

Contexte Bibliothèque**Mission n°2 – Modification site WEB -
Enregistrement des livres dans bdd.**

Propriétés	Description
Intitulé	Modification du site de la bibliothèque.
Présentation Rapide	Les livres doivent être enregistrés dans une bdd.
Evaluations	A chaque fin de sprint : sérieux dans la réalisation – travail fourni Individuelles selon le travail réalisé dans le groupe . <i>GIT, Trello et feuille des durées sont donc primordiaux</i>
A rendre pour	Sprint1 : Mardi 12 mars 2024 15h00 (6 séances + vacances) Point notation pendant les vacances Sprint2 : Mardi 26 mars 2024 15h00 (4 séances) Sprint3 : Mardi 2 avril 2024 15h00 (2 séances) Sprint4 : Mardi 9 avril 2024 15h00 (2 séances)
Équipes	Groupes identiques aux missions précédentes sauf cas particulier

CAHIER DES CHARGES**Définition du besoin**

Afin de simplifier la mise à jour du site par les employés de la bibliothèque, il a été décidé d'enregistrer les livres dans une bdd MySQL.

Les fonctionnalités actuelles du site doivent persister.

Le site.

Il doit donc être modifié afin d'ajouter des nouvelles fonctionnalités:

- Un back office qui permettra à un utilisateur authentifié d'ajouter de nouveaux livres, de modifier un livre existant, de supprimer un livre.
- Un front office qui permettra :
 - les fonctionnalités actuelles (affichage des livres avec pour chacun les informations telles que définies dans la 1ère mission)
 - la recherche multi-critères de livres.
- Il devra être mis en production sur une machine virtuelle linux.

Contraintes générales.

- Temps de réponse d'affichage des pages (Attention à la taille des images).
- Site responsive
- Utilisation d'une méthode Agile.
- Travailler en collaboration en utilisant GIT, Trello sans oublier une feuille répertoriant les durées passées.

Pour le GIT, faire 2 dossiers : doc et appli

- Les différents critères pour la recherche multi-critères (aucun, un, plusieurs ou tous les critères pour la recherche) :
 - genre du livre (roman, BD ...)
 - titre du livre
 - auteur(s) du livre
 - date de sortie du livre
 - la cotation

La mission.

- Modifier le site.
- Modifier la documentation technique du site en précisant :
 - Le schéma de la bdd
 - La structure des pages du site.
 - Nomenclature des noms de fichiers, fonctions avec leur signature etc...
 - Le rôle de chaque fichier.

Déroulement de la mission. Méthode Agile ⇒ Des Sprints

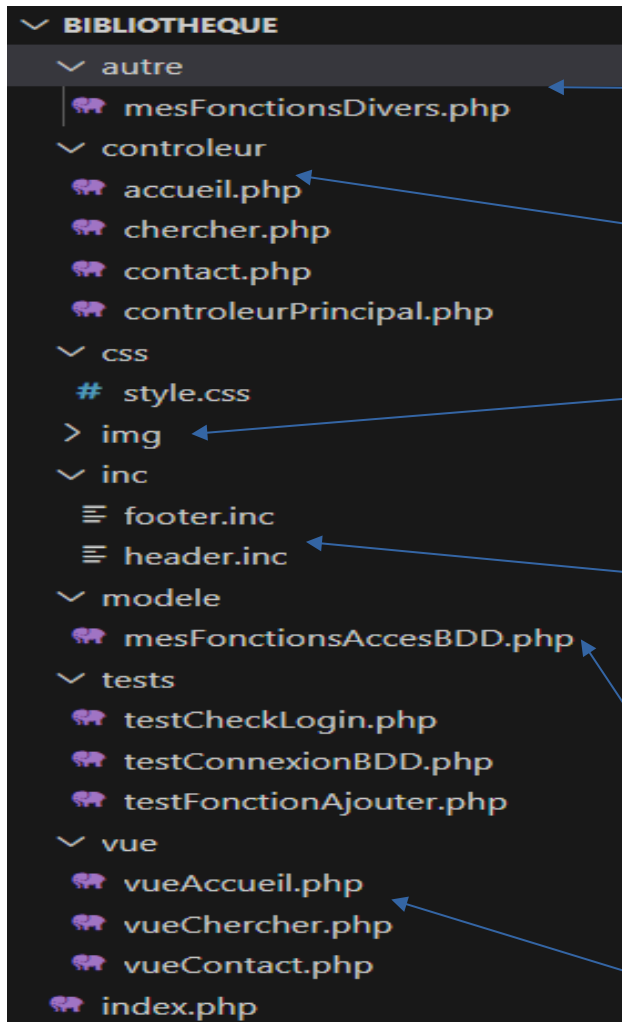
- Sur le GIT :
 - Faire une branche spécifique pour chaque sprint (sprint1 – sprint2 – ...)
 - Travailler sur cette branche pendant le développement du sprint.
 - Fusionner (merge) avec la branche principale en fin de sprint.
- Trello : 3 Colonnes : TO DO DOING DONE
 - Placer toutes les Users Stories dans la colonne 'TO DO'
- Chaque sprint démarre par un 'planning meeting' pour vous répartir les users stories du sprint :
 - Vous notez le nom du responsable de la User Story sur celle-ci
 - Vous déplacez la User Story dans la colonne DOING
- Chaque user story doit être découpée en différentes tâches. Ce sont ces tâches qui seront renseignées sur **Trello**.
Par exemple, structure à adapter à chaque user story :
 - Création / modification de la base de données
 - Création fonction(s) d'accès aux données
 - Script de test par fonction d'accès aux données
 - Création/modification de l'interface graphique (HTML/CSS)
 - Intégration interface graphique / appel fonctions accès aux données
 - Intégration au site
- Lorsqu'une User Story est terminée, vous la déplacez dans la colonne DONE.
- Chaque début de séance : scrum daily (ce qui a été fait – ce que vous allez faire – les pbs rencontrés : à noter au tableau)

Eléments à actualiser tout au long du projet.

- Le site modifié sur GIT, le script de la BDD sur GIT.
- La documentation technique modifiée avec l'adresse du lien git. (format pdf avec mise en page identique à la mission du 1^{er} semestre) sur le ftp.
- Trello et la feuille des durées.

Contraintes de développement :

- Un script de test par fonction d'accès aux données.
- Modèle d'arborescence à respecter :



- Les fonctions PHP sans SQL, de contrôle de saisie par exemple, doivent se trouver dans un autre script, dans le dossier 'autre'
- Tout ce qui est logique de traitement doit se trouver dans des fichiers qui se situent dans le dossier controleur.
- Toutes les images des livres doivent se trouver dans le dossier img. (Choisir une nomenclature de nommage pour les fichiers images)
- Tout ce qui est répété sur les pages doit être mis dans des fichiers à part qui seront 'include' (en-tête, pied etc ...), dans un dossier inc
- Toutes les fonctions d'accès aux données doivent se trouver dans le dossier modele dans un ou plusieurs fichiers (vous pouvez choisir de faire un fichier avec toutes les fonctions qui concernent l'accès à une même table)
- Tout ce qui sera affiché à l'écran doit être dans un fichier 'vue'

Contraintes de sécurité :

- Tous les contrôles de saisie doivent être faits en PHP.
- Utiliser un utilisateur MySQL qui n'a que des droits sur la base de données biblio et utiliser cet utilisateur dans votre script.
- Empreintes stockées dans la base de données (et pas le mot de passe en clair) => faire un script addUser dans le dossier test (qui ne sera pas utilisé dans le site)
- Eviter les injections SQL et Javascript =>
 - Utilisation de la bibliothèque PDO
 - Utilisation fonction htmlspecialchars en insert et en affichage
 - Contrôle sur la longueur des chaînes (doit correspondre aux tailles des champs dans la bdd)
 - Contrôle des types des données saisies.

Projet agence Biblio

Product backlog (Liste des users stories)

sprint backlog 1

un bibliothécaire doit pouvoir se connecter et voir « son » menu

un internaute doit pouvoir visualiser la liste des livres (ref – titre – auteur – 50 premiers caractères du résumé)

un internaute doit pouvoir rechercher et visualiser un ou des livre(s) sur 1 ou 2 critères : titre et genre

sprint backlog 2

un bibliothécaire doit pouvoir ajouter un livre.

un internaute doit pouvoir visualiser le détail d'un livre.

un internaute doit pouvoir rechercher un livre sur 0 à 4 critères : titre, genre, un auteur, année de sortie.

sprint backlog 3

un bibliothécaire doit pouvoir modifier les informations d'un livre existant.

un internaute doit pouvoir afficher le pdf d'un livre.

un internaute doit pouvoir rechercher un livre sur 0 à 5 critères : titre, genre, auteur, année de sortie, cotation

sprint backlog 4

un bibliothécaire doit pouvoir supprimer un livre existant

un internaute doit pouvoir trier la liste des livres (ref – titre – auteur – 50 premiers caractères du résumé) selon la ref, le titre ou l'auteur.

un internaute doit pouvoir rechercher un livre sur sa référence.

	Back office	Front Office :Visu	Front Office :Recherche
Sprint 1	un bibliothécaire doit pouvoir se connecter et voir « son » menu	un internaute doit pouvoir visualiser la liste des livres (ref – titre – auteur – 50 premiers caractères du résumé)	un internaute doit pouvoir rechercher un livre sur 1 ou 2 critères : titre et genre
Sprint 2	un bibliothécaire doit pouvoir ajouter un livre.	un internaute doit pouvoir visualiser le détail d'un livre.	un internaute doit pouvoir rechercher un livre sur 0 ou 4 critères : titre, genre, un auteur, année de sortie.
Sprint 3	un bibliothécaire doit pouvoir modifier les informations d'un livre existant.	un internaute doit pouvoir afficher le pdf d'un livre.	un internaute doit pouvoir rechercher un livre sur 0 ou 5 critères : titre, genre, auteur, année de sortie, cotation
Sprint 4	un bibliothécaire doit pouvoir supprimer un livre existant	un internaute doit pouvoir trier la liste des livres (ref – titre – auteur – résumé) selon la ref, le titre ou l'auteur.	un internaute doit pouvoir rechercher un livre sur sa référence.

Vous devrez prendre une User Story dans chaque colonne lors des différents sprints.

Vous changez de colonne à chaque sprint :

ce qui vous oblige à modifier un code non créé par vous !