1. Flujio:

a-Nuestra API recibirá un GET a un endpoint /items/prices/[item-id]. b-La api llamará al servicio de moneda

curl -X GET -H 'Authorization: Bearer \$ACCESS_TOKEN' https://api.mercadolibre.com/currency_conversions/search?from=ARS&to=CLP

Con new Relic podemos saber el promedio en función a estadísticas el tema de las llamadas

Tomando en cuenta los re intentos a dicha api, se deberán hacer los cálculos correspondientes y devolver el item con su precio final.

Si llegan demasiadas request habría que limitar desde nuestra api la cantidad de requests que puede tomar y devolver el código correspondiente.

Si las cotizaciones cambian una vez al día se podría hacer un proceso que corra a determinada hora para que tome todas las cotizaciones posibles y las guarde en memoria o en una DB propia (dependiendo de la cantidad de información a guardar. Considero que no debería ser mucha y se podría manejar en memoria con redis por ejemplo). Se puede usar Cache.

Necesitamos algo que NO sea eventualmente consistente. un KVS del tipo CACHE.