

Dougs Test Technique

Hey ! Félicitations, tu as passé la première étape d'entretiens avec succès ! L'équipe technique Dougs souhaite te soumettre un petit challenge pour apprendre à te connaître. Rien de trop compliqué, cela ne devrait te prendre que quelques heures.

N'hésite pas à nous contacter si tu as besoin de clarifications.

Contact: test-technique@dougs.fr

Contexte :

En tant que cabinet d'expertise-comptable, Dougs a besoin de récupérer les opérations bancaires de ses clients. Cette synchronisation bancaire est gérée par des prestataires externes.

Ces prestataires font du scrapping pour récupérer les transactions depuis le site des banques. Cette technique n'est pas infallible, et il arrive parfois que certaines opérations soient remontées en double ou qu'il en manque quelques-unes.

Il est primordial de nous assurer de l'intégrité de cette synchronisation, sans quoi la comptabilité de nos clients pourrait être faussée.

Problème :

Comment s'assurer que les opérations remontées par la synchronisation bancaire soient correctes ? Nous demandons au client de nous fournir les relevés bancaires de sa banque pour chaque mois. Nous pouvons ainsi avoir des **points de contrôle** tout au long de l'année grâce au solde indiqué en fin de période de chaque relevé. Le solde indiqué sur un relevé bancaire est juste.

En cas d'anomalie, un comptable effectue un contrôle afin de supprimer les doublons ou de créer manuellement les opérations manquantes.

Étant donné une liste d'**opérations bancaires**, ainsi que des **points de contrôle**, il faut définir un algorithme et son interface qui permette de valider ou non l'intégrité de la synchronisation, et le cas échéant, de simplifier au maximum le contrôle manuel du comptable.

Opération bancaire : { id: number, date: Date, wording: string, amount: number }

Point de contrôle : { date: Date, balance: number }

Spec Technique :

Cette spec n'est là qu'à but indicatif, tu es libre de proposer ta propre solution afin de répondre au mieux au problème.

API:

/POST /movements/validation

Request:

```
{  
  "movements": [{ id, date, label, amount }],  
  "balances": [{ date, balance }]  
}
```

Responses:

Code 2XX: { "message": "Accepted" }

Code 4XX: { "message": "Validation failed", "reasons": [{ ... }] }

A toi de définir l'interface des "reasons" avec tous les détails que tu juges nécessaire.

Livrable

Nous analyserons avec soin ta proposition, merci de nous envoyer une archive avec tes sources (NestJS), ainsi que ton analyse de ta production à test-technique@dougs.fr .
Nous reviendrons rapidement vers toi pour débriefer.

Bon courage, à bientôt.