

1 INFORMATIONS GENERALES

| | | |
|------------------------|---|-----------|
| Apprenti : | Nom: | Prénom : |
| Lieu de travail : | ETML / SEBEILLON 12 / 1004 LAUSANNE | |
| Chef de projet : | Nom: | Prénom: - |
| | Du lundi ?? . ?? .2024 au vendredi ?? . ?? .2024 | |
| Dates de réalisation : | ? périodes par semaine, ? semaines (moins les jours fériés !) | |
| Horaire de travail : | 24 périodes environ | |
| Temps total : | | |

2 PROCÉDURE

- Tous les apprentis réalisent le projet sur la base d'un cahier des charges.
- Le cahier des charges est présenté, commenté et discuté.
- Les apprentis sont entièrement responsables de la sécurité de leurs données.
- En cas de problèmes graves, les apprentis avertissent leur chef de projet au plus vite.
- Les apprentis ont la possibilité d'obtenir de l'aide externe, mais ils doivent le mentionner.
- Une auto-évaluation est demandée à chaque apprenti.
- Ce projet est réalisé par des groupes de 4 apprentis.
- En cas d'imprévu, le temps total doit être géré par l'apprenti.
- Chaque apprenti travaille sur tous les aspects du projet (code, documentation, tests, etc)

3 TITRE

Passion lecture

4 SUJET

Réaliser le backend d'une application permettant de partager sa passion pour la lecture.

5 MATÉRIEL ET LOGICIEL À DISPOSITION

- Microsoft Windows 7
- vscode
- Serveur local
- Navigateurs Web
- Accès à Internet

6 PRÉREQUIS

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

7 CAHIER DES CHARGES

7.1 Objectif et portée du projet (objectifs SMART)

Ce projet vise à mettre en œuvre les connaissances apprises dans le module 295.

Au final, l'application réalisée devra être exploitable et livrable.

Dès lors, on attend un rendu professionnel et un soin particulier dans la documentation du projet.

7.2 Maquette

Ce projet consiste à réaliser uniquement le backend de l'application. Néanmoins, pour avoir une vision complète du projet, vous devez déjà réfléchir à la maquette de votre site. Cela vous aidera à développer une API REST complète.

7.3 Ecoconception Web

Votre application comprendra au moins un point d'écoconception Web. Il devra être présenté dans le rapport.

7.4 Fonctionnalités requises (techniques)

L'application backend consiste à la mise en place d'une API REST permettant à un consommateur de développer le frontend de l'application.

Cette API REST doit être la plus complète possible avec toutes les notions vues en cours, à savoir :

- Un ensemble de routes permettant la gestion des livres, des catégories, des utilisateurs, etc
- Une validation de toutes les données fournies par le consommateur de l'API
- Une gestion des statuts http (200, 3xx, 4xx, 5xx) et des erreurs
- Une recherche sur les livres, catégories, etc
- Un système d'authentification basé sur les jetons JWT
- Une documentation Swagger la plus complète possible
- Des tests de votre API avec Insomnia ou Postman
- Des tests automatisés avec vitest
- Une intégration continue dans github actions
- Une « dockerisation » du backend

7.5 Fonctionnalités requises (d'un point de vue du client)

Le frontend de l'application devra avoir :

- 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

PS : le pied-de page du site doit faire mention de la personne qui a créé l'application ainsi que le moyen de la contacter.

7.6 Détails des vues

Ce site web permet de visualiser la liste des ouvrages stockés dans une base de données, de pouvoir en ajouter, modifier et en supprimer.

Un ouvrage est représenté par diverses informations, notamment :

- Un titre
- Une catégorie (bande dessinée, manga, roman, livre etc.)
- Un nombre de pages
- Un extrait (lien relatif vers un fichier pdf d'une page de l'ouvrage)
- Un résumé de l'ouvrage
- Les nom et prénom de l'écrivain qui est l'auteur de l'ouvrage
- Le nom de l'éditeur
- L'année d'édition
- La moyenne des appréciations des utilisateurs
- Les commentaires
- Une image de couverture

Un utilisateur est défini par un pseudo, une date d'entrée dans le site, du nombre d'ouvrages proposés et du nombre d'appréciations et de commentaires faits.

Une appréciation pour un ouvrage consiste à lui attribuer une note parmi la liste suivante : {0, 1, 2, 3, 4, et 5}

Vue liste

Sur la page de la liste des ouvrages, il doit être possible de visualiser le titre, l'auteur ainsi que le pseudo de la personne qui a posté l'ouvrage. Ceci doit être possible par catégorie.

Pour chaque ouvrage, il est possible en cliquant sur le titre ou le pseudo, de voir les détails de l'ouvrage ou de la personne.

Vue détail (accessible seulement pour les membres connectés ou les admin)

Pour l'ouvrage, il est possible de voir les informations, le nombre d'appréciations ainsi que la moyenne obtenue.

Pour la personne, il est possible de voir les informations, le nombre d'ouvrages proposés.

Sur la page de la liste des ouvrages, il doit être possible de visualiser le titre, l'auteur ainsi que le pseudo de la personne qui a posté l'ouvrage. Ceci doit être possible par catégorie.

7.7 Les technologies

Pour réaliser ce projet, vous devez utiliser :

- Node.js et npm
- Express.js
- L'ORM Sequelize
- MySQL
- Swagger
- Insomnia ou Postman

7.8 Travail à réaliser par les apprentis dans le cadre de ce projet

Les parties suivantes doivent figurer dans le rapport¹ du projet :

- Introduction

Comprend une brève explication du projet (½ page)²

- Analyse

Planification de toutes les tâches à réaliser : Vous devez utiliser un outil permettant au chef de projet de voir l'évolution des tâches (github projet ou trello).

Contiendra une analyse de l'API REST présentant pour chaque route :

Le verbe http, l'URI et si nécessaire le json ou autre information à transmettre (1 page)

Contiendra une analyse de la base de données à réaliser (MCD, MLD, MPD) (1 page)

Contiendra une analyse de la structure du code qui sera effectuée (Organisation du code, ...) (1 page)

¹ La documentation est écrite en police Century Gothic et avec une taille de 11. Les titres des différentes parties sont écrits en police Century Gothic et avec une taille de 14. Les sous-titres sont écrits en police Century Gothic et avec une taille de 12. Les notes de bas de page seront utilisées à bon escient. Toutes les sources seront citées.

² Le nombre de page en () est le minimum à effectuer.

Schéma de l'architecture représentant les interactions entre le frontend et le backend tout en situant les différents composants (API REST, la base de données, l'ORM, etc)

- Réalisation

Comprend une explication de l'algorithme utilisé pour gérer l'authentification et la gestion des rôles (1 page)

Comprend une explication des mesures prises pour les aspects de sécurité. (1 page)

Comprend une explication sur chaque fonctionnalité technique demandée.

- Test

Comprend une explication des tests réalisés (test automatique) (1 page)

- Conclusion

Comprend une explication de l'organisation du groupe quant à la gestion du code avec git/github.

Comprend une conclusion générale sur le projet (½ page)

Comprend une conclusion personnelle sur le projet (½ page)

Comprend une critique constructive sur la planification du projet (½ page)

- Webographie / Bibliographique / Glossaire

Concernant la méthode de projet, elle devra être annexée au rapport et sera imposée par le chef de projet.

Remarque : Le recours à des outils en ligne d'intelligence artificielle (ex. : Chat GPT) doit être mentionné et ne peut servir que d'inspiration à la réalisation. En cas d'abus, l'évaluation du projet en tiendra compte.

8 VALIDATION

| | Lu et approuvé le : | Signature : |
|------------------|---------------------|-------------|
| Candidat : | | |
| Candidat : | | |
| Candidat : | | |
| Candidat : | | |
| Chef de Projet : | | |