

Noël Jérémie

Développeur-se logiciel

Nom : Noël
Prénom : Jérémie
Date de Naissance : 30/01/1996
Adresse : 93 Bd. Du Montparnasse,
75006 Paris

Dossier de projet
Titre RNCP 5927
DEVELOPPEUR-SE LOGICIEL
NIVEAU III

Sommaire

1. <u>Remerciements</u>	Page x
2. <u>Résumé du projet</u>	Page x
3. <u>Liste des compétences du référentiel</u>	Page x
4. <u>Cahier des charges / expressions de l'application à développer</u>	Page x
▪ a) Gestion Back-End	Page x
• b) Gestion Front-end	Page x
• c) Gestion de l'api	Page x
5. <u>Spécifications fonctionnelles</u>	Page x
6. <u>Spécifications techniques</u>	Page x
7. <u>Réalisations</u>	Page x
8. <u>Conclusions</u>	Page x

1 – Remerciements

Je souhaite remercier l'entreprise Simplon.co, entreprise/école dans laquelle j'ai pu suivre une formation de 6 mois, ainsi qu'une alternance de 12 mois.

Merci à Sara, Sandro, Rodolphe, Norbert, pour leurs soutiens en étant dans le staff pédago ainsi que leur aides pendant tout ce temps.

Merci à Jerome Lavancier, Laure Arricastre et Jean-baptiste Gabin de m'avoir accueilli au sein de leur équipe à Société Générale pour mon contrat de professionnalisation.

Merci aux autres apprenants de ma promo, une entraide géniale sur l'ensemble du temps de formation.

Yoann, mon coéquipier pour ce projet afin de valider le titre professionnel.

2 – Résumé du projet

French version below

The project is called "Mangathèque", it is a responsive web application in which the user will be able to have a collection of manga, and note to which chapter it is. We can compare that to a page mark to know what page we are in a book.

He will therefore be able to have a note for all his manga, available from a public API.

For this project, we used HTML, CSS, Javascript, with framework and libraries: AngularJS. All the back-end is written with NodeJS, express, as well as MongoDB for the database. We use Git and GitHub for versionning, Trello for the team synchronisation and Balsamiq for the wireframe

Le projet s'appelle "Mangathèque", c'est une application Web responsive dans laquelle l'utilisateur va pouvoir avoir une collection de manga, et noter à quel chapitre il est. On peut comparer ça à un marque page pour savoir à quelle page nous sommes dans un livre.

Il va donc pouvoir avoir une note pour tout ses manga, disponible à partir d'une api publique.

Pour ce projet, nous avons utilisé HTML, CSS, Javascript, avec comme framework et librairies : AngularJS. Tout le back-end est écrit avec NodeJS, express, ainsi que MongoDB pour la base de données. Nous avons utilisé git et gitHub pour le versionning, Trello pour la synchronisation dans l'équipe ainsi que Balsamiq pour les maquettes / wireframes.

3 – Liste des compétences du référentiel

Pour les compétences utilisés dans le projet présenté pour le titre professionnel :

- Compétence N°1 : Développer une application simple de mobilité numérique. En effet, notre application web est disponible sur tout type d'écran.
- Compétence N°2 : Concevoir une base de données, en effet, pour notre application nous avons besoin de concevoir une base de données, ainsi que ses diagrammes.
- Compétence N°3 : Mettre en place une base de données, afin de stocker les données utilisateurs.
- Compétence N°4 : Développer des composants d'accès aux données. Dans notre projet, l'utilisateur va avoir besoin de s'inscrire ou alors de se connecter afin d'avoir accès à l'application et aux données.
- Compétence N°5 : Développer des pages web en lien avec une base de données. En effet, l'utilisateur pourra consulter ses données qui seront affichées sur des pages web.
- Compétence N°6 : Utiliser l'anglais dans son activité professionnelle en informatique.

4 - Cahier des charges / expressions de l'application à développer

Le projet de Mangathèque s'est fait dans deux grands temps, le premier qui était la gestion du Back-End, l'authentification de l'utilisateur, son accès en base de données. Dans un second temps la réalisation de la partie Front-End

a) Gestion Back-End

L'objectif dans cette partie était de créer à partir de zéro toute une partie authentification sécurisée pour les utilisateurs ainsi que la création de la base de données, et la gestion des requêtes.

L'utilisateur en lui-même aura donc la possibilité de :

- se créer un compte, allouer ses informations sur la base de données
- s'y connecter

Le réelle difficulté là-dedans était de réussir avec b-crypt et JsonWebToken à faire une authentification sécurisée, à savoir que le mot de passe stocké dans la base de donnée est crypté pour protéger le mot de passe de l'utilisateur.

La partie back-end servira notamment à :

- Consulter la liste de Manga de l'utilisateur,
- Ajouter/supprimer des Manga de sa bibliothèque via des requêtes,
- Editer le nombre de Tome lu pour chaque Manga dans la base de données
- Se déconnecter à tout moment.

b) Gestion Front-end

L'objectif dans cette partie était de créer et construire toute la partie Front-End. Avec cette partie, l'utilisateur aura accès à :

- Toute la partie visuelle du site et sur tout les supports disponibles.
- La possibilité de s'inscrire / se connecter à l'application via des composants écrits en html/css/Javascript
- Toutes les animations, systèmes de routes pour afficher tel ou tel élément.

c) Gestion de l'api

En effet, pendant notre projet, nous avons besoin de consulter une api publique pour avoir une liste très détaillée de Manga, cette api est consultable ici : <http://www.mangaeden.com/api/>

Cette api a une documentation qui était très intéressante et bien détaillée, ce qui nous a vite permis de faire quelques essais dessus et se rendre compte de son efficacité.

Celle-ci va nous servir à avoir :

- Une liste complète de Manga,
- Un synopsis, éventuellement leurs couvertures

Elle nous servira aussi éventuellement pour une future traduction en français des composants de l'api avec i18next.

5- Spécifications fonctionnelles

Pour créer les différentes fonctions nous avons travailler avec le diagramme UML afin de trouver les différents liens entre les tables et les réunir pour faire fonctionner les différentes requêtes.

[SCREEN UML]

6 – Spécifications techniques

Pour la réalisation du projet nous avons utilisé HTML/CSS, Javascript avec comme framework AngularJS

Nous avons aussi utilisé Gulp pour différentes tâches (lancement du serveur, concaténation).

Pour la gestion de la base de donnée nous avons utilisé MongoDB.

Pour l'organisation à l'intérieur du groupe et la coordination du travail nous avons utilisé Trello, nous avons fonctionné dans un principe d'agilité avec une livraison continue sur la branche de développement.

Pour le versionning et l'assemblage du code nous avons utilisé Git et GitHub.

Ci joint une capture d'écran de Trello
[SCREEN TRELLO]

Ci-joint une capture d'écran de notre repository GitHub
[Screen GitHub]

7 - Réalisations