

# Rapport Développement Avancé - TP1

## Mes démarches et choix

Pour aborder ce TP, j'ai décidé de suivre l'ordre des exercices afin de bien faire chaque étape l'une après l'autre. Je ne souhaitais pas m'avancer particulièrement sur une partie qui me semblait plus intéressante ou facile à faire, car je préférais réussir à faire fonctionner chaque étape du projet, de manière atomique.

## Ce qui a marché (facilement)

### Postman

Installer et utiliser Postman n'a pas été un problème. J'avais déjà pu le manipuler pour un autre cours donc j'avais les bases de son fonctionnement, et l'utilisation que j'en ai faite pour ce TP était simple et rapide donc cela a été facile.

### Les fonctions

**findBlocks** : Je n'ai pas eu trop de difficultés à programmer la fonction `findBlocks()` qui est la première fonction à faire. Le principe et l'exécution étaient simples.

**createBlock** : Concernant la fonction `createBlock()`, je n'ai pas eu des facilités sur toutes les parties à coder. La partie de la fonction qui nécessite de créer un nouveau bloc à l'aide des infos données a été plutôt facile pour moi. J'ai pu sans problème récupérer les données et les mettre dans un objet, et y ajouter l'identifiant, la date et le hash, en important des bibliothèques.

**findLastBlock** : Ensuite, ayant réussi à faire la fonction `findBlocks()`, il était facile pour moi de coder la fonction `findLastBlock()` à partir de cette dernière. Cela s'est donc fait rapidement.

**sha256** : Enfin, j'ai réussi à faire facilement le hachage de l'identifiant, tout simplement en important et utilisant l'algorithme sha256.

## Mes difficultés et comment je les ai surmontés

### Se retrouver dans les fichiers

Au début j'ai eu un peu de difficultés à me retrouver dans les différents fichiers du projet ainsi que comprendre leur lien entre eux. Cette difficulté a été accrue par l'architecture tournée vers Node.js, avec des fichiers `.json` servant à stocker des données ou encore ceux servant aux bibliothèques importées. Au fil du TP, en manipulant les différents fichiers et fonctions présentes afin de faire les exercices, j'ai pu mieux comprendre la structure des fichiers entre eux, ainsi que l'utilité et la fonction de chacun.

## Le path

Lors du développement de la méthode `createBlock()` j'ai rencontré plusieurs problèmes avec le path du fichier *blockchain.json* que j'utilisais lors de la lecture et écriture dans ce fichier. En effet, l'utilisation que j'en faisais marchait dans la partie lecture mais pas dans la partie écriture. Ce problème inexplicable m'a valu beaucoup de questionnements et d'essais à changer le path et son utilisation. Finalement après de nombreuses tentatives j'ai trouvé la solution, que j'ai pu utiliser aux deux endroits.

## Utiliser et gérer les `await/async`

Comme cela était pour moi la première fois que je manipulais les fonction `await` et `async` en JavaScript, j'ai pu parfois oublier de les utiliser à certains endroits, avant de m'en rendre compte plus tard et de me corriger.

## Ce qui pourrait être amélioré

Comme j'ai pris du temps à comprendre le projet, comprendre les nouvelles méthodes et technologies, et résoudre mes problèmes, je n'ai pas eu le temps d'arriver jusqu'à la partie « Pour aller plus loin » du TP. Cette partie demande notamment de faire les fonctions `verifBlocks()` et `findBlock(id)`, que j'aurais pu donc rajouter pour améliorer le projet. Je pense que j'aurais implémenté la fonction `findBlock(id)` sans problèmes. En revanche, la fonction `verifBlocks()` m'aurait sûrement donné plus de difficultés étant donné qu'elle concerne le hachage, que je ne maîtrise pas autant que le reste.

Enfin, je n'ai pas non plus eu le temps de créer et connecter un repo GitHub à mon projet pendant le TP car j'étais concentré sur la réalisation des exercices. J'ai donc dû m'occuper de cela après le TP. Pour la prochaine fois, maintenant que le GitHub est mis en place, je m'occuperais de cela en amont afin que je puisse commit pendant la séance.