Coding Challenge

Backend Engineering

Introduction

Thank you for applying for a Software Architect position at eBay Classifieds Group, and congratulations on the previous successful steps. Prior to your in-house interviews, we kindly ask you to complete the assignment described below. This exercise is designed to assess your development skills.

We will evaluate the work you send us against the criteria explained below.

This is not a timed assignment. However, please make sure that you send back the assignment by the deadline agreed upon with your interviewer.

Context

You are working at eBay and your team is asked to build a microservice that will calculate real-time statistics of item sales on eBay marketplace platform. This microservice will feed data to a dashboard installed in a business team's room.

The microservice shall have a REST interface with two endpoints. The first endpoint will be called by the checkout service whenever a new payment is received and the second endpoint will provide statistics about the total order amount and average amount per order for the last 60 seconds. (Orders between *t* and *t* - 60 sec, given *t* = request time)

Specifications for the requested endpoints are as follows:

Transactions

URL: /sales
Method: POST
Content-Type: application/x-www-form-urlencoded

Parameters:

Name	Required	Type
sales_amount	Yes	Number String (e.g. “10.00”)

Return HTTP Code: 202 Accepted
Return Body: *blank*

Statistics

URL: /statistics
Method: GET
Parameters: *none*
Return HTTP Code: 200 OK
Return Body:

```
{
   total_sales_amount: “1000000.00”,
   average_amount_per_order: “45.04”
}
```

Facts

This might seem to be an easy task from functional perspective, but like the most of engineering problems you would face at eBay, there are constraints you should keep in account:

- Business people are only interested in seeing the statistics for the last minute. They are not interested in seeing historical data, since they use this service only to create a real-time dashboard. The dashboard is updated once every second.

- In the eBay marketplace, around 250.000 items are sold each minute. So, your service must expect a high amount of transaction data per minute and on several TCP connections in parallel.

- For your coded solution, you don’t need to worry about the service running on several nodes. You can keep and process data in main memory. You are not allowed to use persistent storage due to internal INFOSEC policies. If the service should be restarted, losing this in-memory data is not a big problem since the risk is only to lose sales statistics for a couple of seconds, then new orders will arrive and your service

will again show statistics.

- The SiteOps team allocated limited cloud resources for this project. Use the main memory wisely. Note that, a high quantity of sales numbers will be delivered to your endpoint and you will need to think about a way to operate over this data with minimum memory.

- Your engineering manager wants the statistics endpoint to return results very fast. Since your CPU will be mainly busy with handling transactions, you should use also your CPU wisely while calculating the statistics. Think about ways of delivering the expected result with low time complexity.

How would you design and implement an efficient solution for these requirements?

Using the Java programming language, create the project from scratch. Decisions about choosing a build tool, a REST framework, testing approach etc. are left to you, but be prepared to defend these decisions during your in-person interviews.

Please incorporate your first name and last name initial in the project name (artifactId), e.g. salesstats-robertk.

Next steps

Please zip your code and any instructions the reviewer might need to run your application. Upload the file to a storage service (e.g. Google Drive) and send the link to DL-eCG-MoVe-Hiring-Backend@ebay.com. Include any instructions to build and run in a README file.

We will review your code and evaluate it against criteria such as **functional completeness**, **time and space efficiency**, **code structure and architecture**. Complete this assignment like you would complete a task at your workplace, documents parts that you think it is hard to understand for others.

After reviewing your assignment, we will share our feedback with you and communicate our decision on whether or not to proceed with in-person interviews.

When you come in for your interview, please be prepared to discuss your experience and answer questions regarding the work you provided and your approach to complete this project. We look forward to meeting you!