Full stack developer task

Mission

Your mission is to **build a system that will store crypto-currency exchange rates** over time (for example: Bitcoin to USD). The system will be used for backup purposes, and enable the users of the system to track the values of crypto-currencies.

Step 1 – Get and store the data

Build a Node.js process that gets the current exchange rates every 1 minute and stores them in a file. This can be stored as a JSON file, there is no need to use a database.

This file will be used to store the exchange rate data (there is no need to get the history from before the system is up).

You should store the BTC→USD, BTC→EUR, ETH→USD and ETH→EUR exchange rates.

There are multiple free APIs to get the exchange rates and you can choose the one that fits you best. We recommend using the Coinbase API.

Step 2 – Provide API access to data

Build another Node.js process which acts as an API server. It should have two endpoints:

1. Get data. An endpoint that allows to query the saved data from step 1. This endpoint should return all entries in the last 5 minutes.
   The returned entries should include the date, exchange rate, and anything else you see fit.
   It should allow the caller to:

   1. Choose the currency. Example: "BTC to USD" (default), "BTC to EUR", etc.

   2. Choose the time frame in minutes. Example: 5 minutes (default), 10 minutes, 15 minutes, etc.

2. Delete data. An endpoint for deleting a specific exchange-rate entry. After this API is called, the data should no longer be returned if the first endpoint

is being called again.

FAQ

- Should I use a framework?

  You can choose to use a framework or not, it is not required or expected.

- Are there any design limitations?

  No. You can design the API in any way you think is best.

- What should my solution include?

  A Node.js process for syncing currency-rates, and a Node.js server (express, nest, hapi or any other server).