

Evaluación técnica

Necesitaríamos que crees una web api en .NET que exponga dos endpoints:

1) Uno que devuelva la cotización del día de una moneda, donde el identificador de la moneda sea input del endpoint. Solo se aceptarán las monedas "dolar" y "real". Cuando se pida la cotización de cualquier otra cosa, devolver un error con un mensaje apropiado.

La cotización del dólar se obtendrá del siguiente servicio externo:

<http://www.bancoprovincia.com.ar/Principal/Dolar>.

La cotización del real será la cuarta parte de la cotización del dólar.

Tener en cuenta que en un futuro se pretenderá obtener la cotización del real de otro servicio externo que está siendo desarrollado por un equipo externo. También se pretende incorporar en un futuro una nueva moneda, el dólar canadiense, como moneda aceptada por nuestra Api

2) Otro que realice una compra de una moneda. Dados un id de usuario, monto a comprar en pesos argentinos e identificador de moneda, el endpoint realizará una transacción que almacenará en la base de datos dicha compra registrando el usuario y los montos comprados en la unidad de la moneda (dólar o real). Por ejemplo, dado el monto 1000 y la moneda "dolar", se registrará en la base de datos una compra con un valor resultado de $1000 / \{\text{cotización dolar del dia}\}$ dólares.

Se necesita validar los montos a comprar. Para el dólar, el límite es 200. Para el real, el límite es 300. Todos los límites son en la moneda extranjera, por usuario y por mes. Cualquier otra moneda solicitada deberá responderse con un error y mensaje apropiado.

Para la base de datos, hacer el diseño lo más simple posible. Asumir que los usuarios existen y no hay que aplicar validaciones sobre el usuario. Solo validar las compras mensuales por usuario. Es decir solo las transacciones. Incluir scripts de base de datos que sean necesarios, como la creación de la DB y tabla.

2a) ¿Qué opina de pasar el id de usuario como input del endpoint? ¿Cómo mejoraría la transacción para asegurarnos de que el usuario que pidió la compra es quien dice ser?

Se valorará:

- Que la web API sea en NET Core.
- Desarrollo orientado a servicios
- Buenas prácticas para el desarrollo de endpoints (verbos HTTP, rutas, códigos de status HTTP, etc)
- Buen manejo de excepciones
- Desarrollo orientado a objetos
- Agregado de unit tests para los servicios
- Logueo de errores

Angular - Opcional

Hacer una app muy sencilla en Angular 2+ que tenga dos páginas/rutas:

- a) /cotizaciones, que muestre las cotizaciones de las monedas real y dólar utilizando la Api desarrollada en el punto anterior. Agregar un botón para actualizar las cotizaciones.
- b) /compra, que liste las mismas cotizaciones de a) y un mini formulario de compra, con tres campos de texto para capturar ID de usuario, moneda y monto a comprar en pesos; y un botón para efectivizar la compra a través de nuestra Api. Mostrar el resultado de la transacción.

Mostrar un loading wheel mientras se realiza la compra

Se valorará la utilización de NgRx, manejo de estado, en la aplicación.