

BAB V

SIMPULAN, IMPLIKASI, DAN REKOMENDASI

5.1 Simpulan

Berdasarkan hasil penelitian yang dilakukan, didapatkan kesimpulan sebagai berikut:

1. Berdasarkan hasil pengujian performa untuk aplikasi presensi pegawai UNISMA yang menggunakan arsitektur REST API, *framework* Gin menunjukkan performa yang paling konsisten dan unggul dibandingkan dengan Express.js dan Nest.js. Gin memiliki rata-rata waktu respons tercepat sebesar 691,07 ms dan *throughput* tertinggi sebesar 37,27 *request/s*, meskipun dengan variabilitas yang lebih besar. Gin sangat cocok untuk aplikasi yang membutuhkan waktu respons cepat dan kemampuan menangani banyak permintaan secara bersamaan, seperti *check-in* pada aplikasi presensi. Sementara itu, Express.js dan Nest.js, meskipun memiliki waktu respons dan *throughput* yang lebih lambat dibandingkan Gin, tetap menunjukkan performa yang memadai. Di sisi lain, Nest.js memiliki penggunaan CPU dan memori yang paling stabil, menjadikannya pilihan yang baik untuk aplikasi yang membutuhkan stabilitas tinggi dalam penggunaan sumber daya. Untuk kebutuhan *endpoint check-in* pada aplikasi presensi pegawai UNISMA, Gin adalah pilihan yang paling optimal karena menawarkan waktu respons yang lebih cepat dan *throughput* yang lebih tinggi, memastikan pengalaman pengguna yang lebih baik serta kemampuan penanganan beban kerja yang lebih besar.
2. Berdasarkan analisis performa, dapat disimpulkan bahwa Gin menunjukkan efisiensi tertinggi dalam *response time* dan *throughput*, meskipun memiliki variabilitas yang lebih besar. Gin juga unggul dalam penggunaan memori dengan efisiensi yang baik dan stabilitas yang tinggi. Pada metrik *CPU load*, Nest.js menunjukkan efisiensi tertinggi, meskipun perbedaan tersebut tidak signifikan dibandingkan dengan *framework* lainnya. Gin adalah pilihan yang paling efisien untuk digunakan pada aplikasi presensi pegawai UNISMA karena

Fathoni Zikri Nugroho, 2024
ANALISIS PERBANDINGAN PERFORMA BACK-END EXPRESS.JS, NEST.JS, DAN GIN PADA APLIKASI PRESENSI PEGAWAI UNISMA

Universitas Pendidikan Indonesia | repository.upi.edu | perpustakaan.upi.edu

memberikan performa yang lebih cepat dan stabil dalam penanganan permintaan serta penggunaan memori yang lebih rendah. Sementara itu, Nest.js dapat menjadi alternatif yang baik jika efisiensi penggunaan CPU menjadi prioritas utama. Meskipun Express.js stabil, efisiensinya lebih rendah dibandingkan dengan Gin dan Nest.js.

5.2 Implikasi

Implikasi yang dapat diambil adalah bahwa penelitian ini memberikan landasan yang kuat bagi UNISMA dalam memilih *framework back-end* yang paling efisien untuk aplikasi presensi pegawai khususnya kebutuhan *check-in*. Dengan informasi rinci tentang perbandingan performa Express.js, Nest.js, dan Gin, UNISMA dapat membuat keputusan yang didasarkan pada data untuk memastikan efisiensi dan kinerja optimal dari aplikasi presensi.

5.3 Rekomendasi

Beberapa rekomendasi atau saran yang dapat diberikan untuk penelitian selanjutnya adalah sebagai berikut:

1. Mengimplementasikan *framework* Gin sebagai *back-end* untuk aplikasi presensi pegawai UNISMA karena efisiensi dan kinerjanya yang unggul.
2. Perlu adanya pemantauan dan optimasi berkala untuk menjaga aplikasi tetap optimal dan sesuai dengan kebutuhan operasional yang terus berkembang.
3. Lakukan penelitian terhadap *framework back-end* lain yang belum diuji untuk mengetahui apakah ada alternatif yang lebih efisien.
4. Penelitian ini hanya menguji *endpoint check-in* saja, ke depannya bisa menguji *endpoint* terkait presensi yang lainnya.
5. Jika ingin melakukan penelitian dari sisi *front-end*, diperlukan penelitian lebih lanjut sebelum melakukan pengujian sisi *client* aplikasi ini.