Alert Monitor

Problem Statement

Consider a microservice env where several systems use 1 central system for all their alert use cases: Design and implement a system which can capture events generated by any system or user triggered (for simplicity), and raise an alert according to alert configuration.

Type of configurations:

SIMPLE_COUNT (which consists of count)
TUMBLING_WINDOW (which consists of count and windowSizeInSecs, Eg: 10 events in 1 hour bucket, you can assume that the bucket starts at 00:00 hour of the day)
SLIDING_WINDOW (which consists of count and windowSizeInSecs)

Example client configuration:

```
alertConfigList:
[{
      client: X
      eventType: PAYMENT_EXCEPTION,
      alertConfig {
            type: TUMBLING_WINDOW,
            count: 10
            windowSizeInSecs: 10
      }
      “dispatchStrategyList”: [{
            “type”: CONSOLE,
            “message” : “issue in payment”
      },{
            “type”: EMAIL,
            “subject” : “payment exception threshold breached”
      }]
},{
      client: X
      eventType: USERSERVICE_EXCEPTION,
      alertConfig {
            type: SLIDING_WINDOW,
            count: 10
            windowSizeInSecs: 10
      }
```

```
     “dispatchStrategyList”: [{
          “type”: CONSOLE,
          “message” : “issue in user service”
     }]
}]
```

Sample Execution

- Alert will be dispatched if the config thresholds are met
- You can design your own config.
- You need to keep a placeholder for defining different dispatch mechanisms. (You may provide implementation for console dispatch)
- Simulate the dispatch of alerts by generating events in a main class / test case

Sample Output

```
[INFO] MonitoringService: Client X PAYMENT_EXCEPTION TUMBLING_WINDOW starts
[INFO] MonitoringService: Client X PAYMENT_EXCEPTION TUMBLING_WINDOW ends
[INFO] MonitoringService: Client X PAYMENT_EXCEPTION TUMBLING_WINDOW starts
[INFO] MonitoringService: Client X PAYMENT_EXCEPTION threshold breached
[INFO] AlertingService: Dispatching to Console
[WARN] Alert: `issue in payment`
[INFO] AlertingService: Dispatching an Email
[INFO] MonitoringService: Client X USERSERVICE_EXCEPTION threshold breached
[INFO] AlertingService: Dispatching to Console
[WARN] Alert: `issue in user service`
[INFO] MonitoringService: Client X PAYMENT_EXCEPTION TUMBLING_WINDOW ends
```

Evaluation

- Code modularity, readability and testability
- Refactored code
- Abstraction
- Language proficiency.
- Functional coverage
- Separation of concerns
- Application of OO design principles

- Data structures and optimal time complexity