectifs du projet*** | ***Demande du client***  ***(NOUS)*** | ***Conditions*** |
| * Notre objectif est de satisfaire les besoins du client (Nous). * Nous Permettre une bonne gestion de notre bibliothèque   (BibYaso). | -les clients qui utiliseront l’application sont :  - Nous  **-Avantage :**    Ne plus utilisé de feuille pour gérer les emprunts mais plutôt une application simple à l’utilisation. | -une application monotone pratique à l’utilisation  -Plusieurs choix pour un maximum de plaisir concernant vos lectures. | -Construire de manière collective ce projet pour pouvoir atteindre le résultat final ;  -Notre application est qualifiée de plateforme bien structurer et fluide à l’utilisation.  Permettre une bonne gestion de notre bibliothéque et eviter | - Il faut s’identifier pour accéder aux fonctions de l’outil.   * Qui contient 7 pages * Connexion * Gestion livres * Nos adhérents * Ajout d’adhérent * Gestion prêts ( qui affiche un tableau des prêts effectué par nos adhérents) * Gestion des emprunts ( qui permet de récupérer nos adhérents inscrit dans notre bibliothèque pour ensuite effectué un emprunt)   - Qui permet de créer des utilisateurs( Membre de la bibliothèque )  -Qui permet de référencer des nouveaux livres, auteurs, collections, éditeurs.  -Qui permet de référencer des adhérents pour l’emprunt de Livre.  -Qui permet de gérer l’emprunt, la restitution d’un livre.  -Qui montre la liste des livres empruntés.   * Qui affiche la date de l’emprunt de livre. * Qui affiche 4 Bouton Image   -Gestion de livres  -Adhérents  -Gestion des prêts  -Deconnexion | **-livre :**  Livre existe en 1 ou plusieurs exemplaires dans une ou plusieurs Collections (chez 1 ou plusieurs éditeurs).  -A**dhérent :**  Un adhérent ne peut pas emprunter plusieurs livre à la fois pour pour la gestion de notre bibliothèque et aussi éviter que l’adhérent dispose de plusieurs livres sans restituer celui qu’il à emprunter.  **-adhérent** : peut emprunter un ou plusieurs livres.  En revanche, ne peut pas emprunter plusieurs exemplaires (même livre, de la même  collection) |

# Trello (Répartition des tâches)

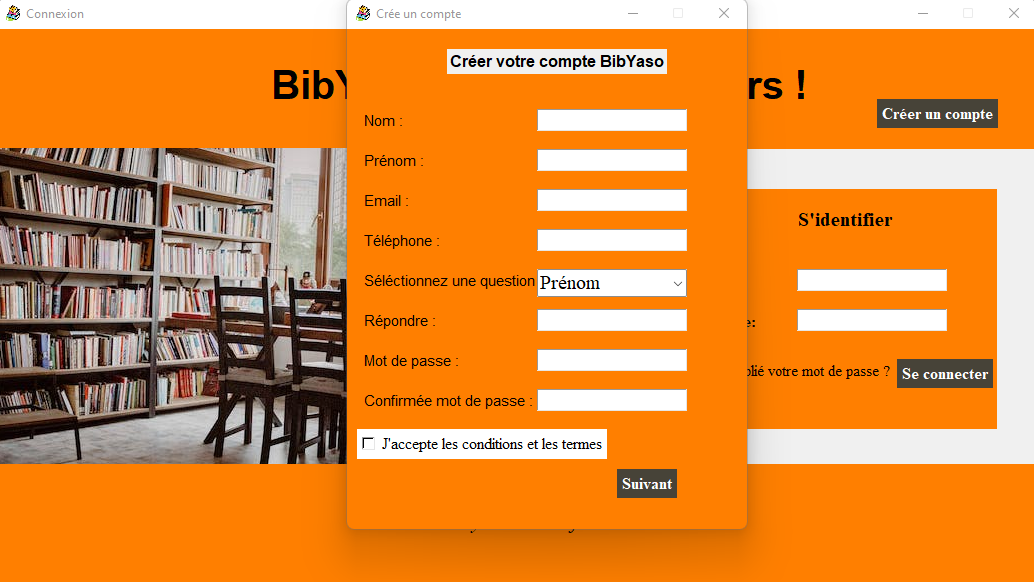
* **Trait jaune : Yassine Ait Alla. - Trait Vert : Sofiane Zaion**

### *Spécification*

***Maquettes :***

Voici notre parti conception ou figure dans un premier temps la maquette (la structure de l’appli) de notre projet.

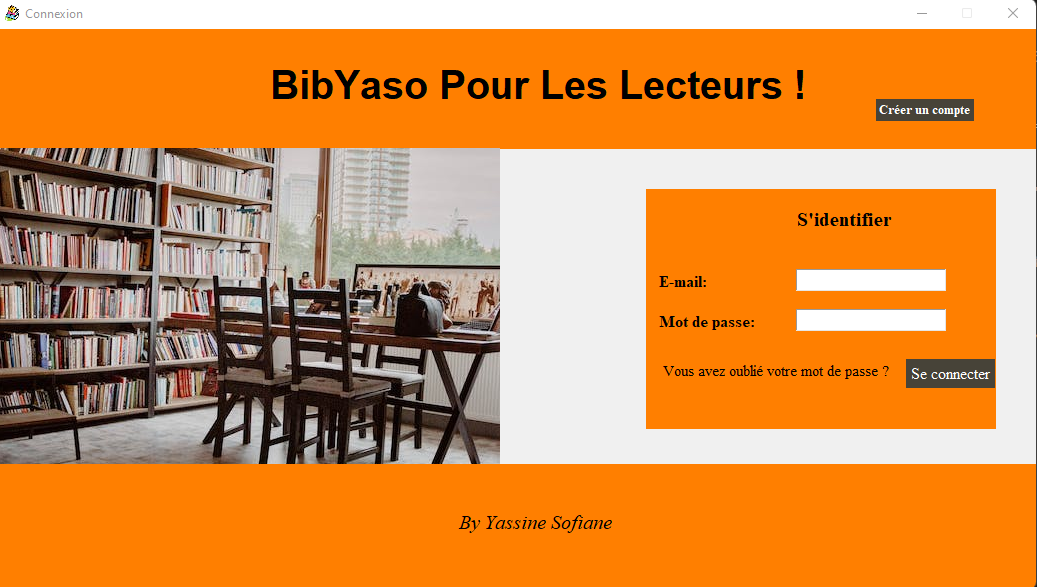
***Partie crée un compte :***

******

#### *Description :*

-Tout d’abord, avant de se connecter puis de lancer l’application, la création du compte est requis pour pouvoir entrer dans celle-ci.

***Page de connexion :***

******

#### *Description :*

* Ensuite, voici la page de connexion de notre application pour pouvoir y avoir accès. Le principe est tout à fait simple :
* Insérer son Email ainsi que son mot de passe après avoir créé son compte pour pouvoir notamment faire partie des membres de chez BibYaso.

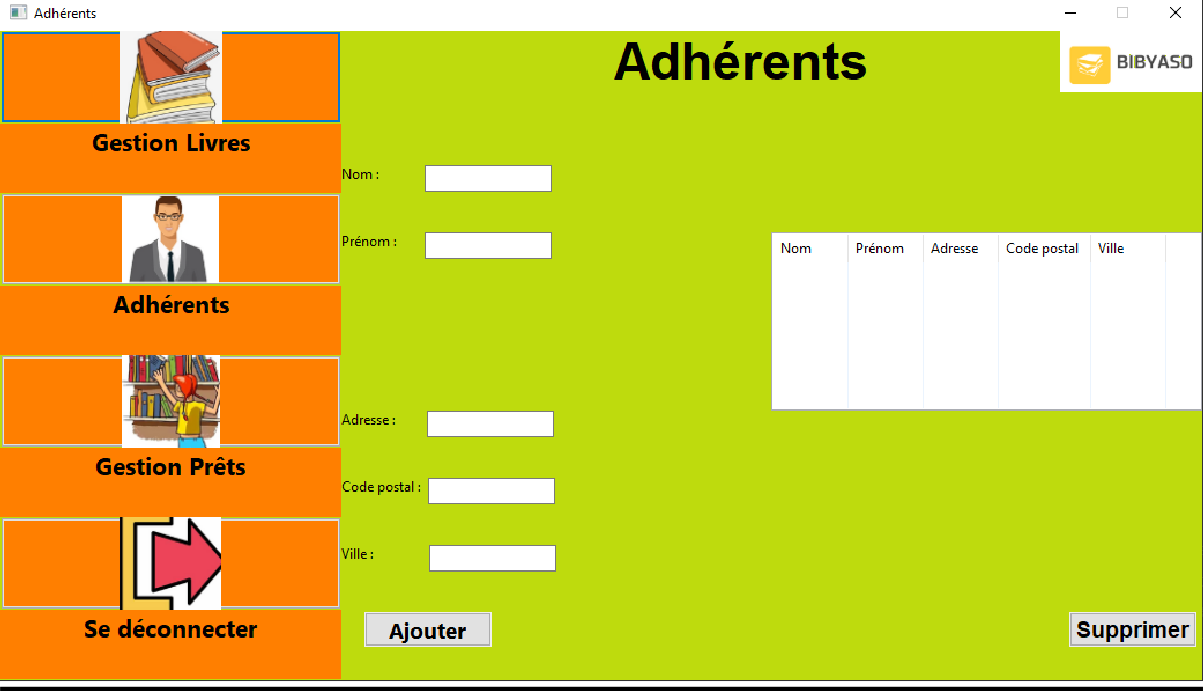
### *Page gestion livres :*



#### *Description :*

* Bienvenue dans la page Gestion livres (la page principale), qui, une fois la connexion de l’utilisateur à l’application avec l’aide de son E-mail ainsi que de son mot de passe sur la page de connexion a été établi avec succès, celle-ci s’affiche directement.
* Elle permet d’ajouter un livre en ligne chez BibYaso pour qu'il puisse être visible auprès des autres utilisateurs qui consulte l’appli, ce qui permettra d’effectuer un emprunt pour un adhérent de notre bibliothèque.

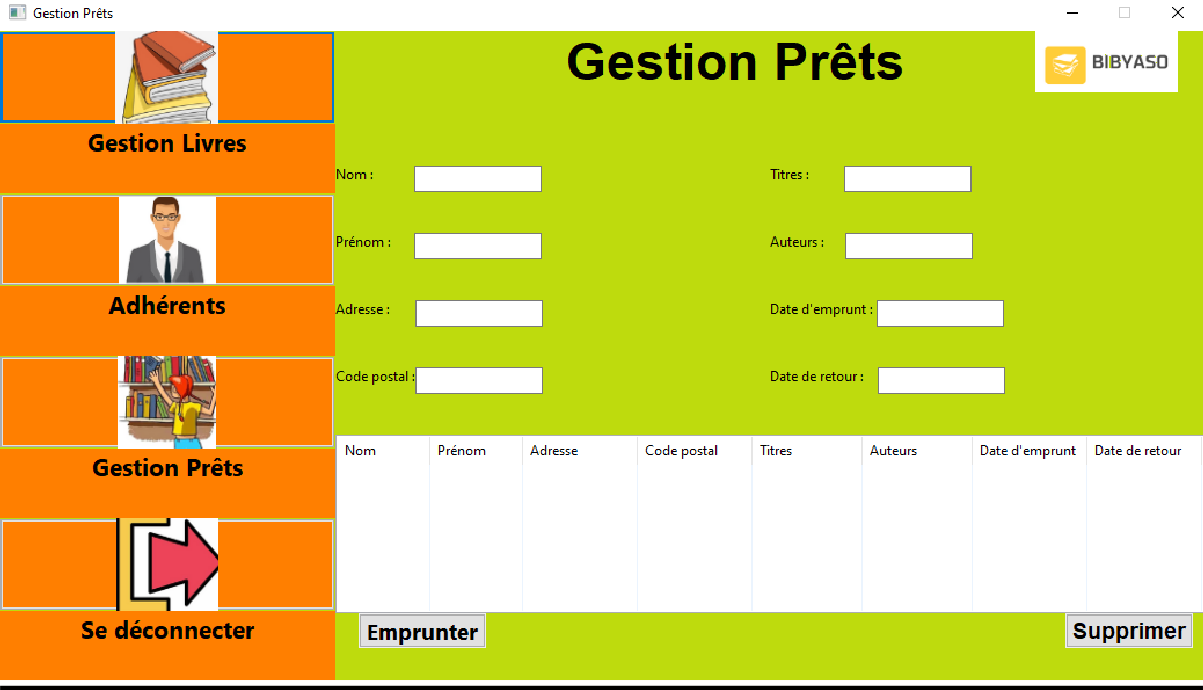
### *Page Adhérents :*



#### *Description :*

* Ensuite, voici la page Adhérents qui permet de gérer l’emprunt des livres en ayant accès à la liste des adhérents de l’application (utilisateur BibYaso qui emprunte un livre).L’adhérents s’affiche sur le tableau lorsque on insère ses coordonnées. Cette page permet donc de référencer des adhérents pour l’emprunt de Livre.

### *Page Gestion Prêts :*

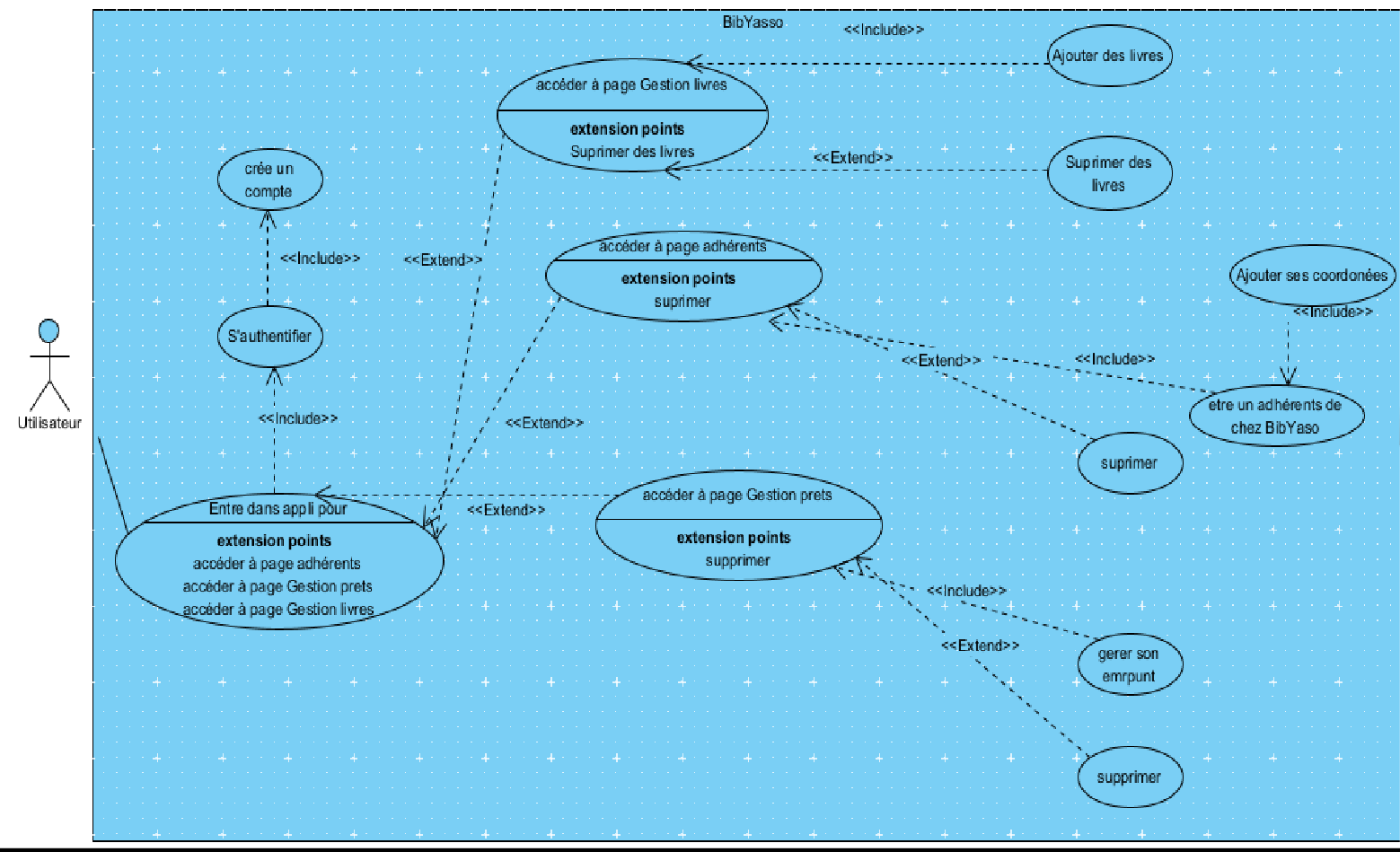


#### *Description :*

* Enfin, la page de prêts quant à elle s’occupe de la gestion de l’emprunt des livres que l’utilisateur choisi. Celle-ci montre notamment la liste des livres empruntés avec l’adhérent concernée avec la date d’emprunt et la date de retour sur le tableau lorsque on insere les informations requis dans les cases.

### *Diagrammes UML :*

Diagramme Use Case



#### *Pre-requis :*

* Pour pouvoir réaliser le graphe UML de notre application, nous avons commencé par installer le logiciel Visual paradigm, un outil UML CASE.

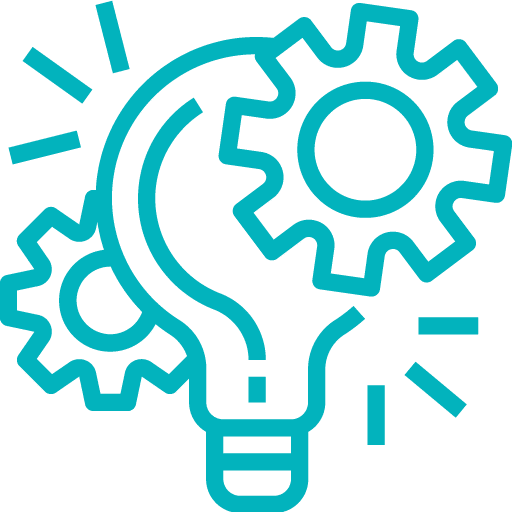
#### *Description :*

* Ci-dessus figure une capture d’écran qui caractérise le diagramme de cas d’utilisation, plus précisément le diagramme UML (use case) de notre application. Celui-ci est utilisés pour pouvoir visualiser le comportement fonctionnel d’un système logiciel en général.

#### *Traitement :*

* Lorsque l’acteur, plus précisément l’utilisateur externe rentre dans l’application BibYaso (relation entre actor et use case”Entre dans appli pour”), il doit impérativement(”includ”) créé un compte (use case «crée un compte”) et par la suite obligatoirement(”includ”)se connecter(use case”s’authentifier”).
* Ensuite, après avoir effectué l’authentification à l’aide de son E-mail et mot de passe, il a accès a l’appli avec ses trois différents pages. Celui-ci a le choix, c’est bien ce que nous montre les extend qui indiquent qu’il peut en sélectionner une d’entre elles a chaque fois puis changer (use case ”accéder a page adhérents, Gestion livres, Gestion prêts”).
* Enfin, dans chaque pages figure deux fonctionnalités qui permettent d’effectuer obligatoirement un ajout par exemple si l’utilisateur le souhaite ou bien supprimer son ajout si le désire. (use case ”Accéder a page Gestion livres”include(obliger)->”Ajout des livres ”extend(choix)->”supprimer des livres”.

**1.6 Conception**



* **Bienvenue à la fabrication de notre application intitulé BibYaSo,spécialement dédié à la bibliothèque BibYaSo. Celle-ci permet la lecture à chacun en empruntant des livres ainsi que toutes autres fonctionnalités.**
* **Nous allons vous présentez nos propositions destiné a notre client qui souhaiterai une application simple et efficace à l’utilisation.**

- Pour concevoir BibYaSo,mon camarade ainsi que moi avons utiliser plusieurs outils pour chaque parti abordées(conception,base de donnés,langage de programmation ).

* - Le langage de programmation de notre application BibYaso choisi est le langage python.Pour ce faire,nous nous sommes appuyé sur l’outil wxPython pour pouvoir crée et générer les fenêtres de l’application ainsi que l’outil Visual Studio Code qui permet de coder celle-ci.

### *Lancement de l'application:*

***-1ère étape:***

* Récupérer le code de l'application puis le copier-coller sur l'outil Visual studio code. Ensuite, une fois sur VSD, tester le code en fessant ctrl f5 puis une fois le contenu de l'appli afficher, cliquez sur le bouton créé un compte pour pouvoir s'authentifier. Vous allez maintenant créer un compte en insérant les coordonnées demandées, Validez puis re basculer sur la page de connexion pour pouvoir se connecter avec l'E-mail et MDP mis durant la création du compte
* Pour consulter la base de données de l'application, il faut utiliser l'outil WampServer et faire un clique-droit sur son icone, une fois que celle-ci apparaitra en couleur verte. Ensuite, cliquer sur phpMyAdmin puis se connecter avec le MDP root
* *Lien sql:(pour page de connexion)*

[http://localhost/phpmyadmin/sql.php?server=1&db=enregistrer](http://localhost/phpmyadmin/sql.php?server=1&db=enregistrer&table=com&pos=0) [&table=com&pos=0](http://localhost/phpmyadmin/sql.php?server=1&db=enregistrer&table=com&pos=0)

Ou crée une table “com” avec phpmyadmin avec

comme ligne de

table(Nom,Prenom,Email,Telephone,MotDepasse,Confirmeemotdepasse)

*-Lien sql:(pour les pages)* [http://localhost/phpmyadmin/sql.php?server=1&db=enregistrer&tabl](http://localhost/phpmyadmin/sql.php?server=1&db=enregistrer&table=gestion&pos=0) [e=gestion&pos=0](http://localhost/phpmyadmin/sql.php?server=1&db=enregistrer&table=gestion&pos=0)

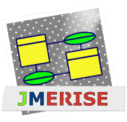
Ou crée une table “com” avec phpmyadmin avec comme ligne de table(Nom,Prenom,Adresse,Codepostal,Titre,Auteurs,Editeurs)

-Enfin, une fois arriver sur phpMyAdmin, crée une table possédant le même nom que celui de la table sur le code de la partie import SQL puis insérer les même lignes de tables

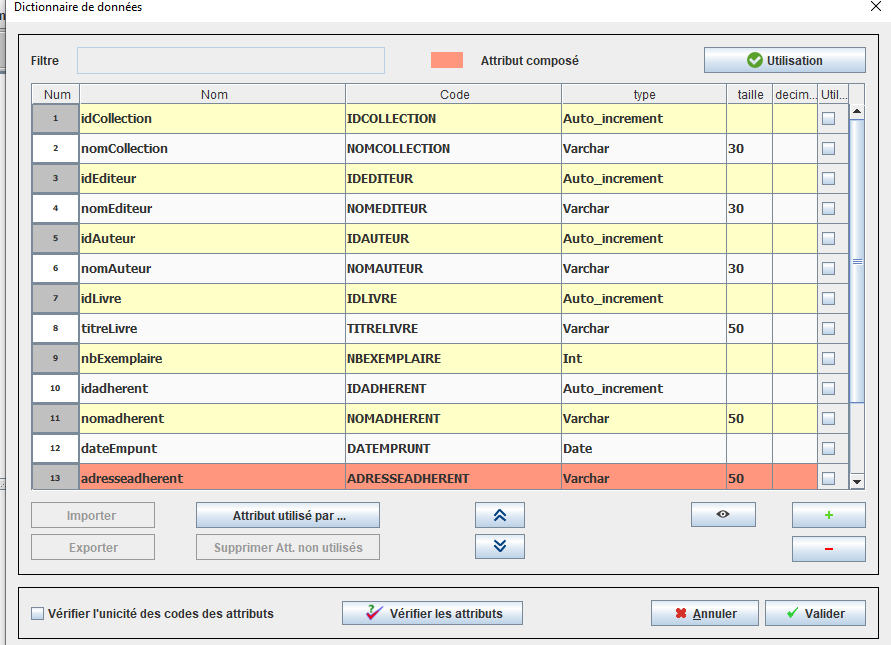
*-A ne pas oublier:*

*Dans les partis sql,changer le numéro de port(ligne 427,752) en fonction de la machine pour pouvoir lié l’application a la base de données.*

### *Conception du diagramme Merise*



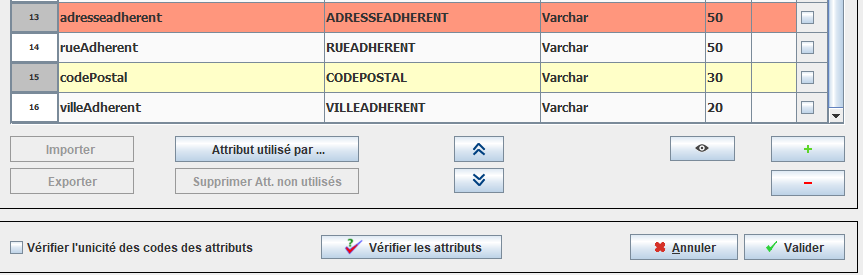
-Cette image est une capture d’écran effectuer sur l’outil Merise concerne le dictionnaire de données,la 1ère étape de notre conception sur Merise.



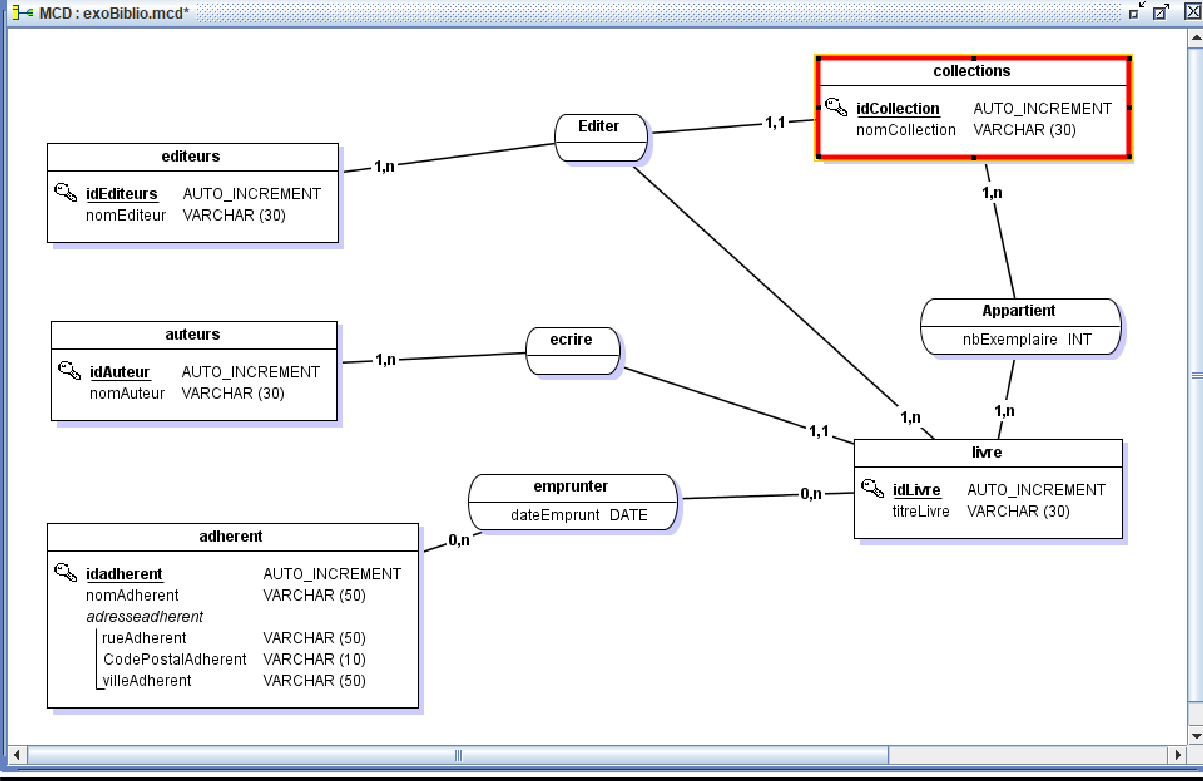
#### *Description :*

-Ici, le dictionnaire de données regroupe toutes les donées qu’on souhaite conserver dans notre base pour pouvoir les inclure dans le MCD.

***-Capture d’écran 2(dictionnaire de données):***

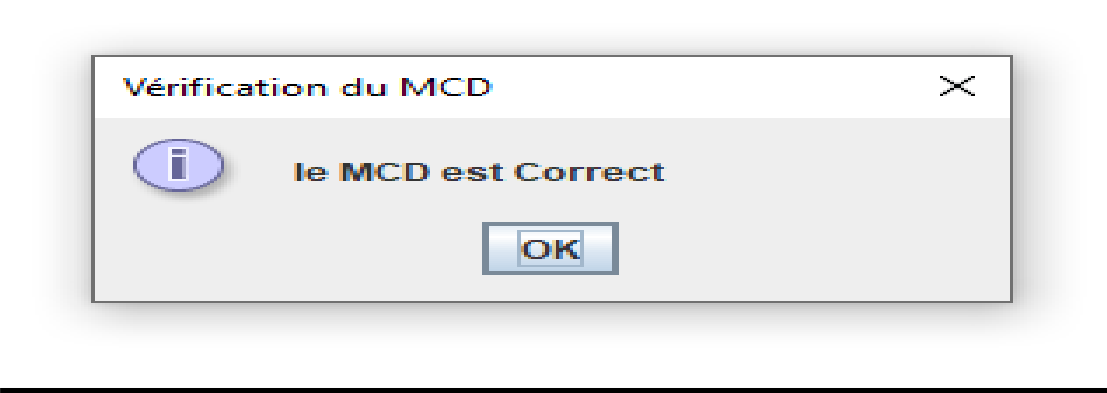


***-Capture d’écran 3(MCD):***



#### *Description :*

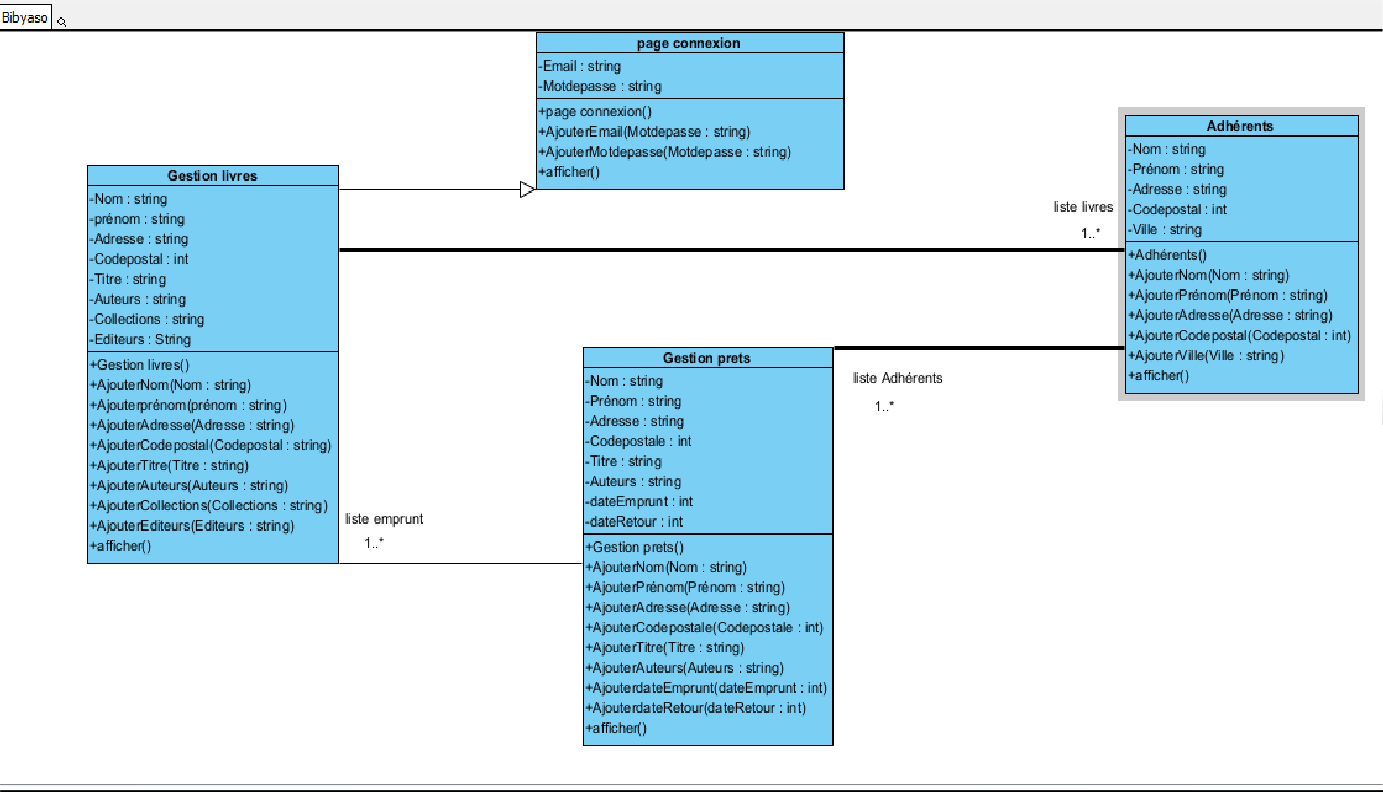
* Ici,nous avons pu réaliser la parti MCD du graphe de dépendance fonctionnel (GDF) dont le but est d’écrire de façon formelle les données qui seront utilisés par notre système d’information.



- La capture ci-dessus nous montre la vérification et validation du MCD qui est correct.

### *Diagrammes UML :*

Diagramme de classe :



-Sur la capture ci-dessus figure le diagramme de classe de l’application.Celui-ci est reparti avec différentes classes qui

correspond aux 4 pages de Bibyaso.

-En effet, on peut retrouver à la fois les attributs qui correspondent aux fonctionnalités de la page(insertion des coordonnées) dans la 1ère parti d’une classe ainsi que les méthodes(reprise des attributs

dédié au code) dans la 2ème parti.

-Enfin,figure des relations(association ci-dessus) qui consiste a reliés les classes avec notamment des cardinalités qui signifie que la

fonctionnalités de l’appli peut se faire par un ou plusieurs utilisateurs.