

Cryptocurrency Price Tracker

Task:

Build a microservice that periodically fetches, caches, and serves cryptocurrency prices from a public API, with the option to use either Node.js or Golang.

Requirements:

Endpoints:

- ***GET /coins***: Fetch the list of available cryptocurrencies.
- ***GET /coins/:symbol***: Fetch the latest price and details of a specific cryptocurrency by its symbol (e.g., BTC, ETH).
- ***GET /coins/:symbol/history***: Fetch the price history for the past 7 days for a specific cryptocurrency.

Data Source:

Fetch cryptocurrency data from a public API like CoinGecko, CoinMarketCap, or any other public and free API you're comfortable with.

Caching:

To reduce the number of requests to the public API and ensure better performance, implement caching for the fetched data. This can be done using in-memory storage or tools like Redis.

Database:

Store the price history of cryptocurrencies in a database of your choice for the ***/coins/:symbol/history*** endpoint.

Error Handling:

Handle potential errors like rate limits from the public API, cache misses, or issues with the database.

Rate Limiting:

Implement rate limiting for your API to avoid misuse.

Bonus:

- Implement a simple notification system where users can subscribe to a specific cryptocurrency, and if its price increases or decreases by more than 10% in 24 hours, they receive a notification (this can be a simulated notification, e.g., a log message).
- Containerize the application using Docker.
- Write unit tests for key components.
- Provide endpoints to convert between cryptocurrencies (e.g., converting 1 BTC to ETH based on current rates).

Important:

Please do not publish the solution or any part of it in any public repository or platform. The assignment is meant to evaluate individual skills and should remain confidential.