Calculate average latency

Your task is to create a web service that takes a period as input and calculates the average latency for different services (id 1 - 5), in theory it could be 1…N different services. This test should take between two and three hours to complete. We don't expect your solution to be perfect but still it should show good coding standards.

Your service should consist of only one endpoint and have startDate and endDate parameters:
GET /latencies?startDate=2021-01-02&endDate=2021-01-04

This endpoint should return an object containing

- The period that was calculated
- A list of objects (one for each service) containing the serviceId, number of requests and the average latency in milliseconds.

Example:

```
{
"period": ["2021-01-02", "2021-01-04"],
"averageLatencies": [
    {"serviceId": 1, "numberOfRequests": 120, "averageResonseTimeMs": 289},
    {"serviceId": 2, "numberOfRequests": 111, "averageResonseTimeMs": 412},
    {"serviceId": 3, "numberOfRequests": 130, "averageResonseTimeMs": 222},
    {"serviceId": 4, "numberOfRequests": 160, "averageResonseTimeMs": 634},
    {"serviceId": 5, "numberOfRequests": 90, "averageResonseTimeMs": 124}
  ]
}
```

For a given date, for example 2021-01-01, the data can be retrieved here:
GET http://latencyapi-env.eba-kqb2ph3i.eu-west-1.elasticbeanstalk.com/latencies?date=2021-01-01
To get data for another date, change the date path variable.

```
[
  {"requestId":69,"serviceId":1,"date":"2021-01-01","milliSecondsDelay":421},
  {"requestId":91,"serviceId":1,"date":"2021-01-01","milliSecondsDelay":404},
  [...]
]
```

Meaning that the above endpoint will for the period [”2021-01-02”, “2021-01-04”] be called 3 times, one time for each day. The service does only work for the year 2021.

NOTE: That the data may contain duplicate requestIds. These need to be filtered out in your service.

Your solution should include tests on the most important parts of the code.