



Summary

Your task is to build an application that supports the requirements outlined below. Outside of the requirements outlined, as well as any language limitations specified by the technical implementation notes below and/or by the hiring manager(s), the application is your own design from a technical perspective.

This is your opportunity to show us what you know! Have fun, explore new ideas, and as noted in the Questions section below, please let us know if you have any questions regarding the requirements!

Requirements

Requirement #1: Store a Purchase Transaction

Your application must be able to accept and store (i.e., persist) a purchase transaction with a description, transaction date, and a purchase amount in United States dollars. When the transaction is stored, it will be assigned a unique identifier.

Field requirements

- Description: must not exceed 50 characters
- Transaction date: must be a valid date
- Purchase amount: must be a valid amount rounded to the nearest cent
- Unique identifier: must uniquely identify the purchase

Requirement #2: Retrieve a Purchase Transaction in a Specified Country's Currency

Based upon purchase transactions previously submitted and stored, your application must provide a way to retrieve the stored purchase transactions converted to currencies supported by the Treasury Reporting Rates of Exchange API based upon the exchange rate active for the date of the purchase.

<https://fiscaldata.treasury.gov/datasets/treasury-reporting-rates-exchange/treasury-reporting-rates-of-exchange>

The retrieved purchase should include the identifier, the description, the transaction date, the original US dollar purchase amount, the exchange rate used, and the converted amount based upon the specified currency's exchange rate for the date of the purchase.

Currency conversion requirements

- When converting between currencies, you do not need an exact date match, but must use a currency conversion rate less than or equal to the purchase date from within the last 6 months.
- If no currency conversion rate is available within 6 months equal to or before the purchase date, an error should be returned stating the purchase cannot be converted to the target currency.
- The converted purchase amount to the target currency should be rounded to two decimal places (i.e., cent).

Technical Implementation

The technical implementation, including frameworks, libraries, etc. is your own design except for the language. That is, if you are applying for Gateways (written in Java), you should implement the solution in Java. If you are applying for TAG (written in GoLang), you may choose to implement the solution in GoLang or Java.

You should build this application as if you are building an application to be deployed in a Production environment. This should be interpreted to mean that all functional automated testing you would include for a Production application should be expected. Please note that non-functional test (e.g., performance testing) automation is not needed.

Your application repository should be runnable without installing additional software stack components, such as databases, web servers, or servlet containers (e.g., Jetty, Tomcat, etc).

Requirements Questions

Questions may be submitted via an email sent to the TAG and Gateways Hiring distribution list: gl-taghiring@wexinc.com. Please include a prefix of "Product Brief Questions: " in the subject line of your email.

Submission and Due Date

Submit your project by emailing the public GitHub repository to gl-taghiring@wexinc.com. Please include a prefix of "Product Brief Submission: " in the subject line of your email.

Your submission is due five business days (Monday-Friday) after it has been provided, not including holidays.