Backend Code Challenge

In this challenge, no fancy algorithm is necessary, but a clean code that you would do in your everyday life and is ready to be put into a productive environment.

Case study

You have a small company that provides money withdrawals from ATMs through APIs. After a quick registration, with name and email (use the email as the unique identifier), the user can start making withdrawals, with the following rules:

- The first money withdrawal can be a maximum of $50,00;
- The maximum amount for the subsequent money withdrawals must be $300,00;
- The minimum value for any money withdrawal is $1,00;
- The user can make a maximum of 5 money withdrawals in 24 hours;
- For each money withdrawal, the user must pay a fee according to the table below:

Withdrawl value	Fee
$1,00 – $100,99	3%
$101,00 –$ 250,99	2%
$251,00 – $300,00	1%

Important

At least two APIs are required:

- Create a new user
- make a withdraw

It is unnecessary to create/add funds to the user account. You can consider the user has unlimited resources.

The solution must be written in Java 11, using the spring-boot framework. A Gradle project is preferable over a maven, but it is up to you.

Provide a README.md with detailed instructions and commands on how to run your application;

Your solution must run "out of the box", without the need for manual configuration by the reviewer; That means that any external tool, like databases, for example, should be dockerized. And with one single command, the reviewer can start up the containers and the application.

Please do not make your solution available on a public SCM account (Github, GitLab, or similar). Send us a zip file by e-mail or by private cloud (Google Drive or One Drive);

Try not to invest more than 8 hours of your time in this challenge;

From: E&V Smart Money dev team