Performance Testing Report

* Project Name : Contact List App
* Date : 11 Juni 2025
* Tested by : Ahmad Nizar Riyansyah Sidik
* Environment : Public API
* Tools Used : Apache JMeter v5.6.3

Target Endpoint :

1. <https://thinking-tester-contact-list.herokuapp.com/users> (add user)
2. <https://thinking-tester-contact-list.herokuapp.com/users/login> (login)
3. <https://thinking-tester-contact-list.herokuapp.com/users/me> (view user profile)
4. <https://thinking-tester-contact-list.herokuapp.com/contacts> (add contact)
5. [https://thinking-tester-contact-list.herokuapp.com/contacts/${contactid}](https://thinking-tester-contact-list.herokuapp.com/contacts/$%7bcontactid%7d) (view detail contact)
6. [https://thinking-tester-contact-list.herokuapp.com/contacts/${contactid}](https://thinking-tester-contact-list.herokuapp.com/contacts/$%7bcontactid%7d) (Update contact)
7. [https://thinking-tester-contact-list.herokuapp.com/contacts/${contactid}](https://thinking-tester-contact-list.herokuapp.com/contacts/$%7bcontactid%7d) (patch data contact)
8. [https://thinking-tester-contact-list.herokuapp.com/contacts/${contactid}](https://thinking-tester-contact-list.herokuapp.com/contacts/$%7bcontactid%7d) (delete contact)
9. [https://thinking-tester-contact-list.herokuapp.com/contacts/${contactid}](https://thinking-tester-contact-list.herokuapp.com/contacts/$%7bcontactid%7d) (delete user)

## Objective

## Tujuan dari pengujian performa ini adalah untuk mengevaluasi kondisi API pada contact list App saat menerima beban pengguna yang meningkat, mulai dari 25 hingga 75 Pengguna Virtual (VU) dalam durasi tetap 20 detik per skenario. Tujuan utamanya adalah untuk menilai skalabilitas, konsistensi respons, dan batas kapasitasnya.

## Test Scenario

|  |  |  |  |
| --- | --- | --- | --- |
| **Scenario** | **Virtual User** | **Rump-up Periode (s)** | **Duration (ms)** |
| 1 | 25 | 20 | 2000 |
| 2 | 50 | 20 | 2000 |
| 3 | 75 | 20 | 2000 |

## Key Matric Captured

* **Average Response Time (ms)**
  + Rata-rata waktu yang dibutuhkan server untuk merespons setiap permintaan.
  + Semakin **rendah** nilainya, semakin **baik**.
  + Contoh: **200 ms** berarti server biasanya merespons dalam waktu **0,2 detik**.
* **90th Percentile Response Time (ms)**
* 90% dari total permintaan selesai **pada atau di bawah waktu ini**, hanya 10% yang lebih lambat.
* Memberikan gambaran **lebih realistis tentang pengalaman pengguna** dibanding rata-rata saja.
* Contoh: Jika P90 = 1180 ms, artinya 90% pengguna mendapat respons lebih cepat dari 1180 ms.
* **Max Response Time (ms)**
* Waktu respon **terlama** yang tercatat selama pengujian.
* Menunjukkan skenario **terburuk**, bisa jadi karena bottleneck atau lonjakan beban.
* Contoh: **2500 ms** berarti ada satu permintaan yang butuh **1.5 detik** untuk dijawab.
* **Throughput (requests/second)**
  + Jumlah permintaan yang dapat ditangani server setiap detik.
  + Mencerminkan **kapasitas sistem di bawah beban**.
  + Contoh: **1000 req/s** berarti sistem memproses **1000 permintaan per detik**.
* **Error Rate (%)**
  + Persentase permintaan yang **gagal** (misalnya timeout, HTTP 500).
  + Harus **mendekati 0%** untuk menunjukkan kestabilan sistem.
  + Contoh: **2% dari 10.000 permintaan = 200 permintaan gagal**.

## Test Result Summary

## Detailed Test Result

### Scenario 1 25 - VU

#### Request Summary

#### Response Time Percentiles

#### Response Time Overview

#### Response Time Percentiles Over Time (successful response)

#### Throughput Hits Per Second

#### Transaction Per Second

#### Response Time vs request

## 2️⃣ **Observation**

* **CPU Usage (Blue Line)**
* **Fluktuasi tinggi**, menunjukkan bahwa CPU load naik-turun secara signifikan selama pengujian.
* Nilai tertinggi mencapai:
* ±36% peak CPU usage.
* Nilai terendah sempat turun sampai:
* ±3% - 5% usage.
* Pola grafik memperlihatkan:
* Ada burst load (lonjakan CPU usage) yang terjadi berkala.
* Ketika request load sedang tinggi, CPU usage spike secara tajam.
* Setelah spike, CPU usage kembali turun dengan cepat — menunjukkan resource CPU mampu recovery secara baik.
* **Interpretasi:**  
  Beban CPU masih dalam kapasitas wajar (belum mencapai over-utilization). Server masih memiliki cukup idle capacity. Tidak ditemukan bottleneck CPU.
* **🔴 Memory Usage (Red Line)**
* **Relatif stabil** selama pengujian.
* Penggunaan memory berkisar di sekitar:
* ±53% - 54% sepanjang load test.
* Tidak terdapat spike / leak memory selama proses berjalan.
* Kenaikan minor sekitar menit ke-2 hingga ke-4, namun kembali stabil.
* **Interpretasi:**  
  Memory management cukup baik, tidak terjadi memory leak atau growth yang abnormal. Garbage Collection berjalan normal.
* **3️⃣ Conclusion**
* CPU dan Memory masih dalam kapasitas aman.
* Tidak ditemukan indikasi resource starvation selama test berlangsung.
* Fluktuasi CPU disebabkan oleh sifat burst request dari load test yang memang intermiten.
* Monitoring ini menunjukkan sistem server target masih capable menangani load dengan jumlah VU yang digunakan saat ini.
* Untuk pengujian lebih lanjut disarankan:
* Menambahkan jumlah virtual users.
* Melakukan long-duration test (endurance test).
* Monitoring resource lain seperti disk I/O & network bandwidth.