Home assignment

We are going to launch a new product soon and we need to create a new simple and robust RESTful service which provides functionality to transfer funds between our customers' accounts. Therefore we have high hopes for you to make it in time (no pressure)!

Functional requirements:

- Service should expose an HTTP API providing the following functionality:
    - Given a client id return list of accounts (each client might have 0 or more accounts with different currencies)
    - Given an account id return transaction history (last transactions come first) and support result paging using "offset" and "limit" parameters
    - Transfer funds between two accounts identified by ids
- Balance must always be positive (>= 0).
- Currency conversion should take place when transferring funds between accounts with different currencies
    - For currency exchange rates you can use any service of your choice, e.g. https://api.exchangerate.host/latest
    - You may limit the currencies supported by your implementation based on what the currency exchange rate service supports
    - Currency of funds in transfer operation must match receiver's account currency (e.g. system should return error when requesting to transfer 30 GBP from USD account to EUR account, however transferring 30 GBP from USD to GBP is a valid operation - corresponding amount of USD is exchanged to GBP and credited to GBP account).

Non-functional requirements:

1. Test coverage should be not less than 80%
2. Implemented web service should be resilient to 3rd party service unavailability
3. DB schema versioning should be implemented

Expected result:

- Task developed in PHP, preferably using the Symphony framework
- We expect that business logic is implemented by you rather than using ready made packages/libraries
- Clear instructions on how to set up the project (we would appreciate containers very much)

Evaluation criteria:

1. Feature-completeness
2. Code quality, structure
3. Test quality
4. Non-functional requirement implementation
5. Ease of setup

Please send the code (archive or link to github) and instructions how to set it up once completed.