



Virtual Internship Experience

Task 5 Report

-

Nelly Yulia



Outline

- 1 Question**
- 2 Result**
- 3 Report**
- 4 Conclusion**



1. Soal

Task 5

- API Create
 - Base url : <https://reqres.in>
 - Path url : /api/users
 - Method : POST
 - Header : application/json
 - Request body :

```
{  
  "name": "morpheus",  
  "job": "leader"  
}
```
- API Update
 - Base url : <https://reqres.in>
 - Path url : /api/users/2
 - Method : PUT
 - Header : application/json
 - Request body :

```
{  
  "name": "morpheus",  
  "job": "zion resident"  
}
```

Task yang akan dikerjakan :

1. Buat scenario test untuk integration test dari 2 API tersebut dan implementasikan ke dalam k6 serta tambahkan assertion dari tiap test yang dilakukan dalam 1 file test
2. Buat scenario test untuk menguji performance dari 2 API tersebut dengan total 1000 virtual user, 3500 iterasi dan batas maksimum toleransi response APInya 2 second serta tambahkan assertion dari tiap test yang dilakukan dalam 1 file test
3. Generate report dalam bentuk HTML dari soal no. 2 untuk mendapatkan visualisasi dari performance test yang dilakukan

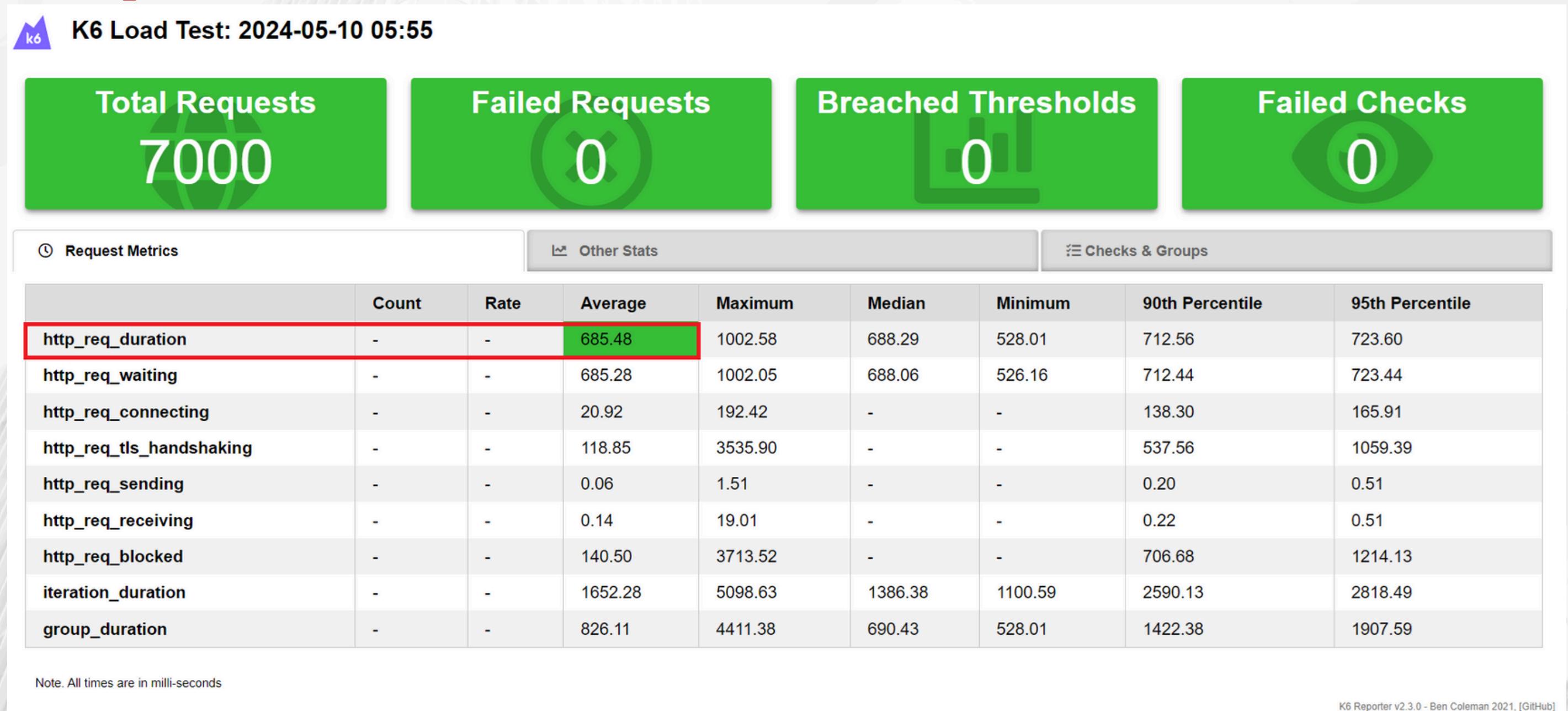
2. Result

Hasil: <https://github.com/nellyyuliaa/task5-QA-Evermos>

Dalamnya terdapat beberapa file yaitu `http_post.js`, `http_put.js`, `integration.js`, dan `performance.js` – masing-masing memiliki fungsi khusus dalam *k6 load test* dan bersama-sama menghasilkan `report.html`. File `http_post.js` menguji permintaan HTTP POST, `http_put.js` menguji pembaruan data melalui HTTP PUT, `integration.js` menangani pengujian integrasi antar komponen, dan `performance.js` fokus pada pengujian performa keseluruhan sistem. Hasil dari semua pengujian ini disajikan dalam `report.html`, yang memberikan ringkasan komprehensif dan analisis dari berbagai tes yang dilakukan.

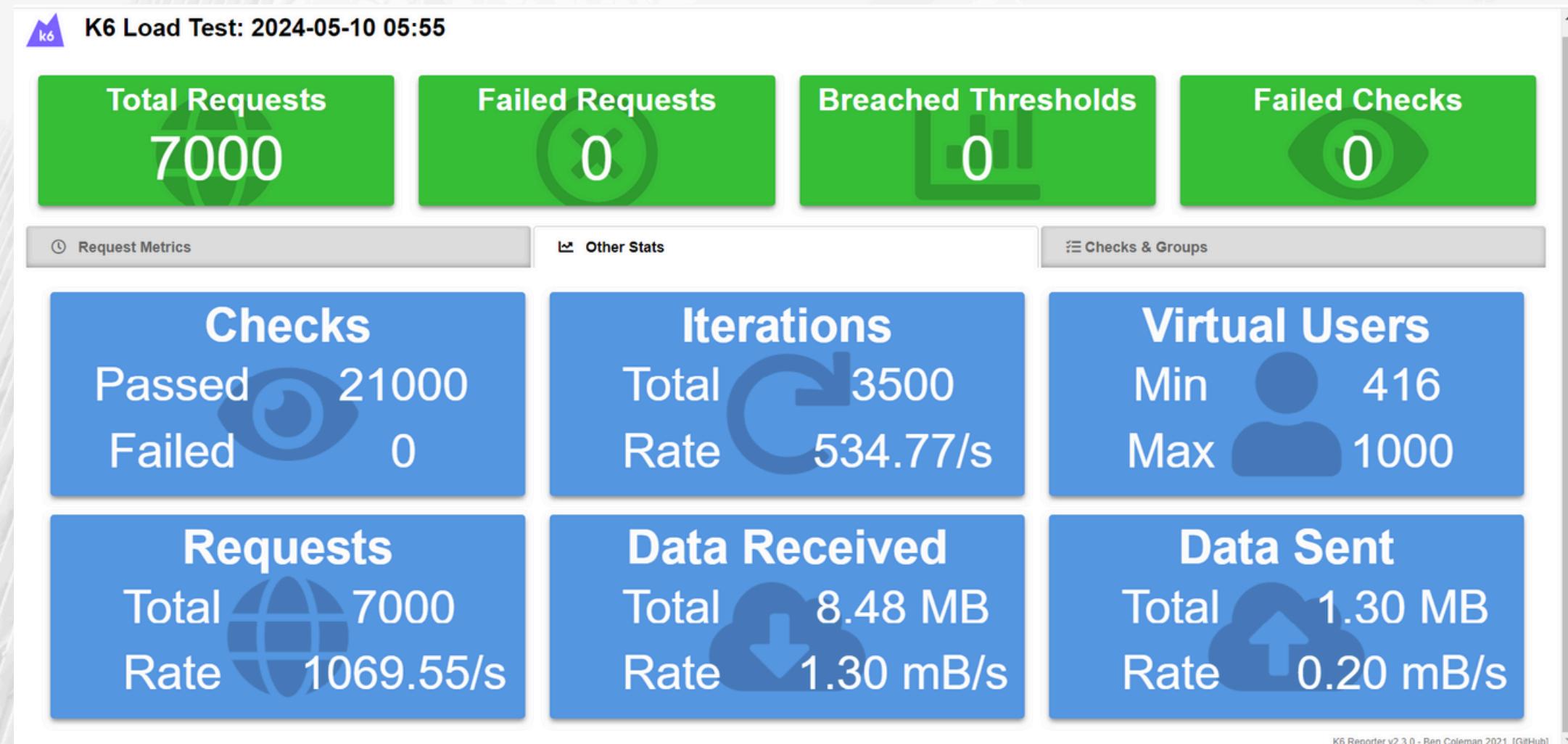


3. Report



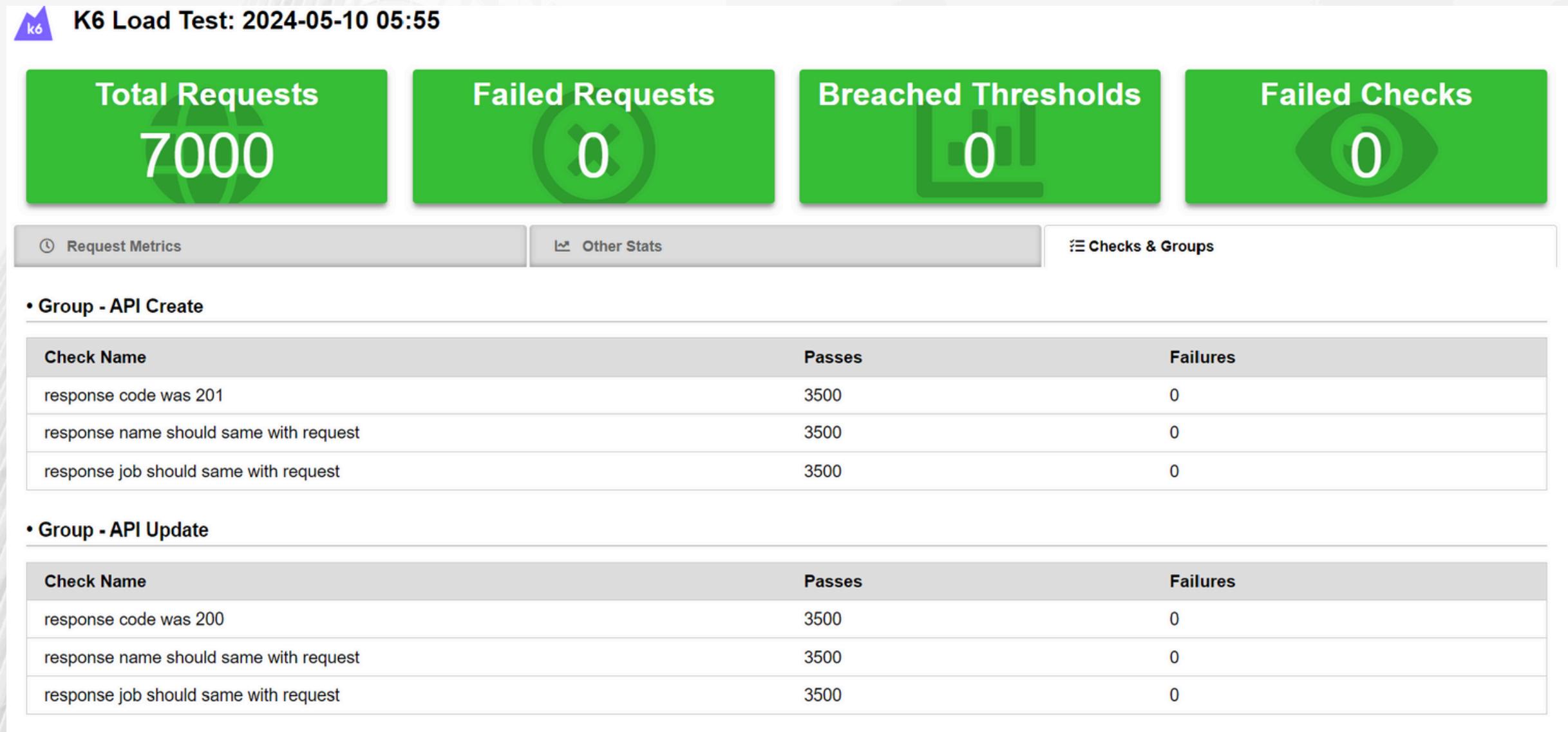
Hasilnya menunjukkan sistem yang aman dan stabil dengan 7000 permintaan tanpa kegagalan, waktu respons rata-rata 685.48 milidetik yang cepat dan konsisten, serta durasi maksimum di bawah 1.5 detik. Performa di percentil ke-90 dan ke-95 menandakan sistem mampu menangani beban mendekati puncak dengan kapasitas yang cukup untuk lonjakan trafik lebih besar.

4. Report



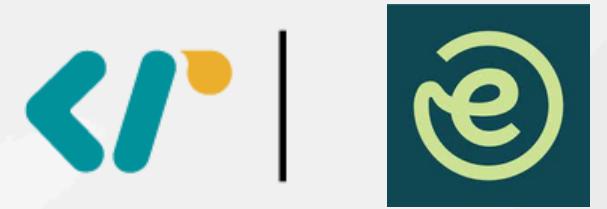
Hasil menunjukkan sistem yang efisien dan stabil, dengan 7000 permintaan diproses pada rate 1069.55/s dan 3500 iterasi pada rate 534.77/s. Jumlah pengguna virtual berkisar antara 416 hingga 1000, menunjukkan kapasitas sistem untuk menangani banyak pengguna secara simultan. Sistem berhasil menangani 21000 checks tanpa kegagalan. Data yang diterima selama tes adalah 8.48 MB pada rate 1.30 MB/s, sementara data yang dikirim adalah 1.30 MB pada rate 0.20 MB/s, menunjukkan pengelolaan data yang baik dalam kondisi beban tinggi.

4. Report



Hasil ini menunjukkan bahwa semua checks termasuk kode respons, kesesuaian nama, dan kesesuaian pekerjaan lulus tanpa kegagalan, dengan masing-masing grup mencatat 3500 passes. Hal ini mengonfirmasi keandalan dan konsistensi API dalam menghasilkan respons yang diharapkan selama tes, menandakan bahwa sistem telah memenuhi permintaan yang dibuat dengan tepat.

5. Conclusion



Hasil pengujian menggunakan K6 ini menunjukkan sistem yang efisien dan stabil. Melalui serangkaian tes menggunakan file `http_post.js`, `http_put.js`, `integration.js`, dan `performance.js`, sistem berhasil menangani 7000 permintaan tanpa kegagalan, dengan waktu respons rata-rata 685.48 milidetik. Hasil pengujian mencatat kemampuan sistem untuk mengelola beban puncak dan data dengan baik, menampilkan performa yang memadai bahkan di bawah kondisi beban tinggi. Keseluruhan tes memverifikasi keandalan dan konsistensi API dalam menghasilkan respons yang diharapkan, memastikan bahwa sistem beroperasi dengan efektif.



Thank You!

