P\_Web294



Amir Cyril Mateja Yohan – CID2A

Vennes

24 périodes

G. Charmier

ETML

Table des matières

[1 Spécifications 3](#_Toc308526316)

[1.1 Titre 3](#_Toc308526317)

[1.2 Description 3](#_Toc308526318)

[1.3 Matériel et logiciels à disposition 3](#_Toc308526319)

[1.4 Prérequis 3](#_Toc308526320)

[1.5 Cahier des charges 3](#_Toc308526321)

[1.5.1 Objectifs et portée du projet (objectifs SMART) 3](#_Toc308526322)

[1.5.2 Caractéristiques des utilisateurs et impacts 3](#_Toc308526323)

[1.5.3 Fonctionnalités requises (du point de vue de l’utilisateur) 3](#_Toc308526324)

[1.5.4 Contraintes 3](#_Toc308526325)

[1.5.5 Travail à réaliser par l'apprenti 4](#_Toc308526326)

[1.5.6 Si le temps le permet … 4](#_Toc308526327)

[1.5.7 Méthodes de validation des solutions 4](#_Toc308526328)

[1.6 Les points suivants seront évalués 4](#_Toc308526329)

[1.7 Validation et conditions de réussite 4](#_Toc308526330)

[2 Planification Initiale 4](#_Toc308526331)

[3 Analyse 4](#_Toc308526332)

[3.1 Opportunités 4](#_Toc308526333)

[3.2 Document d’analyse et conception 4](#_Toc308526334)

[3.3 Conception des tests 5](#_Toc308526335)

[3.4 Planification détaillée 5](#_Toc308526336)

[4 Réalisation 5](#_Toc308526337)

[4.1 Dossier de Réalisation 5](#_Toc308526338)

[4.2 Modifications 5](#_Toc308526339)

[5 Tests 5](#_Toc308526340)

[5.1 Dossier des tests 5](#_Toc308526341)

[6 Conclusion 5](#_Toc308526342)

[6.1 Bilan des fonctionnalités demandées 5](#_Toc308526343)

[6.2 Bilan de la planification 5](#_Toc308526344)

[6.3 Bilan personnel 5](#_Toc308526345)

[7 Divers 6](#_Toc308526346)

[7.1 Journal de travail 6](#_Toc308526347)

[7.2 Bibliographie 6](#_Toc308526348)

[7.3 Webographie 6](#_Toc308526349)

[8 Annexes 6](#_Toc308526350)

# Spécifications

## Titre

Passion lecture

## Description

Réaliser le frontend d’une application permettant de partager sa passion pour la lecture

## Matériel et logiciels à disposition

Microsoft Windows 10 • vscode • Serveur local • Navigateurs Web • Accès à Internet

## Prérequis

Modules 106, 162, 164, 231, 293, 295, 319, 320, 322 et 426

## Les points suivants seront évalués

* Le rapport
* Le code et les commentaires

# iNTRODUCTION

Ce projet consiste à créer un application web vue.js sur la base d’un backend d’un précédent projet ou d’un backend donné par le chef de projet (notre professeur).

C’est un projet qui se fait par groupe de 4 ou 5 personnes.

Maintenant pour être plus précis le résultat final du projet doit être un site web qui contient c’est fonctionnalité :

* Une page d’accueil comprenant une explication de l’utilité du site ainsi que les cinq derniers ouvrages ajoutés (accès tout public).
* Une page comprenant la liste des ouvrages par catégorie (accès tout public avec restrictions sur les liens).
* Une page d’ajout d’un ouvrage (accès utilisateur).
* Une page de modification d’un ouvrage (accès utilisateur pour ses ouvrages)
* Une page de suppression d’un ouvrage (accès utilisateur pour ses ouvrages)
* Une page (vue détail d’un livre) permettant d’ajouter une appréciation et un commentaire à un ouvrage (accès utilisateur).
* L’utilisateur admin peut réaliser toutes les actions

Pour faire ce projet nous disposons de 32 périodes

# 

# Analyse

## Planification(trello)

Pour la planification des taches nous avons utilisé l’outil Trello

Disponible avec ce lien :

<https://trello.com/invite/b/PvSiD7YC/ATTI5990fb5a2cb3c2d0d799ddc14a297e02902803DC/pweb294>

Nous avons approché assez simple de l’attribution des taches, nous les avons simplement toutes listé dans une liste “à faire” puis chacun prend une tâche pour la faire un fois terminer il l’a déplacée dans terminer puis il en reprend une et ainsi de suite.

Afin que chacun puisse être noté sur ce qu’il a fait, en description de la tache terminée, l’élève écrit tout ce qu’il a lors de cette ainsi que la date ou il l’aura terminée et commencé si possible.

# Réalisation

## Github

La méthode Trunk-Based Développement, que nous avons choisie pour notre projet en équipe, privilégie l'intégration continue des changements sur la branche principale.

Contrairement aux méthodes basées sur des branches prolongées, cela favorise des commits fréquents directement sur le tronc principal (main), facilitant la détection précoce des conflits et des problèmes d'intégration.

Cette approche réduit les risques de divergences majeures entre les codes des membres de l'équipe, encourageant une collaboration étroite et une meilleure compréhension du code global.

De plus, en minimisant la durée de vie des branches, le Trunk-Based développement permet une livraison plus rapide des fonctionnalités, contribuant ainsi à maintenir un cycle de développement agile et réactif aux besoins changeants du projet.

Adopter cette méthode témoigne de notre engagement envers des pratiques de développement efficaces, axées sur la collaboration et la livraison continue.

## Durabilité

Pour améliorer la durabilité, après une évaluation à 80%, le chef de projet nous a demandé de mettre en place le lazy loading pour toutes les pages afin d'utiliser moins de ressources.

Mais avant tout, c'est quoi le lazy loading ?

Le lazy loading consiste à différer le chargement de certains éléments d'une page web jusqu'à ce qu'ils soient réellement nécessaires. Grâce à un lazy loading bien implémenté, sur un site avec un grand nombre d'utilisations par jour, les ressources économisées peuvent être très importantes.

# TEST

# Conclusion

## Bilan des fonctionnalités demandées

Il s’agit de reprendre point par point les fonctionnalités décrites dans les spécifications de départ et de définir si elles sont atteintes ou pas, et pourquoi.

Si ce n’est pas le cas, estimer en « % » ou en « temps supplémentaire » le travail qu’il reste à accomplir pour terminer le tout.

## Bilan de la planification

Distinguer et expliquer les tâches qui ont généré des retards ou de l'avance dans la gestion du projet. Indiquer les différences entre les planifications initiales et détaillées avec le journal de travail.

## Bilan personnel

### Amir

### Cyril

### Mateja

### Yohan