



Enoncé

Nous avons un système de queue FIFO.

La queue est une liste d'action en attente d'exécution.

Chaque type d'action possède un nombre de crédits d'exécution.

Les différents types d'actions sont définis par le développeur.

Les crédits d'exécution ont une valeur maximale définie aussi par le développeur.

Ils seront calculés de manière aléatoire entre 80% et 100% cette valeur maximale.

Ils sont recalculés si au moins 24 heures sont passées depuis le dernier calcul.

Ajouter une action à la queue ne coûte pas de crédit, l'exécution en consomme un.

Via une interface, l'utilisateur peut ajouter autant d'actions qu'il souhaite dans sa queue. (Même s'il n'a plus de crédits).

Toutes les 2 minutes la première action disponible est exécutée.

Exemple :

Je suis dans un système avec 3 types d'action, A, B, C ayant respectivement 20, 20, 30 en valeur maximale de crédits.

L'utilisateur, après le premier calcul, possède en crédits pour les types d'actions A, B, C, dans l'ordre, 18, 20, 29.

Il ajoute 2 actions de type A, puis 3 de B puis 1 de A puis 2 de C.

Il aura donc une queue : First-In \rightarrow [A, A, B, B, B, A, C, C] \rightarrow Last-In

2 minutes plus tard, la première action de type A s'exécute et consomme un crédit.

La queue sera donc devenue: First-Out \rightarrow [A, B, B, B, A, C, C] \rightarrow Last-Out, et les crédits seront passés à 17, 20, 29 (A, B, C)

Objectif :

Tu devras rendre un code mettant en place le système décrit ci-dessus.

Il devra comporter un frontend (React) et un backend (Express, Node).

L'interface devra donc afficher au moins la queue d'action à exécuter, les crédits disponibles et donner la possibilité de rajouter des actions à exécuter à la queue.

L'architecture de la codebase devra être scalable et une attention particulière sera portée dessus.

Il faudra utiliser TypeScript.

Des tests backends devront être écrits.

Le code devra être exécutable via un `npm run start`.

Les tests devront être exécutables via un `npm run test`.

Aucune limite de temps.

Une review du code avec des questions pourra avoir lieu.

L'énoncé peut cacher des subtilités.

N'hésite pas si tu as des questions