

Rapport Développement Avancé – TP2

Mes démarches et choix

J'ai décidé de faire les étapes du TP les unes après les autres, sans sauter d'étapes pour passer à une plus simple ou plus rapide par exemple. Il était important pour moi de comprendre chaque partie, avec ses nouvelles technologies et méthodes, afin de passer à la suivante.

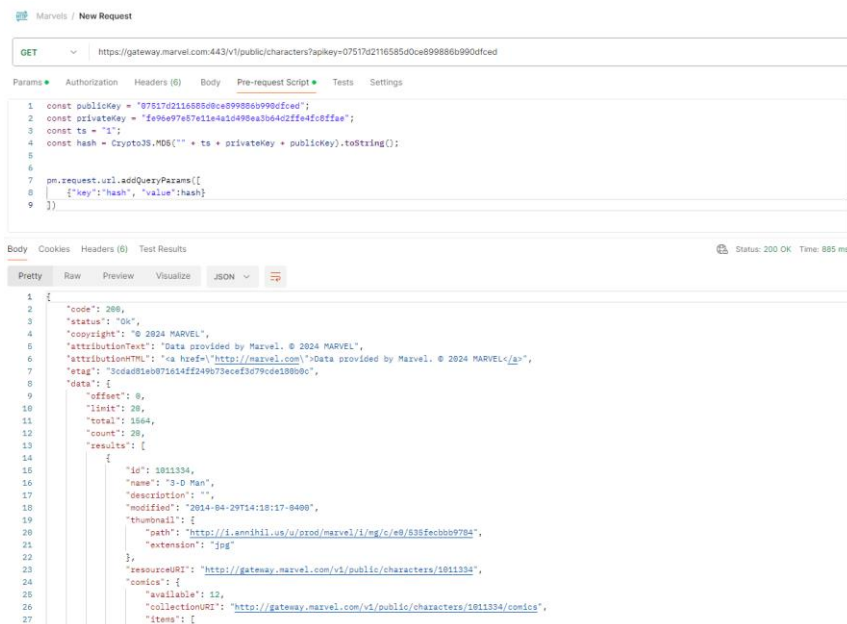
Ce qui a marché (facilement)

L'API Marvel

La prise en main de l'API sur le site de Marvel est plutôt facile à comprendre et à utiliser. J'ai réussi à utiliser le "Try it out" de l'API présent sur le site et lire les données que ça renvoie directement sous format JSON dans le site.

Postman

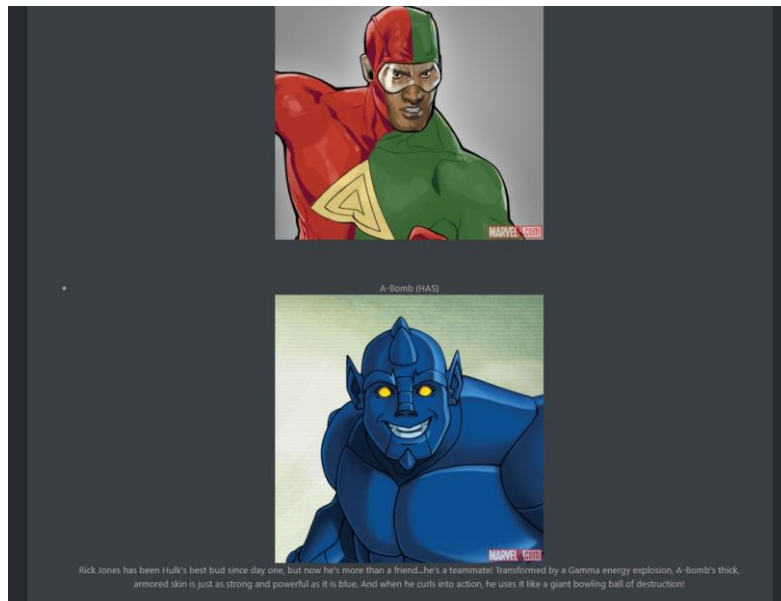
J'ai dû utiliser Postman pour l'étape 2. Comme je l'avais déjà utilisé au TP précédent, je n'ai eu aucun problème à le manipuler dans les grandes, mais j'ai eu quelques difficultés avec le Pre-request Script, que je n'avais jamais utilisé.



Capture d'écran d'une requête à l'API Marvel sur Postman

Le tableau et le site web

Lors de l'étape 3, j'ai pu récupérer les données de la requête sur PHPStorm. Avec cela, je n'ai eu aucun problème à faire le tableau contenant les héros avec leur nom, leur description et leur image, et à utiliser Handlebars pour ensuite envoyer ces données sur le site web.



Capture d'écran du site de l'étape 3 et 4

Mes difficultés et comment je les ai surmontés

Pre-request Script

J'ai eu du mal à prendre en main les paramètres nécessaires à la requête de l'API Marvel sur Postman, en particulier à travers le Pre-request Script car cela demandait des méthodes que je ne connaissais pas. Notamment lorsque j'ajoutais les paramètres depuis le Pre-request Script, cela ne fonctionnait pas. Cela était sûrement dû à la façon dont j'implémentais les paramètres de la requête qui n'étaient donc pas conformes au format de requête. Finalement, je me suis rendu compte que la variable `ts` était un Integer et que cela ne la reconnaissait donc pas lors du hachage. En la transformant en String cela a débloqué le problème et ma requête GET a fonctionné.

Programmer getData()

J'ai également eu des problèmes avec le fait de recevoir les données sur PHPStorm, par la méthode `getData()`. Une bonne partie des problèmes venait du hash qui n'était pas bien fait, dû à des problèmes liés aux clés publique et privée.

Utiliser Falsify

J'ai eu quelques difficultés à comprendre Falsify, et comment l'intégrer et l'utiliser dans le projet. Mais en persévérant, j'ai pu finalement comprendre, et à partir de là, son utilisation était assez facile et bien pratique.

Ce qui pourrait être amélioré

Docker

Je suis allé jusqu'à l'étape 5 du TP, dans laquelle j'ai installé Docker et j'ai fait les manœuvres permettant de générer une image pour Docker. Tout cela a fonctionné, avec un peu de difficultés car c'était la première fois que j'utilisais Docker.

```
PS D:\BUT 3\Dev Avance\LesMarvels> docker build . -t monimagedocker  
[+] Building 32.4s (11/11) FINISHED
```

Capture d'écran du build de l'image docker réussi

Mais lorsque j'ai lancé l'image obtenu, cela m'a fait une erreur à la place du site web. Je n'ai malheureusement pas eu assez de temps et de ressources pour corriger cette erreur. Je pense qu'elle pourrait venir des path mentionnés pour le header, footer, et index, dans le fichier serveur.js. C'est donc quelque chose qui devrait être amélioré.