

Practical PHP test

As part of the selection process, we ask each candidate to complete a tech task to ensure that they possess the skills necessary for the position.

Disclaimer

The only purpose of this task is to assess candidate skills. Therefore, everything sent as a result of the task will not be used for commercial purposes.

The Objective

- Using MySQL create a database that will hold all data required to be stored.
- Write a web service in PHP that will do any data retrieval and processing. AJAX must be used from the browser to call at least one web service.
- Create a web application that will consume the web service and present the user with a workable user experience.

Tools

To complete this task, please use Laravel.

We condone the use of 3rd party PHP frameworks and libraries and while we like to see good usage of them, this test is to assess the candidate's abilities in writing bespoke software.

What will be assessed?

These are the main things that will be assessed.

- Database structure.
- Understanding of OOP.
- Usage of application layering.
- Knowledge of design patterns and best practices that inspires maintainable code.
- Web service (API) design.
- Installation instructions for application (README). We will be deploying your code locally to see that it runs.

What is not required?

- A “pretty” UI, though it must be user-friendly. Thus colors, fonts, images, etc are not needed but feel free to use JS and CSS frameworks if that’s what you are comfortable with.
- User registration and authentication.

Acceptance Criteria

- A web application must be created that will allow a user to purchase foreign currencies.
- The page should display the available currencies for selection by the user.
- The page should have input where the user can get a quote by entering the amount of the currency that they wish to exchange.
- Once the user has entered either amount and selected the foreign currency, the necessary calculation needs to be done that will display the amount they need to pay in USD
- With the calculated amount displayed, the user can then select to “purchase” the foreign currency. An “order” for the currency must be saved to the database and the user must be shown a confirmation.

Details

- The currency used for payment will be US Dollars (USD).
- The currencies that can be purchased are:
 - Japanese Yen (JPY)
 - British Pound (GBP)
 - Euro (EUR)
- Use the following exchange rates:
 - USD to JPY: 107.17
 - USD to GBP: 0.711178
 - USD to EUR: 0.884872
- A surcharge must be added to orders and differs for the currencies:
 - JPY: 7.5%
 - GBP: 5%
 - EUR: 5%
- The following information must be saved in an order:
 - Foreign currency purchased.
 - Exchange rate for foreign currency.
 - Surcharge percentage.
 - Amount of surcharge.
 - Amount of foreign currency purchased.
 - Amount paid in USD.
 - Discount percentage
 - Discount amount
 - Date created.
- When an order is saved the following extra actions need to be taken for the different currencies:
 - JPY: No action.
 - GBP: Send an email with order details. This can be a basic text or html email to any configurable email address.



- EUR: Apply a 2% discount on the total order amount (this needs to be configurable for the currency).

Special note

Use a foreign exchange API to get exchange rates. Results from the API should be “cached” by updating the foreign exchange rates in the database for each currency. The updating of exchange rates only needs to happen periodically in the real world, thus the update should be triggered via a URL or the command line (cron-ready). You can use <https://currencylayer.com/> to fetch currency rates.

If you have any questions or need further information, please feel free to contact us.

Good Luck! 🍀