

Université de Mons
Faculté des sciences
Département d'Informatique

Energenius

Rapport de projet - prototype

Professeur :

Tom MENS

Assistants :

Pierre HAUWEELE

Sébastien BONTE

Auteurs :

Godwill LOUHO

Gilles JAUNART

Jérémy DELNATTE

Louis DASCOTTE



Année académique 2022-2023

Table des matières

1	Introduction	2
2	Installation	2
3	Présentation	2
4	Fonctionnement	3
5	Conclusion	3

1 Introduction

Ce court rapport présentera l'état actuel du projet Energenius, ainsi que les fonctionnalités actuellement disponibles et implémentées. Il sera également présenté un petit mode d'emploi afin de permettre de tester le projet localement.

2 Installation

Pour tester le projet localement, il faut donc récupérer le code depuis le github si vous y avez accès, en exécutant la commande suivante dans le terminal :

```
git clone https://github.com/LouisDascotte/ProjetGenieLogiciel2022.git
```

Ensuite, il faudra se déplacer dans le dossier obtenu. Naviguez ensuite jusqu'aux dossiers **back-end** et **front-end**.

Dans **back-end**, il faudra exécuter la commande :

```
gradle bootRun
```

pour démarrer le serveur de l'API Rest. Pour nous, le l'API démarrait sur le port 8080. Si ce n'est pas le cas, il faudra remplacer le port dans la partie front-end pour ne pas avoir d'erreur. Une fois que l'application est lancée, il suffira de lancer l'application client.

Pour cela, il faut donc aller dans **front-end**. Assurez-vous d'avoir la dernière version de NodeJS installée sur votre machine. Exécutez la commande

```
npm install
```

pour installer les packages nécessaires au fonctionnement de l'application. Une fois ces packages installés, lancez le serveur avec :

```
npm start
```

Si tout se passe bien, vous devriez obtenir la page de connexion de l'application.

3 Présentation

L'application client est développée en React JS. Nous avons choisi cette technologie car elle est très répandue, et permet d'avoir rapidité et stabilité. Du fait de sa popularité, il est possible de trouver un grand nombre de ressources. De nombreuses librairies permettent d'accélérer le développement de l'application.

Dans le cadre de ce projet, nous utilisons la librairie MUI pour l'aspect pratique et visuel de celle ci. Cela nous permet d'ajouter facilement des composants et de les assembler les uns les autres sans difficultés, en ayant un rendu fonctionnel et agréable à utiliser.

Nous utilisons également d'autres librairies permettant d'ajouter certaines fonctionnalités, comme *react-router-dom* ou encore *axios*.

Pour le back-end, nous avons choisi Spring pour la création de l'API RESTful, et MongoDB Atlas en tant que base de données, qui est donc No-SQL.

4 Fonctionnement

Lors du démarrage de l'application client, le navigateur par défaut s'ouvre sur l'adresse localhost :port (3000 pour nous). On arrive directement sur la page de login.

Pour l'instant, nous n'avons pas encore réussi à implémenter la connexion avec identifiants vérifiés par le serveur mais c'est en cours de développement.

Comme présenté par la maquette, on peut également choisir de s'enregistrer en tant que nouveau client, en remplissant un formulaire.

Chaque input de ce formulaire est vérifié par l'application client pour correspondre aux règles instaurées au préalable (avec les regex). Une fois que tous les champs sont remplis, on peut cliquer sur le bouton d'enregistrement, et si les adresses et ports sont configurés correctement, l'API recevra la requête d'enregistrement de nouvel utilisateur, et créera un nouveau client en envoyant les données à la base de données. Cela enregistrera un nouveau client, mais également des nouveaux identifiants, avec le mot de passe chiffré pour plus de sécurité.

Une fois enregistré, vous serez redirigé vers la page de succès de création de compte, puis en cliquant, sur la page d'accueil.

En choisissant un portfolio dans le sélecteur du menu latéral gauche, il sera possible d'afficher des graphiques (avec des données hardcodées pour le moment) qui représentent la consommation du portfolio. Cette partie n'est pas encore implémentée pour communiquer avec le serveur, mais elle sera implémentée dès que le login sera terminé.

Il est possible d'interagir avec les différentes sections de la barre latérale pour aller sur les autres pages, qui ne contiennent pas encore le contenu final.

Un menu d'utilisateur est disponible en haut à droite, avec quelques pages également vides.

Comme le login n'est pas encore terminé, les différentes pages sont accessibles en modifiant manuellement l'URL dans la barre de recherche et il est possible d'accéder à l'application sans être un utilisateur, mais cela sera remplacé lorsque le login sera implémenté.

5 Conclusion

Pour le moment, le projet est encore loin d'être terminé, et de nombreuses parties clés sont encore en phase d'implémentation. Certains choix ne sont pas encore définitifs et optimaux, et seront potentiellement remplacés dans le futur.

Cela dit, une fois certaines fonctionnalités comme l'authentification implémentées, il sera plus facile et plus rapide d'avancer dans le reste. Beaucoup des fonctionnalités sont redondantes dans leur fonctionnement et implémentation, comme par exemple la création de page ou les requêtes pour obtenir des données. Il sera donc possible de compléter ces fonctionnalités plus rapidement.