Tribal Coding Challenge - Backend

Thank you for taking the time to complete this challenge and considering joining our team.

The purpose of this challenge is to outline your programming skills.
We expect the project will take you a couple of hours to complete.

Once you finish it, please let us know to review the repository.

In case of any question or comment you can reach our team at engineering@tribal.credit.

Required Tools

We will provide you with a GitHub repository to push your code

Feel free to pick your preferred programming language and testing framework

The credit line

1. The problem

Determine the credit line of our customers is a key aspect on our daily basis.
Aimed to facilitate the process, Tribal needs a microservice to speed up the process of determining the credit line that better suits our customers.

2. The inputs

The credit line model has 5 inputs:

- Business type:
  It can be:
  - Startup
  - SME
- Cash balance
  The customer's bank account balance
- Monthly revenue
  The total sales revenue for the month
- Requested credit line
- Requested date
  Represents when request was made

Example:

```
{
  "foundingType": "SME",
  "cashBalance": 435.30,
  "monthlyRevenue": 4235.45,
  "requestedCreditLine": 100,
  "requestedDate": "2021-07-19T16:32:59.860Z"
}
```

3. Business rules

Risk management is a key aspect at Tribal, therefore to define the recommended credit line of a customer is defined using the following rules:

One third of the cash balance (3:1 ratio)
One fifth of the monthly revenue (5:1 ratio)

And depending on the founding type the rules are:

When it is SME, it's only needed to calculate the recommended credit line based one the monthly revenue.

And when it is Startup, the recommended credit line is the maximum value between the rule based on the cash balance and the monthly revenue.

Once the recommended credit line has been calculated, the next step is to determine whether the application is accepted or not.

If the recommended credit line is higher, then the application is accepted; otherwise, it is rejected

4. Non functional requeriments

- Once the application is accepted:
    - The accepted credit line will be the same regardless of the inputs.
    - If the system receives 3 or more requests within two minutes, return the http code 429.
- If the application was rejected:
    - Don't allow a new application requests within 30 seconds next to the previous one, if so, return HTTP code 429.
    - After failing 3 times, return the message "A sales agent will contact you".

5. The outputs

When accepted, the result should indicate that it was accepted and the credit line authorized, and when rejected, indicate it.

Upon Completion

Add your repo to GitHub and reach out to your point of contact at Tribal Credit
with the URL.
Include a README file explaining the general aspects of your code design and the instruction on how to run your application

Evaluation Criteria

- Understanding of coding best practices
- Good unit test cases