|  |  |  |
| --- | --- | --- |
| **TEKNIKAL TEST**  DEVELOPER | **Tanggal** | 18 July 2023 |
| **Nama** | Ahmad Dzaky Syaddad |
|  | |

1. What is your most mastered skill? Please elaborate.

|  |
| --- |
| Java, particularly in Spring Boot Rest API development, I have previously handled two projects: the "Dbank Registration app" and the "General Ledger System." The "Dbank Registration app" was designed to onboard potential Bank Danamon customers, allowing them to become Danamon account holders. The application utilized web and mobile interfaces as the front-end, while the back-end was built using Spring Boot (REST API) and PostgreSQL. On the other hand, the "General Ledger System" was an accounting bookkeeping application used by users at Mandiri Utama Finance. This application featured a web front-end and utilized Spring Boot (REST API) as the back-end, with Oracle as the database. Throughout 2021 up to the present, my primary programming language has been Java, and I have gained significant expertise in this area. These projects have allowed me to sharpen my Java skills, and I am well-versed in building Rest APIs using Spring Boot for various applications. |

1. Analysis - Elaborate your way to work out the alert.

Given a transactional table that has 300 million of records and insert rate of 40 tps (transaction/second). The transactions status varied from success, failed, decline, and suspect. There is condition of alert:

- within 5 minutes, if the sum of failed transactions > 100 records, send alert to officer on duty that tells

- ex: there are 120 failed transactions between now and (now-5 minutes) by telegram automatically.

|  |
| --- |
| Based on the information provided, we can implement a solution that processes and stores data with the 'failed' status in a temporary table. The temporary table will store the failed rows for a duration of 5 minutes. After 5 minutes, if the condition is met, the application will integrate with the Telegram bot to send an alert containing the failed transaction details. Once the alert is sent, the application will clean up the data in the temporary table.  Here's an outline of the solution:  Create a Temporary Table:  Set up a temporary table in the database to store the failed transaction details. The table should have columns similar to the original transactional table to store relevant information.  Data Processing:  Continuously monitor new transaction records and identify those with a 'failed' status. Insert the failed transaction details into the temporary table.  Time-Based Cleanup:  Implement a scheduled task or background process to perform cleanup in the temporary table. Set the schedule to execute every 5 minutes. The cleanup process should remove all records from the temporary table that are older than 5 minutes or after send all details row to telegram bot, effectively maintaining only the recent failed transactions.  Alert Condition Check:  When a new transaction with a 'failed' status is inserted into the temporary table, check if the sum of failed transactions in the table exceeds 100. If the condition is met in 5 minutes, trigger the integration with the Telegram bot.  Telegram Bot Integration:  Create a Telegram bot and obtain the necessary authentication token to interact with the Telegram API.  Implement a method to send messages containing the failed transaction details to the designated officer on duty.  Alerting Process:  When the condition for sending an alert is met, retrieve all the failed transaction details from the temporary table. Format the data and compose the alert message. Send the alert message to the Telegram bot using the previously implemented method.  Data Cleanup:  After sending the alert, clean up the temporary table by removing the processed failed transaction records. By following this approach, the application will efficiently process and manage failed transaction data, send timely alerts, and ensure the temporary table remains relevant for a 5-minute window, maintaining a clean and efficient process. |

1. Native Java Program on folder name “SignOnReminder.”
2. Database – PL/SQL on file name “PRC\_INSERT\_DATA\_TABLE\_A\_B\_TO\_C”
3. REST API – GUI SIMULATOR on folder name “GUIandSample”