

**MITIGASI NOISY NEIGHBOR DAN ISOLASI DATA
PADA ARSITEKTUR MULTI-TENANT
MENGUNAKAN QUEUE-BASED LOAD LEVELING
DAN GLOBAL QUERY FILTERS**

SKRIPSI SARJANA INFORMATIKA

Oleh

Timotius Lucky Darmawan

227064516004



**PROGRAM STUDI INFORMATIKA
FAKULTAS TEKNOLOGI KOMUNIKASI DAN
INFORMATIKA
UNIVERSITAS NASIONAL
2025**

**MITIGASI NOISY NEIGHBOR DAN ISOLASI DATA
PADA ARSITEKTUR MULTI-TENANT
MENGUNAKAN QUEUE-BASED LOAD LEVELING
DAN GLOBAL QUERY FILTERS**

SKRIPSI SARJANA

Karya ilmiah sebagai salah satu syarat untuk memperoleh gelar
Sarjana Informatika dari Fakultas Teknologi Komunikasi dan Informatika

Oleh

Timotius Lucky Darmawan

227064516004



**PROGRAM STUDI INFORMATIKA
FAKULTAS TEKNOLOGI KOMUNIKASI DAN
INFORMATIKA
UNIVERSITAS NASIONAL
2025**


HALAMAN PERNYATAAN ORISINALITAS

Skripsi ini adalah hasil karya saya sendiri, dan semua sumber baik yang dikutip maupun dirujuk telah saya nyatakan dengan benar. Bilamana di kemudian hari ditemukan bahwa karya tulis ini menyalahi peraturan yang ada berkaitan etika dan kaidah penulisan karya ilmiah yang berlaku, maka saya bersedia dituntut dan diproses sesuai dengan ketentuan yang berlaku

Yang Menyatakan,

Nama : Timotius Lucky Darmawan

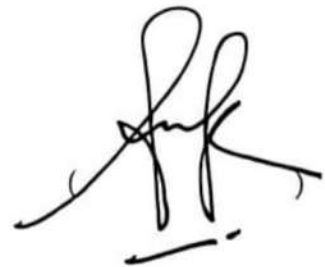
NPM : 227064516004

Tanda Tangan : 

Tanggal : 27 Februari 2026

Mengetahui

Dosen Pembimbing I : Dr. Ucuk Darusalam, S.T., M.T



Dosen Pembimbing II : Rima Tamara Aldisa, S.Kom., M.Kom



HALAMAN PENGESAHAN

TUGAS SARJANA

MITIGASI NOISY NEIGHBOR DAN ISOLASI DATA PADA ARSITEKTUR MULTI-TENANT MENGUNAKAN QUEUE-BASED LOAD LEVELING DAN GLOBAL QUERY FILTERS



Pembimbing I

Dr. Ueuk Darusalam, S.T., M.T

NIDN. 0327047903

Pembimbing II

Rima Tamara Aldisa, S.Kom., M.Kom

NIDN. 0318019401

KATA PENGANTAR

Puji syukur penulis ucapkan kehadirat Tuhan Yang Maha Esa atas segala rahmat dan karunia-Nya, sehingga penulis dapat menyelesaikan penyusunan skripsi dengan judul “Mitigasi *Noisy Neighbor* dan Isolasi Data pada Arsitektur *Multi-Tenant* Menggunakan *Queue-Based Load Leveling* dan *Global Query Filters*”.

Penyusunan skripsi ini dimaksudkan untuk memenuhi salah satu syarat guna memperoleh gelar Sarjana Komputer pada Program Studi Informatika, Fakultas Teknologi Komunikasi dan Informatika, Universitas Nasional.

Penulis menyadari bahwa selesainya skripsi ini tidak terlepas dari bantuan, bimbingan, dan dukungan dari berbagai pihak. Oleh karena itu, dengan segala kerendahan hati, penulis ingin menyampaikan rasa terima kasih yang setulus-tulusnya kepada:

1. Kedua orang tua tercinta, yang senantiasa memberikan doa tulus serta dukungan tanpa henti bagi penulis dalam menempuh pendidikan hingga tahap ini.
2. Abang dan Kakak tercinta, atas segala bantuan moril dan materil yang telah diberikan kepada penulis.
3. Bapak Dr. El Amry Bermawi Putera, M.A., selaku Rektor Universitas Nasional, yang telah memberikan kesempatan kepada penulis untuk menimba ilmu di almamater tercinta ini.
4. Bapak Dr. Agung Triayudi, S.Kom., M.Kom., selaku Dekan Fakultas Teknologi Komunikasi dan Informatika, atas dukungannya dalam penyelenggaraan kegiatan akademik yang kondusif.
5. Bapak Dr. Ucuk Darusalam, S.T., M.T, selaku Dosen Pembimbing I, dan Ibu Rima Tamara Aldisa, S.Kom., M.Kom, selaku Dosen Pembimbing II. Terima kasih atas kesabaran, kesempatan waktu yang diberikan, arahan ilmu, serta

bimbingan berharga yang telah diberikan selama proses penyusunan skripsi ini.

6. Staf Sekretariat Fakultas Teknologi Komunikasi dan Informatika (FTKI), yang telah membantu kelancaran administrasi dalam pembuatan surat izin penelitian sehingga penulis dapat melaksanakan wawancara dengan lancar.
7. Kepala Sekolah, Wakil Kepala Sekolah Bidang Kurikulum, serta jajaran Staf IT dan Guru dari lima sekolah di wilayah Jakarta Selatan yang telah bersedia meluangkan waktu sebagai narasumber wawancara dan memberikan data yang sangat berharga bagi identifikasi masalah dalam penelitian ini.
8. Seluruh teman-teman seperjuangan angkatan 2022, khususnya teman-teman di Program Studi Informatika, atas informasi dan diskusi yang telah diberikan selama masa perkuliahan.

Penulis menyadari bahwa skripsi ini masih jauh dari kata sempurna karena keterbatasan pengetahuan dan pengalaman penulis. Oleh karena itu, kritik dan saran yang membangun sangat penulis harapkan demi penyempurnaan penelitian ini di masa mendatang. Akhir kata, semoga skripsi ini dapat memberikan manfaat bagi pengembangan ilmu pengetahuan, khususnya di bidang rekayasa perangkat lunak dan teknologi pendidikan.

Jakarta, Februari 2025

Penulis

ABSTRAK

Institusi pendidikan menghadapi tantangan operasional akibat fragmentasi data serta masalah isolasi data dan degradasi performa (*Noisy Neighbor*) pada adopsi Sistem Informasi Akademik (SIKAD) berbasis *Software as a Service (SaaS) multi-tenant*. Penelitian ini bertujuan untuk merancang arsitektur *SaaS* berbasis *Shared Database*. Menggunakan metode *Research and Development (R&D)*, arsitektur dikembangkan di atas ekosistem ASP.NET Core 9.0 dan Microsoft SQL Server 2022. Solusi diusulkan dengan mengimplementasikan *Global Query Filters (GQF)* untuk memitigasi masalah isolasi data dan pola *Queue-Based Load Leveling* untuk masalah *Noisy Neighbor*. Hasil pengujian membuktikan *GQF* secara efektif menjaga isolasi data antar *tenant* dengan mencegah akses ilegal antar lintas *tenant*. Pada beban kerja ekstrem, arsitektur yang diusulkan berhasil memitigasi fenomena *Noisy Neighbor* dengan meningkatkan tingkat keberhasilan status "GURU OK" dari 0% menjadi 70%. Selain itu, penerapan mekanisme antrian berhasil mereduksi tingkat kegagalan sistem dari 9,39% menjadi 0,52%. Temuan ini mengonfirmasi bahwa integrasi *GQF* dan pola *Queue-Based Load Leveling* merupakan solusi arsitektural yang valid untuk membangun layanan *SaaS* pendidikan yang aman, andal, dan responsif.

Kata Kunci: *SaaS Multi-Tenant, Shared Database, Noisy Neighbor, Global Query Filters, Queue-Based Load Leveling.*

ABSTRACT

Educational institutions face operational challenges due to data fragmentation, as well as data isolation issues and performance degradation (Noisy Neighbor) when adopting multi-tenant Software as a Service (SaaS)-based Academic Information Systems (SIKAD). This research aims to design a Shared Database-based SaaS architecture. Using the Research and Development (R&D) method, the architecture was developed on the ASP.NET Core 9.0 and Microsoft SQL Server 2022 ecosystem. The proposed solution implements Global Query Filters (GQF) to mitigate data isolation issues and the Queue-Based Load Leveling pattern for the Noisy Neighbor problem. Test results prove that the GQF effectively maintains inter-tenant data isolation by preventing illegal cross-tenant access. Under extreme workloads, the proposed architecture successfully mitigates the Noisy Neighbor phenomenon by increasing the success rate of the "GURU OK" status from 0% to 70%. Furthermore, the implementation of the queueing mechanism managed to reduce the system failure rate from 9.39% to 0.52%. These findings confirm that the integration of GQF and the Queue-Based Load Leveling pattern is a valid architectural solution for building secure, reliable, and responsive educational SaaS services.

Keywords: Multi-Tenant SaaS, Shared Database, Noisy Neighbor, Global Query Filters, Queue-Based Load Leveling.

DAFTAR ISI

ABSTRAK	vi
ABSTRACT	vii
DAFTAR ISI	viii
DAFTAR GAMBAR	x
DAFTAR TABEL	xi
DAFTAR LAMPIRAN	xii
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	3
1.3 Tujuan Penelitian	4
1.4 Manfaat Penelitian	4
1.5 Batasan Masalah	5
BAB II TINJAUAN PUSTAKA	6
2.1 Landasan Teori	6
2.1.1 <i>Software as a Service (SaaS)</i>	6
2.1.2 <i>Model Shared Database</i>	8
2.1.3 <i>Item-Level Isolation</i>	9
2.1.4 <i>Entity Framework Core dan Global Query Filters</i>	12
2.1.5 Fenomena <i>Noisy Neighbor</i>	15
2.1.6 Pola <i>Queue-Based Load Leveling</i>	17
2.2 Tinjauan Teknologi Pengembangan	19
2.2.1 <i>ASP.NET Core dan Dependency Injection (DI)</i>	19
2.2.2 <i>Arsitektur Middleware Pipeline</i>	21
2.2.3 <i>Layanan Latar Belakang Terkelola (Managed Hosted Services)</i> ..	23
2.2.4 <i>Struktur Data Antrean Asinkron (System.Threading.Channels)</i> ..	24
2.2.5 <i>Microsoft SQL Server</i>	25
2.3 Tinjauan Teknologi dan Metrik Pengujian	26
2.3.1 <i>Grafana k6</i>	26
2.3.2 <i>Postman</i>	27
2.4 <i>Penelitian Terdahulu</i>	28
2.5 <i>Kerangka Berpikir</i>	34
BAB III Metode penelitian	37
3.1 <i>Alur Penelitian</i>	37
3.2 <i>Studi Pendahuluan dan Identifikasi Masalah</i>	38
3.2.1 <i>Lokasi Penelitian</i>	38
3.2.2 <i>Subjek dan Objek Penelitian</i>	38
3.2.3 <i>Temuan Masalah di Lapangan</i>	39
3.3 <i>Studi Literatur</i>	39
3.4 <i>Penetapan Fokus dan Analisis Kebutuhan</i>	40
3.4.1 <i>Fokus Penelitian</i>	40

3.4.2 Analisis Kebutuhan Fungsional	40
3.4.3 Analisis Kebutuhan Nonfungsional	44
3.5 Perancangan Sistem	45
3.5.1 Perancangan Arsitektur Sistem	45
3.5.1.1. Komponen <i>Middleware Pipeline</i>	46
3.5.1.2. Desain Mekanisme Isolasi Data (<i>Read Path</i>)	46
3.5.1.3. Desain Mekanisme <i>Load Leveling (Write Path)</i>	47
3.5.2 Perancangan Basis Data	47
3.5.2.1. Modul <i>Platform</i> dan Manajemen <i>Tenant</i>	48
3.5.2.2. Modul Data Master dan Pengguna	49
3.5.2.3. Modul Transaksi Akademik	50
3.6 Implementasi Sistem	51
3.6.1 Stack Teknologi	51
3.6.2 Implementasi <i>Middleware</i>	52
3.6.3 Implementasi Isolasi Data (<i>Global Query Filters</i>)	54
3.6.4 Implementasi <i>Queue-Based Load Leveling</i>	55
3.6.3.1 Implementasi Pemrosesan Sinkron	56
3.6.3.1 Implementasi Pemrosesan Asinkron	57
3.7 Pengujian dan Analisis	58
3.7.1 Lingkungan Pengujian	58
3.7.2 Skenario Pengujian Integritas Arsitektur	59
3.7.3 Skenario Pengujian Isolasi Data	60
3.7.4 Skenario Pengujian Performa	61
BAB IV Hasil & pembahasan	63
4.1 Persiapan Lingkungan dan Data Uji	63
4.1.1 Inisialisasi Data Target (Sekolah B)	63
4.1.2 Implementasi Logika Pengambilan Data	64
4.2 Hasil Pengujian Integritas Arsitektur	65
4.2.1 Skenario ARC-01: Integritas Konteks	65
4.2.2 Skenario ARC-02: Validasi Jalur Asinkron	66
4.3 Hasil Pengujian Isolasi Data	67
4.3.1 Skenario ISO-01: Akses Data Sah	67
4.3.2 Skenario ISO-02: Akses Lintas Tenant	68
4.3.3 Skenario ISO-03: Akses Tanpa Token/Header	69
4.4 Hasil Pengujian Performa	70
4.4.1 Parameter dan Skenario Uji Beban	71
4.4.2 Skenario PRF-01: Arsitektur Sinkron (<i>Baseline</i>)	72
4.4.3 Skenario PRF-02: Arsitektur Asinkron (<i>Proposed</i>)	73
4.4.4 Analisis Komparatif dan Pembahasan	74
BAB V Kesimpulan & saran	76
5.1 Kesimpulan	76
5.2 Saran	77
DAFTAR PUSTAKA	78
LAMPIRAN	83

DAFTAR GAMBAR

Gambar 2.1	Arsitektur <i>Multi-Tenant</i>	6
Gambar 2.2	Arsitektur Akses Data pada Model <i>Shared Database</i>	8
Gambar 2.3	Konsep <i>Item-Level Isolation</i>	10
Gambar 2.4	Penerapan Kebijakan Isolasi menggunakan <i>LeadingKeys</i> pada <i>DynamoDB</i>	11
Gambar 2.5	Konfigurasi Model pada <i>Entity Framework Core</i>	12
Gambar 2.6	Mekanisme Intersepsi <i>Global Query Filter</i> pada Kueri LINQ	13
Gambar 2.7	Ilustrasi Interferensi Performa pada Arsitektur Sinkron	16
Gambar 2.8	Arsitektur <i>Queue-Based Load Leveling</i> dengan konfigurasi <i>Capacity Bounded</i>	17
Gambar 2.9	Mekanisme <i>Dependency Injection</i> dan Validasi <i>Service Scope</i>	20
Gambar 2.10	Alur Pemrosesan <i>Request</i> pada <i>Middleware Pipeline</i>	22
Gambar 2.11	Kerangka Berpikir	35
Gambar 3.1	Diagram <i>Flowchart</i> Alur Penelitian	37
Gambar 3.2	Diagram <i>Use Case</i> Aktor Admin Platform	41
Gambar 3.3	Diagram <i>Use Case</i> Aktor Admin Tenant	42
Gambar 3.4	Diagram <i>Use Case</i> Aktor Guru	43
Gambar 3.5	Diagram <i>Use Case</i> Aktor Siswa	44
Gambar 3.6	Diagram Arsitektur Sistem	45
Gambar 3.7	Skema Tabel <i>Platform</i> dan <i>Tenant</i>	48
Gambar 3.8	Skema Tabel Master Data dan Pengguna	49
Gambar 3.9	Skema Tabel Transaksi Akademik	50
Gambar 3.10	Implementasi <i>Tenant Resolution Middleware</i>	53
Gambar 3.11	Konfigurasi <i>Global Query Filter</i> pada <i>AppDbContext</i>	55
Gambar 3.12	Implementasi Pemrosesan Sinkron	56
Gambar 3.13	Implementasi Asinkron	57
Gambar 4.1	Kode SQL Input Data Sekolah B	63
Gambar 4.2	Kode <i>cek-nilai</i>	64
Gambar 4.3	Hasil Pengujian ARC-01	65
Gambar 4.4	Hasil Pengujian ARC-02	66
Gambar 4.5	Hasil Pengujian ISO-01	68
Gambar 4.6	Hasil Pengujian ISO-02	69
Gambar 4.7	Hasil Pengujian ISO-03	70
Gambar 4.8	Kode <i>loadtest.js</i>	71
Gambar 4.9	Hasil Pengujian PRF-01	72
Gambar 4.10	Hasil Pengujian PRF-02	73

DAFTAR TABEL

Tabel 2.1 Representasi Data pada Tabel Berbasis <i>Shared Schema</i>	9
Tabel 2.2 Penelitian Terdahulu	31
Tabel 3.1 Skenario Uji Integritas Arsitektur	59
Tabel 3.2 Skenario Uji Isolasi Data	60
Tabel 3.3 Skenario Uji Performa	62
Tabel 4.1 Perbandingan Performa Arsitektur Sinkron vs Asinkron	74



DAFTAR LAMPIRAN

Lampiran 1 Dokumentasi dengan Responden	83
---	----

