

BAB V KESIMPULAN & SARAN

5.1 Kesimpulan

Berdasarkan seluruh rangkaian tahapan penelitian yang telah dilaksanakan, penelitian ini berhasil membangun dan memvalidasi arsitektur *Software as a Service (SaaS) multi-tenant* untuk Sistem Informasi Akademik (SIKAD). Merujuk pada hasil analisis data dan pembahasan pada bab sebelumnya, maka dapat ditarik kesimpulan sebagai berikut:

1. Penelitian ini telah berhasil merancang arsitektur *SaaS multi-tenant* berbasis *Shared Database* di atas *stack* teknologi ASP.NET Core 9.0. Melalui hasil pengujian integritas (ARC-01) dan akses data (ISO-01), arsitektur ini terbukti mampu menjamin keamanan isolasi data yang setara dengan model *Database-per-Tenant*. Hal ini dicapai melalui penerapan mekanisme *Global Query Filters* dan lapisan *middleware*.
2. Mekanisme *Global Query Filters (GQF)* terbukti efektif dalam menegakkan isolasi data antar *tenant* secara otomatis dan sistematis di tingkat basis data. Berdasarkan hasil pengujian fungsional dan isolasi (ISO-02), sistem terbukti mampu mencegah upaya akses lintas *tenant* (sekolah) dengan memberikan respons *404 Not Found*.
3. Penerapan pola *Queue-Based Load Leveling* terbukti menjadi solusi yang valid untuk memitigasi fenomena *Noisy Neighbor*. Data menunjukkan arsitektur asinkron (*proposed*) mampu mencapai tingkat keberhasilan status "GURU OK" sebesar 70% (1.151 permintaan berhasil) dibandingkan dengan arsitektur sinkron (*baseline*) yang memiliki tingkat keberhasilan 0% pada kondisi beban kerja ekstrem. Dengan demikian, arsitektur asinkron mampu menangani operasi penulisan data massal (*Heavy Write*) secara efektif sembari menjaga ketersediaan layanan tetap stabil bagi pengguna lain.

5.2 Saran

Berdasarkan temuan penelitian, penulis merumuskan tiga saran teknis utama untuk pengembangan sistem lebih lanjut. Pertama, disarankan untuk mengganti mekanisme antrian berbasis memori (*In-Memory Channel*) dengan *distributed message broker* seperti RabbitMQ atau Redis guna menjamin persistensi data dan mencegah *data loss*. Kedua, disarankan untuk mengintegrasikan notifikasi *real-time* melalui SignalR agar sistem dapat memberikan umpan balik status akhir pemrosesan asinkron kepada pengguna yang saat ini hanya menerima respons kode status *HTTP*. Terakhir, diperlukan pengembangan dasbor pemantauan (*Dashboard Monitoring Job*) untuk memudahkan administrator dalam memantau kesehatan mekanisme antrian serta menangani kegagalan pemrosesan data.

