

## **BAB 5** **KESIMPULAN DAN SARAN**

### **5.1 Kesimpulan**

Berdasarkan hasil penelitian dan analisis terhadap sistem *microservices* API berbasis metadata pada sistem ERP PT Sucofindo dengan *framework* FastAPI dan Gin, diperoleh kesimpulan sebagai berikut:

1. FastAPI dan Gin berhasil mengimplementasikan *microservices* API dan lulus uji fungsionalitas. Dari sisi kinerja, Gin menghasilkan response time dan latency lebih rendah di semua tingkat beban, sedangkan FastAPI lebih tinggi namun tetap stabil karena proses validasi dan pemrosesan metadata.
2. Keduanya skalabel saat beban meningkat. Gin lebih efisien per request dengan penggunaan CPU lebih rendah, sementara FastAPI mencapai throughput lebih tinggi pada beban puncak melalui mekanisme asynchronous.
3. FastAPI unggul dalam kemudahan pengembangan, dukungan *async*, dan validasi data, tetapi lebih berat dalam penggunaan sumber daya. Gin unggul dalam kecepatan dan efisiensi runtime, namun validasi bawaannya lebih minimal.
4. Pengujian menunjukkan bahwa jalur pemrosesan request yang lebih ringkas menghasilkan latency lebih rendah, sedangkan pipeline yang lebih lengkap meningkatkan waktu respons namun memberi kontrol dan stabilitas lebih baik.

5. Model concurrency asynchronous terbukti meningkatkan throughput total pada beban puncak, meskipun tidak selalu menghasilkan waktu respons per request tercepat, sebagaimana terlihat pada FastAPI.
6. Semakin kompleks pipeline pemrosesan, semakin tinggi penggunaan CPU, tetapi tidak selalu meningkatkan penggunaan memori. FastAPI lebih tinggi di CPU namun stabil di memori, sedangkan Gin lebih hemat CPU tetapi lebih besar penggunaan memorinya.
7. Gin lebih sesuai untuk layanan ERP berlatensi rendah dan efisien sumber daya, sedangkan FastAPI cocok untuk layanan dengan validasi kuat dan kebutuhan throughput tinggi. Pemilihan framework perlu disesuaikan dengan karakteristik beban layanan.

## 5.2 Saran

Berdasarkan hasil penelitian dan pengujian yang telah dilakukan, terdapat beberapa saran untuk pengembangan dan penelitian selanjutnya:

1. Pengujian lanjutan dilakukan pada lingkungan *cloud/container orchestration* untuk mengukur skalabilitas horizontal.
2. Gunakan skenario *mixed workload* agar pola uji lebih mendekati trafik ERP nyata.
3. Tambahkan *dashboard* Grafana untuk visualisasi metrik Prometheus.
4. Perluasan metrik uji seperti I/O, *garbage collection*, dan *fault tolerance*.
5. Mekanisme autentikasi API Key sebaiknya ditingkatkan menjadi JWT atau OAuth2 agar keamanan akses lebih kuat.