

React Fullstack Engineer (React/PHP)

Context:

Imagine you have joined a fast growing e-commerce company which serves more than 2 million customers. The company has been running for several years and is now going through its growth stage. The current system consists of various sub components referring to the same database. The Engineering team is required to support the management team with various reports regularly. As of now, these reports do not follow any particular format and the expectations are changing at a high frequency.

You are required to build:

1. A REST API using **any PHP framework** and given data files
2. Dashboard using react js that displays data using above **API with ReactJS**

Your program should be in final production quality and it should demonstrate your understanding and skills in:

- clean code & best practices
- expectations of edge cases
- exception handling
- security
- design and architecture
- Unit testing

1 . REST API requirements

The Engineering team has decided to build an internal reporting tool. You will have to read data from another source which will provide data in a json format. You can choose any php framework that you prefer. The API should be very well documented and unit tested.

You are required to build an API that sends different types of data:

1. Report Type (Only 2 report types are defined as of now. there can be 100s of report types in the future)
2. Report Format (at the moment it supports only CSV, in future there will be a few more types like json etc.)

3. Each Report type may have different additional parameters.

Report Type 1: Turnover per brand (all brands).

Parameters: Start Date, End Date, max date range is 1 week, Report format CSV

Report Type 2: Turnover per day (of all brands) :

Parameters: Start Date, End Date, max date range is 1 week, Report format CSV

Bonus points:

- Include Turnover excluding VAT per brand in your report

2. Dashboard Requirements

Users should be able to select report type and required additional fields (e.g. report format, start date, end date) should appear upon the selection of the report type. (Some Report types may not require any parameter)

The response data should be displayed in a data table. You are required to write your own CSS/HTML without any css framework. Errors should be very well expected and handled.

Data:

Attached (in the email) you'll find 2 json files containing dummy data:

- Brands: containing data of fictive brands.
- GMV: containing the total turnover per day for all brands.

The prices in the database are 21% VAT included.

Before you Submit:

Make sure to include documentation around the setup of your project. Do a thorough code review yourself. Assume you are the Chief Architect and you have a very high coding standard. Please fix ALL bugs and errors before you submit. We hope to see quality work of a very high, professional standard