

Ontop's Challenge for backend software engineer with Java

We need to design and develop a REST API to release the functionalities shown in the following screens:

We need some information about your recipient's bank account

Please fill the recipient's bank information

First name

👉 Looks good!

Last name

Routing number

🚫

National identification number

Account number


Continue

How much money do you want to send?

Please fill in the money you want to transfer to the receiving account


🏠 \$99,924.00 Available wallet funds

You send

  USD

- \$2.00 Transaction fee

Recipient gets

  USD

Continue

Recent transactions



Bank transfer to your account

✅ Transaction completed

-33 USD

05 Jan 2023



Refund from bank transfer

Refunded to your wallet

200 USD

05 Jan 2023



Bank transfer to your account

❌ Transaction failed

-200 USD

05 Jan 2023



Bank transfer to your account

🕒 Transaction in progress

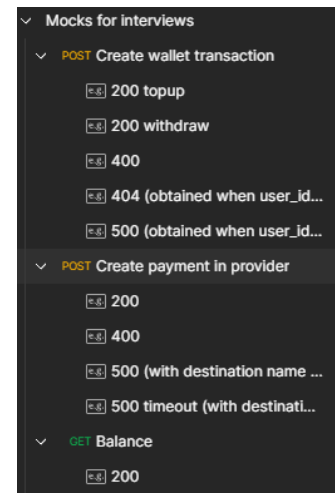
-123 USD

04 Jan 2023

The functionality consists of the user being able to withdraw their money from the wallet and transfer it to their bank.

To do this, we have to use the provided APIs (see the attached mocks) which include:

- Wallet transactions API that is used to update the wallet balance through top ups and withdraws operations. (The mock will not actually update the balance because it's a mock)
- Payments is an external provider API that is used to execute the transfer between Ontop's bank account and the users' bank account, but it doesn't update the wallet balance. Ontop's bank account will always be the same but different depending on the environment.
- Wallet balance API that is used to fetch the user's balance. (The mock will always return the same balance)



It's worth clarifying that the users' wallets are not bank accounts, but are virtual accounts whose funds are inside one unique bank account, the Ontop's one.

To have the best user experience, we want the user to enter their bank details only once, and then be able to carry out all the transactions they want, as long as they have the balance in their wallet to do so.

The transaction must have:

- A fee of 10% applied when the transaction is carried out.
- A status (we leave the decision of which statuses to use up to you).

To configure the account, we need the following attributes to be entered:

- Name and surname
- Routing number
- National identification number
- Account number
- Bank name

We also want to be able to query the transactions ordered by descending "creation date" in a paginated table and filtered by amount and date.

The tasks

For this challenge we want you to:

- Briefly design the solution of the previously described functionality using any diagrams that would help you explain it and your teammates understand it.
- Code ONLY the endpoint that does the transaction between the bank accounts and updates the balance.

Review

The challenge will be evaluated focusing on **good coding practices** and the solution design.

Points to include:

- Unit testing (at least one per layer and one for exceptions)
- Exception handling
- Readme with instructions to build the project locally (or upload it to the cloud)

Technologies

- Java 8+
- Spring boot 2+
- Git
- Any database without restrictions
- Any testing framework

Finished work

Once you've completed the challenge, please send an email that includes the following information:

- The public repository url for us to review the code.
- A postman collection with the developed endpoint and an **example of execution**.

Since we know that the challenge takes some time to be completed, it's okay if you send us a detailed description about things you would've done differently but didn't because of the time limit.