

Assignment Description

Your task is to build a two service system for product reviews.

Product service will expose RESTful API interfaces for manipulation with products and reviews.

Review processing service will handle calculation of average review product rating.

You are free to use any frameworks, databases, caching mechanisms and messaging brokers to develop a solution.

Service 1: Product service

- RESTful API for products with create, delete, edit, list and get by identifier actions
 - product information should not return reviews, only average product rating
- RESTful API for reviews with create, delete and edit actions
- Endpoints to show product reviews
- Service should notify review service when new review is added, modified or deleted

Service 2: Review processing service

- Service receives events from product service
- Each time review is received, it calculates average rating and stores it into persistent storage
- Running in 2 separate instances
 - design must be able to handle more than 2 instances
- Service must be able to process multiple events concurrently

Product reviews and average product ratings should be cached.

Product structure

- name
- description
- price
- list of reviews
- average rating (does not have to be directly a part of product)

Review structure

- first name
- last name
- review text
- rating (1 - 5)

What is provided as part of the implementation

- project setup using docker compose
- typescript configuration
- linter configuration

How to deliver the result

- A public Git repository with your code and configuration files
- Documentation that explains the thought process behind your implementation, including any design decisions or trade-offs you made, this can be done as part of project in [README.md](#) or as a PDF

Considerations

- the project itself should be easy to setup, extendable and maintainable by other developers
- pay attention to your commit messages

Please send the completed assignment to alena.blokhina@cloudtalk.io within 5 business days. Good luck!