

Rancang Bangun Model Logis Basis Data BERKAH dengan Integritas Referensial sebagai Kebijakan Tata Kelola Data

Sandra Islama Putra*

Jurusan Teknik Informatika, Fakultas Teknik, Universitas Pasundan
Jln. Dr. Setiabudhi no. 193 Bandung, Jawa Barat
[*sandra@unpas.ac.id](mailto:sandra@unpas.ac.id)

Abstrak : Platform BERKAH (*Business Engine for Retail, Kitchen, Accommodation & Hospitality*) menuntut konsistensi data lintas modul agar informasi yang dihasilkan sah diperlakukan sebagai bukti digital yang dapat dipertanggungjawabkan. Dalam praktik sistem terintegrasi, kerusakan paling serius umumnya bukan disebabkan oleh ketiadaan data, melainkan oleh relasi yang dibiarkan longgar sehingga melahirkan data yatim (orphan), duplikasi identitas, dan anomali pembaruan. Berangkat dari persoalan tersebut, artikel ini merancang model logis basis data BERKAH dengan menempatkan integritas referensial sebagai kebijakan tata kelola data, bukan sekadar rincian implementasi. Penelitian dilakukan melalui pendekatan rancang bangun berbasis artefak: mengidentifikasi entitas inti lintas modul, mentransformasikan model konseptual ke model logis yang ternormalisasi, membakukan matriks integritas referensial (aturan foreign key beserta perilaku restrict/cascade/set null), serta menyusun rancangan uji pelanggaran (negative testing) untuk memastikan kendala benar-benar menolak inkonsistensi. Hasilnya mencakup kerangka model logis, prinsip penetapan perilaku integritas referensial berdasarkan makna bisnis, matriks kebijakan (Tabel 2), dan rancangan uji pelanggaran (Tabel 3) yang dapat dijadikan standar desain lintas modul. Kontribusi utama artikel ini adalah memformalkan integritas referensial sebagai kebijakan tata kelola yang dapat diaudit, sehingga konsistensi tidak bergantung pada kedisiplinan aplikasi semata, melainkan menjadi sifat sistemik pada tingkat basis data.]

Kata Kunci : BERKAH, model logis, integritas referensial, tata kelola data, normalisasi, kualitas data.

I. PENDAHULUAN

Platform terintegrasi seperti BERKAH pada dasarnya “hidup” dari relasi antar entitas. Ketika relasi itu dibiarkan longgar—misalnya transaksi tidak diwajibkan merujuk data induk yang sah—sistem memang masih tampak berjalan, tetapi biaya sesungguhnya muncul pada tahap pelaporan dan audit: data yatim, identitas ganda, serta konflik makna data menjadi warisan struktural yang sulit dibereskan di hilir. Dari perspektif tata kelola data, mutu dan nilai data justru meningkat ketika organisasi menegakkan kebijakan yang mengikat definisi data, mengendalikan perubahan, dan mencegah kesalahan struktural sejak sumbernya, bukan setelah masalah terlanjur menyebar. [2], [3]

Dalam kerangka itu, basis data relasional menyediakan instrumen kontrol formal melalui kendala integritas, termasuk integritas referensial. Literatur kendala integritas menempatkan kendala sebagai fondasi konsistensi pada sistem relasional, terutama ketika operasi insert/update/delete berlangsung pada skala besar dan melewati banyak jalur integrasi. [8] Namun pada praktik sistem terintegrasi, integritas referensial kerap “dilonggarkan” demi percepatan implementasi lalu dialihkan ke logika aplikasi. Strategi ini membuat konsistensi menjadi rapuh, karena jalur integrasi, proses impor, atau koreksi manual dapat melewati kontrol aplikasi dan meninggalkan ketidakonsistenan yang tidak lagi mudah ditelusuri sumbernya.

Berangkat dari masalah tersebut, penelitian ini memposisikan integritas referensial sebagai kebijakan tata kelola data dalam rancangan model logis BERKAH. Konsekuensinya, relasi kunci didefinisikan secara eksplisit, perilaku perubahan data (*restrict*, *cascade*, *set null*) dipilih berdasarkan makna bisnis alih-alih preferensi teknis, dan konsistensi diverifikasi melalui rancangan uji pelanggaran. Pendekatan ini sejalan dengan gagasan tentang “materialitas” basis data: struktur dan kendala bukan sekadar detail implementasi, melainkan cara organisasi memaksa data tetap benar secara sistemik. [1]

A. Rumusan Masalah

1. Bagaimana merancang model logis basis data BERKAH yang ternormalisasi, namun tetap operasional untuk mendukung transaksi lintas modul tanpa menambah friksi implementasi dan pemeliharaan?
2. Bagaimana membakukan integritas referensial sebagai kebijakan tata kelola data—melalui penetapan relasi kunci dan perilaku perubahan data yang eksplisit—agar mampu mencegah data yatim, duplikasi identitas, dan konflik makna antartransaksi maupun antarmodul?

3. Bagaimana menyusun rancangan uji pelanggaran (*negative testing*) yang terstandar sehingga kebijakan integritas referensial dapat diverifikasi secara konsisten, dapat diulang, dan menyediakan bukti bahwa sistem menolak inkonsistensi sebelum berdampak pada pelaporan dan audit?

B. Tujuan Penelitian

Penelitian ini bertujuan:

1. Menghasilkan model logis BERKAH yang memuat entitas inti, relasi, serta penerapan normalisasi yang memadai untuk menekan redundansi dan mengurangi anomali pembaruan pada data transaksi. [7], [10]
2. Menyusun matriks integritas referensial sebagai kebijakan tata kelola, yang merumuskan aturan *foreign key*, kardinalitas relasi, serta perilaku perubahan data (misalnya *restrict/cascade/set null*) yang dipilih berdasarkan makna bisnis dan konsekuensi historis pelaporan (Tabel 2). [8], [11]
3. Merancang skenario uji pelanggaran (*negative testing*) untuk membuktikan bahwa kendala integritas benar-benar menolak pola inkonsistensi yang lazim pada sistem terintegrasi—misalnya referensi induk tidak sah, penghapusan induk yang masih direferensikan, atau pembaruan kunci yang memutus rantai rujukan (Tabel 3). [8], [11]

C. Batasan

Penelitian ini dibatasi pada perancangan model logis relasional, meliputi struktur tabel logis, definisi kunci, relasi antartabel, prinsip normalisasi, serta perumusan kebijakan integritas referensial sebagai mekanisme formal pengendalian konsistensi data. Dengan batasan tersebut, fokus kajian diarahkan pada bagaimana relasi bisnis diturunkan menjadi kendala relasional yang eksplisit dan dapat diverifikasi, bukan pada isu implementasi teknis yang bersifat spesifik platform. [8], [11]

Pembahasan terkait optimasi fisik—seperti perancangan indeks secara rinci, partisi, dan *tuning* kueri—diberikan secara terbatas, yakni hanya pada aspek yang memiliki konsekuensi langsung terhadap keputusan normalisasi dan penerapan integritas. Dengan demikian, optimasi diposisikan sebagai pertimbangan pendukung untuk menjaga kinerja tanpa mengorbankan konsistensi pada data sumber. [7]

II. METODE PENELITIAN

A. Identifikasi Kebutuhan Data dan Entitas Inti

Bagian ini menegaskan bahwa model logis disusun sebagai turunan dari kebutuhan informasi lintas modul, bukan hasil “otomatis” dari kebiasaan perancangan tabel. Logikanya sederhana: jika kebutuhan pelaporan dan audit bersifat lintas domain, maka struktur relasional juga harus dibangun dari peta kebutuhan lintas domain—bukan dari preferensi teknis semata. Untuk itu, kebutuhan data diinventarisasi secara sistematis per domain BERKAH—meliputi identitas organisasi, pengguna, cabang, unit, rekening/bank, dokumen, transaksi, serta bukti digital—agar cakupan entitas dan atribut benar-benar merepresentasikan alur layanan. Selanjutnya, ketergantungan antarmodul dipetakan untuk mengidentifikasi relasi-relasi “wajib” yang tidak boleh dibiarkan longgar, terutama relasi yang langsung memengaruhi konsistensi pelaporan dan kekuatan jejak audit. Dengan pendekatan ini, integritas referensial tidak diperlakukan sebagai pelengkap skema, melainkan sebagai pengaman struktural yang memastikan data tetap utuh ketika berpindah lintas proses dan lintas modul.

B. Transformasi Model Konseptual ke Model Logis dan Normalisasi

Bagian ini menjembatani ranah konseptual ke ranah implementatif dengan menurunkan hubungan-hubungan utama ke dalam skema relasional yang konsisten, sekaligus menekan potensi anomali pembaruan yang kerap muncul ketika definisi entitas dan relasi diterjemahkan secara longgar. Fokusnya bukan sekadar “mengubah diagram menjadi tabel”, melainkan memastikan bahwa makna bisnis yang sudah dipakukan pada tahap konseptual tetap utuh ketika diwujudkan sebagai struktur data operasional. Transformasi skema dilakukan menggunakan pendekatan extended entity-relationship agar aspek semantik—seperti relasi, atribut kompleks, serta struktur supertipe-subtipe—diturunkan secara konsisten ke bentuk relasional, sehingga tidak terjadi distorsi makna ketika data mulai beredar lintas proses dan lintas modul. [10] Selanjutnya, normalisasi diterapkan hingga tingkat yang memadai bagi transaksi operasional (hingga *third normal form*/3NF*) untuk mengurangi redundansi dan mencegah anomali pembaruan, sembari tetap mempertimbangkan kebutuhan pelaporan melalui artefak turunan (misalnya *view* atau ringkasan terkelola) agar ketertiban pada data sumber tidak dikorbankan demi kenyamanan laporan. [7], [9]

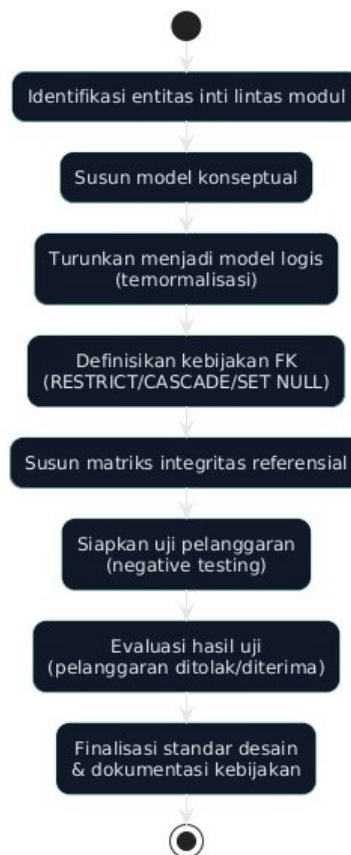
C. Perumusan Kebijakan Integritas Referensial

Bagian ini menempatkan integritas referensial sebagai poros kontribusi penelitian, yakni diposisikan sebagai kebijakan tata kelola data yang mengikat, bukan sekadar pilihan implementasi. Konsekuensinya, setiap relasi yang dinilai krusial dibakukan sebagai *foreign key* dan dilengkapi perilaku perubahan data yang eksplisit—*restrict*, *cascade*, atau *set null*—yang diturunkan dari makna bisnis serta konsekuensi operasionalnya, alih-alih preferensi teknis semata. [8]

Dalam kerangka tersebut, kebijakan integritas mencakup sedikitnya tiga ketentuan pokok: (1) konsistensi tipe data pada atribut kunci untuk mencegah anomali relasi lintas tabel, (2) pelarangan penghapusan data induk (*master*) yang masih direferensikan oleh transaksi agar data historis tidak kehilangan konteks, dan (3) pengendalian perubahan referensi yang berpotensi menggeser makna historis pelaporan—misalnya ketika pembaruan relasi membuat angka lama “terlihat benar” tetapi sebenarnya sudah berpindah konteks. Dengan demikian, integritas referensial berfungsi sebagai pagar kebijakan yang menjaga konsistensi identitas dan akurasi interpretasi data dari waktu ke waktu. [2], [8]

D. Rancangan Uji Pelanggaran (Negative Testing)

Validasi pada penelitian ini sengaja dirancang dengan strategi “memancing pelanggaran”, karena di titik inilah terlihat apakah integritas referensial benar-benar bekerja sebagai kebijakan, bukan sekadar klaim di dokumen. Logikanya sederhana: bila kendala mampu menolak kondisi yang salah sejak awal, maka konsistensi menjadi sifat sistemik—bukan bergantung pada kedisiplinan pengguna atau keberuntungan integrasi. Skenario uji yang digunakan mencakup: (1) pembuatan transaksi yang merujuk entitas induk (*master*) tidak sah, (2) percobaan menghapus entitas induk yang masih direferensikan oleh transaksi, (3) pembaruan kunci induk yang berpotensi memutus rantai referensi, serta (4) pola struktur referensial yang rawan membuka celah manipulasi data. [8], [11] Ukuran keberhasilan dibakukan secara tegas: setiap skenario pelanggaran harus ditolak oleh kendala yang berlaku, dan jejak historis transaksi tetap mempertahankan konteks identitas secara utuh—sehingga data lama tidak berubah menjadi “angka tanpa asal-usul” ketika terjadi operasi perubahan data pada entitas induk. Gambar 1 menampilkan siklus rancang bangun kebijakan integritas referensial yang bergerak dari tahap perancangan hingga verifikasi. Alur ini menegaskan bahwa integritas tidak diperlakukan sebagai keputusan teknis belaka, melainkan diturunkan dari makna bisnis ke dalam aturan relasi dan perilaku perubahan data, lalu dibuktikan melalui pengujian yang secara sengaja menantang sistem dengan skenario pelanggaran.



Gambar 1. Siklus kebijakan integritas referensial sebagai tata kelola data

III. HASIL DAN PEMBAHASAN

A. Artefak Kerangka Model Logis Entitas Inti

Hasil utama pertama dari penelitian ini adalah tersusunnya kerangka model logis yang secara tegas membedakan peran entitas ke dalam tiga kelas: entitas induk (*master*), entitas referensi (acuan klasifikasi), dan entitas transaksi. Pembedaan ini bukan sekadar pilihan pengelompokan, melainkan mekanisme untuk menjaga “sumber kebenaran tunggal” bagi identitas dan atribut kunci. Dengan sumber identitas yang dipakukan pada entitas induk, integrasi lintas modul tidak lagi bergantung pada interpretasi lokal yang berpotensi melahirkan identitas ganda maupun data yang saling bertentangan. [2], [3]

Kerangka entitas inti yang digunakan dalam model logis BERKAH dirangkum pada Tabel 1. Ringkasan tersebut sekaligus menampilkan risiko tata kelola yang muncul apabila integritas referensial tidak ditempatkan sebagai kebijakan—misalnya transaksi kehilangan rujukan induk yang sah, klasifikasi tidak seragam antarmodul, dan konsistensi lintas proses menjadi rapuh karena tidak ada kendala formal yang memaksa keterhubungan data.

Kelompok	Entitas	Peran Logis	Kunci Utama	Risiko jika tanpa integritas referensial
Master	Pengguna	Identitas pelaku aksi	id_pengguna	Transaksi tidak jelas pelakunya; audit melemah
Master	Cabang	Identitas lokasi operasional	kode_cabang	Transaksi salah konteks; laporan bias
Master	Unit	Identitas unit kerja	kode_unit	Konsolidasi unit tidak konsisten
Master	Bank	Identitas bank	kode_bank	Arus kas/saldo sulit ditelusuri
Master	Rekening	Identitas rekening internal	kode_rekening	
Referensi	Jenis transaksi	Klasifikasi transaksi	id_jenis_transaksi	Makna transaksi berubah antar periode
Referensi	Status dokumen	Tahap proses & kontrol	id_status	Dokumen “loncat status”; kontrol proses rusak
Transaksi (Header)	Dokumen transaksi	Kejadian bisnis utama	id_dokumen	Detail terlepas dari induk; orphan detail
Transaksi (Detail)	Baris transaksi	Rincian nilai & akun	id_detail	Total dokumen tidak dapat direkonsiliasi
BuktiDigital	Lampiran/bukti	Akuntabilitas	id_bukti	Klaim transaksi tanpa bukti valid

Tabel 1. Ringkasan entitas inti pada model logis BERKAH

B. Artefak Matriks Kebijakan Integritas Referensial

Hasil utama kedua penelitian ini adalah matriks kebijakan integritas referensial yang membakukan relasi antartabel—meliputi pasangan tabel, atribut kunci, kardinalitas, serta perilaku perubahan (*on delete/on update*). Pada tataran tata kelola, integritas referensial diposisikan sebagai “pagar” formal: kondisi yang bertentangan dengan aturan organisasi ditolak oleh sistem secara otomatis, sehingga konsistensi tidak bergantung pada kedisiplinan aplikasi semata. [8]

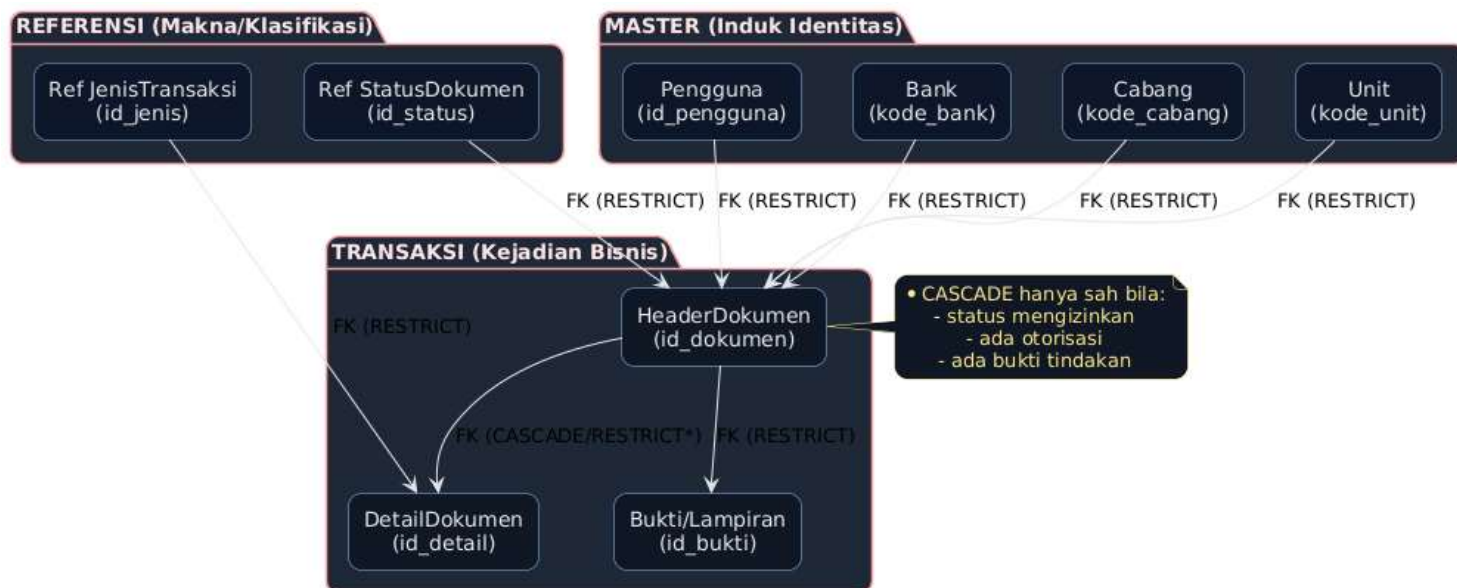
Ringkasan matriks kebijakan yang berfungsi sebagai “pola minimal wajib” disajikan pada Tabel 2.

Relasi	Induk	Turunan	Kardinalitas	ON Update	ON Delete	Rasional Kebijakan
Cabang → HeaderDokumen	Master Cabang	HeaderDokumen	1 : N	RESTRICT	RESTRICT	Cabang adalah konteks historis; tidak boleh hilang
Unit → HeaderDokumen	Master Unit	HeaderDokumen	1 : N	RESTRICT	RESTRICT	Menjaga konsistensi konsolidasi pelaporan
Pengguna → HeaderDokumen	Master Pengguna	HeaderDokumen	1 : N	RESTRICT	RESTRICT	Audit trail wajib punya identitas pelaku yang valid
HeaderDokumen → DetailDokumen	Header Dokumen	DetailDokumen	1 : N	CASCADE	RESTRICT*	Boleh cascade hanya jika penghapusan tindakan sah
Referensi StatusDokumen → HeaderDokumen	Ref Status Dokumen	HeaderDokumen	1 : N	RESTRICT	RESTRICT	Status adalah makna proses; tidak boleh berubah seenaknya
Referensi JenisTransaksi → HeaderDokumen	Ref Jenis Transaksi	HeaderDokumen	1 : N	RESTRICT	RESTRICT	Klasifikasi mempengaruhi pelaporan historis

Tabel 2. Matriks kebijakan integritas referensial

Penggunaan CASCADE pada relasi HeaderDokumen–DetailDokumen hanya diperkenankan untuk menjamin konsistensi struktur header–detail, dan tidak boleh digunakan untuk menghapus historis transaksi tanpa dasar tindakan bisnis yang sah dan jejak bukti yang dapat diaudit.

Untuk memperkuat argumentasi visual bahwa integritas referensial adalah kebijakan lintas domain (master–referensi–transaksi), Gambar 2 menampilkan peta relasi kebijakan secara ringkas.



Gambar 2. Integritas referensial sebagai kebijakan tata kelola data BERKAH

C. Normalisasi dan Konsekuensi pada Transaksi serta Pelaporan

Normalisasi tetap dipertahankan pada skema transaksi untuk menekan redundansi sekaligus mencegah anomali pembaruan. Temuan empiris pada konteks sistem dukungan keputusan (*decision support*) menunjukkan bahwa pilihan normalisasi/denormalisasi tidak selalu menghasilkan dampak kinerja yang sejalan dengan intuisi “mengurangi *join* pasti lebih cepat”; di sisi lain, redundansi justru menambah beban volume data dan biaya pemeliharaan jangka panjang. [7] Oleh karena itu, pendekatan yang lebih aman adalah menjaga data sumber transaksi tetap ternormalisasi dan berintegritas, sementara kebutuhan pelaporan dipenuhi melalui artefak turunan—misalnya *view* atau ringkasan terkelola—tanpa melonggarkan integritas pada sumber data. [7] Dengan strategi tersebut, organisasi tidak “membayar” kemudahan laporan dengan degradasi konsistensi. Pelaporan boleh dibuat lebih praktis, tetapi data sumber harus tetap ketat—sebab ketika konsistensi rusak, biaya koreksinya pada akhirnya tetap kembali ke tim sendiri.

D. Rancangan Uji Pelanggaran dan Implikasi Tata Kelola

Rancangan uji pelanggaran berperan sebagai penghubung yang operasional antara kebijakan (yang tertulis dalam dokumen) dan kontrol (yang dieksekusi oleh sistem). Argumennya sederhana: ketika setiap jalur integrasi diwajibkan tunduk pada kendala yang sama, maka konsistensi tidak lagi bergantung pada kedisiplinan pengguna, melainkan menjadi sifat bawaan (*systemic property*) dari sistem itu sendiri. Pandangan ini sejalan dengan literatur tentang kendala integritas yang menempatkan kendala sebagai mekanisme formal untuk menjaga kebenaran relasi saat terjadi operasi perubahan data, sehingga inkonsistensi dapat dicegah secara deterministik di tingkat basis data [8].

Tabel 3 merupakan rancangan uji pelanggaran (negative testing) untuk memverifikasi kebijakan integritas referensial pada tingkat basis data; pada tahap implementasi, setiap skenario dieksekusi untuk menghasilkan bukti penolakan DBMS (misalnya pelanggaran foreign key) sebagai artefak audit atas efektivitas kebijakan.

ID	Tujuan Uji	Aksi Pelanggaran	Kendala Penolak	Hasil yang diharapkan	Dampak Tata Kelola
N1	Mencegah cabang fiktif pada transaksi	Insert HeaderDokumen dengan kode_cabang yang tidak ada	FK Cabang → HeaderDokumen	Ditolak oleh DBMS (foreign key violation)	Menjaga validitas konteks cabang dan mencegah orphan record.
N2	Menjaga histori transaksi per cabang	Delete Cabang yang masih direferensikan HeaderDokumen	ON DELETE RESTRICT pada Cabang → HeaderDokumen	Ditolak oleh DBMS	Histori cabang tidak hilang; laporan historis tetap konsisten.

ID	Tujuan Uji	Aksi Pelanggaran	Kendala Penolak	Hasil yang diharapkan	Dampak Tata Kelola
N3	Menjaga pertanggungjawaban pelaku	Delete Pengguna yang direferensikan HeaderDokumen	ON DELETE RESTRICT pada Pengguna HeaderDokumen →	Ditolak oleh DBMS	Audit trail tetap utuh; pelaku tidak bisa “menghilang”.
N4	Menjaga konsistensi struktur header–detail	Delete HeaderDokumen saat masih ada DetailDokumen	Kebijakan HeaderDokumen → DetailDokumen (RESTRICT atau CASCADE bersyarat)	Ditolak (RESTRICT) atau hanya diizinkan bila prosedur bisnis sah dan jejak audit lengkap (CASCADE bersyarat)	Mencegah detail yatim dan mencegah penghapusan historis secara serampangan.
N5	Mencegah pergeseran makna histori dokumen	Delete/Update RefStatusDokumen atau RefJenisTransaksi yang sedang dipakai HeaderDokumen	ON DELETE/UPDATE RESTRICT pada Ref → HeaderDokumen	Ditolak oleh DBMS	Menjaga makna historis status/jenis; pelaporan tidak berubah makna karena referensi diutak-atik.
N6	Menjaga keterlacakan bukti	Delete BuktiDigital yang masih direferensikan HeaderDokumen atau VerifikasiBukti	ON DELETE RESTRICT pada BuktiDigital → HeaderDokumen dan VerifikasiBukti → BuktiDigital	Ditolak oleh DBMS	Bukti dan rekam verifikasi tidak hilang; audit bukti tetap lengkap.

Tabel 3. Skenario Uji Pelanggaran

Implikasi tata kelola yang paling langsung dari rancangan ini bersifat operasional dan terukur. Pertama, risiko munculnya *data yatim* menurun karena kondisi inkonsisten dicegah sejak awal melalui penolakan pada tingkat basis data. Kedua, konsistensi definisi identitas menjadi lebih terjaga karena setiap transaksi “dipaksa” merujuk pada data induk yang sah, sehingga rujukan tidak bergeser mengikuti interpretasi lokal. Ketiga, auditabilitas meningkat karena konteks historis transaksi tetap utuh—khususnya dengan mencegah penghapusan data induk yang tidak semestinya, yang sering kali membuat jejak transaksi kehilangan keterkaitan maknanya.

IV.KESIMPULAN

Penelitian ini merumuskan rancangan model logis basis data BERKAH dengan menempatkan integritas referensial sebagai kebijakan tata kelola data, bukan sekadar preferensi teknis pada tahap implementasi. Kontribusi utamanya mencakup: (1) kerangka model logis ternormalisasi yang dirangkum pada Tabel 1; (2) matriks kebijakan integritas referensial yang memformalkan relasi-relasi kritis beserta perilaku perubahan data agar tetap selaras dengan makna bisnis (Tabel 2 dan Gambar 1); serta (3) rancangan uji pelanggaran untuk membuktikan bahwa kendala yang ditetapkan benar-benar menolak kondisi inkonsisten (Tabel 3 dan Gambar 2). Dengan kerangka tersebut, integritas referensial diperlakukan sebagai “aturan organisasi” yang dapat diverifikasi, diulang pengujiannya, dan diaudit secara sistematis. [2], [8]

Ucapan Terima Kasih

Peneliti mengucapkan banyak terima kasih kepada Bapak Dedi Zein selaku Pemilik Zein Corp, Fakultas Teknik dan Program Studi Teknik Informatika Universitas Pasundan, Ketua Program Studi, para dosen dan pihak lain yang telah mendukung berjalannya kegiatan penelitian ini.

REFERENSI

- [1] A. K. Thomer dan K. M. Wickett, “Relational data paradigms: What do we learn by taking the materiality of databases seriously?,” *Big Data & Society*, vol. 7, no. 1, 2020, doi: 10.1177/2053951720934838.
- [2] B. M. V. Bernardo, H. S. Mamede, J. M. P. Barroso, dan V. Santos, “Data governance & quality management—Innovation and breakthroughs across different fields,” *Journal of Innovation & Knowledge*, vol. 9, no. 4, art. 100598, 2024, doi: 10.1016/j.jik.2024.100598.

- [3] D. Acev, S. Biyani, F. Rieder, T. T. Aldenhoff, M. Blazevic, D. M. Riehle, dan M. A. Wimmer, “Systematic analysis of data governance frameworks and their relevance to data trusts,” *Management Review Quarterly*, 2025, doi: 10.1007/s11301-025-00545-1.
- [4] E. F. Codd, “A relational model of data for large shared data banks,” *Communications of the ACM*, vol. 13, no. 6, hlm. 377–387, 1970, doi: 10.1145/362384.362685.
- [5] E. F. Codd, *The Relational Model for Database Management: Version 2*. Reading, MA: Addison-Wesley, 1990.
- [6] K. Kumar dan S. Kumar, “Relational database design: A review,” *International Journal of Computer Applications*, vol. 176, no. 6, hlm. 14–18, 2017, doi: 10.5120/ijca2017915626.
- [7] M. Fotache, M.-I. Cluci, T. Taipalus, dan G. Talaba, “The effects of database normalization on decision support system performance,” *Information Systems*, art. 102636, 2026 (tersedia daring lebih awal 2025), doi: (lihat laman artikel penerbit).
- [8] P. W. P. J. Grefen dan P. M. G. Apers, “Integrity control in relational database systems—An overview,” *Data & Knowledge Engineering*, vol. 10, no. 2, hlm. 187–223, 1993, doi: 10.1016/0169-023X(93)90008-D.
- [9] R. Elmasri dan S. B. Navathe, *Fundamentals of Database Systems*, ed. ke-7. Boston: Pearson, 2015.
- [10] T. J. Teorey, D. Yang, dan J. P. Fry, “A logical design methodology for relational databases using the extended entity-relationship model,” *ACM Computing Surveys*, vol. 18, no. 2, hlm. 197–222, 1986, doi: 10.1145/7474.7475.
- [11] V. M. Markowitz, “Safe referential integrity structures in relational databases,” dalam *Proceedings of the 17th International Conference on Very Large Data Bases (VLDB)*, Barcelona, 1991, hlm. 123–132.