

Node JS - Activités N02

► Activité N01

◆ Discussion

1. *Qu'apportes-tu en termes de clarté et de maintenabilité ?*
 - Le projet est devenu plus claire, plus simple et facile a comprendre et a maintenir par la separation de chaque fonctionnalité (routage, donnees, logique ...)
2. *Quels fichiers changeraient le plus souvent si l'API évolue ?*
 - les fichier qui sont responsable de de routages

► Activité N02

◆ Discussion

1. *Qu'est-ce qu'un endpoint REST bien conçu ?*
 - unique, claire, chaque route est spécifiée par une méthode http
2. *Pourquoi l'idempotence est-elle importante dans une API ?*
 - l'idempotence garantit qu'une même requête répétée donne le même résultat (sécurité et prévisibilité).

► Activité N03

◆ Discussion

1. *Quelle est la différence entre un middleware global et local ?*
 - middleware global peut être appliqué globalement sur plusieurs module, et souvent définit comme un module séparé.
 - middleware local est appliqué sur le même module, et souvent définit dans ce même module la.
2. *Pourquoi la réutilisabilité est-elle une force des middlewares ?*
 - la réutilisabilité peut être appliqué plusieurs fois sans l'implémenter à chaque fois.

► Activité N04

◆ Discussion

1. *Pourquoi séparer la logique d'erreur du code métier ?*
 - pour faciliter la maintenance du code et garder le rôle de chaque phase/code.
2. *Qu'apporte la cohérence des messages d'erreur côté client ?*
 - pour éviter l'affichage des erreurs originaux aux clients pour des raisons de sécurité ... et de faire comprendre à l'utilisateur d'où vient le problème (côté client ou côté serveur)

► Activité N05

◆ Discussion

1. *Comment éviter le blocage du thread lors de la lecture de fichier ?*
 - éviter d'utiliser les fonctions synchrones comme `readFileSync`, et d'utiliser `async/await`
2. *Quelle est la différence entre `readFileSync` et `fs.promises.readFile` ?*
 - `readFileSync` bloque le thread et `fs.promises.readFile` est non bloquant.