JAVA Online Test

Backend Engineer

Background

Nowadays, for most eCommerce application, there is a "e-Wallet". End users can easily check their balance, do transfer to other users or retrieve their latest transaction list.

In this simple task, the security check can be omitted, but in the documentation if you can explain the security implications, you can gain extra review marks.

Your Task

Develop an API server that does simple "e-Wallet" based on the User Stories below.

You are required to:

- Design proper database schema to support these user stories
- Design and implement APIs to support these user stories
- Deploy an instance of the API server on the public cloud or provide a 1-step command to run your API server locally, e.g. using a Makefile or Docker Compose) for us to test run the APIs
- Write sufficient documentation for the APIs and explain your technical choices

User Stories

- ### As a user, I need an API to register new account and earn 10000 signup credits.

The API should receive the following JSON request:

```
{

"email" : "test@test.com"

}
```

The API should return the following JSON response on success:

```
{
 "success": true,

 "balance" : 10000.0
}
```

Please propose JSON responses for any errors that might occur.

- ### As a user, I need an API to check my latest balance

The API should receive the following JSON request:

```
{

"email" : "test@test.com"

}
```

The API should return the following JSON response on success:

```
{
 "success": true,

 "balance" : 10000.0
}
```

Please propose JSON responses for any errors that might occur.

- **As a user, I need an API to transfer credits to an existing user**

The API should receive the following JSON request:

```
{

"email" : "test@test.com",

"transferee": "test1@test.com",

"amount": 20

}
```

The API should return the following JSON response on success:

```
{
 "success": true
}
```

Please propose JSON responses for any errors that might occur.

- **As a user, I need an API to retrieve latest transaction list**

The API should receive the following JSON request:

```
{

"email" : "test@test.com"

}
```

The API should return the following JSON response on success:

```
{
 "success": true,

 "transactions": [

      {

"id": "1",
        "from" : "test@test.com",
        "to": "test1@test.com",
        "type": "transfer",
        "amount": 100.0,
         "datetime":  "2018-04-16T19:20+08:00"
     },
     {
"id": "2",
        "from" : "test1@test.com",
        "to": "test@test.com",
        "type": "transfer",
        "amount": 8.0,
         "datetime":  "2018-04-17T12:20+08:00"
      }
  ]
}
```

Please propose JSON responses for any errors that might occur.

Constraints

Time

Please feel free to submit your work any time, before the deadline. Please timebox yourself to a maximum of 12 hours for this activity.

Technology

You are required to use any of these languages: Java or Python.

You are allowed to use any frameworks for the language you chose. For Java, Spring boot is preferred. For Python, flask is preferred.

Testing

Please approach this exercise as you would in your day-to-day development workflow.

If you write tests in your daily work, we would love to see them in this exercise too.

Git and Commit History

Zip all the source code include the .git folder.

Please maintain a descriptive and clear Git commit history as it would allow us to better understand your thought process.

Follow Up

In the event that you are selected for the next round of interview (onsite chat), please be expected to discuss your work further with us during the interview.

Looking forward to seeing your code!