

Implementasi Teknis Sistem Manajemen Asesmen Karir Multi-Instrumen Berarsitektur Multi-Tenant: Studi Rekayasa Perangkat Lunak

Muhammad Zirlida Prairi¹, Maulana Ardhiansyah^{2*}

¹²Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Pamulang, Tangerang Selatan, Indonesia

*Corresponding author: maulana.ardhiansyah@unpam.ac.id

Published: July , 2026

ABSTRAK

Sistem manajemen asesmen karir berbasis web yang melayani banyak institusi sekaligus (multi-tenant) menghadapi sejumlah tantangan rekayasa perangkat lunak, antara lain konsistensi hasil kalkulasi pada eksekusi berulang (idempotency), ekstensibilitas terhadap penambahan jenis instrumen, dan isolasi data antar institusi. Penelitian ini melaporkan studi implementasi teknis sebuah sistem manajemen asesmen karir yang mengintegrasikan enam instrumen psikometrik dan satu mekanisme rekomendasi program studi berbasis pembobotan multi-dimensi, dengan spesifikasi instrumen, skema pembobotan, dan data katalog program studi yang seluruhnya ditetapkan oleh institusi mitra pengembang sebagai bagian dari kurikulum dan kekayaan intelektual mereka. Kontribusi penelitian ini terletak pada pola implementasi teknis yang diterapkan untuk merealisasikan spesifikasi tersebut, meliputi pola pure function pada modul scoring untuk menjamin determinisme dan testability, pola registry untuk ekstensibilitas penambahan instrumen, mekanisme idempotent upsert untuk konsistensi data pada kondisi retry atau resubmit, dan middleware isolasi tenant untuk mencegah kebocoran data antar institusi. Pengujian fungsional dilakukan pada lima skenario teknis inti yang seluruhnya berhasil lulus, serta satu studi kasus pengujian pada akun pengguna dengan kelengkapan data asesmen parsial yang mengungkap perilaku sistem berupa nilai fallback seragam ketika dimensi input dengan bobot dominan belum terisi. Penelitian ini diposisikan sebagai studi rekayasa perangkat lunak yang melaporkan pola implementasi dan temuan pengujian, bukan sebagai penelitian atau validasi terhadap algoritma maupun instrumen psikometrik itu sendiri, mengingat keduanya merupakan domain kerja pihak mitra yang berada di luar lingkup kerja penulis.

Kata Kunci: *rekayasa perangkat lunak; sistem multi-tenant; idempotency; pola desain perangkat lunak; sistem informasi pendidikan*

ABSTRACT

Multi-tenant web-based career assessment management systems face several software engineering challenges, including consistency of calculation results across repeated executions (idempotency), extensibility for adding new assessment instruments, and data isolation between institutions. This study reports a technical implementation case study of a career assessment management system integrating six psychometric instruments and a multi-dimensional weighted recommendation mechanism, where the instrument specifications, weighting scheme, and study program catalog data were entirely determined by the partner institution as part of their curriculum and intellectual property. The contribution of this research lies in the technical implementation patterns applied to realize these specifications, including a pure function pattern in the scoring module to ensure determinism and testability, a registry pattern for instrument extensibility, an idempotent upsert mechanism for data consistency under retry or resubmission conditions, and tenant isolation middleware to prevent cross-institution data leakage. Functional testing was conducted on five core technical scenarios, all of which passed, along with one case study test on a user account with partial assessment data completeness that revealed a uniform fallback value behavior when dominant-weighted input dimensions were unavailable. This research is positioned as a software engineering study reporting implementation patterns and testing findings, not as research or validation of the algorithm or psychometric instruments themselves, as both fall under the partner institution's domain and are outside the author's scope of work.

Keywords: *software engineering; multi-tenant system; idempotency; software design pattern; educational information system*

© The Author(s). 2021 Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's

PENDAHULUAN

Platform digital yang melayani banyak institusi pendidikan sekaligus (arsitektur multi-tenant) semakin banyak digunakan untuk mengelola proses bimbingan karir berbasis data di sekolah menengah [1]. Tren digitalisasi layanan berbasis platform ini juga tercermin pada pengembangan sistem informasi berbasis web di berbagai skala institusi non-pendidikan, termasuk pemerintahan desa [10]. Pengembangan platform semacam ini melibatkan dua domain kerja yang berbeda namun saling bergantung: domain konten dan metodologi asesmen (perancangan instrumen psikometrik, skema pembobotan, dan basis pengetahuan rekomendasi), serta domain rekayasa perangkat lunak (bagaimana spesifikasi tersebut diimplementasikan menjadi sistem yang berjalan secara andal, konsisten, dan dapat diperluas).

Penelitian ini berfokus secara eksklusif pada domain kedua. Sebagai disiplin ilmu, rekayasa perangkat lunak mencakup keseluruhan siklus analisis dan desain sistem hingga pemeliharaannya [15], termasuk tahap rekayasa kebutuhan yang menjadi fondasi keberhasilan proyek pengembangan perangkat lunak [16]. Spesifikasi instrumen psikometrik, skema pembobotan algoritma rekomendasi, formula perhitungan persentase kecocokan, serta data katalog program studi yang digunakan dalam penelitian ini seluruhnya ditetapkan oleh institusi mitra pengembang sebagai bagian dari kurikulum dan kekayaan intelektual mereka, dan tidak dirancang maupun divalidasi secara ilmiah oleh penulis. Penulis berperan sebagai pelaksana teknis (developer) yang menerima spesifikasi tersebut dan bertanggung jawab atas implementasinya ke dalam sistem perangkat lunak yang berjalan.

Tantangan rekayasa perangkat lunak yang muncul dalam mengimplementasikan spesifikasi semacam ini meliputi: bagaimana memastikan hasil kalkulasi skor konsisten meski proses dijalankan berulang kali (idempotency), bagaimana merancang struktur kode yang memungkinkan penambahan instrumen baru di masa depan tanpa mengubah logika inti (ekstensibilitas), dan bagaimana mencegah data milik satu institusi terekspos ke institusi lain pada sistem yang melayani banyak tenant sekaligus (isolasi data). Penelitian ini melaporkan pola implementasi yang diterapkan untuk menjawab tantangan tersebut, beserta temuan dari pengujian fungsional yang dilakukan, sebagai kontribusi pada domain rekayasa perangkat lunak pendidikan.

TINJAUAN PUSTAKA

2.1 Arsitektur Multi-Tenant

Aziza & Nugraha [2] menganalisis trade-off tiga model arsitektur basis data pada lingkungan multi-tenant software as a service (SaaS): separate database, shared database dengan schema terpisah, dan shared database dengan schema bersama yang dibedakan melalui kolom tenant identifier. Hasil evaluasi menunjukkan bahwa tidak ada satu model yang secara mutlak unggul, karena masing-masing memiliki kelebihan dan kekurangan yang saling bertukar terkait efisiensi biaya, pemisahan data, dan skalabilitas. Pendekatan ketiga umum dipilih pada sistem skala menengah karena keseimbangan antara efisiensi operasional dan kompleksitas pemeliharaan, namun menuntut disiplin tinggi pada lapisan akses data untuk mencegah kebocoran lintas tenant. Penelitian terkait lainnya menerapkan kombinasi arsitektur multi-tenant dengan role-based access control pada sistem manajemen waralaba [11], serta strategi mitigasi noisy neighbor dan isolasi data menggunakan queue-based load leveling dan global query filters [13]. Adopsi layanan cloud computing secara umum, termasuk model software as a service, juga banyak dikaji sebagai konteks pendukung arsitektur multi-tenant [12], [14].

2.2 Idempotency pada Sistem Pemrosesan Data

Idempotency adalah properti suatu operasi yang menghasilkan keluaran identik meskipun dijalankan berulang kali dengan input yang sama [3]. Properti ini penting pada sistem yang rentan terhadap kondisi retry akibat kegagalan jaringan atau resubmission yang tidak disengaja oleh pengguna, karena tanpa idempotency, eksekusi berulang dapat menghasilkan duplikasi data atau inkonsistensi state. Pada basis data relasional, idempotency pada operasi tulis umumnya direalisasikan melalui mekanisme upsert (INSERT ... ON CONFLICT DO UPDATE), yang memperbarui baris yang sudah ada alih-alih menyisipkan baris baru ketika terjadi konflik pada kunci unik.

2.3 Pola Desain Registry dan Pure Function

Pola registry adalah pola desain perangkat lunak yang menyediakan satu titik pendaftaran terpusat untuk memetakan suatu identifier ke implementasi yang sesuai, umum digunakan untuk mendukung ekstensibilitas sistem tanpa memodifikasi kode yang sudah ada (prinsip open-closed) [4]. Pure function adalah fungsi yang keluarannya hanya ditentukan oleh nilai input yang diberikan, tanpa efek samping seperti pembacaan atau penulisan ke basis data, sehingga lebih mudah diuji secara terisolasi dan lebih mudah dipahami perilakunya secara deterministik [5].

2.4 RESTful API sebagai Lapisan Komunikasi

Vallemoni [6] menunjukkan bahwa pola konektivitas berbasis API — mencakup API gateway, autentikasi, serta observability — menjadi lapisan penting yang menghubungkan penyedia layanan dan konsumen layanan pada arsitektur berbasis HTTP, termasuk penanganan idempotency pada permintaan yang berpotensi diulang. Sutara & Gunawan [7] menunjukkan penerapan praktis RESTful API menggunakan kerangka kerja Express.js dibandingkan dengan Hapi.js, dengan hasil pengujian performa yang relevan sebagai pertimbangan pemilihan basis arsitektur sistem dalam penelitian ini. Ardhiansyah & Arifin [8] turut menyediakan modul praktis pembuatan REST API menggunakan Express.js dan MongoDB yang menjadi salah satu rujukan teknis implementasi pada penelitian ini.

METODE PENELITIAN

3.1 Spesifikasi dari Institusi Mitra

Tabel 1 merangkum komponen spesifikasi fungsional yang ditetapkan oleh institusi mitra dan menjadi input bagi proses implementasi. Seluruh komponen pada tabel ini merupakan kurikulum dan kekayaan intelektual mitra; penulis tidak terlibat dalam perancangan, validasi, maupun penelitian terhadap komponen-komponen tersebut.

Tabel 1. Spesifikasi Fungsional dari Institusi Mitra

Komponen Spesifikasi	Sumber dan Deskripsi
Instrumen asesmen	Ditetapkan oleh tim psikometri mitra: RIASEC, Multiple Intelligences, MBTI, DISC, Gaya Belajar, dan Adaptasi Sekolah
Skema pembobotan	Ditetapkan oleh mitra: RIASEC primary 3,0; RIASEC secondary 1,5; MI 2,0; MBTI 1,0; DISC 0,5
Formula skor kecocokan	Ditetapkan oleh mitra: $\text{match_pct} = \text{round}((\text{skor_aktual} / 11,5) \times 100)$
Katalog program studi	Disediakan oleh mitra, terdiri lebih dari 100 entri program studi dengan atribut kecocokan per dimensi

3.2 Pola Implementasi Teknis

Berdasarkan spesifikasi pada Tabel 1, empat pola implementasi teknis diterapkan untuk merealisasikan sistem sebagaimana dirangkum pada Tabel 2. Pemilihan pola-pola ini didasarkan pada kebutuhan untuk menjamin determinisme hasil kalkulasi, ekstensibilitas terhadap instrumen baru, konsistensi data pada kondisi eksekusi berulang, dan isolasi data pada arsitektur multi-tenant.

Tabel 2. Pola Implementasi Teknis

Pola Implementasi	Tujuan Teknis	Cara Kerja
Pure Function Scorer	Memastikan hasil scoring deterministik dan mudah diuji	Setiap scorer menerima item bank dan jawaban sebagai input, tanpa membaca/menulis basis data secara langsung
Registry Pattern	Memungkinkan penambahan instrumen baru tanpa mengubah kode inti	Dispatcher tunggal memetakan tipe asesmen ke fungsi scorer yang sesuai berdasarkan key terdaftar
Idempotent Upsert	Mencegah duplikasi data akibat retry atau resubmit	Penyimpanan skor menggunakan INSERT ... ON CONFLICT DO UPDATE pada kunci komposit (student_id, type)
Tenant Isolation Middleware	Mencegah kebocoran data antar institusi pada arsitektur multi-tenant	Middleware memvalidasi dan menyisipkan institution_id dari JWT pada setiap query sebelum mencapai handler

Modul scoring untuk setiap instrumen diimplementasikan sebagai pure function yang menerima item bank dan kumpulan jawaban sebagai parameter, dan mengembalikan raw score, normalized score, serta interpretation tanpa melakukan operasi baca/tulis basis data secara langsung di dalam fungsi tersebut. Pemisahan ini memungkinkan setiap scorer diuji secara terisolasi dengan unit test berbasis input-output semata, tanpa memerlukan mock basis data. Registry scoring.registry.js bertindak sebagai dispatcher tunggal yang memetakan tipe instrumen ke scorer yang sesuai, sehingga penambahan instrumen baru di masa depan hanya membutuhkan penambahan satu entri pada registry tanpa mengubah logika pemanggilan di lapisan controller.

Penyimpanan hasil skor ke tabel scores dan cumulative_scores menerapkan mekanisme upsert pada kunci komposit, sehingga eksekusi scoring berulang untuk attempt yang identik menghasilkan baris data yang sama tanpa duplikasi. Middleware requireInstitutionScope memvalidasi dan menyisipkan institution_id yang diekstraksi dari payload JWT pengguna ke setiap query basis data pada lapisan akses data, sebelum permintaan mencapai handler bisnis logika, sehingga isolasi tenant ditegakkan secara terpusat alih-alih bergantung pada implementasi manual di setiap endpoint, sejalan dengan pendekatan role-based access control pada lingkungan multiuser yang dilaporkan Ramadhani dkk. [9].

3.3 Desain Pengujian

Pengujian fungsional dirancang untuk memverifikasi lima aspek teknis inti dari pola implementasi pada Tabel 2: idempotency scoring, batas normalisasi nilai output, penanganan error pada registry, dan isolasi data tenant. Selain itu, satu pengujian tambahan dilakukan sebagai studi kasus tunggal pada akun pengguna nyata di lingkungan staging yang baru menyelesaikan sebagian instrumen asesmen, untuk mengamati perilaku sistem pada kondisi data input tidak lengkap yang lazim terjadi pada penggunaan dunia nyata. Pengujian studi kasus ini bersifat observasional dan tidak dimaksudkan sebagai validasi statistik, mengingat hanya melibatkan satu akun pengguna.

HASIL DAN PEMBAHASAN

4.1 Hasil Pengujian Fungsional Teknis

Hasil pengujian pada lima skenario teknis inti disajikan pada Tabel 3.

Tabel 3. Hasil Pengujian Fungsional

ID	Skenario Uji	Aspek yang Diverifikasi	Status
TC-01	Eksekusi scoring 2 kali pada attempt yang sama	Idempotency	LULUS
TC-02	Normalisasi skor pada nilai input ekstrem	Batas nilai output	LULUS
TC-03	Registry dipanggil dengan tipe asesmen tidak terdaftar	Penanganan error	LULUS
TC-04	Akses data lintas institution_id berbeda	Isolasi tenant	LULUS
TC-05	Permintaan rekomendasi dengan kelengkapan data parsial	Perilaku sistem pada input tidak lengkap	Lihat catatan 4.3

Pengujian TC-01 mengonfirmasi bahwa menjalankan proses scoring dua kali untuk attempt yang identik menghasilkan baris data yang sama pada tabel scores, tanpa duplikasi, membuktikan keberhasilan mekanisme idempotent upsert. Pengujian TC-04 mengonfirmasi bahwa permintaan data dengan institution_id yang berbeda dari token JWT pengguna secara konsisten ditolak oleh middleware, membuktikan keberhasilan isolasi tenant. Pengujian TC-05 dibahas secara terpisah pada subbagian 4.2 karena hasilnya bersifat lebih kompleks dan memerlukan interpretasi lebih lanjut.

4.2 Studi Kasus: Perilaku Sistem pada Kelengkapan Data Parsial

Studi kasus pengujian TC-05 dilakukan pada satu akun pengguna nyata yang tercatat menyelesaikan 4 dari 6 instrumen asesmen yang tersedia, sebagaimana dirangkum pada Tabel 4.

Tabel 4. Status dan Hasil Asesmen pada Studi Kasus

Instrumen	Status	Hasil
Kepribadian (MBTI)	Selesai	ISTJ
Minat Karir (RIASEC)	Belum diselesaikan	—
Kecerdasan Utama (MI)	Belum diselesaikan	—
Gaya Perilaku (DISC)	Selesai	C (Conscientiousness)
Gaya Belajar (VARK)	Selesai	A (Auditori)
Adaptasi Sekolah	Selesai	Asrama — dapat beradaptasi

Dua instrumen dengan bobot tertinggi pada skema mitra — RIASEC (bobot primary 3,0) dan Multiple Intelligences (bobot 2,0), dengan kontribusi gabungan sekitar 43% dari skor maksimal teoritis — belum diselesaikan oleh akun studi kasus pada saat pengujian. Hasil permintaan rekomendasi pada kondisi ini tetap menghasilkan keluaran berupa daftar program studi terurut, namun seluruh program studi yang ditampilkan

memperoleh persentase kecocokan yang identik (10%), termasuk pada program studi dengan karakteristik vokasional yang secara konseptual berjauhan.

Dari sudut pandang rekayasa perangkat lunak, temuan ini mengindikasikan bahwa lapisan implementasi saat ini tidak menerapkan penanganan eksplisit untuk kondisi data input parsial pada dimensi berbobot dominan, sehingga skor yang dihasilkan kehilangan daya pembeda (*discriminative power*) antar item. Hal ini bukan merupakan kegagalan pada algoritma atau skema pembobotan milik mitra itu sendiri, melainkan sebuah kesenjangan pada lapisan implementasi teknis yang seharusnya menangani validasi kelengkapan data sebelum menjalankan kalkulasi, atau menerapkan strategi alternatif seperti *exclusion-renormalisasi* pada dimensi yang datanya kosong.

4.3 Implikasi Teknis dan Rekomendasi Implementasi

Berdasarkan temuan pada subbagian 4.2, tiga rekomendasi perbaikan pada lapisan implementasi (bukan pada algoritma atau skema pembobotan itu sendiri) diidentifikasi. Pertama, penambahan langkah validasi kelengkapan data sebelum eksekusi kalkulasi skor, dengan ambang batas minimum yang dapat dikonfigurasi oleh mitra sesuai kebutuhan domain mereka. Kedua, penerapan strategi *exclusion-renormalisasi* pada lapisan kalkulasi, yaitu mengeluarkan dimensi yang datanya kosong dari perhitungan dan menormalisasi ulang terhadap total bobot dimensi yang tersedia, sehingga daya pembeda skor tetap terjaga meski sebagian data belum lengkap. Ketiga, penyajian indikator tingkat kepercayaan (*confidence level*) pada lapisan antarmuka pengguna, sebagai metadata tambahan yang dihasilkan oleh lapisan backend berdampingan dengan skor kecocokan, untuk membantu pengguna memahami tingkat kelengkapan data yang mendasari rekomendasi yang ditampilkan.

Ketiga rekomendasi ini bersifat netral terhadap nilai bobot dan formula yang ditetapkan mitra — rekomendasi berfokus pada bagaimana lapisan implementasi menangani kondisi data tidak lengkap sebagai sebuah keputusan rekayasa perangkat lunak, terlepas dari skema pembobotan spesifik yang digunakan.

4.4 Keterbatasan Penelitian

Penelitian ini memiliki keterbatasan yang signifikan. Pertama, studi kasus pada subbagian 4.2 hanya melibatkan satu akun pengguna, sehingga temuan mengenai nilai fallback tidak dapat digeneralisasi secara statistik untuk seluruh kemungkinan kombinasi kelengkapan data. Kedua, penelitian ini secara sengaja tidak membahas validitas ilmiah instrumen psikometrik maupun ketepatan skema pembobotan yang digunakan, karena keduanya berada di luar lingkup kerja dan kompetensi penulis sebagai pelaksana teknis; pembahasan tersebut memerlukan kajian terpisah oleh pihak yang berkompeten di bidang psikometri. Ketiga, pengujian dilakukan pada lingkungan staging dengan data terbatas, sehingga belum mencakup pengujian beban (*load testing*) pada skala pengguna yang besar.

KESIMPULAN

Penelitian ini melaporkan studi implementasi teknis sebuah sistem manajemen asesmen karir multi-instrumen berarsitektur multi-tenant, dengan spesifikasi instrumen, skema pembobotan, dan data katalog program studi yang seluruhnya ditetapkan oleh institusi mitra. Empat pola implementasi teknis — *pure function* pada modul scoring, *registry pattern* untuk ekstensibilitas, *idempotent upsert* untuk konsistensi data, dan *middleware isolasi tenant* — berhasil diterapkan dan diverifikasi melalui pengujian fungsional pada lima skenario teknis inti yang seluruhnya lulus.

Studi kasus tambahan pada satu akun pengguna dengan data asesmen tidak lengkap mengungkap kesenjangan pada lapisan implementasi terkait penanganan kondisi data parsial, yang berujung pada hilangnya daya pembeda skor rekomendasi. Temuan ini menjadi dasar tiga rekomendasi perbaikan pada lapisan implementasi yang bersifat netral terhadap algoritma dan skema pembobotan milik mitra. Penelitian lanjutan dengan pengujian pada sampel pengguna yang lebih besar serta kolaborasi dengan pihak psikometri mitra diperlukan untuk memperluas dan memvalidasi temuan pada penelitian ini secara lebih menyeluruh.

DAFTAR PUSTAKA

- [1] A. Z. Ibad, “Ruang lingkup sistem informasi pendidikan dalam lembaga pendidikan,” *Promis*, vol. 3, no. 1, hlm. 99–109, 2022.
- [2] S. T. N. Aziza dan Y. Nugraha, “Analisis trade-off arsitektur multi-tenant sebagai dasar justifikasi strategi basis data SaaS,” *Progresif: Jurnal Ilmiah Komputer*, vol. 22, no. 2, 2026, doi: 10.35889/progresif.v22i2.3590.
- [3] K. Dusad, “Taming asynchrony in distributed payment systems: Guarantees, idempotency, and end-to-end reconciliation,” *Journal of Computer Science and Technology Studies*, 2025.
- [4] R. V. Bhupatiraju, “Design patterns for payment system integration in distributed microservices: A supply chain perspective,” *Journal of Computer Science and Technology Studies*, vol. 7, no. 9, hlm. 750–756, 2025, doi: 10.32996/jcsts.2025.4.1.89.
- [5] R. Andriani, P. N. Sabrina, dan H. Ashaury, “Metode refactoring untuk meningkatkan kualitas maintainability pada sistem informasi puskesmas,” *Jurnal Tekno Kompak*, vol. 19, no. 1, hlm. 227, 2024, doi: 10.33365/jtk.v19i1.4686.

- [6] R. K. Vallemoni, "API-led connectivity for gateways, acquirers, and core banking: Idempotency, security, and observability patterns," *International Journal of Innovative Research in Engineering & Multidisciplinary Physical Sciences*, 2025.
- [7] B. Sutara dan S. Gunawan, "Comparative analysis of REST API performance between Express.js framework and Hapi.js using Apache JMeter," *Jurnal Riset Teknik Informatika*, vol. 1, no. 1, hlm. 19–26, 2024.
- [8] M. Ardiansyah dan J. Arifin, *Belajar Membuat REST API dengan Menggunakan Express.JS dan MongoDB*. [Buku/modul]. 2021.
- [9] S. Ramadhani, D. Armiady, dan R. Fajri, "Penerapan role-based access control (RBAC) untuk mengelola hak akses pada sistem informasi sekolah taman kanak-kanak berbasis multiuser," *Jurnal Ilmu Komputer Aceh*, vol. 3, no. 1, hlm. 129–141, 2026.
- [10] I. Basten dan M. Ardiansyah, "Perancangan sistem informasi desa berbasis web menggunakan model waterfall: Studi kasus Desa Banjarsari Kabupaten Lebak," *Scientia Sacra: Jurnal Sains, Teknologi dan Masyarakat*, vol. 2, no. 1, hlm. 147–156, 2022.
- [11] M. I. Athallah dan A. T. Ayunda, "Model arsitektur multi-tenant berbasis role-based access control pada sistem manajemen waralaba," *Progresif: Jurnal Ilmiah Komputer*, 2026.
- [12] T. Duha, W. Setiawan, dan N. Fajriyah, "Analisis layanan cloud computing di era digital," *Jurnal Informatika*, vol. 1, no. 1, hlm. 32–39, 2022.
- [13] T. L. Darmawan, "Mitigasi noisy neighbor dan isolasi data pada arsitektur multi-tenant menggunakan queue-based load leveling dan global query filters," *Skripsi, Universitas Nasional*, 2026.
- [14] M. H. Ratsanjani, I. F. Risnandari, dan T. W. Sulaiman, "Literatur review: Peran aplikasi SAAS dalam kegiatan bisnis e-commerce," *SINOMIKA Journal: Publikasi Ilmiah Bidang Ekonomi dan Akuntansi*, vol. 1, no. 4, hlm. 1009–1020, 2022.
- [15] R. A. Sukamto, *Analisis dan Desain Perangkat Lunak: Rekayasa Perangkat Lunak untuk Pemrograman Terstruktur, Berorientasi Objek dan Agile*. Bandung: Informatika, 2022.
- [16] A. Aditya, "Rekayasa kebutuhan perangkat lunak," dalam *Rekayasa Perangkat Lunak: Konsep, Metode, dan Praktik Terbaik*. [Buku/modul]. 2023.