

BAB 5

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil pengujian performa menggunakan k6 serta pemantauan sumber daya melalui Grafana Cloud, diperoleh bahwa framework AdonisJS menunjukkan performa yang secara konsisten lebih baik dibandingkan dengan Laravel dalam berbagai aspek pengujian.

Dari sisi Request per Second (RPS), AdonisJS mampu mencapai rata-rata 90,70 request/detik, sedangkan Laravel hanya sekitar 8,24 request/detik. Hal ini menunjukkan bahwa throughput AdonisJS sekitar 11 kali lipat lebih tinggi dibandingkan Laravel.

Untuk aspek latensi atau waktu respon, AdonisJS juga menunjukkan hasil yang jauh lebih cepat dan stabil dengan rata-rata waktu respon hanya 0,89 detik, sementara Laravel memiliki rata-rata latensi sebesar 17,25 detik. Perbedaan yang signifikan ini menunjukkan bahwa waktu respon Laravel sekitar 19 kali lebih lambat, atau dengan kata lain AdonisJS hampir 19 kali lebih cepat dalam merespons permintaan meskipun jumlah Virtual User meningkat.

Dari sisi penggunaan sumber daya CPU, AdonisJS mencatat rata-rata penggunaan sebesar 0,98%, sedikit lebih rendah dibandingkan Laravel yang mencapai 2,8%. Jika dilihat secara perbandingan, Laravel menggunakan hampir 3 kali lipat CPU yang dibutuhkan AdonisJS, sehingga dapat dikatakan AdonisJS lebih efisien dalam pengolahan proses dan pemanfaatan CPU.

Perbedaan lain terlihat pada konsumsi memori, di mana Laravel menggunakan rata-rata 126,75 MB, sedangkan AdonisJS hanya sekitar 254,75 MB. Dengan kata lain, penggunaan memori AdonisJS sekitar 2 kali lipat lebih besar dibandingkan Laravel. Walaupun demikian, efisiensi dan kestabilan AdonisJS dalam menangani beban tetap lebih baik, sehingga

peningkatan konsumsi memori tersebut dapat dikompensasikan oleh kemampuan throughput dan latensi yang jauh lebih unggul dibandingkan Laravel.

Secara keseluruhan, dapat disimpulkan bahwa AdonisJS lebih unggul dibandingkan Laravel dalam hal throughput, latensi, serta efisiensi penggunaan sumber daya. Laravel masih dapat digunakan dengan baik pada skala kecil hingga menengah, namun performanya kurang optimal untuk aplikasi dengan beban traffic tinggi seperti sistem lelang agro yang melibatkan banyak transaksi dan permintaan simultan.

5.2 Saran

Berdasarkan hasil penelitian yang telah dilakukan, terdapat beberapa saran yang dapat diberikan, antara lain:

Framework AdonisJS dapat dipertimbangkan sebagai pilihan utama karena terbukti lebih efisien dan responsif dalam menangani beban pengguna yang tinggi. Namun, apabila menggunakan Laravel, disarankan untuk melakukan optimasi khusus pada pengelolaan memori dan kecepatan respon.

Disarankan agar pengujian dilakukan pada lingkungan production server dengan jumlah pengguna virtual yang lebih besar, serta menambahkan parameter uji lain seperti *error rate* dan *throughput stability* agar hasil penelitian semakin komprehensif. Ataupun menambahkan *framework* lain agar hasil lebih variatif dan beragam seperti menambahkan Golang, ExpressJS, maupun NestJS. Dapat juga menambahkan pengujian antar *tools* misalnya seperti menambahkan perbandingan antara penggunaan *tools* k6 dengan Apache Jmeter.