Site web - Livres

(Une image originale représentant le projet)

Julien Leresche et Simon Guggisberg

ETML

Table des matières

[1 Introduction 3](#_Toc58566886)

[2 Analyse 3](#_Toc58566887)

[2.1 HTML 3](#_Toc58566888)

[2.2 MCD/MLD/MPD 3](#_Toc58566889)

[2.3 UML 3](#_Toc58566890)

[3 Réalisation 3](#_Toc58566891)

[3.1 Algorithme login 3](#_Toc58566892)

[3.2 Appréciation moyenne 3](#_Toc58566893)

[3.3 Explication écoconception 3](#_Toc58566894)

[4 Tests 3](#_Toc58566895)

[5 Conclusion 3](#_Toc58566896)

[5.1 Générale 3](#_Toc58566897)

[5.2 Personnelle 3](#_Toc58566898)

[5.3 Critique méthode projet 3](#_Toc58566899)

[6 Divers 3](#_Toc58566900)

[6.1 Bibliographie 3](#_Toc58566901)

[6.2 Webographie 3](#_Toc58566902)

[6.3 Glossaire 3](#_Toc58566903)

# Introduction

Le but de ce projet étant de réaliser un site web dynamique, permettant de se login, d’ajouter des éléments à celui-ci et de voir ces éléments se retrouver sur le site.

# Analyse

## HTML

Après lecture du cahier des charges, nous avons décidé qu’une page principale, celle d’accueil, devrait être celle n’affichant que 5 premiers livres, une mise en bouche, si vous préférez.

La seconde page serait celle contenant tous les livres, mais plutôt similaire à celle d’accueil sinon.

En cliquant sur un livre nous aboutirions alors à la page de détails dudit livre, mais, en découvrant le framework Bootstrap, nous avons découvert un outil, le modal, permettant d’afficher ces détails de manière bien plus élégante qu’une page à part, cela permettrait de rester sur la page principale/liste de livres et de consulter les livres, avec, certes, un coût de calcul pour le serveur plus important, mais cela ne consultant que du texte essentiellement, cela ne nous a pas posé problème.

Reste encore la page d’ajout de livre, accessible uniquement une fois connecté, via le header de la page principale.

La page permettant d’afficher la liste de livres ajoutés par chaque utilisateur, respectivement. Ainsi, l’utilisateur « Paul » aura sa propre page, dédiée, permettant de visionner les livres qu’il a ajouté à la base de données.

De la même manière, en inspectant les détails d’un livre, il est possible de voir les évaluations dudit livre, sur une page dédiée, encore une fois. Là, il est possible de voir les évaluations individuelles ajoutées par les utilisateurs, et d’en écrire une nouvelle, en indiquant le score que l’on souhaite attribuer.

## MCD/MLD/MPD

## UML

# Réalisation

## Algorithme login

## Appréciation moyenne

## Explication écoconception

Pour l’écoconception, nous avons décidé de lister les différentes choses que l’on pouvait faire dans ce sens au début du projet, puis les implémenter au fur et à mesure, ou orienter le projet dans ce sens si cela relève une plus grande organisation.

En effet, les points varient, allant de « Minification des fichiers » à « utilisation d’un framework ». Autant le premier est trivial, et ne demande qu’un peu de temps alloué sur la fin du projet, autant le dernier nécessite du travail en amont et une vision d’ensemble.

Je vais donc lister les différents points que nous avons relevé et ce qui a été fait dans ce sens pour chacun de ceux-ci :

* Minification des fichiers :
* Compression des médias utilisés :
* Images sprite :
* Utilisation d’un framework :
* Limitation du nombre de requêtes http :
* Stockage en local des données statiques :
* Design simple et épuré : Puisque nous utilisons Bootstrap comme base, cela nous donne une bonne ligne directrice en terme de design limpide et simple.
* Élimination des fonctionnalités inutiles :
* Réduction du nombre de vidéos et des animations : Puisqu’aucune vidéo ou animation n’est présente sur le site, à l’exception de quelques effets en Javascript via Bootstrap, Nous considérons ce point comme aisément acquis.
* Limiter le recours aux plugins :

# Tests

# Conclusion

## Générale

## Personnelle

## Critique méthode de gestion de projet

Nous avons décidé d’utiliser une méthode inspirée de Kanban pour suivre la quantité de travails à effectuer, la répartition de celui-ci, et enfin l’évolution au cours du temps du projet.

Comme nous utilisions Gitkraken pour tout ce qui est gestionnaire de versions, nous avons également décidé d’utiliser l’outil directement inclus dans Gitkraken, à savoir, Boards. Sorte de Trello, avec des features en moins, d’autres en plus pensées spécialement pour tout le développement Git.

Nous avons décidé d’un code couleur pour les cartes : rouge, ce qui est nécessaire, orange, ce qui est un bug, bleu, ce qui relève de la documentation, et vert, ce qui est facultatif/serait bien à avoir.

En ayant 4 colonnes, To Do, In Progress, Testing, Done, Nous avons largement su gérer les objectifs par priorité. Mais observer l’évolution du projet sur le temps est plus ardu, certaines fonctionnalités comme des Story Points ou une Burndown Chart manquant à l’outil Boards.

Reste que cette méthode de gestion de projet s’est révélée assez dynamique et adaptée pour un projet de cette ampleur, constitué uniquement de deux personnes et sur une période de moins de six mois.

# Divers

## Bibliographie

## Webographie

## Glossaire